User's Manual

32

SH7734

User's Manual: Hardware

Renesas 32-Bit RISC Microcomputer
SuperH™ RISC engine Family / SH-4A Series

Renesas Electronics
www.renesas.com

Rev.1.00 Jun 2012

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
3. Renesas Electronics does not assume any liability for infringement of patents, copyrights, or other intellectual property rights of third parties by or arising from the use of Renesas Electronics products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from such alteration, modification, copy or otherwise misappropriation of Renesas Electronics product.
5. Renesas Electronics products are classified according to the following two quality grades: “Standard” and “High Quality”. The recommended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below.

   “Standard”: Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots etc.

   “High Quality”: Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; and safety equipment etc.

   Renesas Electronics products are neither intended nor authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems, surgical implantations etc.), or may cause serious property damages (nuclear reactor control systems, military equipment etc.). You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application for which it is not intended. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for which the product is not intended by Renesas Electronics.
6. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
7. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or systems manufactured by you.
8. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
9. Renesas Electronics products and technology may not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You should not use Renesas Electronics products or technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. When exporting the Renesas Electronics products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations.
10. It is the responsibility of the buyer or distributor of Renesas Electronics products, who distributes, disposes of, or otherwise places the product with a third party, to notify such third party in advance of the contents and conditions set forth in this document, Renesas Electronics assumes no responsibility for any losses incurred by you or third parties as a result of unauthorized use of Renesas Electronics products.
11. This document may not be reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

(2012.4)

General Precautions on Handling of Product

1. Treatment of NC Pins

Note: Do not connect anything to the NC pins.
The NC (not connected) pins are either not connected to any of the internal circuitry or are they are used as test pins or to reduce noise. If something is connected to the NC pins, the operation of the LSI is not guaranteed.

2. Treatment of Unused Input Pins

Note: Fix all unused input pins to high or low level.
Generally, the input pins of CMOS products are high-impedance input pins. If unused pins are in their open states, intermediate levels are induced by noise in the vicinity, a pass-through current flows internally, and a malfunction may occur.

3. Processing before Initialization

Note: When power is first supplied, the product's state is undefined.
The states of internal circuits are undefined until full power is supplied throughout the chip and a low level is input on the reset pin. During the period where the states are undefined, the register settings and the output state of each pin are also undefined. Design your system so that it does not malfunction because of processing while it is in this undefined state. For those products which have a reset function, reset the LSI immediately after the power supply has been turned on.

4. Prohibition of Access to Undefined or Reserved Addresses

Note: Access to undefined or reserved addresses is prohibited.
The undefined or reserved addresses may be used to expand functions, or test registers may be allocated to these addresses. Do not access these registers; the system's operation is not guaranteed if they are accessed.

5. Reading from/Writing to Reserved Bit of Each Register

Note: Treat the reserved bit of register used in each module as follows except in cases where the specifications for values which are read from or written to the bit are provided in the description.
The bit is always read as 0. The write value should be 0 or one, which has been read immediately before writing.
Writing the value, which has been read immediately before writing has the advantage of preventing the bit from being affected on its extended function when the function is assigned.

Configuration of This Manual

This manual comprises the following items:

1. General Precautions on Handling of Product
2. Configuration of This Manual
3. Preface
4. Contents
5. Overview
6. Description of Functional Modules
    - CPU and System-Control Modules
    - On-Chip Peripheral Modules

      The configuration of the functional description of each module differs according to the module. However, the generic style includes the following items:

    i) Feature
    ii) Input/Output Pin
    iii) Register Description
    iv) Operation
    v) Usage Note

When designing an application system that includes this LSI, take notes into account. Each section includes notes in relation to the descriptions given, and usage notes are given, as required, as the final part of each section.

7. Electrical Characteristics
8. Appendix
9. Index

Preface

This LSI is a RISC (Reduced Instruction Set Computer) microcomputer which includes a Renesas original RISC CPU as its core, and the peripheral functions required to configure a system.

Target Users: This manual was written for users who will be using this LSI in the design of application systems. Users of this manual are expected to understand the fundamentals of electrical circuits, logical circuits, and microcomputers.

Objective: This manual was written to explain the hardware functions and electrical characteristics of this LSI to the above users.

Notes on reading this manual:

- In order to understand the overall functions of the chip
  Read the manual according to the contents. This manual can be roughly categorized into parts on the CPU, system control functions, peripheral functions and electrical characteristics.
- In order to understand the details on FPU functions and each instructions
  Read the additional volume, SH-4A Extended Functions Software Manual.

Rules:		
	Bit order:	The MSB is on the left and the LSB is on the right.
	Number notation:	Binary is B'xxxx, hexadecimal is H'xxxx, decimal is xxxx.
	Signal notation:	An overbar is added to a low-active signal: $\overline{\text{xxxx}}$

Abbreviations

ALU	Arithmetic Logic Unit
ASID	Address Space Identifier
ATAPI	AT Attachment Packet Interface
BSC	Bus State Controller
CPG	Clock Pulse Generator
CPU	Central Processing Unit
DBSC	DDR-SDRAM Bus State Controller
DMA	Direct Memory Access
DMAC	Direct Memory Access Controller
ETU	Elementary Time Unit
FIFO	First-In First-Out
FPU	Floating point number Processing Unit
H-UDI	User Debugging Interface
IIC	Inter IC bus
INTC	Interrupt Controller
IrDA	Infrared Data Association
JPEG	Joint Photographic Experts Group
JTAG	Joint Test Action Group
LCDC	LCD Controller
LRU	Least Recently Used
LSB	Least Significant Bit
MMC	Multi Media Card

MMU	Memory Management Unit
MPEG	Motion Picture Experts Group
MSB	Most Significant Bit
PC	Program Counter
PFC	Pin Function Controller
RISC	Reduced Instruction Set Computer
RTC	Realtime Clock
SCIF	Serial Communication Interface with FIFO
STIF	Stream Interface
TAP	Test Access Port
TLB	Translation Lookaside Buffer
TMU	Timer Unit
TPU	Timer Pulse Unit
UART	Universal Asynchronous Receiver/Transmitter
UBC	User Break Controller
USB	Universal Serial Bus
VEU	Video Engine Unit
WDT	Watchdog Timer

Contents

Section 1 Overview

1.1 Introduction

1.1.1 SH7734

The SH7734 is a next-generation SH-4A series SOC featuring the basic functions for next-generation graphics and display applications.

The SH7734 includes the following.

- A 533-/400-MHz SH-4A CPU core,
- memory controller for DDR-SDRAM,
- graphic engine,
- camera interface,
- TFT panel display function,
- SD card interface,
- USB2.0 interface, and
- gigabit Ethernet interface.

Also, a full implementation of the extremely expandable and flexible SuperHyway bus has been adopted as the internal bus for the SH7734. This bus structure is optimized for maximum system performance. The SH7734 also supports booting from various types of memory (including NAND-type or serial flash memory), and low-power supply modes.

The SH7734 can be used to realize display for a graphical user interface (GUI), specifically in the form of 2D-graphical displays for easy operability. The chip provides the basis for realizing high-performance compact systems with excellent cost-performance ratios.

Note: Frequency notation
Frequencies are given as integers unless greater accuracy is required.
In this section, however, frequencies are given to one decimal place (e.g. 533.3 MHz).
Frequencies in mode names are given as integers (e.g. 533-MHz mode).
Frequencies in descriptions related to the CPG and so on are given to the second decimal place.

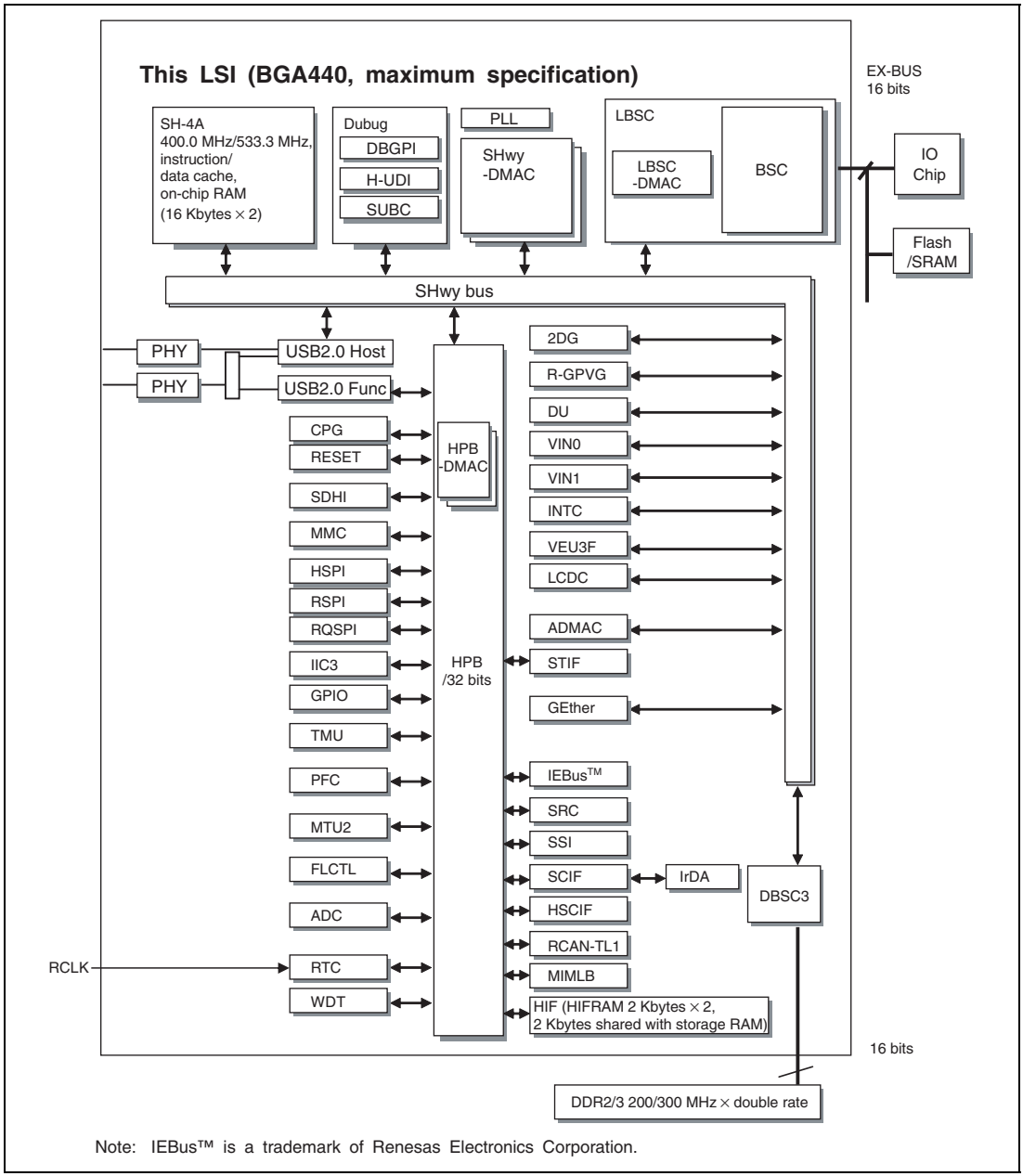
1.2 Correspondence between Type Number and Function

Table 1.1 List of Type Numbers (SH7734G)

Abbreviation	Package Type	Cryptographic Function	Temperature Range (°C)	Operating Frequency (MHz)	Renesas Type Number	Marking	Main Fields of Application
SH7734G	BGA440	No	–40 to 85	533.3	R8A77343PAxxBG	R8A77343P533BG	—
			–40 to 85	533.3	R8A77343DAxxBG	R8A77343D533BG	Industrial and Consumer

Note: Axx refers to the inclusion of functions in products as listed in section 1.5, Product Type Numbers (e.g. A00, A01). The type numbers of products for export from Japan are the same as those listed under “Marking” above.

1.3 System Configuration Diagram



Note: IEBus™ is a trademark of Renesas Electronics Corporation.

Figure 1.1 System Configuration of This LSI

1.4 List of Specifications

Specifications of the modules in this LSI are listed below.

1.4.1 SH-4A Core

Item	Description
Maximum operating frequency	• 533.3 MHz or 400.0 MHz
Performance	• 960 MIPS (for operation at 533.3 MHz), 3.73 GFLOPS (for operation at 533.3 MHz) • 720 MIPS (for operation at 400.0 MHz), 2.8 GFLOPS (for operation at 400.0 MHz)
FPU	• Incorporates floating-point processor • Single-precision (32 bits), double-precision (64 bits) supported
Memory management unit	• 4-GByte address space, 256 address spaces (ASID 8 bits) • Single virtual memory mode and multiple virtual memory mode • Supports multiple page sizes: 1 K, 4 K, 8 K, 64 K, 256 K, 1 M, 4 M, 64 Mbytes • Fully associative TLB with four entries for commands • Fully associative TLB with 64 entries for commands and operands • Supports software-controlled replacement and randomcounter replacement algorithm • TLB contents can be directly accessed through address mapping • In addition to conventional 29-bit physical address mode, 32-bit physical address mode also supported

Item	Description
Cache memory	• Instruction cache: — 32-Kbyte, 4-way set-associative — 256-entry, 32-byte block length • Operand cache: — 32-Kbyte, 4-way set-associative — 256-entry, 32-byte block length — Selectable write method (copy-back/write-through) • One-stage copy-back buffer, one-stage write-through buffer • Store queue: 32 bytes × two entries
LRAM	• ILRAM 16 Kbytes of high-speed access memory Three independent read/write ports Instruction fetch access from the CPU 8-/16-/32-/64-bit access from the CPU 8-/16-/32-/64-bit and 16-/32-byte access by external requests • OLRAM 16K bytes of high-speed access memory Three independent read/write ports Operand access from the CPU 8-/16-/32-/64-bit access from the CPU 8-/16-/32-/64-bit and 16/32-byte access by external requests
User break controller	• Supports debugging through user break interrupts • Two break channels • Addresses, data values, access types, data size, can all be set as break conditions • Supports a sequential break function

1.4.2 CPU Core Peripherals

Item	Description
Operating clock pulse generation circuit (CPG)	• CPU clock: 12/16/24/32 times EXTAL • Clock modes: For 400-MHz mode (1) — CPU frequency: 2/3 (max. 400.00 MHz) — Internal clocks: 2/3, 1/3, 1/6, 1/12 — SHwy-bus frequency: 1/3, 1/6 (max. 200.00 MHz) — HPB-bus frequency: 1/6, 1/12 (max. 100.00 MHz) — DDR controller: Asynchronously connected with the SHwy bus. (max. 300.00 MHz) — EX-BUS frequency: 1/12 (max. 50.00 MHz) For 400-MHz mode (2) — CPU frequency: 1/1 (max. 400.00 MHz) — Internal clock: 1/1, 1/2, 1/4, 1/8 — SHwy-bus frequency: 1/2, 1/4 (max. 200.00 MHz) — HPB-bus frequency: 1/4, 1/8 (max. 100.00 MHz) — DDR controller: Asynchronously connected with the SHwy bus (max. 200.00 MHz) — EX-BUS frequency: 1/8 (max. 50.00 MHz) For 533-MHz mode — CPU frequency: 1/1 (max. 533.33 MHz) — Internal clock: 1/1, 1/3, 1/6, 1/12 — SHwy-bus frequency: 1/3, 1/6 (max. 177.77 MHz) — HPB-bus frequency: 1/6, 1/12 (max. 88.88 MHz) — DDR controller: Asynchronously connected with the SHwy bus (max. 266.66 MHz) — EX-BUS frequency: 1/12 (max. 44.44 MHz) • Provided with module standby control registers to stop or enable clock supply to individual on-chip modules • Power-down modes: Stops clock supply to individual peripheral modules (module standby), sleep mode, software standby mode, deep standby mode

Item	Description
Reset (RESET)	• One reset-signal external output port for external modules provided
Interrupt controller (INTC)	• Five independent external interrupts: NMI, IRQ3 to IRQ0 • External interrupts encoded at 15 levels supported: IRL3 to IRL0 • On-chip peripheral interrupts: priority levels set for each module • GPIO interrupts
Watchdog timer (WDT)	• Monitors for system crashes by using a timer counting a specified interval • Has two different modes in response to an overflow of the counter: the internal modules are reset in watchdog timer mode; an interrupt is generated in interval timer mode. • Generates a reset for on-chip peripheral modules when counter overflow occurs in watchdog timer mode • A power-on reset or a manual reset is selectable • In interval timer mode, an interval timer interrupt is generated on counter overflow. • To prevent easy rewriting of the registers related to the WDT, settings must be made with a code value in the 8 higher-order bits. • The maximum time until the counter overflows differs according to the HPB-bus clock (Pck) frequency as follows: approximately 1546 seconds (when the Pck frequency is 44.44 MHz), approximately 1648 seconds (when the Pck frequency is 41.66 MHz)
Realtime clock (RTC)	• Clock, calendar, and alarm functions • 32.768-MHz crystal oscillator provides resolution of up to 1/256-sec (interrupt cycle) • Input of RCLK is not necessary when the RTC is not in use.

Item		Description
Direct memory access controller (DMAC)	LBSC-DMAC	• 3-channel physical address DMA controller (Refer to section 1.8, Direct Memory Access Controllers.)
	SHwy-DMAC	• 2-channel physical address DMA controller (Refer to section 1.8, Direct Memory Access Controllers.)
	HPB-DMAC	• 28-channel physical address DMA controller (Refer to section 1.8, Direct Memory Access Controllers.)
Local bus state controller (LBSC)		• EX-BUS interface Max. 44.44 MHz (for 533-MHz mode) /16-bit bus Max. 50.00 MHz (for 400-MHz mode (1) and (2)) /16-bit bus Synchronized to a bus clock frequency which is 1/12th or 1/8th of the CPU operation frequency. • External area divided into up to eight areas and managed — Allocation to space of area 0, area 1, and area 6 or allocation to space of area 0 only is selected at startup time. — Area 0 supports 128-Mbyte memory space (startup mode). — Area 6 space is further divided into up to six areas (capacity of each area variable) and managed (when the MD9 pin = 0). — Space of area 0 is divided into up to seven areas (capacity of each area variable) and managed (when the MD7 pin = 0 and the MD9 pin = 1). — I/F settings, bus width settings, wait state insertion possible for each area • SRAM interface — Wait states can be inserted through register settings Period of waiting is set in cycle unit, and the maximum value is 15. — EX_WAIT pin can be used for wait state insertion — Connectable bus widths: 16 bits or 8 bits

Item	Description
Local bus state controller (LBSC)	• Burst ROM interface — Wait states can be inserted through register settings — Number of bursts can be set through register — Possible bus widths: 16 bits or 8 bits • Byte-control SRAM interface (available with areas 1 and 6 only) — Byte-control SRAM interface — Wait states can be inserted through register settings — EX_WAIT pin can be used for wait state insertion — Connectable bus widths: 16 bits or 8 bits • ATA interface — Wait states can be inserted through register settings — Supports PIO modes 0 through 4 — Supports multi-word modes 0 through 2 — Ready timeout detection (detection time (ns) = EX-BUS operating frequency (ns) × 100 clock cycles) • Supports external buffer enable/direction control
Others	• Supports H-UDI (User Debugging Interface) • Supports AUD (Advanced User Debugger)

1.4.3 Memory Control Unit

Item	Description
DDR2-SDRAM and DDR3-SDRAM controller (DBSC3)	• Multi-bank support: Supports multi-bank operation with eight banks • Supports either four (DDR2-SDRAM) or eight (DDR2-SDRAM and DDR3-SDRAM) banks • Supports an external bus width of 16 bits • Advance precharge-activate function • Supported operating mode: Burst length: DDR2-SDRAM 8 (fixed) or 4 (fixed) DDR3-SDRAM 8 (fixed) Burst type: Sequential (fixed) • Power-down mode Supports self-refresh mode, power-down mode, SDRAM power buck-up mode, and deep standby mode. • Timing settings: The following timing settings can be made CAS latency, CAS write latency, ACT-READ/WRITE minimum period, PRE period, ACT-ACT/REF minimum period, ACT-PRE minimum period, ACT(A)-ACT(B) minimum period, 4-active-window minimum period, READ-PRE minimum period, write recovery period, READ-WRITE minimum period, WRITE-READ minimum period, REF-ACT/REF minimum period, CKE-Hi minimum period, CKE-Low minimum period, short calibration period. Only 0 is supported for additive latency (AL). • Refresh operation Average interval and the maximum post count are set by a register. If an empty cycle for request is found, preceding refresh can be performed. • This controller supports both DDR2-SDRAM and DDR3-SDRAM (connect a DDR-SDRAM for operation within the available range of frequencies). DDR2-SDRAM and DDR3-SDRAM operation at up to 600 Mbps

Item	Description	
Memory connections	8- or 16-bit bus (16 bits recommended)	• (Pins MA13 to MA0) One memory unit with 32 Mwords × 16 bits (512 Mbits), one memory unit with 64 Mwords × 16 bits (1 Gbit), one memory unit with 128 Mwords × 16 bits (2 Gbits) • (Pins MA12 to MA0) One memory unit with 32 Mwords × 16 bits (512 Mbits), one memory unit with 64 Mwords × 16 bits (1 Gbit)

1.4.4 Internal Bus Configuration and Arbitration Specifications

Item	Description
SHwy basic bus	Of the slave devices within and outside this LSI, slave devices on the external bus (accessed via an LBSC or a DDR-SDRAM controller) and the SH-4A core internal memory can be accessed by all modules having a master function in this LSI chip (the SH-4A core, debug module, LBSC, HPB, SHwy-DMAC, 2DG, DU, VIN0, VIN1, VEU3F, USB, GETHER, A-DMAC, LCDC). Slave devices within this LSI other than the above (primarily registers in this chip) can be accessed only from the SH-4A core and debug module.
HPB bridge	200.00 MHz/177.77 MHz/166.66 MHz 64 bits → 100.00 MHz /88.88 MHz/83.33 MHz 32 bits This bridge is connected to the slave interfaces for the following modules. 2DG, DU, VIN0, VIN1, SSI, SCIF, HSCIF, RCAN, USB, CPG, RESET, SDHI, MMC, HSPI, RSPI, RQSPI, I2C, MIMLB, GPIO, TMU, PFC, ADC, RTC, IEBus, STIF, MTU2, FLCTL, and SRC
Arbitration specifications	A round-robin method is used for arbitration of the register bus. A weighted round-robin method is used for arbitration of DDR access. During exception and interrupt handling by the SH-4A (i.e. while the block bit, BL, in the status register of the SH-4A is 1), the access by the SH-4A has the higher priority than the register bus and DDR access by other modules.* Note: * Continuous access at the highest priority during exception or interrupt handling by the SH-4A (BL = 1) may affect the handling of real-time tasks such as the display system or audio data transfer.

1.4.5 Graphics Units

Item		Description
Maximum operating clock frequency	Graphics engine operation	This module can operate either at the SHwy frequency or at half of the SHwy frequency (clks1). Example: 177.77/88.88 MHz (for 533-MHz mode) 200.00/100.00 MHz (for 400-MHz mode)
	Display system, maximum internal operating frequency	This module operates at half the SHwy frequency (clks1). Example: 88.88 MHz (for 533-MHz mode) 100.00 MHz (for 400-MHz mode)
Operating clock		The maximum frequency differs with the type of package. HPB frequency: The HPB operates at half the frequency of clks1. Example: 44.44 MHz (for 533-MHz mode) 50.00 MHz (for 400-MHz mode) SHwy frequency: Example: 177.77 MHz (for 533-MHz mode) 200.00 MHz (for 400-MHz mode)

Item			Description
Graphics engine basic functions (2DG)	2D function	2DG	• Drawing functions: Four-vertex surface drawing, polygon drawing, line drawing, highly functional thick line drawing, anti-aliasing, BitBLT with raster operations/α blending • Color display: Source: 1, 8, 16 bits/pixel, Drawing: 8, 16 bits/pixel • Work: 2 bits/pixel • Screen coordinates: X direction: 0 to 4095, Y direction: 0 to 4095
		R-GPVG Note: If you intend to use this module, select an applicable product from among those listed in section 1.5, Product Type Numbers.	• Supports OpenVG, an open API for two-dimensional vector graphics • 2048-by-2048-dot data can be created in the memory • Command FIFO

Item		Description
Display functions (DU) Note: The four modules VIN0, VIN1, DU, and LCDC cannot be in use at the same time due to the bus performance (however, up to three of these modules may be in use at the same time with DDR memory operating at 600 Mbps).	Recommended screen size and number of composite planes	• Up to WXGA (1280 × 768) planes settable • Recommended screen size: WVGA planes (832 × 496 dots, 32 bits/pixel) (only when the VIN is not in use) • Composition of displays from 8 planes (used in common as α planes) is possible.
	Extended video display	Capable of performing field interpolation and interlaced-scanning centroid matching for input PAL/NTSC and other video images, thereby eliminating screen noise from the video display.
	CRT scanning method	• Non-interlaced, interlacing, interlacing & video
	External sync	• Master, TV sync
	Internal color palettes	Contains four color palette planes which can display 256 out of 260,000 colors at the same time.
	Digital RGB output	• One output stream • 8-bit resolution for each RGB color
	Blend ratio settings	Number of color palette planes with blend ratio: 4
	Buffer size	128 bytes × 3 planes
	Dot clock	Switchable between external input and internal clock (dividing ratio: 1 to 64) Tentative performance: Whichever is lower of half the SHwy frequency or 80 MHz is selected. Example: 80 MHz or less when the SHwy frequency is 200 MHz.
Video input (VIN0) Note: See the Note in the entry on display functions (DU).	Input interface	ITU-R BT.601, 656 or 1358 interface
	Horizontal-direction scaling	Uses a 9-tap multi-phase filter. Up to maximum 2 ×
	Vertical-direction scaling	Scaling by linear interpolation Up to maximum 3 ×
	Input format	YUV422 in 8-bit YC format YUV422 in 16-bit YC format 18-bit RGB666
	Output format	RGB-565, ARGB-1555, YUV420, YUV422
	YC separation	Data is separated into Y and UV components

Item	Description
Video input (VIN1) Note: See the Note in the entry on display functions (DU).	• ITU-R BT.656 interface compliant • Scaling up (up to two-fold) or down in the horizontal and vertical direction • Data format for image capture: YCbCr4:2:2 • Output format: YCbCr4:2:2 • Capturing field images with up to 720 pixels × 480 lines is possible

1.4.6 Video

Item	Description
Video engine unit (VEU3F)	• One channel • Image processing functions for data within memory — Video processing functions Image processing functions: Input image format: YCbCr, RGB Output image format: YCbCr, RGB Shrunk/enlarged-image generation filtering function YCbCr→RGB/RGB→YCbCr conversion function Dithering function (for RGB color reduction) — Filtering functions Deblocking filter Median filter Picture quality enhancement FIR filter — Combined video image processing and filter processing functions

1.4.7 Sound Interface

Item	Description
Serial sound interface (SSI)	• Four internal channels • Supports versatile serial audio formats • Supports master/slave functions • Programmable word clock, bit clock generation functions • Multichannel format functions • Supports 8-/16-/18-/20-/22-/24-/32-bit data formats
Sampling rate converter (SRC)	• Data size: stereo 32 bits (L/R 16 bits each), monaural 16 bits • Input sampling rate: 8, 11.025, 12, 16, 22.05, 24, 32, 44.1, or 48 kHz is selectable. • Output sampling rate: either 44.1 or 48 kHz is selectable.

1.4.8 Peripheral Modules

Item	Description
USB2.0 host interface	• Supports EHCI ver 1.0, OCHI ver 1.0a • Transfer rate: high speed, full speed, low speed • Two ports One of two ports can be replaced with the USB 2.0 function interface
USB2.0 function interface	• Internal USB device controller (UDC) conforming to the USB2.0 specifications Automatic processing of USB standard commands (except for some commands) Get Descriptor/Class/Vendor commands are processed by microprocessor firmware • Control, bulk, interrupt transfers supported • Transfer rate: High speed, full speed

Item	Description
Timer unit (TMU)	• Nine channels • Each channel equipped with auto-reloading 32-bit count-down counter • Channels 2 and 5 only equipped with input capture function • Channels 0 to 5 only enable selection of rising edge/falling edge as external clock input edge when external clock is selected or during use of input capture function • Each channel provided with 32-bit timer constant register for auto-reload use, readable/writable at any time, and 32-bit down-counter • Channels 0 to 5 only enable selection from six counter input clocks — External clock (TCLK) — Five peripheral clocks (clkp/4, clkp/16, clkp/64, clkp/256, clkp/1024) (where clkp is peripheral clock) • Channels 6 to 8 enable selection from five counter input clocks — Five peripheral clocks (clkp/4, clkp/16, clkp/64, clkp/256, clkp/1024) (where clkp is peripheral clock) • Two types of interrupt source — Underflow × one source (each channel), input capture × one source (channels 2 and 5) provided

Item	Description
Multi-function timer pulse unit 2 (MTU2)	• Up to sixteen types of pulse input and output are possible based on the 5-channel 16-bit timer • Eighteen output compare/input capture registers • Input capture function • Pulse output mode Toggle/PWM/complimentary PWM/reset synchronous PWM • Multiple timer counters can be written to simultaneously • Complimentary PWM output mode — 3-phase non-overlapped waveform output for controlling inverters — Automatic dead time setting — PWM duty ratio settable as required within the range from 0 to 100 % — A/D converter start request delaying function — Interrupts at the crest and trough can be skipped • Reset synchronous PWM mode Three positive-/inverse-phase PWM waveform outputs with required duty cycle. • Phase counting mode 2-phase encoder counting is possible
I^2C bus interface 3 (IIC3)	Two channels • Master/slave functions
Serial peripheral interface (HSPI)	One channel • Master/slave functions • Full-duplex communication is possible • Programmable data rate

Item	Description
Renesas Serial Peripheral Interface (RSPI)	• One channel (and booting from an SPI is possible) • SPI operation • Support for master and slave modes • Programmable bit length and active sense of the serial transfer clock, and selectable clock phase • Transfer can be executed sequentially. • MSB-first or LSB-first selectable • Maximum transfer rate: 12.5 Mbps (when the CPU frequency is 400 MHz and this module is directly connected to flash memory)
Renesas Quad Serial Peripheral Interface (RQSPI)	• One channel • Connectable to serial flash memory with multiple (single/dual/quad) types of I/O • Programmable bit length and active sense of the serial transfer clock, and selectable clock phase • Transfer can be executed sequentially • MSB-first or LSB-first selectable • Maximum transfer rate: 50 Mbps (when the CPU frequency is 400 MHz and this module is directly connected to flash memory)
RCAN-TL1 (RCAN) Note: If you intend to use this module, select an applicable product from among those listed in section 1.5, Product Type Numbers.	• Two channels • Supports CAN specification 2.0B • Bit timing compliant with ISO-11898-1 • Thirty-two mailbox versions • Thirty-one programmable mailboxes for transmit/reception and one programmable receive-only mailbox • Sleep mode for low power consumption and automatic recovery from sleep mode by detecting CAN bus activity • Programmable receive filter mask (standard and extended identifier) supported by all mailboxes • Programmable CAN data rate up to 1 Mbit/s • Transmit message queuing with internal priority sorting mechanism against the problem of priority inversion for real-time applications

Item	Description
Serial communication interfaces (SCIF)	• Six channels (SCIF0 to SCIF5) • Asynchronous, clock-synchronized modes • Asynchronous serial communication Serial data is transmitted using an asynchronous method in which synchronization is in character units. Serial data communication with standard asynchronous communication LSIs, such as a Universal Asynchronous Receiver/Transmitter (UART) and Asynchronous Communication Interface Adapter (ACIA), is possible. The serial data communication format can be selected from among eight formats. — Data length: 7 bits or 8 bits — Stop bits: 1 bit or 2 bits — Parity: Even parity, odd parity, no parity — Reception error detection: Parity error, framing error, overrun error detection — Break detection: When a framing error occurs, and then there is a space (low level) of one frame length or longer in succession, a break is detected. A break can also be detected at the time of a framing error when the level of the Rxn pin is read directly from the serial port register (SCSPTRn). • Clock-synchronized serial communication Serial data communication with clock synchronization is performed. Serial data communication with other LSIs having a clock-synchronized communication function is possible. There is one serial data communication format. — Data length: 8 bits — Reception error detection: Overrun error detection • Full-duplex communication is possible An independent transmission unit and reception unit are provided, so that simultaneous transmission and reception are possible. Both the transmission unit and the reception unit have a 16-stage FIFO buffer, so that serial data can be continuously transmitted and received.

Item	Description
Serial communication interfaces (SCIF)	• An internal baud rate generator can be used to select an arbitrary bit rate Transmission/reception clock source can be selected: the clock generated by the internal baud rate generator using the LSI internal clock as the reference clock or the external clock • The transmission/reception clock source can be selected from the internal clock from the baud rate generator, or the SCK0, 1, 2, external clocks (for channels 0, 1 and 2 only). • Eight interrupt sources There are eight interrupt sources (transmission FIFO data empty, break, reception FIFO data full, reception error, receive data ready, transmission end, overrun error, and timeout), and interrupts can be requested independently. • When the transmission FIFO is empty or when there is received data in the reception FIFO, the DMA controller (DMAC) can be activated and data can be transferred by requesting a DMA transfer. • A modem control function (RTS, CTS) in asynchronous mode is provided (channels 0 and 1 only) • The amount of data in the transmission/reception FIFO registers and the number of reception errors for received data in the reception FIFO register can be determined. • During reception in asynchronous mode, receive data ready (DR) and timeout errors (TO) can be detected.
IrDA	Used in combination with an SICF (channel 3); this module modulates and demodulates the data format for the serial communications interface to produce the data format for IrDA infrared communications.
High-speed serial communication interface (HSCIF)	• One channel (available in all package types) • Incorporates 128-stage FIFOs. • Supports an asynchronous mode.

Item	Description
SD card host interface (SDHI) Note: If you intend to use this module, select an applicable product from among those listed in section 1.5, Product Type Numbers.	• Three channels (one of three interfaces is for booting.) • Supports SD memory/SDIO interface (1-/4-bit SD buses). • Error check function: CRC7 (command/response), CRC16 (data) • Card detection function • Supports write protection.
MMC interface (MMC)	• Controls the Multimedia Card (MMC). • Data bus: 1 bit/4 bits/8 bits • Error check function: CRC7 and CRC16 • Supports MMC mode (not SPI mode). • Supports block transfer (not stream transfer). • Block size for multi-block transfer: 512 bytes • MMC 4.3 compliant for storage • Bootable from MMC 4.4

Item	Description
MOST Interface Module MediaLB (MIMLB)	Media Local Bus (MediaLB) • Supports MediaLB available for connection with INIC made by SMSC. (including OS62400-Ver.1.4 module from SMSC) • 3-pin interface • MediaLB Ver.2.0-compliant and supports data transfer rates of up to 50 Mbps. • Stream transfer (synchronous) Transferring data between the CPU and MediaLB. (A maximum of two lines of transfers can be performed.) • Packet transfer (asynchronous) Transferring data simultaneously with one transmission line and one reception line. • Control transfer Transferring data simultaneously with one transmission line and one reception line. • Does not support the DTCP function. Note: For this module, a clock signal with a frequency greater than 66 MHz needs to be supplied as clks1. For example, if the frequency of the CPU clock is 400 MHz in 400-MHz modes (1) and (2), this is satisfied when the frequency of the signal supplied as clks1 is 100 MHz.
IEBus	• Supports IEBus protocol control (layer 2) • Half-duplex asynchronous communications • Multi-master operation • Broadcasting • Two modes with different transfer rates are selectable. Mode 0: Approximately 4.1 Kbps (for operation at 6.29 MHz), transfer of up to 16 bytes/frame Mode 1: At approximately 18 Kbps (for operation at 6.29 MHz), transfer of up to 32 bytes/frame • 32-byte buffers for both transmission and reception • Consecutive transfer of up to 32 bytes, i.e. the maximum number of bytes per frame in mode 1 • External driver and receiver • IEBus frequency: 6.29 MHz

Item	Description
Gigabit ethernet controller (GETHER)	• E-DMAC (Direct Memory Access Controller for Ethernet controller) function — Data transfer between GETHER and external/internal memory — 32-byte burst transfer — Supports single-frame/single-descriptor operation and single-frame/multi-descriptor (multibuffer) operation — Transfer data width: 32 bits — Transmit/receive FIFO (for transmission: 2 Kbytes, for reception: 4 Kbytes) • MAC (Media Access Control) function — 1 channel — Constructs/deconstructs data frames (frame format conforming to IEEE802.3, 2000 Edition) — Variable transfer rate: supports transfer at 10, 100, and 1000 Mbps — Supports full-duplex and half-duplex modes — Flow control conforming to IEEE802.3x is possible. A PAUSE frame can be transmitted automatically or manually for flow control. — IEEE802.1Q (VLAN) compliant — Supports three PHY interfaces conforming to IEEE802.3: GMII (Gigabit Media Independent Interface), MII (Media Independent Interface), and RMII (Reduced Media Independent Interface) — Upper-layer protocol support (checksum) function
LCD controller (LCDC) Note: See the Note in the entry on display functions (DU).	• Supports LCD-panel sizes from 16 × 1 to 1024 × 1024 • Supports 4-/8-/15-/16-bpp color modes • Supports 1-/2-/4-/6-bpp color grayscale modes • Supports TFT/DSTN/STN displays • Signal polarity switchable • 24-bit color palette memory (16 out of 24 bits are valid; R:5/G:6/B:5) • Unified memory architecture adopted for the graphics memory

Item	Description
Host interface (HIF)	• A total of 4-Kbytes of buffer RAM (2 Kbytes × 2 banks) • Buffer RAM is connected with external devices in parallel through 16 data pins. • Buffer RAM and on-chip CPU are connected in parallel through the internal bus. • Access to a desired register can be obtained from an external device by setting the index register (automatic incrementation of addresses is possible during access to consecutive addresses of the buffer RAM). • Endian mode selectable • Interrupt requests to external devices are possible. • Internal interrupt requests to the on-chip CPU are possible. • Booting from the buffer RAM is possible if an external device stores the instruction codes in the buffer RAM in advance.
NAND flash memory controller (FLCTL) (booting available, pins multiplexed with the LBSC)	• Interface directly connectable to NAND-type flash memory • Read or write in sector units • Two transfer modes: command access mode and sector access mode • Interrupt requests, direct memory access controller transfer requests • Supports flash memory with address ranges of 2 Gbits and even more by extension to 5-byte addresses
A/D converter (ADC)	• Input: eight channels • Resolution: 10 bits • Activation of A/D conversion by an external trigger or a timer trigger is possible

Item	Description
Stream Interface (STIF) Note: If you intend to use this module, select an applicable product from among those listed in section 1.5, Product Type Numbers.	• Two interfaces. Operation is linked with a dedicated DMAC, the A-DMAC • Either serial or parallel mode is selectable for each interface. • MPEG2-TS transfer mode and MPEG-PS transfer mode • Each device supports both push transfer and pull transfer • A PWM timer and related output pins for each channel can be used to control an external VCO • The interfaces can output a common stream clock. Each interface has its own input for a stream clock.
Power-down modes Operating modes	• Sleep mode • Software standby mode • Module standby mode • Deep standby mode HIFRAM (with a single 2-KB bank) for storage when power supply to the CPU core is shut off
Booting	• Supports various boot modes — Booting from memory connected to the CS0 area — Booting from NAND-type flash memory — Booting from serial flash memory — MMC 4.4 boot (only for booting; interface module otherwise supports MMC 4.3) — eSD boot • Applicable devices Devices to which the eSD (Embedded SD) Addendum Version 2.10 applies — HIF boot
General-purpose I/O (GPIO)	• General-purpose I/O ports: 171 • Supports GPIO interrupts.

Others

JTAG	• A boundary scan function is not added to the DDR and USB pins.
Package	BGA 440-pin (0.8-mm pitch, 21 mm × 21 mm)

1.5 Product Type Numbers

Type number for the SH7734 series: R8A77343xAxxBx

Axx differs according to the available functions of the products.

Enabling the functions in the table requires settings by software. Since the way of doing this is not for disclosure, we'll provide it when exchanging delivery specifications.

STIF	RCAN	SDHI	R-GPVG	Code
0	0	0	0	A00
0	0	0	1	A01
0	0	1	0	A02
0	0	1	1	A03
0	1	0	0	A04
0	1	0	1	A05
0	1	1	0	A06
0	1	1	1	A07
1	0	0	0	A08
1	0	0	1	A09
1	0	1	0	A0A
1	0	1	1	A0B
1	1	0	0	A0C
1	1	0	1	A0D
1	1	1	0	A0E
1	1	1	1	A0F

1.6 Power Supply Voltages and Temperature Range

Power Supply Voltage: 3.3 V ± 0.3 V (3.3 V IO)

1.15 V to 1.30 V (Core)

1.8 V ± 0.1 V (DDR2-SDRAM IO)

1.5 V ± 0.075 V (DDR3-SDRAM IO)

Temperature range: -40 °C to 85 °C (Wide-temperature- range products)

1.7 Area Map

The physical address space of this LSI is shown in figure 1.2, and figures 1.3 to 1.8 are LSI internal area maps. Area 6 is a bus bridge and is further divided into six, allocated as expansion areas. Area 7 is allocated as a space for the registers in this bus bridge and in the internal modules.

Switching from the 29-bit address mode to the 32-bit address mode (extended mode) of the SH-4A is accomplished using the SE bit of the physical address space control register (PASCR). As the power-on reset startup mode, either the 29-bit mode or the 32-bit mode can be selected. However, switching from the 32-bit address mode to the 29-bit address mode is not possible.

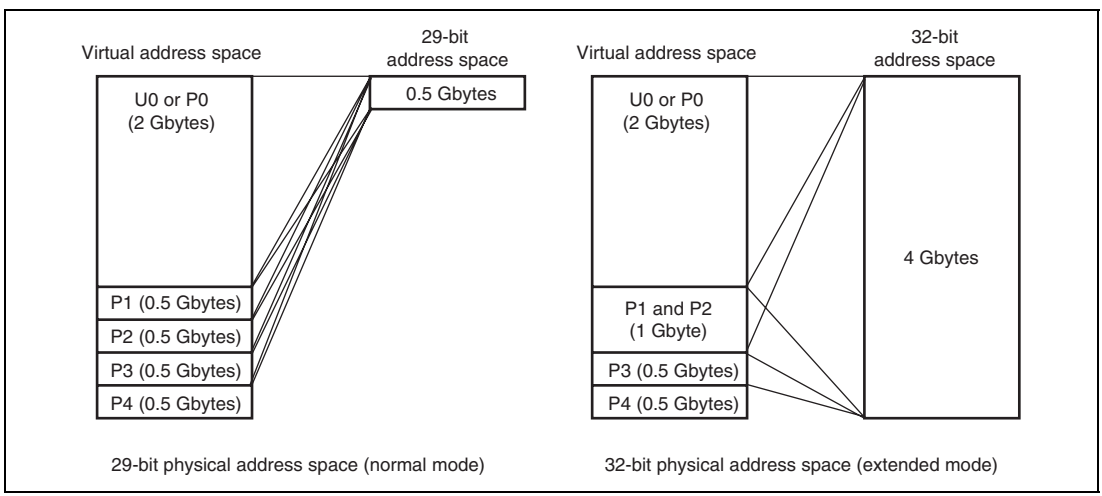


Figure 1.2 Physical Address Space

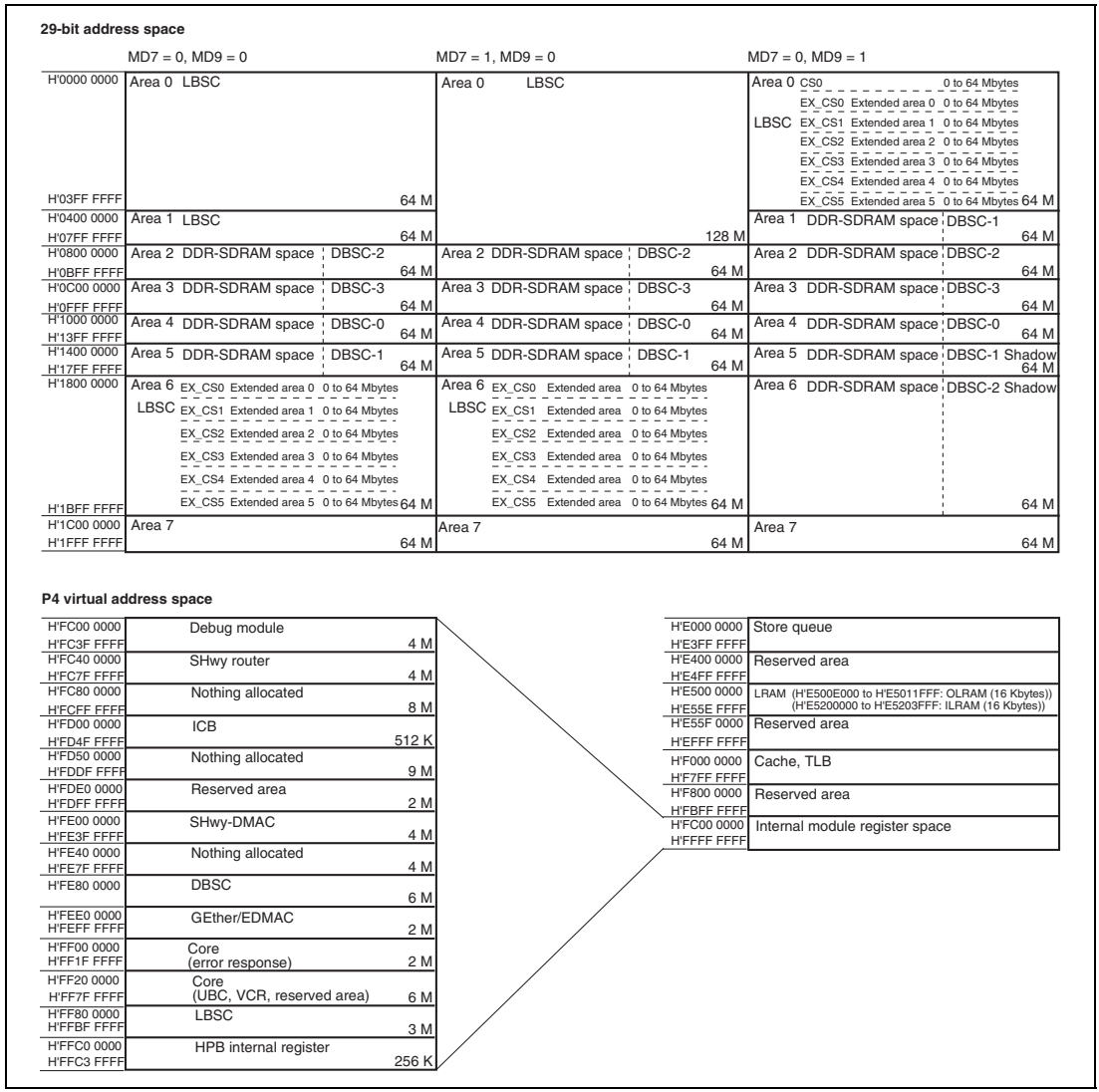


Figure 1.3 Memory Map (in 29-bit Address Mode)

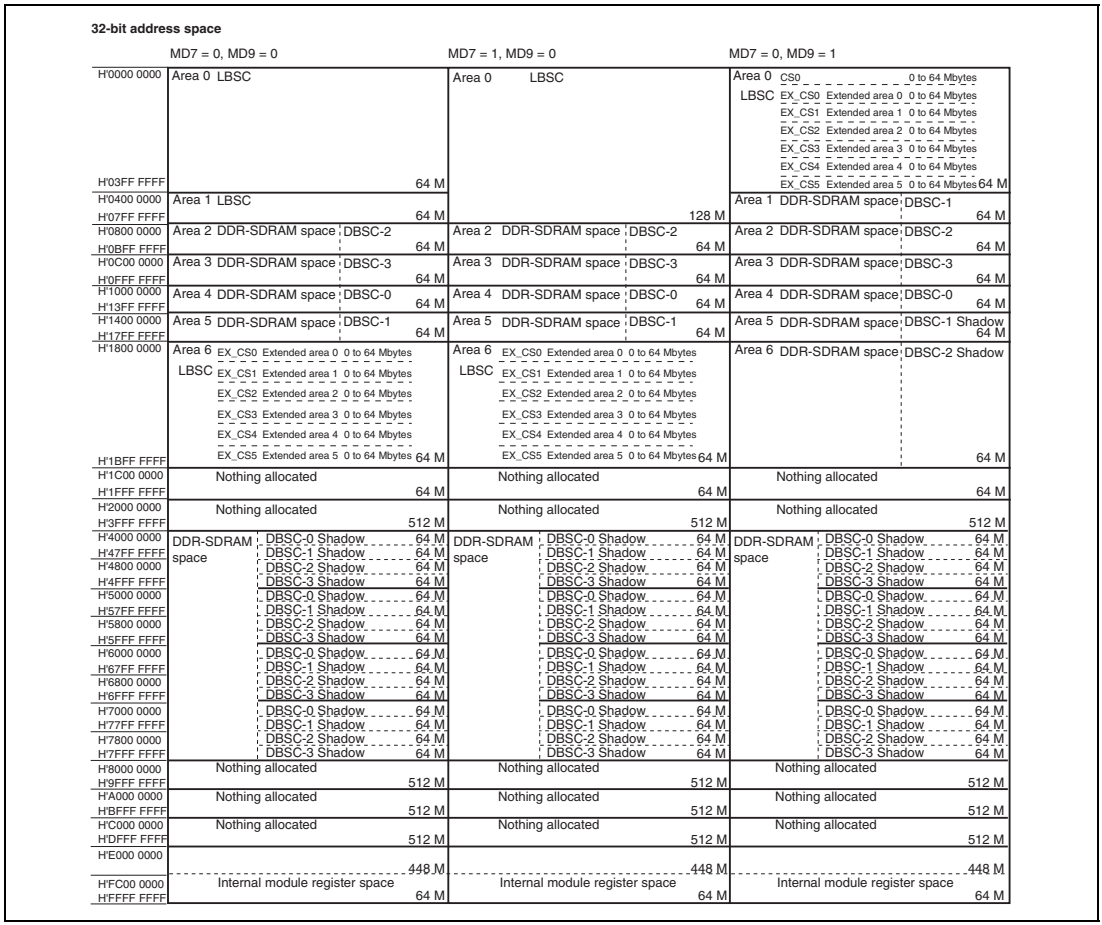


Figure 1.4 Memory Map (1) (in 32-bit Address Mode)

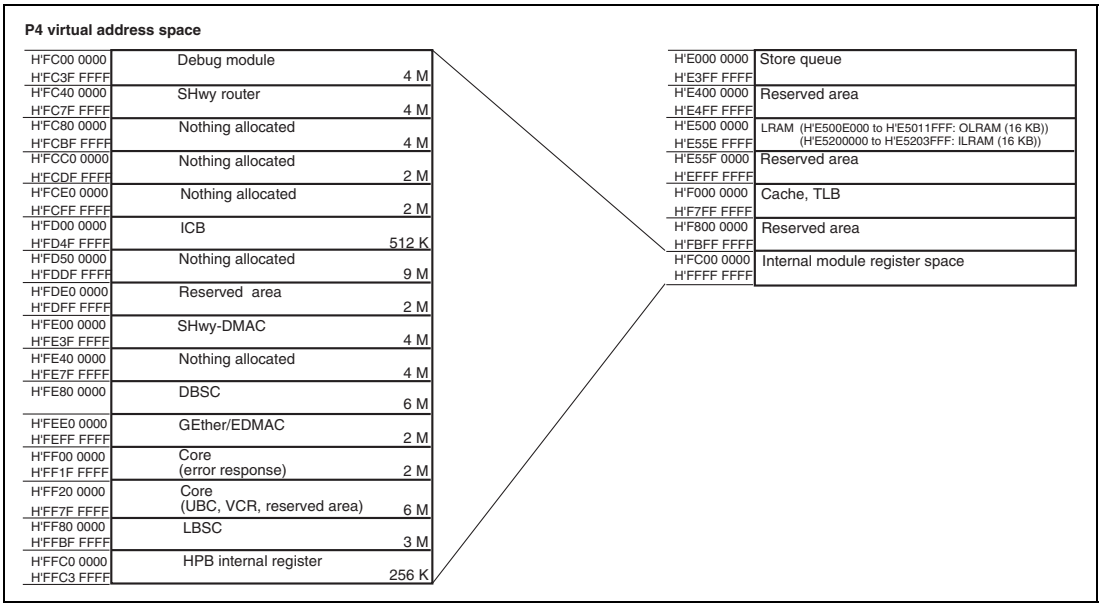


Figure 1.5 Memory Map (2) (in 32-bit Address Mode)

H'FD00 0000 H'FD11 FFFF	Nothing allocated	1152K
H'FD12 0000 H'FD12 3FFF	VEU3F	16K
H'FD12 4000 H'FD14 FFFF	Nothing allocated	176K
H'FD15 0000 H'FD15 FFFF	LMB	64K
H'FD16 0000 H'FD3F FFFF	Nothing allocated	2688K
H'FD40 0000 H'FD47 FFFF	ICB control	512K
H'FD48 0000 H'FD4F FFFF	Nothing allocated	512K

Figure 1.6 Memory Map for ICB Register Space

H'FF80 0000 H'FF80 01FF	VCR/system-related register	512
H'FF80 0200 H'FF80 04FF	BSC internal register	768
H'FF80 0500 H'FF80 0FFF	Nothing allocated	2816
H'FF80 1000 H'FF80 1FFF	DMAC	4 K
H'FF80 2000 H'FF80 3FFF	INTC	8 K
H'FF80 4000 H'FF80 5FFF	INTC2	8 K
H'FF80 6000 H'FF81 FFFF	Nothing allocated	104 K
H'FF82 0000 H'FF82 FFFF	HIF-RAM	64 K
H'FF83 0000 H'FF83 FFFF	HIF (REG)	64 K
H'FF84 0000 H'FF87 FFFF	Reserved area	192 K
H'FF88 0000 H'FF88 1FFF	ADMAC	8 K
H'FF88 2000 H'FF8F FFFF	Nothing allocated	504 K
H'FF90 0000 H'FFBF FFFF	LBSC reserved area	3 M

Figure 1.7 Memory Map for LBSC Register Space

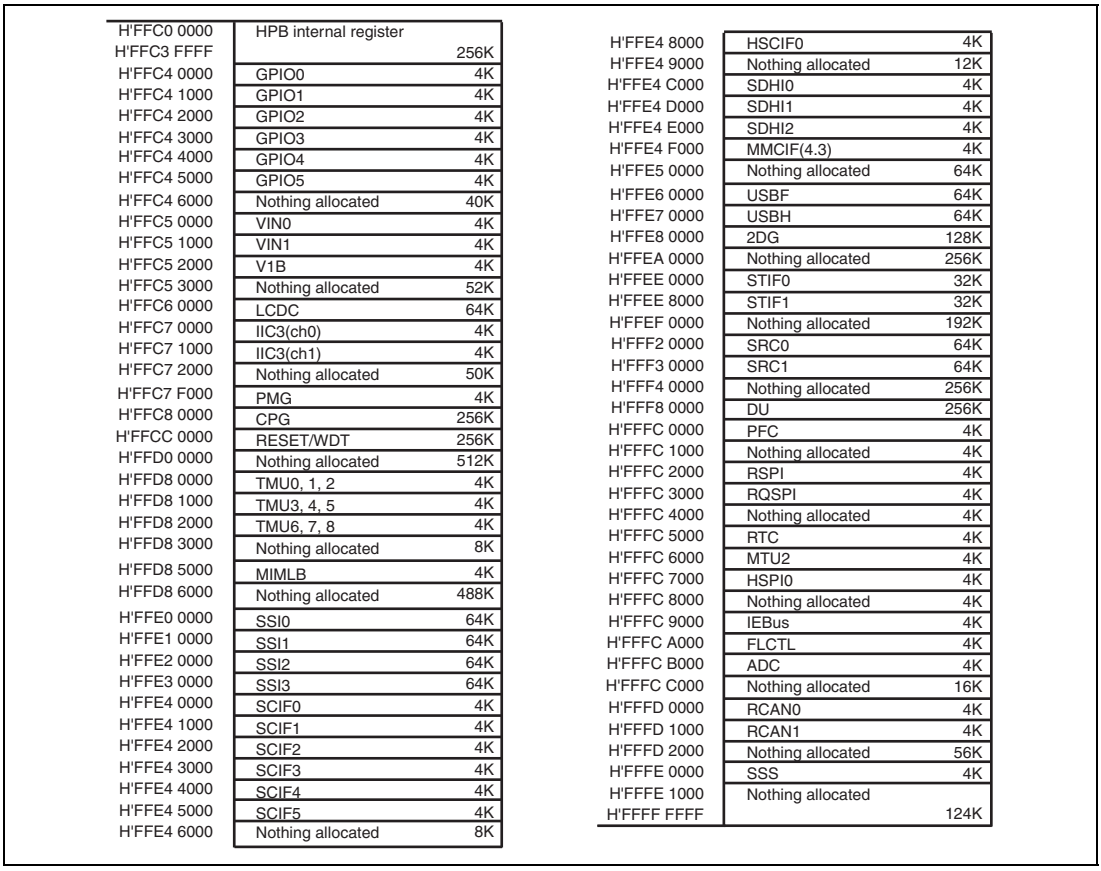


Figure 1.8 Memory Map for HPB Register Space

1.7.1 Memory Access and Register Access

Accessing Unified Memory (DDR-SDRAM)

The UM is a common area for the CPU, display unit, graphic unit, and peripheral modules. Data storage in the UM is performed such that the data arrangement agrees with the CPU endian mode.

If the CPU is in big-endian mode, byte, word, and longword data are positioned such that addresses increase in moving from the upper to the lower side of the UM bus.

If the CPU is in little-endian mode, byte, word, and longword data are positioned such that addresses increase in moving from the lower to the upper side of the UM bus.

Operation of the display and graphic units is little endian. If the data for these units are big endian, accomplish conversion by using features for the support of endian conversion under software control.

Linear tiling conversion is not performed.

1.7.2 Accessing Registers

Access space for all registers is allocated in the space of area 7. In this space there are registers for image transfer; data written to these registers undergoes color space conversion and other processing, and is finally stored in the UM. With respect to arrangement in the UM, there is a support function for endian conversion. When endian conversion is necessary, this function may be used to perform conversion under software control.

1.8 Direct Memory Access Controllers

1.8.1 Basic Specifications

This LSI incorporates a three-channel DMAC with an external request mode (LBSC-DMAC), a 28-channel dedicated DMAC for modules connected to the HPB (HPB-DMAC), and a two-channel dedicated DMAC for handling transfer between the SH-4A core and memory (SHwy-DMAC).

By using DMACs, the load on the CPU can be reduced.

(1) LBSC-DMAC and HPB-DMAC

See section 6A, Direct Memory Access Controller for Local Bus and Peripheral Modules (LBSC-DMAC/HPB-DMAC).

(2) SHwy-DMAC

See section 5, Direct Memory Access Controller (SHwy-DMAC).

1.9 Interrupt Controllers

- Interrupt controllers using an SH4-compatible direct jump method
- Five independent external interrupts: NMI, IRQ3 to IRQ0, and GIPO interrupts accepted from outside
- 15-level encoded external interrupts possible: IRL3 to IRL0
- Internal peripheral interrupts: priority levels set per module
- Detailed interrupt sources for internal peripheral function block indicated by the INTC2 internal register
- INTEVT codes have been gathered in a set of detailed source indication registers.

For details, see section 7, INTC/INTC2.

1.10 Data Format

The CPU core of this LSI, the SH-4A supports both big and little endian data formats. Switching between formats is performed using an external pin (MD8) when power-on reset is executed with the $\overline{\text{PRESET}}$ pin.

Table 1.2 shows the method for data alignment conversion for each module. For details, refer to the sections concerning the modules.

Table 1.2 Data Alignment Conversion for Each Module

Module/Internal Endian		Endian Change Method	Remarks
SH-4A	—	MD8 pin linked	
SHwy-DMAC	—	MD8 pin linked	
DBSC3	—	MD8 pin linked	
LBSC	—	MD8 pin linked	
LBSC-DMAC	Big	• SHwy-connected device MD8 pin linked /register setting • LBSC-connected device Register setting	• SHwy-connected device Not linked to MD8, can be set through a register Initial setting: Data alignment conversion mode according to the endian mode • LBSC-connected device Initial setting: Data not converted
HPB	Big	MD8 pin linked	
HPB-DMAC	Big	• SHwy-connected device MD8 pin linked /register setting	• SHwy-connected device Not linked to MD8, can be set through a register Initial setting: Data alignment conversion mode according to the endian mode
VIN0	Little	Register setting	
VIN1	Little	Register setting	
DU	Little	Register setting	
R-GPVG	Little	Register setting	
2DG	Little	Register setting	

1.11 List of Registers

The registers are described in the relevant module specifications.

The register which is not described in any module specifications is described below.

1.11.1 Processor Version Register (PVR) (Address: H'FF00 0030)

PVR provides version information to outside the CPU core.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CHIP								VER							
Initial value:	0	0	0	1	0	0	0	0	0	0	1	1	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CUT								—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	CHIP	H'10	R	The value for this LSI is H'10.
23 to 16	VER	H'30	R	The value for this LSI is H'30.
15 to 8	CUT	H'0B	R	The value for this LSI is H'0B.
7 to 0	—	All 0	R	Reserved

1.12 Initial Values of Registers and Pin State

1.12.1 Initial Values of Registers

See register descriptions in each section.

1.12.2 Pin State

See section 1.16, Pin Function Table for pin states.

1.13 Processing State

There are four processing states: reset state, exception handling state, program execution state, and power-down state.

1.13.1 Reset State

The CPU is reset by a power-on reset or a manual reset. Sources of power-on reset and manual reset are listed below.

- Sources of power-on reset
  - (a) When a low level signal is input from the $\overline{\text{PRESET}}$ pin
  - (b) When WDTCNT overflows while the WT/IT bit in WDTCSR is 1 and the RSTS bit in WDTCSR is 0
  - (c) When an H-UDI reset occurs
- Sources of manual reset
  - (d) When a general exception except for user break occurs while the BL bit in SR is 1
  - (e) When WDTCNT overflows while the WT/IT bit and the RSTS bit in WDTCSR are both 1

Reset operation is described below.

(1) Power-On Reset Operation

Power-on resetting initializes all modules in this LSI. The registers for the CPG module are initialized by power-on reset using both a $\overline{\text{PRESET}}$ pin and WDT (see items (a) and (b) above). The registers for the RESET/WDT module are initialized by power-on reset using only a $\overline{\text{PRESET}}$ pin (see item (a) above). The modules might include some registers that are not initialized by a reset. For details, see the register descriptions in the relevant sections.

(2) Manual Resetting Operation

Manual resetting basically initializes all components while retaining the clock, reset, and pin-multiplexing settings. The manual resetting does not affect any SDRAM refreshing operation. For details, see the register descriptions in the relevant sections.

(3) Operation during Display Resetting (the DRES bit of the display unit system control register (DSYSR) of the display unit (DU))

Display control stops. In combination with the DEN bit of the display unit system control register (DSYSR), this resetting initiates and stops display operation without initializing any register value.

(4) Operation during HSPI Module Resetting (when one of the bits, FBS, CLKP, IDIV, or CLKC of the control register (SPCR), or one of the bits, FFEN, LMSB, CSA, or MASL of the system control register (SPSCR) for the HSPI module is changed)

By means of a software reset, the HSPI module is reset to a pre-defined state and the FIFO pointers for transmission and reception can be initialized. A software reset is generated when the settings of SPCR and control bits, with the exception of the interrupt/DMA enable bits and the chip select value (CSV) bit in SPSCR, are changed.

When $\overline{\text{HSPI\_CS}}$ is to be forced to the low level except when the HSPI is in slave mode and the master device is transmitting data, CS must be reset after the software reset, in order to prevent error receptions of data.

(5) Operation during LBSC-DMAC/HPB-DMAC Resetting (the SRST bit of the LBSC-DMAC/HPB-DMAC software-reset registers (LSRSTR0 to LSRSTR2, HSRSTR0 to HSRSTR28))

Irrespective of the DMA transfer status, resets the DMAC module by writing 1 to the SRST bit. The scope of the resetting is the same as in power-on resetting or manual resetting.

Accordingly, it is presumed that a software reset will only be issued during system debugging and so on when DMA transfer is not in progress.

If DMA operation is on, specify either forced halting or temporary halting instead of software resetting.

In the registers used by all channels in common (DMA transfer end interrupt status register (DINTSR) and DMA transfer end interrupt enable register (DINTMR)), only the bits corresponding to the target channel of the software reset are initialized.

1.13.2 Exception Handling State

Exception handling refers to the execution of required action in a program different from normal action, due to exception sources such as resetting, a general exception, or an interrupt.

This LSI provides three types of exception handling: resettings, general exceptions, and interrupts.

In the case of resetting, the system branches to H'A000 0000 and starts the execution of a user-provided exception handling program.

In the case of a general exception or an interrupt, the system saves the program counter (PC) to the saved program counter (SPC), the status register (SR) to the saved status register (SSR), and register R15 to the saved general register 15 (SGR).

The system branches to the start address of a user-provided exception handling routine, determined as the sum of a vector base address and vector offset, and starts executing the program. For details on resetting, a general exception, and interrupts, see section 2.5, Exception Handling.

1.13.3 Program Execution State

In this state, the CPU sequentially executes programs.

1.13.4 Power-Down States

This LSI has the following power-down states: sleep mode, module standby mode (which stops the clocks for major modules), software standby mode, deep standby mode, and SDRAM power supply backup mode.

For details on the sleep mode, module standby mode, software standby mode, and deep standby mode, see section 9, Operating Modes and Power-Down Modes.

For details on the SDRAM power supply backup mode, see section 4, Memory Controller (DBSC3).

1.14 Pin Settings

Input fixed values for the TEST1, TEST2, MPMD and BSMODE pins. These values cannot be changed after power is supplied. The values of pins MD0 to MD15 are input upon power-on reset using the $\overline{\text{PRESET}}$ pin. Power-on reset results in switching to a different function.

TEST2	TEST1	Test Mode Switching
0	0	Normal operation
0	1	Setting prohibited
1	0	Setting prohibited
1	1	Setting prohibited

MPMD	BSMODE	JTAG Pin Operating Mode Switching
0	0	Operates in emulation support mode
0	1	Setting prohibited
1	0	Normal operation
1	1	Operates in boundary scan mode When $\overline{\text{PRESET}}$ is at low level, pin IO control is disabled.

MD0	Free-Running Mode or Step-Up Mode
0	Free-running mode
1	Step-up mode

MD2	MD1	Clock Mode
0	0	400-MHz mode (2) (DDR2-400)
0	1	533-MHz mode (DDR2-533)
1	0	400-MHz mode (1) (DDR2/3-600)
1	1	Setting prohibited

MD4	MD3	External Bus Clock (CLKOUT) Frequency Setting
0	0	Normal operation
0	1	Setting prohibited
1	0	Setting prohibited
1	1	Setting prohibited

MD6	MD5	Data Bus Width of EX-BUS in Area 0
0	0	Setting prohibited
0	1	8-bit bus
1	0	16-bit bus
1	1	Setting prohibited

MD9	MD7	Area Division
0	0	Area 0: 64 Mbytes, areas 2 to 5: DDR mode
0	1	Area 0: 128 Mbytes, areas 2 to 5: DDR mode
1	0	Area 0: 64 Mbytes, areas 1 to 6: DDR mode
1	1	Setting prohibited

MD8	Big/Little Endian
0	Big endian
1	Little endian

MD10	EXTAL/XTAL Pin Setting
0	Inputs an external clock to the EXTAL pin
1	Connects a crystal resonator to the EXTAL/XTAL pin

MD12	MD11	PLL Multiplication Ratio
0	0	× 12
0	1	× 16
1	0	× 24
1	1	× 32

MD13	29-/32-Bit Address Mode
0	29-bit mode
1	32-bit mode

MD15	PLL Feedback Route Selection
0	Normal operation
1	Setting prohibited

MD19	MD14	MD18	MD17	MD16	Boot Mode
0	0	0	0	0	CS0 boot (NOR flash memory, etc.)
0	0	0	0	1	Setting prohibited
0	0	0	1	0	Booting up from NAND flash memory (SLC)
0	0	0	1	1	Serial boot
0	0	1	0	0	Setting prohibited
0	0	1	0	1	Setting prohibited
0	0	1	1	0	Setting prohibited
0	0	1	1	1	Setting prohibited
0	1	0	0	0	MMC boot
0	1	0	0	1	Setting prohibited
0	1	0	1	0	eSD boot
0	1	0	1	1	Setting prohibited
0	1	1	0	0	Setting prohibited
0	1	1	0	1	Setting prohibited
0	1	1	1	0	Setting prohibited
0	1	1	1	1	Setting prohibited
1	X	X	X	X	HIF boot

1.15 Pin Assignments (Top View)

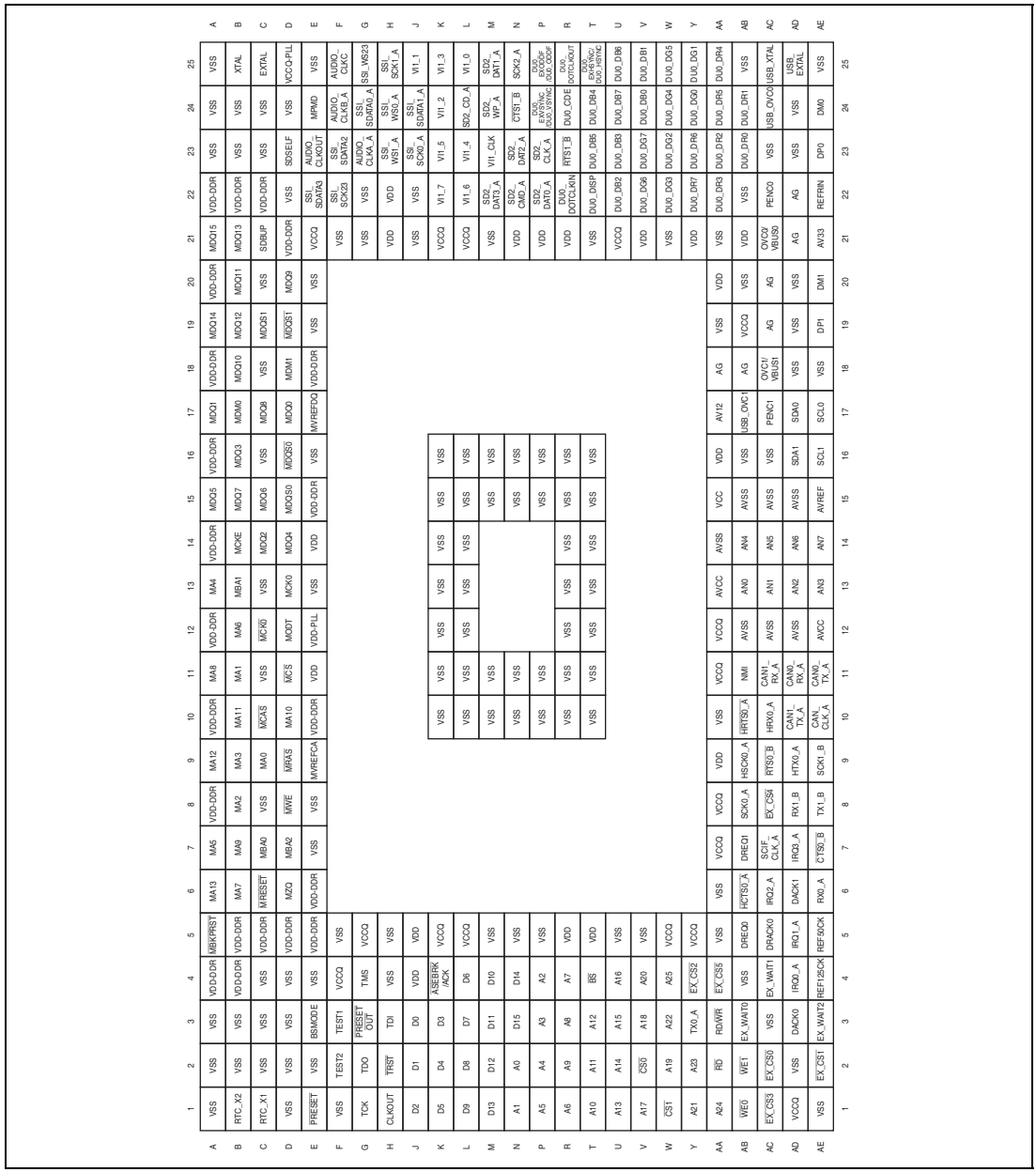


Figure 1.9 Pin Assignments (BGA440)

1.16 Pin Function Table

1.16.1 Pin Characteristics for the BGA440

Table 1.3 Pin Characteristics (for the BGA440)

Signal Name	√: GPIO Included GPI: GPI Included ×: No GPIO	(GPIO/Peripheral Function Select Bit)	(Driving Ability of I/O Pins Powered by VCCQ)	Grouping of Driving Ability	Pull-up Control (Pull-up*: Pulling up is enabled from the reset state. *: Pulling up is disabled in the reset state. —: No pull-ups)	Mode Signal
MCK0	×		—	—	—	—
$\overline{\text{MCK0}}$	×		—	—	—	—
MCKE	×		—	—	—	—
$\overline{\text{MCS}}$	×		—	—	—	—
$\overline{\text{MWE}}$	×		—	—	—	—
$\overline{\text{MRAS}}$	×		—	—	—	—
$\overline{\text{MCAS}}$	×		—	—	—	—
MA0	×		—	—	—	—
MA1	×		—	—	—	—
MA2	×		—	—	—	—
MA3	×		—	—	—	—
MA4	×		—	—	—	—
MA5	×		—	—	—	—
MA6	×		—	—	—	—
MA7	×		—	—	—	—
MA8	×		—	—	—	—
MA9	×		—	—	—	—
MA10	×		—	—	—	—
MA11	×		—	—	—	—
MA12	×		—	—	—	—
MA13	×		—	—	—	—
MBA0	×		—	—	—	—
MBA1	×		—	—	—	—
MBA2	×		—	—	—	—

Signal Name	√: GPIO Included GPI: GPI Included ×: No GPIO	(GPIO/Peripheral Function Select Bit)	(Driving Ability of I/O Pins Powered by VCCQ)	Grouping of Driving Ability	Pull-up Control (Pull-up*: Pulling up is enabled from the reset state. *: Pulling up is disabled in the reset state. —: No pull-ups)	Mode Signal
MDQ0	×		—	—	—	—
MDQ1	×		—	—	—	—
MDQ2	×		—	—	—	—
MDQ3	×		—	—	—	—
MDQ4	×		—	—	—	—
MDQ5	×		—	—	—	—
MDQ6	×		—	—	—	—
MDQ7	×		—	—	—	—
MDQ8	×		—	—	—	—
MDQ9	×		—	—	—	—
MDQ10	×		—	—	—	—
MDQ11	×		—	—	—	—
MDQ12	×		—	—	—	—
MDQ13	×		—	—	—	—
MDQ14	×		—	—	—	—
MDQ15	×		—	—	—	—
MDQS0	×		—	—	—	—
$\overline{\text{MDQS0}}$	×		—	—	—	—
MDQS1	×		—	—	—	—
$\overline{\text{MDQS1}}$	×		—	—	—	—
MDM0	×		—	—	—	—
MDM1	×		—	—	—	—
MODT	×		—	—	—	—
$\overline{\text{MBKPRST}}$	×		—	—	—	—
MZQ	×		—	—	—	—
MVREFDQ	×		—	—	—	—
MVREFCA	×		—	—	—	—
$\overline{\text{MRESET}}$	×		—	—	—	—
SDBUP	×		—	—	—	—

Signal Name	√: GPIO Included GPI: GPI Included ×: No GPIO	(GPIO/Peripheral Function Select Bit)	(Driving Ability of I/O Pins Powered by VCCQ)	Grouping of Driving Ability	Pull-up Control (Pull-up*: Pulling up is enabled from the reset state. *: Pulling up is disabled in the reset state. —: No pull-ups)	Mode Signal
SDSELF	√		6 mA	—	pull-up*	—
TEST1	×		—	—	—	—
TEST2	×		—	—	—	—
BSMODE	×		—	—	—	—
EXTAL	×		—	—	—	—
XTAL	×		—	—	—	—
$\overline{\text{PRESET}}$	×		—	—	—	—
$\overline{\text{PRESETOUT}}$	√	GP0[21]	6/8 mA	Independent of the TSIF	*	—
A0	√	GP0[22]	6 mA	—	*	—
A1	√	GP0[23]	6 mA	—	*	—
A2	√	GP0[24]	6 mA	—	*	—
A3	√	GP0[4]	6 mA	—	*	—
A4	√	GP0[5]	6 mA	—	*	—
A5	√	GP0[6]	6 mA	—	*	MD18
A6	√	GP0[7]	6 mA	—	*	MD19
A7	√	GP0[8]	6 mA	—	*	MD0
A8	√	GP0[9]	6 mA	—	*	MD1
A9	√	GP0[10]	6 mA	—	*	MD2
A10	√	GP0[11]	6 mA	—	*	MD3
A11	√	GP0[12]	6 mA	—	*	MD4
A12	√	GP0[13]	6 mA	—	*	MD5
A13	√	GP0[14]	6 mA	—	*	MD6
A14	√	GP0[15]	6 mA	—	*	MD7
A15	√	GP0[16]	6 mA	—	*	MD8
A16	√	GP0[17]	6 mA	—	*	MD9
A17	√	GP0[18]	6 mA	—	*	MD11
A18	√	GP0[19]	6 mA	—	*	MD12

Signal Name	√: GPIO Included GPI: GPI Included ×: No GPIO	(GPIO/Peripheral Function Select Bit)	(Driving Ability of I/O Pins Powered by VCCQ)	Grouping of Driving Ability	Pull-up Control (Pull-up*: Pulling up is enabled from the reset state. *: Pulling up is disabled in the reset state. —: No pull-ups)	Mode Signal
A19	√	GP0[20]	6 mA	—	*	MD13
A20	√	GP0[0]	6 mA	—	*	—
A21	√	GP0[1]	6 mA	—	*	—
A22	√	GP0[2]	6 mA	—	*	—
A23	√	GP0[3]	6 mA	—	*	—
A24	√	GP0[25]	6 mA	—	*	MD14
A25	√	GP0[26]	6 mA	—	*	MD15
D0	√	GP0[27]	6 mA	—	*	—
D1	√	GP0[28]	6 mA	—	*	—
D2	√	GP0[29]	6 mA	—	*	—
D3	√	GP0[30]	6 mA	—	*	—
D4	√	GP0[31]	6 mA	—	*	—
D5	√	GP1[25]	6 mA	—	*	—
D6	√	GP1[28]	6 mA	—	*	—
D7	√	GP1[29]	6 mA	—	*	—
D8	√	GP1[30]	6/8 mA	Gr#B	*	—
D9	√	GP1[4]	6/8 mA	Gr#B	*	—
D10	√	GP1[5]	6/8 mA	Gr#B	*	—
D11	√	GP1[6]	6/8 mA	Gr#B	*	—
D12	√	GP1[7]	6/8 mA	Gr#B	*	—
D13	√	GP1[8]	6/8 mA	Gr#B	*	—
D14	√	GP1[9]	6 mA	—	*	—
D15	√	GP1[10]	6 mA	—	*	—
CLKOUT	√	GP1[11]	6 mA	—	*	—
$\overline{BS}$	√	GP1[12]	6 mA	—	*	—
$\overline{CS0}$	√	GP1[13]	6 mA	—	*	—
$\overline{CS1}$/A26	√	GP1[14]	6 mA	—	*	—
$\overline{EX\_CS0}$	√	GP1[15]	6 mA	—	pull-up*	—

Signal Name	√: GPIO Included GPI: GPI Included ×: No GPIO	(GPIO/Peripheral Function Select Bit)	(Driving Ability of I/O Pins Powered by VCCQ)	Grouping of Driving Ability	Pull-up Control (Pull-up*: Pulling up is enabled from the reset state. *: Pulling up is disabled in the reset state. —: No pull-ups)	Mode Signal
$\overline{\text{EX\_CS1}}$	√	GP1[16]	6/8 mA	Gr#COM	pull-up*	—
$\overline{\text{EX\_CS2}}$	√	GP1[17]	6/8 mA	Gr#A	pull-up*	—
$\overline{\text{EX\_CS3}}$	√	GP1[18]	6/8 mA	Gr#A	*	MD16
$\overline{\text{EX\_CS4}}$	√	GP1[19]	6/8 mA	Gr#A	*	MD17
$\overline{\text{EX\_CS5}}$	√	GP1[20]	6/8 mA	Gr#A	pull-up*	—
$\overline{\text{RD}}$	√	GP1[21]	6 mA	—	*	—
RD/$\overline{\text{WR}}$	√	GP1[22]	6/8 mA	Gr#COM	pull-up*	—
$\overline{\text{WE0}}$	√	GP1[23]	6 mA	—	*	—
$\overline{\text{WE1}}$	√	GP1[24]	6 mA	—	*	—
EX_WAIT0	√	GP1[0]	6 mA	—	*	—
EX_WAIT1	√	GP1[26]	6/8 mA	Gr#A	pull-up*	—
EX_WAIT2	√	GP1[27]	6/8 mA	Gr#A	pull-up*	—
DRACK0	√	GP1[1]	6/8 mA	Gr#COM	pull-up*	—
DREQ0	√	GP1[2]	6/8 mA	Gr#COM	pull-up*	—
DACK0	√	GP1[3]	6/8 mA	Gr#COM	*	—
DREQ1	√	GP1[31]	6 mA	—	pull-up*	—
DACK1	√	GP2[5]	6 mA	—	pull-up*	—
$\overline{\text{TRST}}$	×		—	—	pull-up*	—
TCK	×		—	—	pull-up*	—
TMS	×		—	—	pull-up*	—
TDI	×		—	—	pull-up*	—
TDO	×		6 mA	—	*	—
MPMD	×		—	—	—	—
$\overline{\text{ASEBRK}}$/ACK	×		6 mA	—	pull-up*	—
NMI	×		—	—	—	—
IRQ0_A	√	GP2[6]	6 mA	—	pull-up*	—
IRQ1_A	√	GP2[7]	6 mA	—	pull-up*	—
IRQ2_A	√	GP2[9]	6 mA	—	pull-up*	—
IRQ3_A	√	GP2[4]	6 mA	—	pull-up*	—

Signal Name	√: GPIO Included GPI: GPI Included ×: No GPIO	(GPIO/Peripheral Function Select Bit)	(Driving Ability of I/O Pins Powered by VCCQ)	Grouping of Driving Ability	Pull-up Control (Pull-up*: Pulling up is enabled from the reset state. *: Pulling up is disabled in the reset state. —: No pull-ups)	Mode Signal
SCIF_CLK_A	√	GP2[0]	6 mA	—	pull-up*	—
SCK0_A	√	GP2[1]	6 mA	—	pull-up*	—
RX0_A	√	GP2[2]	6 mA	—	pull-up*	—
TX0_A	√	GP2[8]	6 mA	—	*	MD10
$\overline{\text{HCTS0\_A}}$	√	GP2[3]	6 mA	—	pull-up*	—
$\overline{\text{HRTS0\_A}}$	√	GP2[10]	6 mA	—	pull-up*	—
HSCK0_A	√	GP2[11]	6 mA	—	pull-up*	—
HRX0_A	√	GP2[12]	6 mA	—	pull-up*	—
HTX0_A	√	GP2[13]	6 mA	—	pull-up*	—
$\overline{\text{CTS0\_B}}$	√	GP2[14]	6 mA	—	pull-up*	—
$\overline{\text{RTS0\_B}}$	√	GP2[15]	6 mA	—	pull-up*	—
SCK1_B	√	GP2[16]	6 mA	—	pull-up*	—
RX1_B	√	GP2[17]	6 mA	—	pull-up*	—
TX1_B	√	GP2[18]	6 mA	—	pull-up*	—
$\overline{\text{CTS1\_B}}$	√	GP2[19]	6 mA	—	pull-up*	—
$\overline{\text{RTS1\_B}}$	√	GP2[20]	6 mA	—	pull-up*	—
SCK2_A	√	GP2[21]	6 mA	—	pull-up*	—
SD2_CLK_A	√	GP2[22]	6 mA	—	pull-up*	—
SD2_CMD_A	√	GP2[23]	6 mA	—	pull-up*	—
SD2_DAT0_A	√	GP2[24]	6 mA	—	pull-up*	—
SD2_DAT1_A	√	GP2[25]	6 mA	—	pull-up*	—
SD2_DAT2_A	√	GP2[26]	6 mA	—	pull-up*	—
SD2_DAT3_A	√	GP2[27]	6 mA	—	pull-up*	—
SD2_CD_A	√	GP2[28]	6 mA	—	pull-up*	—
SD2_WP_A	√	GP2[29]	6 mA	—	pull-up*	—
REF125CK	√	GP2[30]	6 mA	—	*	—
REF50CK	√	GP2[31]	6 mA	—	*	—
DU0_DR0	√	GP3[0]	6 mA	—	*	—
DU0_DR1	√	GP3[1]	6 mA	—	*	—

Signal Name	√: GPIO Included GPI: GPI Included ×: No GPIO	(GPIO/Peripheral Function Select Bit)	(Driving Ability of I/O Pins Powered by VCCQ)	Grouping of Driving Ability	Pull-up Control (Pull-up*: Pulling up is enabled from the reset state. *: Pulling up is disabled in the reset state. —: No pull-ups)	Mode Signal
DU0_DR2	√	GP3[2]	6 mA	—	*	—
DU0_DR3	√	GP3[3]	6 mA	—	*	—
DU0_DR4	√	GP3[4]	6 mA	—	*	—
DU0_DR5	√	GP3[5]	6 mA	—	*	—
DU0_DR6	√	GP3[6]	6 mA	—	*	—
DU0_DR7	√	GP3[7]	6 mA	—	*	—
DU0_DG0	√	GP3[8]	6 mA	—	*	—
DU0_DG1	√	GP3[9]	6 mA	—	*	—
DU0_DG2	√	GP3[10]	6 mA	—	*	—
DU0_DG3	√	GP3[11]	6 mA	—	*	—
DU0_DG4	√	GP3[12]	6 mA	—	*	—
DU0_DG5	√	GP3[13]	6 mA	—	*	—
DU0_DG6	√	GP3[14]	6 mA	—	*	—
DU0_DG7	√	GP3[15]	6 mA	—	*	—
DU0_DB0	√	GP3[16]	6 mA	—	*	—
DU0_DB1	√	GP3[17]	6 mA	—	*	—
DU0_DB2	√	GP3[18]	6 mA	—	*	—
DU0_DB3	√	GP3[19]	6 mA	—	*	—
DU0_DB4	√	GP3[20]	6 mA	—	*	—
DU0_DB5	√	GP3[21]	6 mA	—	*	—
DU0_DB6	√	GP3[22]	6 mA	—	*	—
DU0_DB7	√	GP3[23]	6 mA	—	*	—
DU0_DOTCLKIN	√	GP3[24]	6 mA	—	pull-up*	—
DU0_ DOTCLKOUT	√	GP3[25]	6 mA	—	pull-up*	—
DU0_EXHSYNC/ DU0_HSYNC	√	GP3[26]	6 mA	—	pull-up*	—
DU0_EXVSYNC/ DU0_VSYNC	√	GP3[27]	6 mA	—	pull-up*	—

Signal Name	√: GPIO Included GPI: GPI Included ×: No GPIO	(GPIO/Peripheral Function Select Bit)	(Driving Ability of I/O Pins Powered by VCCQ)	Grouping of Driving Ability	Pull-up Control (Pull-up*: Pulling up is enabled from the reset state. *: Pulling up is disabled in the reset state. —: No pull-ups)	Mode Signal
DU0_EXODDF/ DU0_ODDF	√	GP3[28]	6 mA	—	pull-up*	—
DU0_DISP	√	GP3[29]	6 mA	—	pull-up*	—
DU0_CDE	√	GP3[30]	6 mA	—	pull-up*	—
VI1_CLK_A	√	GP3[31]	6 mA	—	*	—
VI1_0_A	√	GP4[8]	6 mA	—	pull-up*	—
VI1_1_A	√	GP4[9]	6 mA	—	pull-up*	—
VI1_2_A	√	GP4[10]	6 mA	—	pull-up*	—
VI1_3_A	√	GP4[11]	6 mA	—	pull-up*	—
VI1_4_A	√	GP4[4]	6 mA	—	pull-up*	—
VI1_5_A	√	GP4[5]	6 mA	—	pull-up*	—
VI1_6_A	√	GP4[6]	6 mA	—	pull-up*	—
VI1_7_A	√	GP4[7]	6 mA	—	pull-up*	—
SSI_SCK0_A	√	GP4[0]	6 mA	—	pull-up*	—
SSI_WS0_A	√	GP4[1]	6 mA	—	pull-up*	—
SSI_SDATA0_A	√	GP4[2]	6 mA	—	pull-up*	—
SSI_SCK1_A	√	GP4[3]	6 mA	—	pull-up*	—
SSI_WS1_A	√	GP4[12]	6 mA	—	pull-up*	—
SSI_SDATA1_A	√	GP4[13]	6 mA	—	pull—up*	—
SSI_SCK23	√	GP4[14]	6 mA	—	pull—up*	—
SSI_WS23	√	GP4[15]	6 mA	—	pull-up*	—
SSI_SDATA2	√	GP4[16]	6 mA	—	pull-up*	—
SSI_SDATA3	√	GP4[17]	6 mA	—	pull-up*	—
AUDIO_CLKA_A	√	GP4[18]	6 mA	—	pull-up*	—
AUDIO_CLKB_A	√	GP4[19]	6 mA	—	*	—
AUDIO_CLKC	√	GP4[20]	6 mA	—	*	—
AUDIO_CLKOUT	√	GP4[21]	6 mA	—	pull-up*	—
SCL0	GPI	GP4[24]	—	—	—	—
SDA0	GPI	GP4[25]	—	—	—	—

Signal Name	√: GPIO Included GPI: GPI Included ×: No GPIO	(GPIO/Peripheral Function Select Bit)	(Driving Ability of I/O Pins Powered by VCCQ)	Grouping of Driving Ability	Pull-up Control (Pull-up*: Pulling up is enabled from the reset state. *: Pulling up is disabled in the reset state. —: No pull-ups)	Mode Signal
SCL1	GPI	GP4[22]	—	—	—	—
SDA1	GPI	GP4[23]	—	—	—	—
USB_EXTAL	×		—	—	—	—
USB_XTAL	×		—	—	—	—
PENC0	√	GP4[26]	6 mA	—	*	—
PENC1	√	GP4[27]	6 mA	—	pull-up*	—
USB_OVC0	√	GP4[28]	6 mA	—	*	—
USB_OVC1	√	GP4[29]	6 mA	—	pull-up*	—
DP0	×		—	—	—	—
DM0	×		—	—	—	—
DP1	×		—	—	—	—
DM1	×		—	—	—	—
REFRIN	×		—	—	—	—
OVC0/VBUS0	×		—	—	—	—
OVC1/VBUS1	×		—	—	—	—
AV33	×		—	—	—	—
AV12	×		—	—	—	—
AG	×		—	—	—	—
VCCQ-PLL	×		—	—	—	—
VDD-PLL	×		—	—	—	—
RTC_X1	×		—	—	—	—
RTC_X2	×		—	—	—	—
CAN_CLK_A	√	GP4[30]	6 mA	—	pull-up*	—
CAN0_TX_A	√	GP4[31]	6 mA	—	*	—
CAN0_RX_A	√	GP5[10]	6 mA	—	*	—
CAN1_TX_A	√	GP5[11]	6 mA	—	*	—
CAN1_RX_A	√	GP5[0]	6 mA	—	pull-up*	—
AN0	GPI	GP5[2]	—	—	—	—
AN1	GPI	GP5[3]	—	—	—	—

Signal Name	√: GPIO Included GPI: GPI Included ×: No GPIO	(GPIO/Peripheral Function Select Bit)	(Driving Ability of I/O Pins Powered by VCCQ)	Grouping of Driving Ability	Pull-up Control (Pull-up*: Pulling up is enabled from the reset state. *: Pulling up is disabled in the reset state. —: No pull-ups)	Mode Signal
AN2	GPI	GP5[4]	—	—	—	—
AN3	GPI	GP5[5]	—	—	—	—
AN4	GPI	GP5[6]	—	—	—	—
AN5	GPI	GP5[7]	—	—	—	—
AN6	GPI	GP5[8]	—	—	—	—
AN7	GPI	GP5[9]	—	—	—	—
AVCC	×		—	—	—	—
AVSS	×		—	—	—	—
AVREF	×		—	—	—	—

[Legend for Pull-up Control]

pull-up*: A register of the PFC controls whether or not the level is pulled up. The initial setting is for pulling up (pulling up is applied while $\overline{\text{PRESET}}$ is at the low level and until the relevant register setting in the PFC is modified after release from the power-on reset state).

*: A register of the PFC controls whether or not the level is pulled up. The initial setting is for not pulling up. (pulling up is not applied while $\overline{\text{PRESET}}$ is at the low level and until the relevant register setting in the PFC is modified after release from the power-on reset state).

—: Pulling-up control is not available.

pull-up: Always pulled up.

1.16.2 Pin Function Table for the BGA440

Table 1.4 Pin Function Table (for the BGA440)

Signal	First Function			Second Function			Third Function			Fourth Function			Fifth Function			Sixth Function		
Name	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO
MCK0	DDR2/3 IF 16 bits	MCK0	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
$\overline{\text{MCK0}}$		$\overline{\text{MCK0}}$	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MCKE		MCKE	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
$\overline{\text{MCS}}$		$\overline{\text{MCS}}$	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
$\overline{\text{MWE}}$		$\overline{\text{MWE}}$	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
$\overline{\text{MRAS}}$		$\overline{\text{MRAS}}$	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
$\overline{\text{MCAS}}$		$\overline{\text{MCAS}}$	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MA0		MA0	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MA1		MA1	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MA2		MA2	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MA3		MA3	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MA4		MA4	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MA5		MA5	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MA6		MA6	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MA7		MA7	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MA8		MA8	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MA9		MA9	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MA10		MA10	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MA11		MA11	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MA12		MA12	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MA13		MA13	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MBA0		MBA0	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MBA1		MBA1	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MBA2		MBA2	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MDQ0		MDQ0	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MDQ1		MDQ1	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MDQ2		MDQ2	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MDQ3		MDQ3	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Signal	First Function			Second Function			Third Function			Fourth Function			Fifth Function			Sixth Function		
Name	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO
MDQ4	DDR2/3 IF 16 bits	MDQ4	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MDQ5		MDQ5	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MDQ6		MDQ6	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MDQ7		MDQ7	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MDQ8		MDQ8	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MDQ9		MDQ9	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MDQ10		MDQ10	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MDQ11		MDQ11	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MDQ12		MDQ12	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MDQ13		MDQ13	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MDQ14		MDQ14	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MDQ15		MDQ15	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MDQS0		MDQS0	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
$\overline{\text{MDQS0}}$		$\overline{\text{MDQS0}}$	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MDQS1		MDQS1	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
$\overline{\text{MDQS1}}$		$\overline{\text{MDQS1}}$	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MDM0		MDM0	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MDM1		MDM1	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MODT		MODT	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
$\overline{\text{MBKPRST}}$		$\overline{\text{MBKPRST}}$	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MZQ		MZQ	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MVREFDQ		MVREFDQ	P	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MVREFCA		MVREFCA	P	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
$\overline{\text{MRESET}}$		$\overline{\text{MRESET}}$	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
SDBUP	DDR2 IF	SDBUP	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
SDSELF		SDSELF	O	(SC1 mirror)	$\overline{\text{RTS1\_E}}$	IO	—	—	—	—	—	—	—	—	—	—	—	—
TEST1	SYSTEM	TEST1	I	—			—			—			—			—		
TEST2		TEST2	I															
BSMODE		BSMODE	I															
EXTAL	CPG	EXTAL	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
XTAL		XTAL	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Signal	First Function			Second Function			Third Function			Fourth Function			Fifth Function			Sixth Function		
Name	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO
PRESET	RESET	PRESET	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
PRESETOUT		PRESETOUT	O	—			—			STIF	ST_CLKOUT	O	—	—	—	—	—	—
A0	LBSC	A0	O								ST0_CLKIN	I	LCDC_A	LCD_DATA0_A	O	MTU2 Common (Mirror_C)	TCLKA_C	I
A1		A1	O								ST0_REQ	IO		LCD_DATA1_A	O		TCLKB_C	I
A2		A2	O								ST0_SYC	IO		LCD_DATA2_A	O		TCLKC_C	I
A3		A3	O								ST0_VLD	IO		LCD_DATA3_A	O		TCLKD_C	I
A4		A4	O								ST0_D[0]	IO		LCD_DATA4_A	O	MTU2/CH0	TIOC0A_C	IO
A5		A5	O								ST0_D[1]	IO		LCD_DATA5_A	O		TIOC0B_C	IO
A6		A6	O				—	—	—		ST0_D[2]	IO		LCD_DATA6_A	O	MTU2/CH0	TIOC0C_C	IO
A7		A7	O				—	—	—		ST0_D[3]	IO		LCD_DATA7_A	O		TIOC0D_C	IO
A8		A8	O				—	—	—		ST0_D[4]	IO		LCD_DATA8_A	O	MTU2/CH1	TIOC1A_C	IO
A9		A9	O				—	—	—		ST0_D[5]	IO		LCD_DATA9_A	O		TIOC1B_C	IO
A10		A10	O				—	—	—		ST0_D[6]	IO		LCD_DATA10_A	O	MTU2/CH2	TIOC2A_C	IO
A11		A11	O				—	—	—		ST0_D[7]	IO		LCD_DATA11_A	O		TIOC2B_C	IO
A12		A12	O				—	—	—	—	—	—		LCD_DATA12_A	O	MTU2/CH3	TIOC3A_C	IO
A13		A13	O				—	—	—					LCD_DATA13_A	O		TIOC3B_C	IO
A14		A14	O				—	—	—					LCD_DATA14_A	O		TIOC3C_C	IO
A15		A15	O				—	—	—	STIF	ST0_VCO_CLKIN	I		LCD_DATA15_A	O		TIOC3D_C	IO
A16		A16	O				—	—	—		ST0_PWM	O		LCD_DON_A	O	MTU2/CH4	TIOC4A_C	IO

Signal	First Function			Second Function			Third Function			Fourth Function			Fifth Function			Sixth Function		
Name	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO
A17	LBSC	A17	O	—			—	—	—	STIF	ST1_VCO_CLKIN	I	LCDC_A	LCD_CL1_A	O	MTU2/CH4	TIOC4B_C	IO
A18		A18	O				—	—	—		ST1_PWM	O		LCD_CL2_A	O		TIOC4C_C	IO
A19		A19	O				—	—	—		ST1_CLKIN	I		LCD_CLK_A	I		TIOC4D_C	IO
A20		A20	O				—	—	—		ST1_REQ	IO		LCD_FLM_A	O	—	—	—
A21		A21	O				—	—	—		ST1_SYC	IO		LCD_VCPWC_A	O	—	—	—
A22		A22	O				—	—	—		ST1_VLD	IO		LCD_VEPWC_A	O	—	—	—
A23		A23	O				—	—	—		ST1_D[0]	IO		LCD_M_DISP_A	O	—	—	—
A24		A24	O	SCIF2_D (SC2 mirror)	RX2_D	I	—	—	—		ST1_D[1]	IO	—			—	—	—
A25		A25	O		TX2_D	O	—	—	—		ST1_D[2]	IO				—	—	—
D0		D0	IO	SDHI0_A (for booting)	SD0_DAT0_A	IO	MMC_A (for booting)	MMC_D0_A	IO		ST1_D[3]	IO	FLCTL_A (for booting)	NAF0_A	IO	—		
D1		D1	IO		SD0_DAT1_A	IO		MMC_D1_A	IO		ST1_D[4]	IO		NAF1_A	IO			
D2		D2	IO		SD0_DAT2_A	IO		MMC_D2_A	IO		ST1_D[5]	IO		NAF2_A	IO			
D3		D3	IO		SD0_DAT3_A	IO		MMC_D3_A	IO		ST1_D[6]	IO		NAF3_A	IO			
D4		D4	IO		SD0_CD_A	I		MMC_D4_A	IO		ST1_D[7]	IO		NAF4_A	IO			
D5		D5	IO		SD0_WP_A	I		MMC_D5_A	IO	—				NAF5_A	IO			
D6		D6	IO	RSPI_A (for booting)	RSPI_RSPCK_A	IO		MMC_D6_A	IO	RQSPI (Mirror A)	QSPCLK_A	IO		NAF6_A	IO			
D7		D7	IO		RSPI_SSL_A	IO		MMC_D7_A	IO		QSSL_A	IO		NAF7_A	IO			
D8		D8	IO	SDHI0_A (for booting)	SD0_CLK_A	O		MMC_CLK_A	O		QIO2_A	IO		$\overline{\text{FCE\_A}}$	O	GETHER (Gr#2; mirror B)	ET0_GTX_CLK_B	O
D9		D9	IO		SD0_CMD_A	IO		MMC_CMD_A	IO		QIO3_A	IO		FCLE_A	O		ET0_ETXD1_B	O
D10		D10	IO	RSPI_A (for booting)	RSPI_MOSI_A	IO	—	—	—		QMO/QIO0_A	IO		FALE_A	O		ET0_ETXD2_B	O
D11		D11	IO		RSPI_MISO_A	IO	—	—	—		QMI/QIO1_A	IO		$\overline{\text{FRE\_A}}$	O		ET0_ETXD3_B	O

Signal	First Function			Second Function			Third Function			Fourth Function			Fifth Function			Sixth Function		
Name	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO
D12	LBSC	D12	IO	—			—	—	—	—			FLCTL_A (for booting)	FWE_A	O	GETHER (Gr#2; mirror B)	ET0_ETXD5_B	O
D13		D13	IO	SCIF2_B (SC2 mirror)	RX2_B	I	—	—	—					FRB_A	I		ET0_ETXD6_B	O
D14		D14	IO		TX2_B	O	—	—	—					—	—		ET0_TX_CLK_B	I
D15		D15	IO		SCK2_B	IO	—	—	—				—			—	—	—
CLKOUT		CLKOUT	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
BS		BS	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
CS0		CS0	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
CS1/A26		CS1/A26	O	—	—	—	—	—	—	—	—	—	RQSPI (Mirror B)	QIO3_B	IO	—	—	—
EX_CS0		EX_CS0	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
EX_CS1		EX_CS1	O	SCIF3_B (SC3 mirror)	RX3_B	I	LBSC (ATA)	ATACS0	O	—			RQSPI (Mirror B)	QIO2_B	IO	GETHER (Gr#2 mirror A)	ET0_ETXD0	O
EX_CS2		EX_CS2	O		TX3_B	O		ATACS1	O					QSPCLK_B	IO		ET0_GTX_CLK_A	O
EX_CS3		EX_CS3	O	SDHI1_A	SD1_CD_A	I		ATARD	O	—	—	—		QMO/QIO0_B	IO		ET0_ETXD1_A	O
EX_CS4		EX_CS4	O		SD1_WP_A	I		ATAWR	O	—	—	—		QMI/QIO1_B	IO		ET0_ETXD2_A	O
EX_CS5		EX_CS5	O		SD1_CMD_A	IO		ATADIR	O	—	—	—		QSSL_B	IO		ET0_ETXD3_A	O
RD		RD	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RD/WR		RD/WR	O	TMU	TCLK0	I	—	—	—	RCAN	CAN_CLK_B	I	—	—	—	(Gr#2)	ET0_ETXD4	O
WE0		WE0	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
WE1		WE1	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
EX_WAIT0		EX_WAIT0	I	TMU	TCLK1_B	I	—	—	—	—	—	—	—	—	—	—	—	—
EX_WAIT1		EX_WAIT1	I	SDHI1_A	SD1_DAT0_A	IO	LBSC (ATA)	DREQ2	I	RCAN1_C	CAN1_TX_C	O	GETHER_B (Gr#1,C)	ET0_LINK_C	I	(Gr#2 mirror A)	ET0_ETXD5_A	O
EX_WAIT2		EX_WAIT2	I		SD1_DAT1_A	IO		DACK2	O		CAN1_RX_C	I		ET0_MAGIC_C	O		ET0_ETXD6_A	O
DRACK0		DRACK0	O		SD1_DAT2_A	IO		ATAG	O	TMU	TCLK1_A	I	—				ET0_ETXD7	O
DREQ0		DREQ0	I		SD1_CLK_A	O	—			—							ET0_TX_EN	O

Signal	First Function			Second Function			Third Function			Fourth Function			Fifth Function			Sixth Function		
Name	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO
DACK0	LBSC	DACK0	O	SDHI1_A	SD1_DAT3_A	IO	—			—			—			(Gr#2 mirror A)	ET0_TX_ER	O
DREQ1		DREQ1	I	—			HSPI_B (mirror)	HSPI_CLK_B	IO	SCIF4_B (mirror)	RX4_B	I	GETHER_B (Gr#1,C)	ET0_PHY_INT_C	I		ET0_TX_CLK_A	I
DACK1		DACK1	O					HSPI_CS_B	IO		TX4_B	O	—				ET0_RX_CLK_A	I
TRST	H-UDI	TRST	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
TCK		TCK	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
TMS		TMS	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
TDI		TDI	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
TDO		TDO	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
MPMD	(PMG)	MPMD	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
ASEBRK/ACK	(PMG)	ASEBRK/ACK	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
NMI	INTC	NMI (wake up from deep standby)	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
IRQ0_A		IRQ0_A (wake up from deep standby)	I	—	—	—	HSPI_B (mirror)	HSPI_TX_B	O	SCIF3_E (mirror)	RX3_E	I	—			(Gr#2 mirror A)	ET0_ERXD0	I
IRQ1_A		IRQ1_A (wake up from deep standby)	I	—	—	—		HSPI_RX_B	IO		TX3_E	O					ET0_ERXD1	I
IRQ2_A		IRQ2_A (wake up from deep standby)	I	SCIF (SCIF0_A)	CTS0_A	IO	—			HSCIF_B (mirror)	HCTS0_B	IO					ET0_ERXD2_A	I
IRQ3_A		IRQ3_A (wake up from deep standby)	I		RTS0_A	IO					HRTS0_B	IO					ET0_ERXD3_A	I
SCIF_CLK_A	SCIF (SCIF0_A)	SCIF_CLK_A	I	HSPI_A	HSPI_CLK_A	IO	VIN0 (RGB666)	VI0_CLK	I	—			GETHER (Gr#3, RMII_A)	RMII0_TXD0_A	O		ET0_ERXD4	I
SCK0_A		SCK0_A	IO		HSPI_CS_A	IO		VI0_CLKENB	I					RMII0_TXD1_A	O		ET0_ERXD5	I

Signal	First Function			Second Function			Third Function			Fourth Function			Fifth Function			Sixth Function		
Name	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO
RX0_A	SCIF (SCIF0_A)	RX0_A	I	HSPI_A	HSPI_RX_A	IO	—			—			GETHER (Gr#3, RMII_A)	RMII0_RXD0_A	I	(Gr#2 mirror A)	ET0_ERXD6	I
TX0_A		TX0_A	O		HSPI_TX_A	O	—	—	—	—	—	—	—	—	—	—	—	—
HCTS0_A	HSCIF_A	HCTS0_A	IO	SCIF (SCIF1_A)	CTS1_A	IO	VIN0	VI0_FIELD	I	—	—	—	(Gr#3, RMII_A)	RMII0_RXD1_A	I	(Gr#2 mirror A)	ET0_ERXD7	I
HRTS0_A		HRTS0_A	IO		RTS1_A	IO		VI0_HSYNC	I	—	—	—		RMII0_TXD_EN_A	O		ET0_RX_DV	I
HSCK0_A		HSCK0_A	IO		SCK1_A	IO		VI0_VSYNC	I	—	—	—		RMII0_RX_ER_A	I		ET0_RX_ER	I
HRX0_A		HRX0_A	IO		RX1_A	I		VI0_DATA0/ VI0_B0	I	—	—	—		RMII0_CRS_DV_A	I		ET0_CRS	I
HTX0_A		HTX0_A	O		TX1_A	O		VI0_DATA1/ VI0_B1	I	—	—	—		RMII0_MDC_A	O		ET0_COL	I
CTS0_B	—	—	—	SCIF0_B (SC0 mirror)	CTS0_B	IO		VI0_DATA2/ VI0_B2	I	—	—	—		RMII0_MDIO_A	IO		ET0_MDC	O
RTS0_B		—	—		RTS0_B	IO		VI0_DATA3/ VI0_B3	I	—	—	—	—				ET0_MDIO_A	IO
SCK1_B		—	—	SCIF1_B (SC1 mirror)	SCK1_B	IO		VI0_DATA4/ VI0_B4	I	—	—	—	—			GETHER (Gr#1 mirror A)	ET0_LINK_A	I
RX1_B		—	—		RX1_B	I		VI0_DATA5/ VI0_B5	I	—	—	—					ET0_MAGIC_A	O
TX1_B		—	—		TX1_B	O		VI0_DATA6/ VI0_G0	I	—	—	—					ET0_PHY_INT_A	I
CTS1_B		—	—		CTS1_B	IO		VI0_DATA7/ VI0_G1	I	—	—	—				—		
RTS1_B		—	—		RTS1_B	IO		VI0_G2	I	—	—	—						
SCK2_A		—	—	SCIF2_A	SCK2_A	IO		VI0_G3	I	—	—	—						
SD2_CLK_A	SDHI_A (SD2)	SD2_CLK_A	O		RX2_A	I		VI0_G4	I	—	—	—	—	—	—	(Gr#2 mirror B)	ET0_RX_CLK_B	I
SD2_CMD_A		SD2_CMD_A	IO		TX2_A	O		VI0_G5	I	—	—	—		—	—		ET0_ERXD2_B	I
SD2_DAT0_A		SD2_DAT0_A	IO	SCIF3_A	RX3_A	I		VI0_R0	I	—	—	—		—	—		ET0_ERXD3_B	I
SD2_DAT1_A		SD2_DAT1_A	IO		TX3_A	O		VI0_R1	I	—	—	—		—	—		ET0_MDIO_B	IO

Signal	First Function			Second Function			Third Function			Fourth Function			Fifth Function			Sixth Function		
Name	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO
SD2_DAT2_A	SDHI_A (SD2)	SD2_DAT2_A	IO	SCIF4_A	RX4_A	I	VIN0	VI0_R2	I	—	—	—	—	—	—	GETHER (Gr#1 mirror B)	ET0_LINK_B	I
SD2_DAT3_A		SD2_DAT3_A	IO		TX4_A	O		VI0_R3	I	—	—	—		—	—		ET0_MAGIC_B	O
SD2_CD_A		SD2_CD_A	I	SCIF5_A	RX5_A	I		VI0_R4	I	—				—	—		ET0_PHY_INT_B	I
SD2_WP_A		SD2_WP_A	I		TX5_A	O		VI0_R5	I					—	—	—		
REF125CK	GEther	REF125CK	I	—	$\overline{ADTRG}$	I	SCIF5_C	RX5_C	I	—	—	—		—	—	—	—	—
REF50CK		REF50CK	I	SCIF1	$\overline{CTS1\_E}$	IO	HSCIF_D	$\overline{HCTS0\_D}$	IO	—	—	—		—	—	—	—	—
DU0_DR0	DU0	DU0_DR0	O	SCIF0_B	SCIF_CLK_B	I	HSCIF_D (mirror)	HRX0_D	I	IEBus (Mirror A)	IETX_A	O	MTU2 Common (Mirror_A)	TCLKA_A	I	HIF_B	HIFD00	IO
DU0_DR1		DU0_DR1	O		SCK0_B	IO		HTX0_D	O		IERX_A	I		TCLKB_A	I		HIFD01	IO
DU0_DR2		DU0_DR2	O		RX0_B	I	—			—				TCLKC_A	I		HIFD02	IO
DU0_DR3		DU0_DR3	O		TX0_B	O								TCLKD_A	I		HIFD03	IO
DU0_DR4		DU0_DR4	O	SCIF0_C (SC0 mirror)	$\overline{CTS0\_C}$	IO							MTU2/CH0	TIOC0A_A	IO		HIFD04	IO
DU0_DR5		DU0_DR5	O		$\overline{RTS0\_C}$	IO								TIOC0B_A	IO		HIFD05	IO
DU0_DR6		DU0_DR6	O	SCIF1_C (SC1 mirror)	SCK1_C	IO								TIOC0C_A	IO		HIFD06	IO
DU0_DR7		DU0_DR7	O		RX1_C	I								TIOC0D_A	IO		HIFD07	IO
DU0_DG0		DU0_DG0	O		TX1_C	O	HSCIF_D (mirror)	HSCK0_D	IO	IEBus (Mirror A)	IECLK_A	I	MTU2/CH1	TIOC1A_A	IO		HIFD08	IO
DU0_DG1		DU0_DG1	O		$\overline{CTS1\_C}$	IO		$\overline{HRTS0\_D}$	IO	—	—	—		TIOC1B_A	IO		HIFD09	IO
DU0_DG2		DU0_DG2	O		$\overline{RTS1\_C}$	IO	GETHER_B (Gr#3,RMII_B: mirror)	RMII0_MDC_B	O	—			MTU2/CH2	TIOC2A_A	IO		HIFD10	IO
DU0_DG3		DU0_DG3	O	SCIF2_C (SC2 mirror)	SCK2_C	IO		RMII0_MDIO_B	IO					TIOC2B_A	IO		HIFD11	IO
DU0_DG4		DU0_DG4	O		RX2_C	I		RMII0_CRS_DV_B	I				MTU2/CH3	TIOC3A_A	IO		HIFD12	IO
DU0_DG5		DU0_DG5	O		TX2_C	O		RMII0_RX_ER_B	I					TIOC3B_A	IO		HIFD13	IO
DU0_DG6		DU0_DG6	O	SCIF3_C (SC3 mirror)	RX3_C	I		RMII0_RXD0_B	I					TIOC3C_A	IO		HIFD14	IO
DU0_DG7		DU0_DG7	O		TX3_C	O		RMII0_RXD1_B	I					TIOC3D_A	IO		HIFD15	IO

Signal	First Function			Second Function			Third Function			Fourth Function			Fifth Function			Sixth Function		
Name	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO
DU0_DB0	DU0	DU0_DB0	O	SCIF4_C (SC4 mirror)	RX4_C	I	GETHER_B (Gr#3, RMII_B: mirror)	RMII0_TXD_EN_B	O	—	—	—	MTU2/CH4	TIOC4A_A	IO	HIF_B	$\overline{\text{HIFCS}}$	I
DU0_DB1		DU0_DB1	O		TX4_C	O		RMII0_TXD0_B	O	—	—	—		TIOC4B_A	IO		HIFRS	I
DU0_DB2		DU0_DB2	O	SCIF5_B (SC5 mirror)	RX5_B	I		RMII0_TXD1_B	O	—				TIOC4C_A	IO		$\overline{\text{HIFWR}}$	I
DU0_DB3		DU0_DB3	O		TX5_B	O	—							TIOC4D_A	IO		$\overline{\text{HIFRD}}$	I
DU0_DB4		DU0_DB4	O	SDHI2_B (SD2 mirror)	SD2_CLK_B	O							—				$\overline{\text{HIFINT}}$	O
DU0_DB5		DU0_DB5	O		SD2_CMD_B	IO											HIFDREQ	O
DU0_DB6		DU0_DB6	O		SD2_DAT0_B	IO											HIFRDY	O
DU0_DB7		DU0_DB7	O		SD2_DAT1_B	IO							SSI_B (mirror)	SSI_SCK0_B	IO		HIFEBL_B	I
DU0_DOTCLKIN		DU0_DOTCLKIN	I		SD2_DAT2_B	IO				HSPI_C (mirror)	$\overline{\text{HSPI_CS_C}}$	IO		SSI_WS0_B	IO	—		
DU0_DOTCLKOUT		DU0_DOTCLKOUT	O		SD2_DAT3_B	IO					HSPI_CLK_C	IO		SSI_SDATA0_B	IO			
DU0_EXHSYNC/DU0_HSYNC		DU0_EXHSYNC/DU0_HSYNC	IO		SD2_CD_B	I				HSPI_C	HSPI_TX_C	O		SSI_SCK1_B	IO			
DU0_EXVSYNC/DU0_VSYNC		DU0_EXVSYNC/DU0_VSYNC	IO		SD2_WP_B	I					HSPI_RX_C	IO		SSI_WS1_B	IO			
DU0_EXODDF/DU0_ODDF		DU0_EXODDF/DU0_ODDF	IO	CAN0_B (CAN0 mirror)	CAN0_RX_B	I				HSCIF_B	HSCK0_B	IO		SSI_SDATA1_B	IO			
DU0_DISP		DU0_DISP	O		CAN0_TX_B	O				HSCIF_B (mirror)	HRX0_B	I		AUDIO_CLKA_B	I			
DU0_CDE		DU0_CDE	O	—							HTX0_B	O		AUDIO_CLKB_B	I	LCDC (mirror)	LCD_VCPWC_B	O
VI1_CLK_A	VIN1 (YUV) (Mirror_A)	VI1_CLK_A	I	H-UDI	AUDCK	O	—	—	—	—			FLCTL_B (mirror)	NAF0_B	IO		LCD_DATA0_B	O
VI1_0_A		VI1_0_A	I		AUDSYNC	O		—	—					NAF1_B	IO		LCD_DATA1_B	O
VI1_1_A		VI1_1_A	I		AUDATA0	O		—	—					NAF2_B	IO		LCD_DATA2_B	O
VI1_2_A		VI1_2_A	I		AUDATA1	O		—	—					NAF3_B	IO		LCD_DATA3_B	O

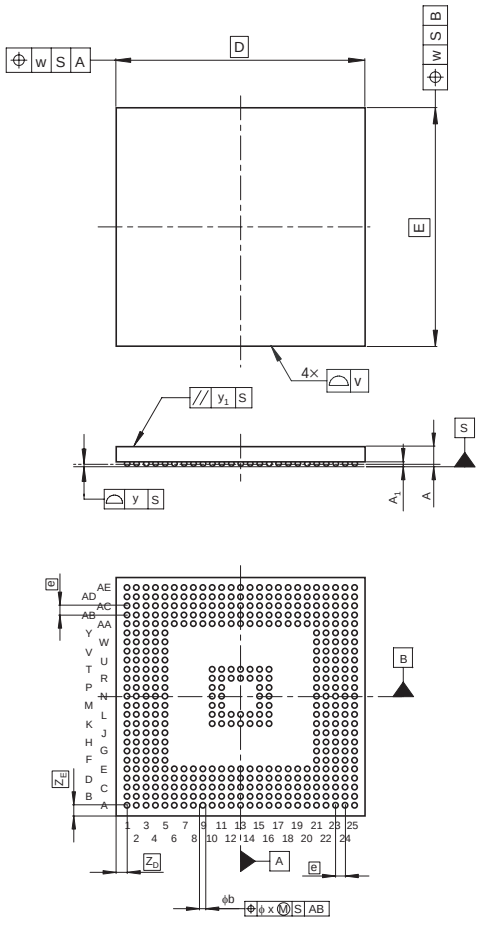
Signal	First Function			Second Function			Third Function			Fourth Function			Fifth Function			Sixth Function		
Name	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO
VI1_3_A	VIN1 (YUV) (Mirror_A)	VI1_3_A	I	H-UDI	AUDATA2	O	—	—	—	—			FLCTL_B (mirror)	NAF4_B	IO	LCDC (mirror)	LCD_DATA4_B	O
VI1_4_A		VI1_4_A	I		AUDATA3	O		—	—					NAF5_B	IO		LCD_DATA5_B	O
VI1_5_A		VI1_5_A	I		AUDATA4	O		—	—					NAF6_B	IO		LCD_DATA6_B	O
VI1_6_A		VI1_6_A	I		AUDATA5	O		—	—					NAF7_B	IO		LCD_DATA7_B	O
VI1_7_A		VI1_7_A	I		AUDATA6	O	—							$\overline{\text{FCE\_B}}$	O		LCD_DATA8_B	O
SSI_SCK0_A	SSI	SSI_SCK0_A	IO		AUDATA7	O	—						MTU2/CH1	TIOC1A_B	IO		LCD_DATA9_B	O
SSI_WS0_A		SSI_WS0_A	IO	—						—	—	—		TIOC1B_B	IO		LCD_DATA10_B	O
SSI_SDATA0_A		SSI_SDATA0_A	IO	VIN1 (YUV) (Mirror_B)	VI1_0_B	I				—	—	—	MTU2/CH2	TIOC2A_B	IO		LCD_DATA11_B	O
SSI_SCK1_A		SSI_SCK1_A	IO		VI1_1_B	I				—				TIOC2B_B	IO		LCD_DATA12_B	O
SSI_WS1_A		SSI_WS1_A	IO		VI1_2_B	I				—	—	—	—				LCD_DATA13_B	O
SSI_SDATA1_A		SSI_SDATA1_A	IO		VI1_3_B	I				—	—	—					LCD_DATA14_B	O
SSI_SCK23		SSI_SCK23	IO		VI1_4_B	I	(SC1 mirror)	RX1_D	I	—			FLCTL_B (mirror)	FCLE_B	O		LCD_DATA15_B	O
SSI_WS23		SSI_WS23	IO		VI1_5_B	I		TX1_D	IO	HSCIF_C (mirror)	HSCK0_C	IO		FALE_B	O		LCD_DON_B	O
SSI_SDATA2		SSI_SDATA2	IO		VI1_6_B	I	—	—	—		HRX0_C	I		$\overline{\text{FRE\_B}}$	O		LCD_CL1_B	O
SSI_SDATA3		SSI_SDATA3	IO		VI1_7_B	I		—	—		HTX0_C	O		$\overline{\text{FWE\_B}}$	O		LCD_CL2_B	O
AUDIO_CLKA_A		AUDIO_CLKA_A	I		VI1_CLK_B	I	(SC1 mirror)	SCK1_D	IO	—	—	—	IEBus (Mirror B)	IECLK_B	I		LCD_FLM_B	O
AUDIO_CLKB_A		AUDIO_CLKB_A	I	—			—			—	—	—	—				LCD_CLK_B	I
AUDIO_CLKC		AUDIO_CLKC	I	SCIF1 (SC1 mirror)	SCK1_E	IO	—	—	—	HSCIF_C (mirror)	$\overline{\text{HCTS0\_C}}$	IO	FLCTL_B (mirror)	FRB_B	I		LCD_VEPWC_B	O
AUDIO_CLKOUT		AUDIO_CLKOUT	O		TX1_E	O		—	—		$\overline{\text{HRTS0\_C}}$	IO		—	—		LCD_M_DISP_B	O

Signal	First Function			Second Function			Third Function			Fourth Function			Fifth Function			Sixth Function		
Name	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO
SCL0	IIC3	SCL0 (Open Drain)	IO	—			—			—			—			—	—	—
SDA0		SDA0 (Open Drain)	IO													HIF_A	HIFEBL_A	I
SCL1		SCL1 (Open Drain)	IO	SCIF0_C	SCIF_CLK_C	I										—		
SDA1		SDA1 (Open Drain)	IO	SCIF1	RX1_E	I												
USB_EXTAL	USB	USB_EXTAL	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
USB_XTAL		USB_XTAL	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
PENC0		PENC0	O	—			—			—			—			—		
PENC1		PENC1	O	SCIF3_D	TX3_D	O	RCAN1_B	CAN1_TX_B	O	SCIF	TX5_D	O	IEBus (Mirror B)	IETX_B	O			
USB_OVC0		USB_OVC0	I	—			—			—			—					
USB_OVC1		USB_OVC1	I	SCIF3_D	RX3_D	I	RCAN1_B	CAN1_RX_B	I	SCIF	RX5_D	I	IEBus (Mirror B)	IERX_B	I			
DP0		DP0	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
DM0		DM0	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
DP1		DP1	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
DM1		DM1	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
REFRIN		REFRIN	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
OVC0/ VBUS0		OVC0/ VBUS0	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
OVC1/ VBUS1		OVC1/ VBUS1	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
AV33		AV33	P	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
AV12		AV12	P	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
AG		AG	P	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
VCCQ-PLL	PLL	VCCQ-PLL	P	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
VDD-PLL		VDD-PLL	P	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RTC_X1	RTC	RTC_X1	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
RTC_X2		RTC_X2	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Signal	First Function			Second Function			Third Function			Fourth Function			Fifth Function			Sixth Function		
Name	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO	Module	Pin Name	IO
CAN_CLK_A	RCAN	CAN_CLK_A	I	SCIF4_D	RX4_D	I	—			—			—			—		
CAN0_TX_A		CAN0_TX_A	O	SCIF4_D	TX4_D	O	MIMLB	MLB_CLK	I									
CAN0_RX_A		CAN0_RX_A (wake up from deep standby)	I	INTC (mirror B)	IRQ0_B (wake up from deep standby)	I		MLB_SIG	IO	—	—	—	—	—	—	—	—	—
CAN1_TX_A		CAN1_TX_A	O	SCIF5_C	TX5_C	O		MLB_DAT	IO	—	—	—	—	—	—	—	—	—
CAN1_RX_A		CAN1_RX_A (wake up from deep standby)	I	INTC (mirror B)	IRQ1_B (wake up from deep standby)	I	—	—	—	—	—	—	—	—	—	—	—	—
AN0	ADC	AN0	I	INTC (mirror B)	IRQ2_B	I	—	—	—	—	—	—	—	—	—	—	—	—
AN1		AN1	I		IRQ3_B	I	—	—	—	—	—	—	—	—	—	—	—	—
AN2		AN2	I	—			—	—	—	—	—	—	—	—	—	—	—	—
AN3		AN3	I				—	—	—	—	—	—	—	—	—	—	—	—
AN4		AN4	I				—	—	—	—	—	—	—	—	—	—	—	—
AN5		AN5	I				—	—	—	—	—	—	—	—	—	—	—	—
AN6		AN6	I				—	—	—	—	—	—	—	—	—	—	—	—
AN7		AN7	I				—	—	—	—	—	—	—	—	—	—	—	—
AVCC		AVCC	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
AVSS		AVSS	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
AVREF		AVREF	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

1.17 Dimensions

JEITA Package Code	RENESAS Code	Previous Code	MASS[Typ.]
P-FBGA440-21x21-0.80	PRBG0440GB-A	—	1.6g



Reference Symbol	Dimension in Millimeters		
	Min	Nom	Max
D	—	21.0	—
E	—	21.0	—
v	—	—	0.15
w	—	—	0.20
A	—	—	2.10
A_1	0.35	0.40	0.45
e	—	0.80	—
b	0.45	0.50	0.55
x	—	—	0.08
y	—	—	0.10
y_1	—	—	0.20
S_D	—	—	—
S_E	—	—	—
Z_D	—	0.9	—
Z_E	—	0.9	—

Figure 1.10 Package Dimensions

Section 2 SH-4A

2.1 Overview

2.1.1 Features

This LSI is a 32-bit RISC (reduced instruction set computer) microprocessor that is upwardly compatible with the SH-1, SH-2, SH-3, and SH-4 microprocessors at the instruction set level. Its 16-bit fixed-length instruction set enables program code size to be reduced by almost 50% compared with 32-bit instructions.

The features of this LSI are listed in table 2.1.

Table 2.1 Features

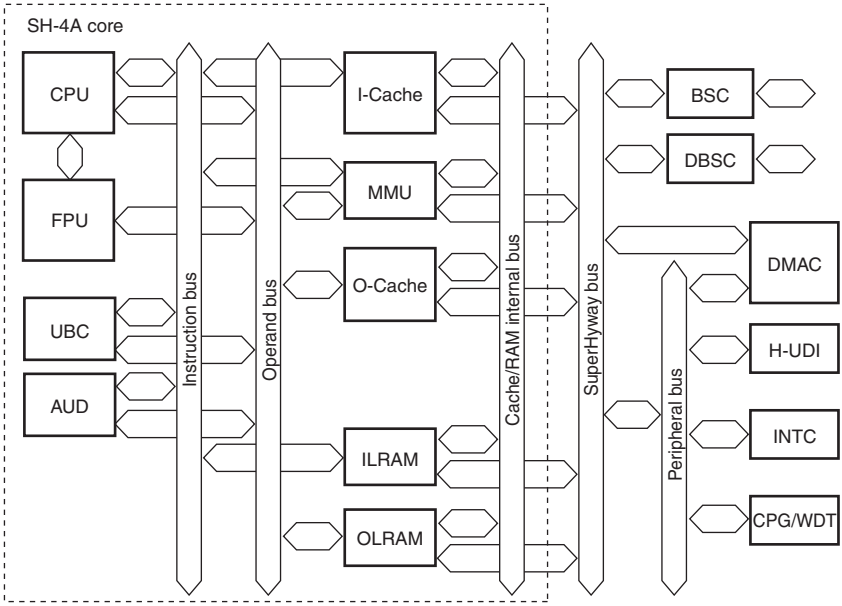
Item	Features
CPU	• Renesas original architecture • 32-bit internal data bus • General-register files: — Sixteen 32-bit general registers (eight 32-bit shadow registers) — Seven 32-bit control registers — Four 32-bit system registers • RISC-type instruction set (upwardly compatible with the SH-1, SH-2, SH-3, and SH-4 microprocessors) — Instruction length: 16-bit fixed length for improved code efficiency — Load/store architecture — Delayed branch instructions — Instructions executed with conditions — Instruction set based on the C language • Superscalar which executes two instructions simultaneously including the FPU • Instruction execution time: Two instructions per cycle (max) • Virtual address space: 4 Gbytes • Address space identifier (ASID): 8 bits, 256 virtual address spaces • On-chip multiplier • Eight-stage pipeline

Item	Features
Floating-point unit (FPU)	• On-chip floating-point coprocessor • Supports single-precision (32 bits) and double-precision (64 bits) • Supports IEEE754-compliant data types and exceptions • Two rounding modes: Round to Nearest and Round to Zero • Handling of denormalized numbers: Truncation to zero or interrupt generation for IEEE754 compliance • Floating-point registers: 32 bits × 16 words × 2 banks (single-precision × 16 words or double-precision × 8 words) × 2 banks • 32-bit CPU-FPU floating-point communication register (FPUL) • Supports FMAC (multiply-and-accumulate) instruction • Supports FDIV (divide) and FSQRT (square root) instructions • Supports FLDI0/FLDI1 (load constant 0/1) instructions • Instruction execution times — Latency (FADD/FSUB): 3 cycles (single-precision), 5 cycles (double-precision) — Latency (FMAC/ FMUL): 5 cycles (single-precision), 7 cycles (double-precision) — Pitch (FADD/FSUB): 1 cycle (single-precision/double-precision) — Pitch (FMAC/FMUL): 1 cycle (single-precision), 3 cycles (double-precision) Note: FMAC is supported for single-precision only. • 3-D graphics instructions (single-precision only): — 4-dimensional vector conversion and matrix operations (FTRV): 4 cycles (pitch), 8 cycles (latency) — 4-dimensional vector (FIPR) inner product: 1 cycle (pitch), 5 cycles (latency) • Ten-stage pipeline

Item	Features
Memory management unit (MMU)	• 4 Gbytes of physical address space, 256 address space identifiers (address space identifier ASID: 8 bits) • Supports single virtual memory mode and multiple virtual memory mode • Supports multiple page sizes: 1 Kbytes, 4 Kbytes, 8 Kbytes, 64 Kbytes, 256 Kbytes, 1 Mbytes, 4 Mbytes, or 64 Mbytes • 4-entry full associative TLB for instructions • 64-entry full associative TLB for instructions and operands • Supports software selection of replacement method and random-counter replacement algorithms • Contents of TLB are directly accessible through address mapping • 32-bit address extended mode
Cache memory	• Instruction cache (IC) — 4-way set associative — 32-byte block length • Operand cache (OC) — 4-way set associative — 32-byte block length — Selectable write method (copy-back or write-through) • Store queue (32 bytes × 2 entries)
IL memory (ILRAM)	• Three independent read/write ports — Instruction fetch access from the CPU — 8-/16-/32-/64-bit operand access from the CPU — 8-/16-/32-/64-bit or 16-/32-byte access requested externally
OL memory (OLRAM)	• Three independent read/write ports — Instruction fetch access from the CPU — 8-/16-/32-/64-bit operand access from the CPU — 8-/16-/32-/64-bit or 16-/32-byte access requested externally

2.1.2 Block Diagram

Figure 2.1 shows an example of a block diagram of a SH-4A product.



[Legend]

AUD:	Advanced user debugger	I-Cache:	Instruction cache
BSC:	Bus state controller	ILRAM:	IL memory (instruction local memory)
CPG/WDT:	Clock pulse generator/watchdog timer	INTC:	Interrupt controller
CPU:	Central processing unit	MMU:	Memory management unit
DBSC:	DRAM bus state controller	O-Cache:	Operand (data) cache
DMAC:	Direct memory access controller	OLRAM:	OL memory (Operand (data) local memory)
FPU:	Floating-point unit	UBC:	User break controller
H-UDI:	User debugging interface		

Figure 2.1 Block Diagram

2.1.3 SH-4A Extended Functions

The SH-4A microprocessors with VER bits in the processor version register (PVR) hold H'30 or greater have upward compatibility with the SH-4A with H’20-valued VER bits.

The SH-4A microprocessors with VER bits hold H’30 or greater have the following extended functions added on to the functionality of the SH-4A with H’20-valued VER bits.

Table 2.2 SH-4A Extended Functions (PVR.VER = H'30)

Item	Features
Instruction set	• The LDTLB operation has been changed.
Pipelining	• A predecode (I3) stage has been added between the I2 stage and the ID stage.
Exception handling	• The non-support detection exception register (EXPMASK) has been added.
Memory management unit (MMU) (some items in this section added and changed)	• Page sizes: 1, 4, 8, 64, or 256 Kbytes, 1, 4, or 64 Mbytes • Further functions for access-right checking have been added. • The PTEA register and the ME bit in the MMU control register have been newly added.
Caches (some items in this section added and changed)	• A low-power function (IC way prediction scheme) has been added. • Functions of instructions for the operand cache manipulation (OCBI, OCBP, and OCBWB) have been added. • A function for fixing to the 2-way set associative method has been added.
IL memory (ILRAM) (section newly added)	• Three independent read/write ports — Instruction fetch access from the CPU — 8-/16-/32-/64-bit operand access from the CPU — 8-/16-/32-/64-bit or 16-/32-byte access requested externally
OL memory (OLRAM) (section name and some material in this section changed)	• With the IL memory newly added, the L memory can be used as operand memory (OL memory). • Three independent read/write ports — Instruction fetch access from the CPU — 8-/16-/32-/64-bit operand access from the CPU — 8-/16-/32-/64-bit or 16-/32-byte access requested externally

2.1.4 Changes from SH-4 to SH-4A (PVR.VER = H'20)

Table 2.3 summarizes the changes made to the SH-4 in the development of the SH-4A with H'20-valued VER bits in the processor version register (PVR) for each of the sections and sub-sections of this manual.

Table 2.3 Changes from SH-4 to SH-4A (PVR.VER = H'20)

Section No. and Name	Sub-section	Sub-section Name	Changes
2.1 Overview	—	—	Modified entirely (Detailed differences are described in the following sections).
2.2 Programming Model	2.2.2	Register Descriptions	The operations in SZ=1 and PR=1 are added to the floating point status/control register (FPSCR).
2.3 Instruction Set	2.3.3	Instruction Set	9 instructions are added as CPU instructions.
			3 instructions are added as FPU instructions.
2.4 Pipelining	2.4.1	Pipelines	The number of stages in the pipeline is changed from five to seven.
	2.4.2	Parallel-Executability	9 instructions are added as CPU instructions.
			3 instructions are added as FPU instructions.
			Instruction group and parallel execution combinations are modified.
	2.4.3	Issue Rates and Execution States	The number of issue rates and execution states is modified.
2.5 Exception Handling	—	—	—
2.6 Floating-Point Unit (FPU)	2.6.3 (2)	Floating-Point Status/Control Register (FPSCR)	Operations in SZ = 1 and PR = 1 and each endian are added
	2.6.5	Floating-Point Exceptions	Specification of FPU exception detection condition with FPU exception enabled is changed.

Section No. and Name	Sub-section	Sub-section Name	Changes
2.7 Memory Management Unit (MMU)	2.7.1 (1)	Address Spaces	Area P4 configuration is modified.
			On-chip RAM space is deleted.
	2.7.2	Register Descriptions	The page table entry assist register (PTEA) is deleted.
			A physical address space control register is added.
	2.7.2 (7)	Physical Address Space Control Register (PASCR)	Newly added
	2.7.2 (8)	Instruction Re-Fetch Inhibit Control Register (IRMCR)	Newly added.
	2.7.3	TLB Functions (TLB Compatible Mode; MMUCR.ME = 0)	Space attribute bits (SA [2:0]) and timing control bit (TC) are deleted from the TLB.
	2.7.5 (5)	Avoiding Synonym Problems	The corresponding bits are modified according to the cache size change and the index mode deletion.
	2.7.6 (1), 2.7.6 (4)	Instruction TLB Multiple Hit Exception and Data TLB Multiple Hit Exception	Multiple hits during the UTLB search caused by ITLB miss handling are changed to be handled as a TLB multiple hit instruction exception.
	2.7.7	Memory-Mapped TLB Configuration	Data array 2 in the ITLB and UTLB is deleted.
	2.7.7 (4)	UTLB Address Array	Associative writes to the UTLB address array are changed to not generate data TLB multiple hit exceptions.
			Memory allocated addresses are changed from H'F6000000–H'F6FFFFFF to H'F6000000–H'F60FFFFF.
	2.7.7 (5)	UTLB Data Array (TLB Compatible Mode)	Memory allocated addresses are changed from H'F7000000–H'F77FFFFF to H'F7000000–H'F70FFFFF.
	2.7.8	32-Bit Address Extended Mode	Newly added.

Section No. and Name	Sub-section	Sub-section Name	Changes
2.8 Caches	2.8.1	Features	Instruction cache capacity is changed to 32 Kbytes.
			The caching method is changed to a 4-way set-associative method.
	2.8.2	Register Descriptions	An on-chip memory control register is added.
	2.8.2 (1)	Cache Control Register (CCR)	Modified. (Descriptions in CCR are modified.)
	2.8.2 (4)	On-Chip Memory Control Register (RAMCR)	Newly added.
	2.8.3	Operand Cache Operation	RAM mode and OC index mode are deleted.
	2.8.3 (6)	OC Two-Way Mode	Newly added.
	2.8.4	Instruction Cache Operation	IC index mode is deleted.
	2.8.4 (3)	IC Two-Way Mode	Newly added.
	2.8.5 (1)	Coherency between Cache and External Memory	The ICBI, PREFI, and SYNCO instructions are added.
	2.8.6	Memory-Mapped Cache Configuration	The entry bits and the way bits are modified according to the size modification and changed into 4-way set associative cache.
	2.8.8	Notes on Using 32-Bit Address Extended Mode	Newly added.
2.9 On-Chip Memory	—	—	Newly added.
11. Instruction Descriptions of the SH-4A Extended Functions Software Manual	—	—	9 instructions are added as CPU instructions.
			3 instructions are added as FPU instructions.

2.2 Programming Model

The programming model of the SH-4A is explained in this section. The SH-4A has registers and data formats as shown below.

2.2.1 Data Formats

The data formats supported in the SH-4A are shown in figure 2.2.

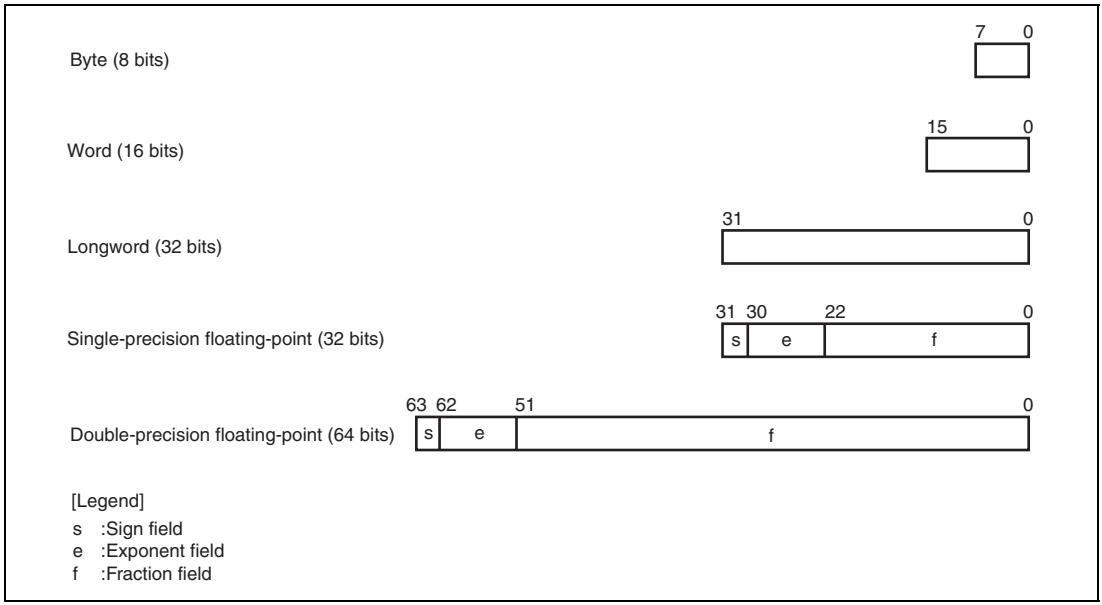


Figure 2.2 Data Formats

2.2.2 Register Descriptions

(1) Privileged Mode and Banks

(a) Processing Modes

This LSI has two processing modes, user mode and privileged mode. This LSI normally operates in user mode, and switches to privileged mode when an exception occurs or an interrupt is accepted. There are four kinds of registers—general registers, system registers, control registers, and floating-point registers—and the registers that can be accessed differ in the two processing modes.

(b) General Registers

There are 16 general registers, designated R0 to R15. General registers R0 to R7 are banked registers which are switched by a processing mode change.

- Privileged mode

  In privileged mode, the register bank bit (RB) in the status register (SR) defines which banked register set is accessed as general registers, and which set is accessed only through the load control register (LDC) and store control register (STC) instructions.

  When the RB bit is 1 (that is, when bank 1 is selected), the 16 registers comprising bank 1 general registers R0_BANK1 to R7_BANK1 and non-banked general registers R8 to R15 can be accessed as general registers R0 to R15. In this case, the eight registers comprising bank 0 general registers R0_BANK0 to R7_BANK0 are accessed by the LDC/STC instructions.
  When the RB bit is 0 (that is, when bank 0 is selected), the 16 registers comprising bank 0 general registers R0_BANK0 to R7_BANK0 and non-banked general registers R8 to R15 can be accessed as general registers R0 to R15. In this case, the eight registers comprising bank 1 general registers R0_BANK1 to R7_BANK1 are accessed by the LDC/STC instructions.
- User mode

  In user mode, the 16 registers comprising bank 0 general registers R0_BANK0 to R7_BANK0 and non-banked general registers R8 to R15 can be accessed as general registers R0 to R15. The eight registers comprising bank 1 general registers R0_BANK1 to R7_BANK1 cannot be accessed.

(c) Control Registers

Control registers comprise the global base register (GBR) and status register (SR), which can be accessed in both processing modes, and the saved status register (SSR), saved program counter (SPC), vector base register (VBR), saved general register 15 (SGR), and debug base register

(DBR), which can only be accessed in privileged mode. Some bits of the status register (such as the RB bit) can only be accessed in privileged mode.

(d) System Registers

System registers comprise the multiply-and-accumulate registers (MACH/MACL), the procedure register (PR), and the program counter (PC). Access to these registers does not depend on the processing mode.

(e) Floating-Point Registers and System Registers Related to FPU

There are thirty-two floating-point registers, FR0–FR15 and XF0–XF15. FR0–FR15 and XF0–XF15 can be assigned to either of two banks (FPR0_BANK0–FPR15_BANK0 or FPR0_BANK1–FPR15_BANK1).

FR0–FR15 can be used as the eight registers DR0/2/4/6/8/10/12/14 (double-precision floating-point registers, or pair registers) or the four registers FV0/4/8/12 (register vectors), while XF0–XF15 can be used as the eight registers XD0/2/4/6/8/10/12/14 (register pairs) or register matrix XMTRX.

System registers related to the FPU comprise the floating-point communication register (FPUL) and the floating-point status/control register (FPSCR). These registers are used for communication between the FPU and the CPU, and the exception handling setting.

Register values after a reset are shown in table 2.4.

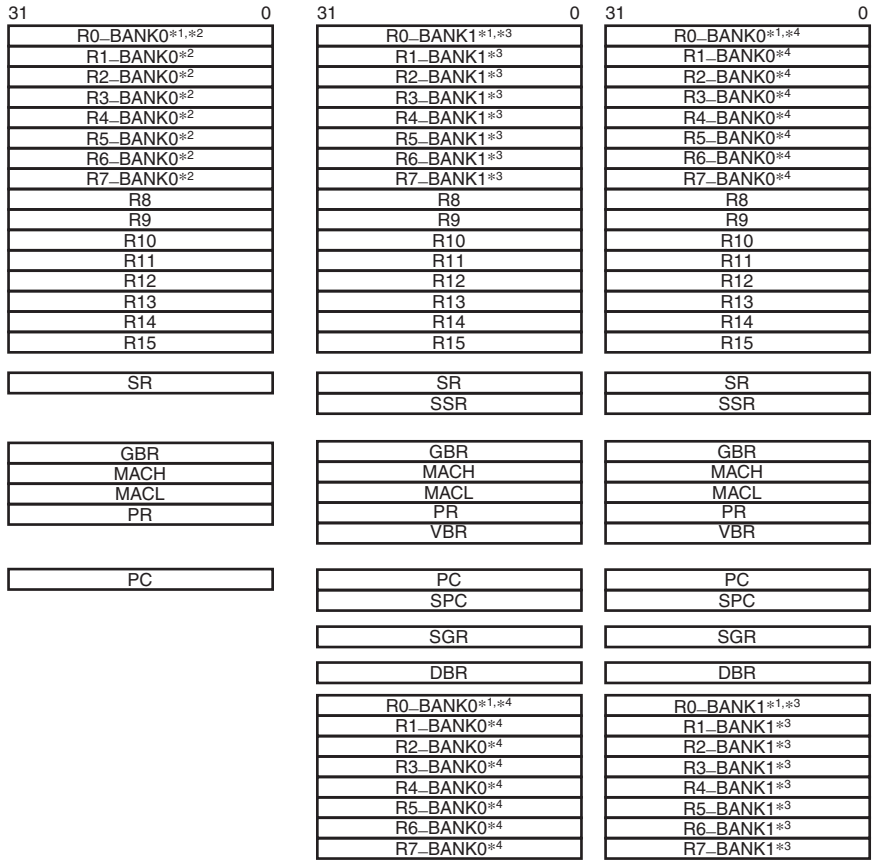
Table 2.4 Initial Register Values

Type	Registers	Initial Value*
General registers	R0_BANK0 to R7_BANK0, R0_BANK1 to R7_BANK1, R8 to R15	Undefined
Control registers	SR	MD bit = 1, RB bit = 1, BL bit = 1, FD bit = 0, IMASK = B'1111, others including reserved bits = 0
	GBR, SSR, SPC, SGR, DBR	Undefined
	VBR	H'00000000
System registers	MACH, MACL, PR	Undefined
	PC	H'A0000000
Floating-point registers	FR0 to FR15, XF0 to XF15, FPUL	Undefined
	FPSCR	H'00040001

Note: * Initialized by a power-on reset and manual reset.

The CPU register configuration in each processing mode is shown in figure 2.3.

User mode and privileged mode are switched by the processing mode bit (MD) in the status register.



Notes: 1. R0 is used as the index register in indexed register-indirect addressing mode and indexed GBR indirect addressing mode.
2. Banked registers
3. Banked registers
Accessed as general registers when the RB bit is set to 1 in SR. Accessed only by LDC/STC instructions when the RB bit is cleared to 0.
4. Banked registers
Accessed as general registers when the RB bit is cleared to 0 in SR. Accessed only by LDC/STC instructions when the RB bit is set to 1.

Figure 2.3 CPU Register Configuration in Each Processing Mode

(2) General Registers

Figure 2.4 shows the relationship between the processing modes and general registers. The SH-4A has twenty-four 32-bit general registers (R0_BANK0 to R7_BANK0, R0_BANK1 to R7_BANK1, and R8 to R15). However, only 16 of these can be accessed as general registers R0 to R15 in one processing mode. The SH-4A has two processing modes, user mode and privileged mode.

- R0_BANK0 to R7_BANK0
  Allocated to R0 to R7 in user mode (SR.MD = 0)
  Allocated to R0 to R7 when SR.RB = 0 in privileged mode (SR.MD = 1).
- R0_BANK1 to R7_BANK1
  Cannot be accessed in user mode.
  Allocated to R0 to R7 when SR.RB = 1 in privileged mode.

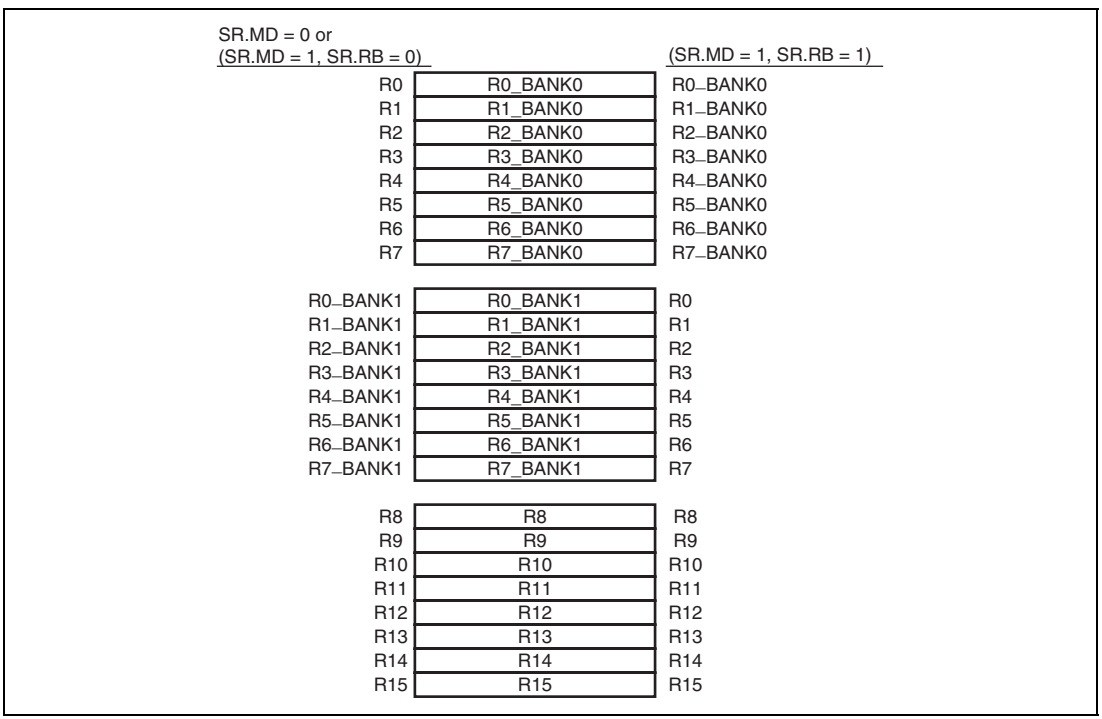


Figure 2.4 General Registers

Note on Programming: As the user's R0 to R7 are assigned to R0_BANK0 to R7_BANK0, and after an exception or interrupt R0 to R7 are assigned to R0_BANK1 to R7_BANK1, it is not necessary for the interrupt handler to save and restore the user's R0 to R7 (R0_BANK0 to R7_BANK0).

(3) Floating-Point Registers

Figure 2.5 shows the floating-point register configuration. There are thirty-two 32-bit floating-point registers, FPR0_BANK0 to FPR15_BANK0, AND FPR0_BANK1 to FPR15_BANK1, comprising two banks. These registers are referenced as FR0 to FR15, DR0/2/4/6/8/10/12/14, FV0/4/8/12, XF0 to XF15, XD0/2/4/6/8/10/12/14, or XMTRX. Reference names of each register are defined depending on the state of the FR bit in FPSCR (see figure 2.5).

1. Floating-point registers, FPRn_BANKj (32 registers)
   FPR0_BANK0 to FPR15_BANK0
   FPR0_BANK1 to FPR15_BANK1
2. Single-precision floating-point registers, FRi (16 registers)
   When FPSCR.FR = 0, FR0 to FR15 are assigned to FPR0_BANK0 to FPR15_BANK0;
   when FPSCR.FR = 1, FR0 to FR15 are assigned to FPR0_BANK1 to FPR15_BANK1.
3. Double-precision floating-point registers or single-precision floating-point registers, DRi (8 registers): A DR register comprises two FR registers.
   DR0 = {FR0, FR1}, DR2 = {FR2, FR3}, DR4 = {FR4, FR5}, DR6 = {FR6, FR7},
   DR8 = {FR8, FR9}, DR10 = {FR10, FR11}, DR12 = {FR12, FR13}, DR14 = {FR14, FR15}
4. Single-precision floating-point vector registers, FVi (4 registers): An FV register comprises four FR registers.
   FV0 = {FR0, FR1, FR2, FR3}, FV4 = {FR4, FR5, FR6, FR7},
   FV8 = {FR8, FR9, FR10, FR11}, FV12 = {FR12, FR13, FR14, FR15}
5. Single-precision floating-point extended registers, XFi (16 registers)
   When FPSCR.FR = 0, XF0 to XF15 are assigned to FPR0_BANK1 to FPR15_BANK1;
   when FPSCR.FR = 1, XF0 to XF15 are assigned to FPR0_BANK0 to FPR15_BANK0.
6. Double-precision floating-point extended registers, XDi (8 registers): An XD register comprises two XF registers.
   XD0 = {XF0, XF1}, XD2 = {XF2, XF3}, XD4 = {XF4, XF5}, XD6 = {XF6, XF7},
   XD8 = {XF8, XF9}, XD10 = {XF10, XF11}, XD12 = {XF12, XF13}, XD14 = {XF14, XF15}

7. Single-precision floating-point extended register matrix, XMTRX: XMTRX comprises all 16 XF registers.

$$XMTRX = \begin{bmatrix} XF0 & XF4 & XF8 & XF12 \\ XF1 & XF5 & XF9 & XF13 \\ XF2 & XF6 & XF10 & XF14 \\ XF3 & XF7 & XF11 & XF15 \end{bmatrix}$$

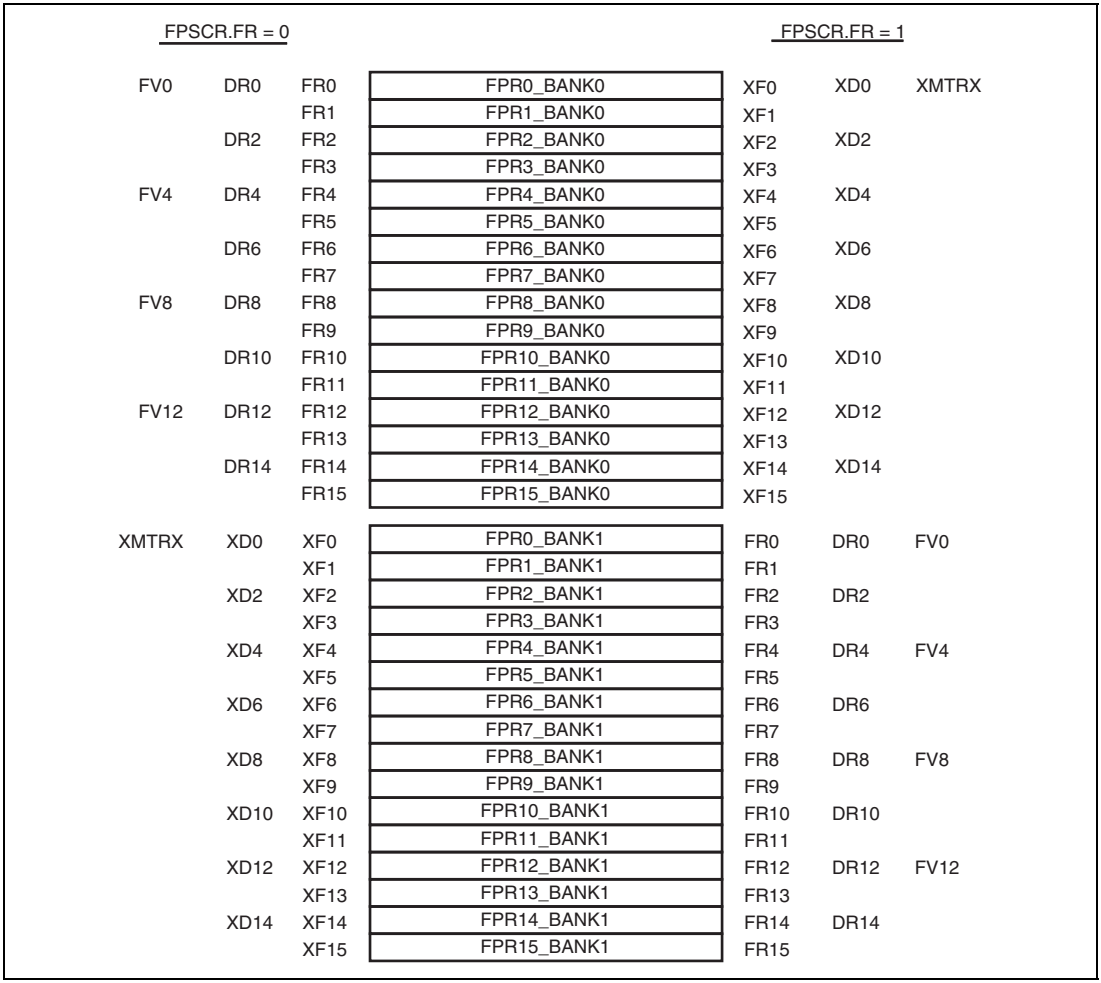


Figure 2.5 Floating-Point Registers

(4) Control Registers

(a) Status Register (SR)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	MD	RB	BL	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FD	—	—	—	—	—	M	Q	IMASK				—	—	S	T
Initial value:	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0
R/W:	R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved For details on reading/writing this bit, see General Precautions on Handling of Product.
30	MD	1	R/W	Processing Mode Selects the processing mode. 0: User mode (Some instructions cannot be executed and some resources cannot be accessed.) 1: Privileged mode This bit is set to 1 by an exception or interrupt.
29	RB	1	R/W	Privileged Mode General Register Bank Specification Bit 0: R0_BANK0 to R7_BANK0 are accessed as general registers R0 to R7 and R0_BANK1 to R7_BANK1 can be accessed using LDC/STC instructions 1: R0_BANK1 to R7_BANK1 are accessed as general registers R0 to R7 and R0_BANK0–R7_BANK0 can be accessed using LDC/STC instructions This bit is set to 1 by an exception or interrupt.
28	BL	1	R/W	Exception/Interrupt Block Bit This bit is set to 1 by a reset, a general exception, or an interrupt. While this bit is set to 1, an interrupt request is masked. In this case, this processor enters the reset state when a general exception other than a user break occurs.

Bit	Bit Name	Initial Value	R/W	Description
27 to 16	—	All 0	R	Reserved For details on reading/writing this bit, see General Precautions on Handling of Product.
15	FD	0	R/W	FPU Disable Bit When this bit is set to 1 and an FPU instruction is not in a delay slot, a general FPU disable exception occurs. When this bit is set to 1 and an FPU instruction is in a delay slot, a slot FPU disable exception occurs. (FPU instructions: H'F*** instructions and LDS (.L)/STS(.L) instructions using FPUL/FPSCR)
14 to 10	—	All 0	R	Reserved For details on reading/writing this bit, see General Precautions on Handling of Product.
9	M	0	R/W	M Bit Used by the DIV0S, DIV0U, and DIV1 instructions.
8	Q	0	R/W	Q Bit Used by the DIV0S, DIV0U, and DIV1 instructions.
7 to 4	IMASK	1111	R/W	Interrupt Mask Level Bits An interrupt whose priority is equal to or less than the value of the IMASK bits is masked. It can be chosen by CPU operation mode register (CPUOPM) whether the level of IMASK is changed to accept an interrupt or not when an interrupt is occurred.
3, 2	—	All 0	R	Reserved For details on reading/writing this bit, see General Precautions on Handling of Product.
1	S	0	R/W	S Bit Used by the MAC instruction.
0	T	0	R/W	T Bit Indicates true/false condition, carry/borrow, or overflow/underflow. For details, see section 2.3, Instruction Set.

(b) Saved Status Register (SSR) (32 bits, Privileged Mode, Initial Value = Undefined)

The contents of SR are saved to SSR in the event of an exception or interrupt.

(c) Saved Program Counter (SPC) (32 bits, Privileged Mode, Initial Value = Undefined)

The address of an instruction at which an interrupt or exception occurs is saved to SPC.

(d) Global Base Register (GBR) (32 bits, Initial Value = Undefined)

GBR is referenced as the base address of addressing @(disp,GBR) and @(R0,GBR).

(e) Vector Base Register (VBR) (32 bits, Privileged Mode, Initial Value = H'00000000)

VBR is referenced as the branch destination base address in the event of an exception or interrupt. For details, see section 2.5, Exception Handling.

(f) Saved General Register 15 (SGR) (32 bits, Privileged Mode, Initial Value = Undefined)

The contents of R15 are saved to SGR in the event of an exception or interrupt.

(g) Debug Base Register (DBR) (32 bits, Privileged Mode, Initial Value = Undefined)

When the user break debugging function is enabled (CBCR.UBDE = 1), DBR is referenced as the branch destination address of the user break handler instead of VBR.

(5) System Registers

(a) Multiply-and-Accumulate Registers (MACH and MACL) (32 bits, Initial Value = Undefined)

MACH and MACL are used for the added value in a MAC instruction, and to store the operation result of a MAC or MUL instruction.

(b) Procedure Register (PR) (32 bits, Initial Value = Undefined)

The return address is stored in PR in a subroutine call using a BSR, BSRF, or JSR instruction. PR is referenced by the subroutine return instruction (RTS).

(c) Program Counter (PC) (32 bits, Initial Value = H'A0000000)

PC indicates the address of the instruction currently being executed.

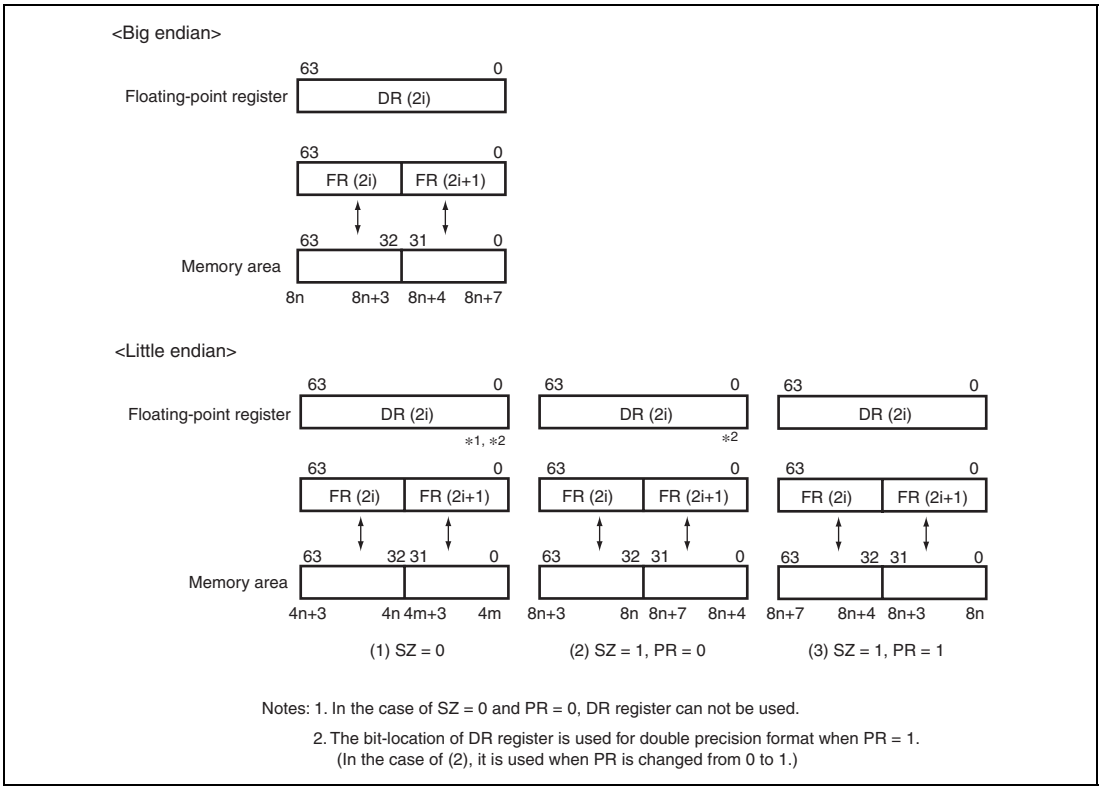
(d) Floating-Point Status/Control Register (FPSCR)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	FR	SZ	PR	DN	Cause	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Cause				Enable (EN)					Flag					RM	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved For details on reading/writing this bit, see General Precautions on Handling of Product.
21	FR	0	R/W	Floating-Point Register Bank 0: FPR0_BANK0 to FPR15_BANK0 are assigned to FR0 to FR15 and FPR0_BANK1 to FPR15_BANK1 are assigned to XF0 to XF15 1: FPR0_BANK0 to FPR15_BANK0 are assigned to XF0 to XF15 and FPR0_BANK1 to FPR15_BANK1 are assigned to FR0 to FR15
20	SZ	0	R/W	Transfer Size Mode 0: Data size of FMOV instruction is 32-bits 1: Data size of FMOV instruction is a 32-bit register pair (64 bits) For relationship between the SZ bit, PR bit, and endian, see figure 2.6.
19	PR	0	R/W	Precision Mode 0: Floating-point instructions are executed as single-precision operations 1: Floating-point instructions are executed as double-precision operations (graphics support instructions are undefined) For relationship between the SZ bit, PR bit, and endian, see figure 2.6.
18	DN	1	R/W	Denormalization Mode 0: Denormalized number is treated as such 1: Denormalized number is treated as zero

Bit	Bit Name	Initial Value	R/W	Description
17 to 12	Cause	000000	R/W	FPU Exception Cause Field FPU Exception Enable Field FPU Exception Flag Field Each time an FPU operation instruction is executed, the FPU exception cause field is cleared to 0. When an FPU exception occurs, the bits corresponding to FPU exception cause field and flag field are set to 1. The FPU exception flag field remains set to 1 until it is cleared to 0 by software. For bit allocations of each field, see table 2.5.
11 to 7	Enable (EN)	00000	R/W	
6 to 2	Flag	00000	R/W	
1, 0	RM	01	R/W	Rounding Mode These bits select the rounding mode. 00: Round to Nearest 01: Round to Zero 10: Reserved (setting prohibited) 11: Reserved (setting prohibited)



Notes: 1. In the case of SZ = 0 and PR = 0, DR register can not be used.

2. The bit-location of DR register is used for double precision format when PR = 1.
(In the case of (2), it is used when PR is changed from 0 to 1.)

Figure 2.6 Relationship between SZ bit and Endian

Table 2.5 Bit Allocation for FPU Exception Handling

Field Name		FPU Error (E)	Invalid Operation (V)	Division by Zero (Z)	Overflow (O)	Underflow (U)	Inexact (I)
Cause	FPU exception cause field	Bit 17	Bit 16	Bit 15	Bit 14	Bit 13	Bit 12
Enable	FPU exception enable field	None	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7
Flag	FPU exception flag field	None	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2

(e) Floating-Point Communication Register (FPUL) (32 bits, Initial Value = Undefined)

Information is transferred between the FPU and CPU via FPUL.

2.2.3 Memory-Mapped Registers

Some control registers are mapped to the following memory areas. Each of the mapped registers has two addresses.

H'1C00 0000 to H'1FFF FFFF
H'FC00 0000 to H'FFFF FFFF

These two areas are used as follows.

- H'1C00 0000 to H'1FFF FFFF

  This area must be accessed using the address translation function of the MMU.

  Setting the page number of this area to the corresponding field of the TLB enables access to a memory-mapped register.

  The operation of an access to this area without using the address translation function of the MMU is not guaranteed.
- H'FC00 0000 to H'FFFF FFFF

  Access to area H'FC00 0000 to H'FFFF FFFF in user mode will cause an address error. Memory-mapped registers can be referenced in user mode by means of access that involves address translation.

Note: Do not access addresses to which registers are not mapped in either area. The operation of an access to an address with no register mapped is undefined. Also, memory-mapped registers must be accessed using a fixed data size. The operation of an access using an invalid data size is undefined.

2.2.4 Data Formats in Registers

Register operands are always longwords (32 bits). When a memory operand is only a byte (8 bits) or a word (16 bits), it is sign-extended into a longword when loaded into a register.

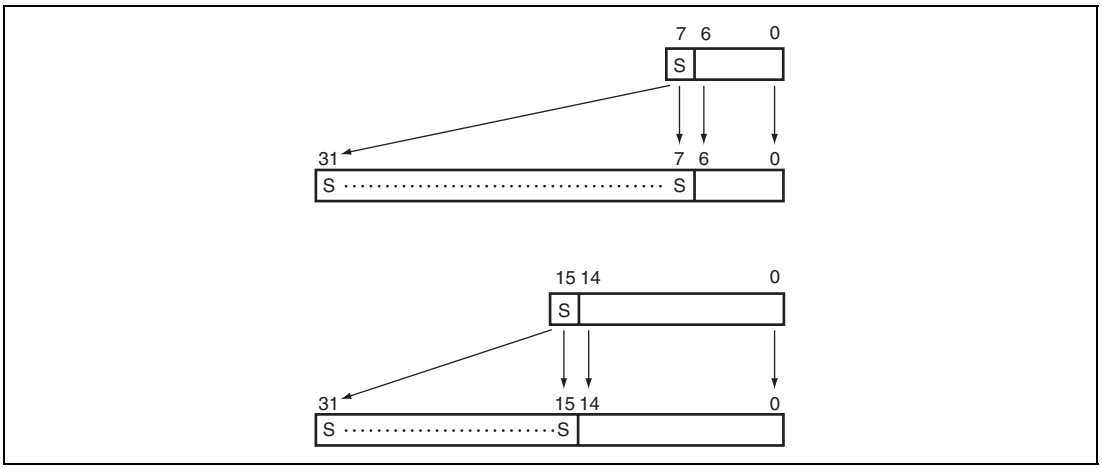


Figure 2.7 Formats of Byte Data and Word Data in Register

2.2.5 Data Formats in Memory

Memory data formats are classified into bytes, words, and longwords. Memory can be accessed in an 8-bit byte, 16-bit word, or 32-bit longword form. A memory operand less than 32 bits in length is sign-extended before being loaded into a register.

A word operand must be accessed starting from a word boundary (even address of a 2-byte unit: address 2n), and a longword operand starting from a longword boundary (even address of a 4-byte unit: address 4n). An address error will result if this rule is not observed. A byte operand can be accessed from any address.

Big endian or little endian byte order can be selected for the data format. The endian should be set with the external pin after a power-on reset. The endian cannot be changed dynamically. Bit positions are numbered left to right from most-significant to least-significant. Thus, in a 32-bit longword, the leftmost bit, bit 31, is the most significant bit and the rightmost bit, bit 0, is the least significant bit.

The data format in memory is shown in figure 2.8.

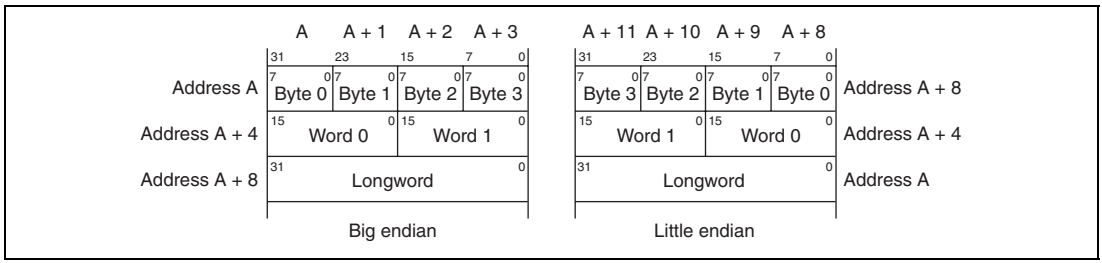


Figure 2.8 Data Formats in Memory

For the 64-bit data format, see figure 2.6.

2.2.6 Processing States

This LSI has major three processing states: the reset state, instruction execution state, and power-down state.

(1) Reset State

In this state the CPU is reset. The reset state is divided into the power-on reset state and the manual reset.

In the power-on reset state, the internal state of the CPU and the on-chip peripheral module registers are initialized. In the manual reset state, the internal state of the CPU and some registers of on-chip peripheral modules are initialized. For details, see register descriptions for each section of the user's manual of the product.

(2) Instruction Execution State

In this state, the CPU executes program instructions in sequence. The instruction execution state has the normal program execution state and the exception handling state.

(3) Power-Down State

In a power-down state, CPU halts operation and power consumption is reduced. The power-down state is entered by executing a SLEEP instruction. The power-down state has the sleep mode, software standby mode and deep standby mode. For details, see section 9, Operating Modes and Power-Down Modes.

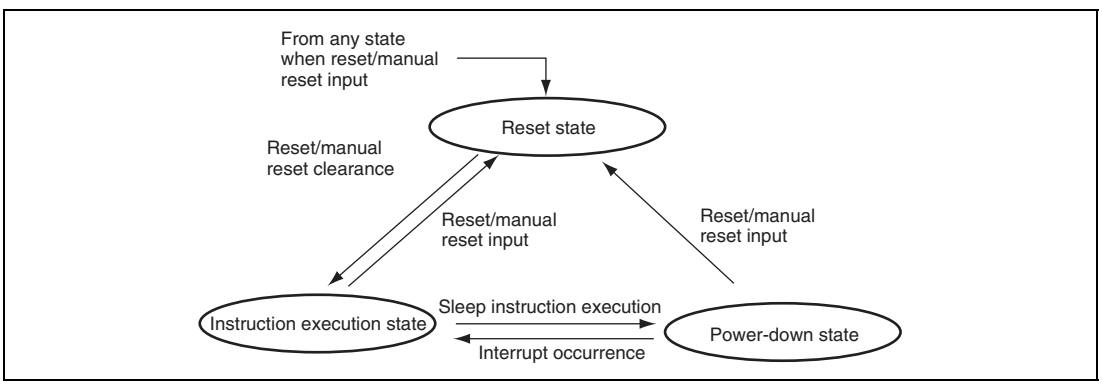


Figure 2.9 Processing State Transitions

2.2.7 Usage Notes

(1) Notes on Self-Modifying Code

To accelerate the processing speed, the instruction prefetching capability of the SH-4A has been significantly enhanced from that of the SH-4. Therefore, in the case when a code in memory is rewritten and attempted to be executed immediately, there is increased possibility that the code before being modified, which has already been prefetched, is executed.

To ensure execution of the modified code, one of the following sequence of instructions should be executed between the code rewriting instruction and execution of the modified code.

(a) When the Codes to be Modified are in Non-Cacheable Area

```
SYNCO
ICBI @Rn
```

The target for the ICBI instruction can be any address within the range where no address error exception occurs.

(b) When the Codes to be Modified are in Cacheable Area (Write-Through)

```
SYNCO
ICBI @Rn
```

All instruction cache areas corresponding to the modified codes should be invalidated by the ICBI instruction. The ICBI instruction should be issued to each cache line. One cache line is 32 bytes.

(c) When the Codes to be Modified are in Cacheable Area (Copy-Back)

```
OCBP @Rm or OCBWB @Rm
SYNCO
ICBI @Rn
```

All operand cache areas corresponding to the modified codes should be written back to the main memory by the OCBP or OCBWB instruction. Then all instruction cache areas corresponding to the modified codes should be invalidated by the ICBI instruction. The OCBP, OCBWB, and ICBI instruction should be issued to each cache line. One cache line is 32 bytes.

Note: Self-modifying code is the processing which executes instructions while dynamically rewriting the codes in memory.

2.3 Instruction Set

The SH-4A's instruction set is implemented with 16-bit fixed-length instructions. The SH-4A can use byte (8-bit), word (16-bit), longword (32-bit), and quadword (64-bit) data sizes for memory access. Single-precision floating-point data (32 bits) can be moved to and from memory using longword or quadword size. Double-precision floating-point data (64 bits) can be moved to and from memory using longword size. When the SH-4A moves byte-size or word-size data from memory to a register, the data is sign-extended.

2.3.1 Execution Environment

(1) PC

At the start of instruction execution, the PC indicates the address of the instruction itself.

(2) Load-Store Architecture

The SH-4A has a load-store architecture in which operations are basically executed using registers. Except for bit-manipulation operations such as logical AND that are executed directly in memory, operands in an operation that requires memory access are loaded into registers and the operation is executed between the registers.

(3) Delayed Branches

Except for the two branch instructions BF and BT, the SH-4A's branch instructions and RTE are delayed branches. In a delayed branch, the instruction following the branch is executed before the branch destination instruction.

(4) Delay Slot

This execution slot following a delayed branch is called a delay slot. For example, the BRA execution sequence is as follows:

Table 2.6 Execution Order of Delayed Branch Instructions

	Instructions			Execution Order
	BRA	TARGET	(Delayed branch instruction)	BRA
	ADD		(Delay slot)	↓
	:			ADD
	:			↓
TARGET	target-inst		(Branch destination instruction)	target-inst

A slot illegal instruction exception may occur when a specific instruction is executed in a delay slot. For details, see section 2.5, Exception Handling. The instruction following BF/S or BT/S for which the branch is not taken is also a delay slot instruction.

(5) T Bit

The T bit in SR is used to show the result of a compare operation, and is referenced by a conditional branch instruction. An example of the use of a conditional branch instruction is shown below.

```
ADD     #1, R0      ; T bit is not changed by ADD operation
CMP/EQ  R1, R0      ; If R0 = R1, T bit is set to 1
BT      TARGET      ; Branches to TARGET if T bit = 1 (R0 = R1)
```

In an RTE delay slot, the SR bits are referenced as follows. In instruction access, the MD bit is used before modification, and in data access, the MD bit is accessed after modification. The other bits—S, T, M, Q, FD, BL, and RB—after modification are used for delay slot instruction execution. The STC and STC.L SR instructions access all SR bits after modification.

(6) Constant Values

An 8-bit constant value can be specified by the instruction code and an immediate value. 16-bit and 32-bit constant values can be defined as literal constant values in memory, and can be referenced by a PC-relative load instruction.

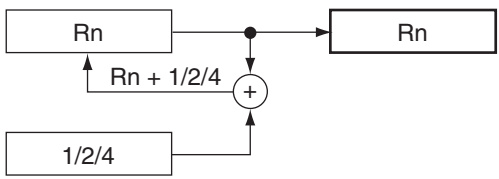
```
MOV.W   @(disp, PC), Rn
MOV.L   @(disp, PC), Rn
```

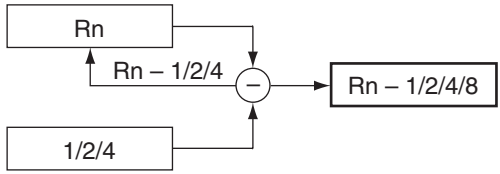
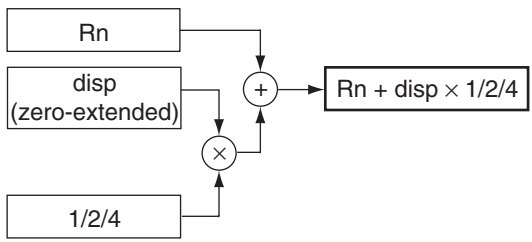
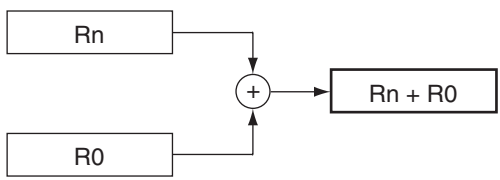
There are no PC-relative load instructions for floating-point operations. However, it is possible to set 0.0 or 1.0 by using the FLDI0 or FLDI1 instruction on a single-precision floating-point register.

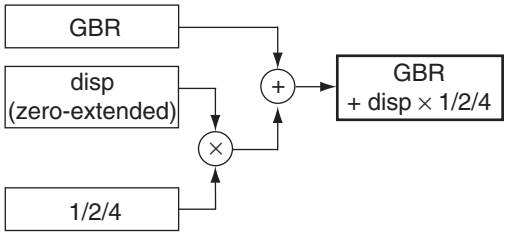
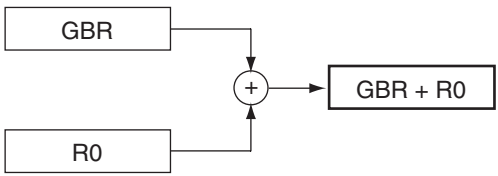
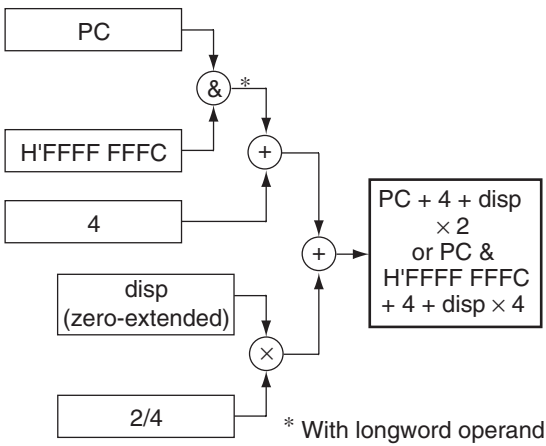
2.3.2 Addressing Modes

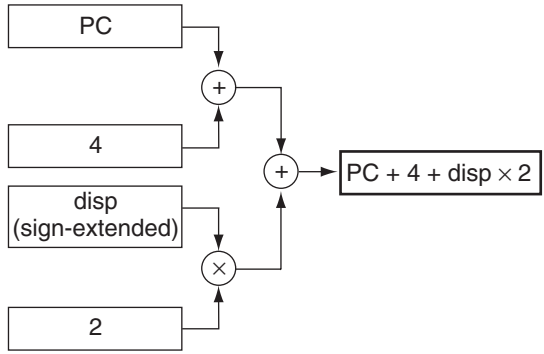
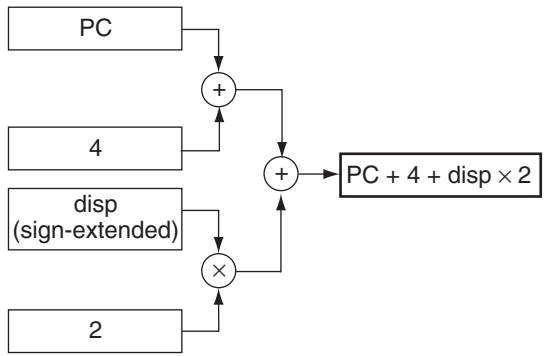
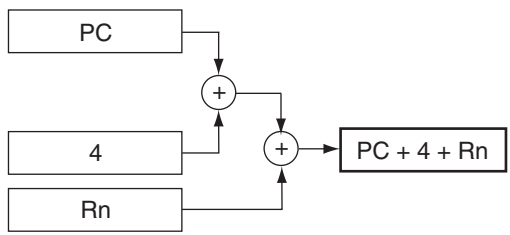
Addressing modes and effective address calculation methods are shown in table 2.7. When a location in virtual memory space is accessed (AT in MMUCR = 1), the effective address is translated into a physical memory address. If multiple virtual memory space systems are selected (SV in MMUCR = 0), the least significant bit of PTEH is also referenced as the access ASID. For details, see section 2.7, Memory Management Unit (MMU).

Table 2.7 Addressing Modes and Effective Addresses

Addressing Mode	Instruction Format	Effective Address Calculation Method	Calculation Formula
Register direct	Rn	Effective address is register Rn. (Operand is register Rn contents.)	—
Register indirect	@Rn	Effective address is register Rn contents. 	Rn → EA (EA: effective address)
Register indirect with post-increment	@Rn+	Effective address is register Rn contents. A constant is added to Rn after instruction execution: 1 for a byte operand, 2 for a word operand, 4 for a longword operand, 8 for a quadword operand. 	Rn → EA After instruction execution Byte: Rn + 1 → Rn Word: Rn + 2 → Rn Longword: Rn + 4 → Rn Quadword: Rn + 8 → Rn

Addressing Mode	Instruction Format	Effective Address Calculation Method	Calculation Formula
Register indirect with pre-decrement	@–Rn	Effective address is register Rn contents, decremented by a constant beforehand: 1 for a byte operand, 2 for a word operand, 4 for a longword operand, 8 for a quadword operand. 	Byte: Rn – 1 → Rn Word: Rn – 2 → Rn Longword: Rn – 4 → Rn Quadword: Rn – 8 → Rn Rn → EA (Instruction executed with Rn after calculation)
Register indirect with displacement	@(disp:4, Rn)	Effective address is register Rn contents with 4-bit displacement disp added. After disp is zero-extended, it is multiplied by 1 (byte), 2 (word), or 4 (longword), according to the operand size. 	Byte: Rn + disp → EA Word: Rn + disp × 2 → EA Longword: Rn + disp × 4 → EA
Indexed register indirect	@(R0, Rn)	Effective address is sum of register Rn and R0 contents. 	Rn + R0 → EA

Addressing Mode	Instruction Format	Effective Address Calculation Method	Calculation Formula
GBR indirect with displacement	@(disp:8, GBR)	Effective address is register GBR contents with 8-bit displacement disp added. After disp is zero-extended, it is multiplied by 1 (byte), 2 (word), or 4 (longword), according to the operand size. 	Byte: GBR + disp → EA Word: GBR + disp × 2 → EA Longword: GBR + disp × 4 → EA
Indexed GBR indirect	@(R0, GBR)	Effective address is sum of register GBR and R0 contents. 	GBR + R0 → EA
PC-relative with displacement	@(disp:8, PC)	Effective address is PC + 4 with 8-bit displacement disp added. After disp is zero-extended, it is multiplied by 2 (word), or 4 (longword), according to the operand size. With a longword operand, the lower 2 bits of PC are masked. 	Word: PC + 4 + disp × 2 → EA Longword: PC & H'FFFF FFFC + 4 + disp × 4 → EA

Addressing Mode	Instruction Format	Effective Address Calculation Method	Calculation Formula
PC-relative	disp:8	Effective address is PC + 4 with 8-bit displacement disp added after being sign-extended and multiplied by 2. 	PC + 4 + disp × 2 → Branch-Target
	disp:12	Effective address is PC + 4 with 12-bit displacement disp added after being sign-extended and multiplied by 2. 	PC + 4 + disp × 2 → Branch-Target
	Rn	Effective address is sum of PC + 4 and Rn. 	PC + 4 + Rn → Branch-Target

Addressing Mode	Instruction Format	Effective Address Calculation Method	Calculation Formula
Immediate	#imm:8	8-bit immediate data imm of TST, AND, OR, or XOR instruction is zero-extended.	—
	#imm:8	8-bit immediate data imm of MOV, ADD, or CMP/EQ instruction is sign-extended.	—
	#imm:8	8-bit immediate data imm of TRAPA instruction is zero-extended and multiplied by 4.	—

Note: For the addressing modes below that use a displacement (disp), the assembler descriptions in this manual show the value before scaling (×1, ×2, or ×4) is performed according to the operand size. This is done to clarify the operation of the LSI. Refer to the relevant assembler notation rules for the actual assembler descriptions.

@ (disp:4, Rn) ; Register indirect with displacement

@ (disp:8, GBR) ; GBR indirect with displacement

@ (disp:8, PC) ; PC-relative with displacement

disp:8, disp:12 ; PC-relative

2.3.3 Instruction Set

Table 2.8 shows the notation used in the SH instruction lists shown in tables 2.9 to 2.18.

Table 2.8 Notation Used in Instruction List

Item	Format	Description	
Instruction mnemonic	OP.Sz SRC, DEST	OP:	Operation code
		Sz:	Size
		SRC:	Source operand
		DEST:	Source and/or destination operand
		Rm:	Source register
		Rn:	Destination register
		imm:	Immediate data
		disp:	Displacement
Operation notation		→, ←	Transfer direction
		(xx)	Memory operand
		M/Q/T	SR flag bits
		&	Logical AND of individual bits
		|	Logical OR of individual bits
		^	Logical exclusive-OR of individual bits
		~	Logical NOT of individual bits
		<<n, >>n	n-bit shift
Instruction code	MSB ↔ LSB	mmmm:	Register number (Rm, FRm)
		nnnn:	Register number (Rn, FRn)
		0000:	R0, FR0
		0001:	R1, FR1
		:	
		1111:	R15, FR15
		mmm:	Register number (DRm, XDm, Rm_BANK)
		nnn:	Register number (DRn, XDn, Rn_BANK)
		000:	DR0, XD0, R0_BANK
		001:	DR2, XD2, R1_BANK
		:	
		111:	DR14, XD14, R7_BANK
		mm:	Register number (FVm)
		nn:	Register number (FVn)
		00:	FV0
		01:	FV4
		10:	FV8
		11:	FV12
		iiii:	Immediate data
		dddd:	Displacement

Item	Format	Description
Privileged mode		"Privileged" means the instruction can only be executed in privileged mode.
T bit	Value of T bit after instruction execution	—: No change
New	—	"New" means the instruction which has been newly added in the SH-4A with H'20-valued VER bits in the processor version register (PVR).

Note: Scaling (×1, ×2, ×4, or ×8) is executed according to the size of the instruction operand.

Table 2.9 Fixed-Point Transfer Instructions

Instruction		Operation	Instruction Code	Privileged	T Bit	New
MOV	#imm,Rn	imm → sign extension → Rn	1110nnnniiiiiiii	—	—	—
MOV.W	@(disp*,PC), Rn	(disp × 2 + PC + 4) → sign extension → Rn	1001nnnnddddddd	—	—	—
MOV.L	@(disp*,PC), Rn	(disp × 4 + PC & H'FFFF FFFC + 4) → Rn	1101nnnnddddddd	—	—	—
MOV	Rm,Rn	Rm → Rn	0110nnnnmmmm0011	—	—	—
MOV.B	Rm,@Rn	Rm → (Rn)	0010nnnnmmmm0000	—	—	—
MOV.W	Rm,@Rn	Rm → (Rn)	0010nnnnmmmm0001	—	—	—
MOV.L	Rm,@Rn	Rm → (Rn)	0010nnnnmmmm0010	—	—	—
MOV.B	@Rm,Rn	(Rm) → sign extension → Rn	0110nnnnmmmm0000	—	—	—
MOV.W	@Rm,Rn	(Rm) → sign extension → Rn	0110nnnnmmmm0001	—	—	—
MOV.L	@Rm,Rn	(Rm) → Rn	0110nnnnmmmm0010	—	—	—
MOV.B	Rm,@-Rn	Rn-1 → Rn, Rm → (Rn)	0010nnnnmmmm0100	—	—	—
MOV.W	Rm,@-Rn	Rn-2 → Rn, Rm → (Rn)	0010nnnnmmmm0101	—	—	—
MOV.L	Rm,@-Rn	Rn-4 → Rn, Rm → (Rn)	0010nnnnmmmm0110	—	—	—
MOV.B	@Rm+,Rn	(Rm)→ sign extension → Rn, Rm + 1 → Rm	0110nnnnmmmm0100	—	—	—
MOV.W	@Rm+,Rn	(Rm) → sign extension → Rn, Rm + 2 → Rm	0110nnnnmmmm0101	—	—	—
MOV.L	@Rm+,Rn	(Rm) → Rn, Rm + 4 → Rm	0110nnnnmmmm0110	—	—	—
MOV.B	R0,@(disp*,Rn)	R0 → (disp + Rn)	10000000nnnndddd	—	—	—
MOV.W	R0,@(disp*,Rn)	R0 → (disp × 2 + Rn)	10000001nnnndddd	—	—	—
MOV.L	Rm,@(disp*,Rn)	Rm → (disp × 4 + Rn)	0001nnnnmmmmdddd	—	—	—

Instruction		Operation	Instruction Code	Privileged	T Bit	New
MOV.B	@(disp*,Rm),R0	(disp + Rm) → sign extension → R0	10000100mmmmdddd	—	—	—
MOV.W	@(disp*,Rm),R0	(disp × 2 + Rm) → sign extension → R0	10000101mmmmdddd	—	—	—
MOV.L	@(disp*,Rm),Rn	(disp × 4 + Rm) → Rn	0101nnnnmmmmdddd	—	—	—
MOV.B	Rm,@(R0,Rn)	Rm → (R0 + Rn)	0000nnnnmmmm0100	—	—	—
MOV.W	Rm,@(R0,Rn)	Rm → (R0 + Rn)	0000nnnnmmmm0101	—	—	—
MOV.L	Rm,@(R0,Rn)	Rm → (R0 + Rn)	0000nnnnmmmm0110	—	—	—
MOV.B	@(R0,Rm),Rn	(R0 + Rm) → sign extension → Rn	0000nnnnmmmm1100	—	—	—
MOV.W	@(R0,Rm),Rn	(R0 + Rm) → sign extension → Rn	0000nnnnmmmm1101	—	—	—
MOV.L	@(R0,Rm),Rn	(R0 + Rm) → Rn	0000nnnnmmmm1110	—	—	—
MOV.B	R0,@(disp*,GBR)	R0 → (disp + GBR)	11000000dddddddd	—	—	—
MOV.W	R0,@(disp*,GBR)	R0 → (disp × 2 + GBR)	11000001dddddddd	—	—	—
MOV.L	R0,@(disp*,GBR)	R0 → (disp × 4 + GBR)	11000010dddddddd	—	—	—
MOV.B	@(disp*,GBR),R0	(disp + GBR) → sign extension → R0	11000100dddddddd	—	—	—
MOV.W	@(disp*,GBR),R0	(disp × 2 + GBR) → sign extension → R0	11000101dddddddd	—	—	—
MOV.L	@(disp*,GBR),R0	(disp × 4 + GBR) → R0	11000110dddddddd	—	—	—
MOVA	@(disp*,PC),R0	disp × 4 + PC & H'FFFF FFFC + 4 → R0	11000111dddddddd	—	—	—
MOVCO.L	R0,@Rn	LDST → T If (T == 1) R0 → (Rn) 0 → LDST	0000nnnn01110011	—	LDST	New
MOVLI.L	@Rm,R0	1 → LDST (Rm) → R0 When interrupt/exception occurred 0 → LDST	0000mmmm01100011	—	—	New
MOVUA.L	@Rm,R0	(Rm) → R0 Load non-boundary alignment data	0100mmmm10101001	—	—	New
MOVUA.L	@Rm+,R0	(Rm) → R0, Rm + 4 → Rm Load non-boundary alignment data	0100mmmm11101001	—	—	New

Instruction		Operation	Instruction Code	Privileged	T Bit	New
MOVT	Rn	T → Rn	0000nnnn00101001	—	—	—
SWAP.B	Rm,Rn	Rm → swap lower 2 bytes → Rn	0110nnnnmmmm1000	—	—	—
SWAP.W	Rm,Rn	Rm → swap upper/lower words → Rn	0110nnnnmmmm1001	—	—	—
XTRCT	Rm,Rn	Rm:Rn middle 32 bits → Rn	0010nnnnmmmm1101	—	—	—

Note: * The assembler of Renesas uses the value after scaling (×1, ×2, or ×4) as the displacement (disp).

Table 2.10 Arithmetic Operation Instructions

Instruction		Operation	Instruction Code	Privileged	T Bit	New
ADD	Rm,Rn	Rn + Rm → Rn	0011nnnnmmmm1100	—	—	—
ADD	#imm,Rn	Rn + imm → Rn	0111nnnniiiiiiii	—	—	—
ADDC	Rm,Rn	Rn + Rm + T → Rn, carry → T	0011nnnnmmmm1110	—	Carry	—
ADDV	Rm,Rn	Rn + Rm → Rn, overflow → T	0011nnnnmmmm1111	—	Overflow	—
CMP/EQ	#imm,R0	When R0 = imm, 1 → T Otherwise, 0 → T	10001000iiiiiiii	—	Comparison result	—
CMP/EQ	Rm,Rn	When Rn = Rm, 1 → T Otherwise, 0 → T	0011nnnnmmmm0000	—	Comparison result	—
CMP/HS	Rm,Rn	When Rn ≥ Rm (unsigned), 1 → T Otherwise, 0 → T	0011nnnnmmmm0010	—	Comparison result	—
CMP/GE	Rm,Rn	When Rn ≥ Rm (signed), 1 → T Otherwise, 0 → T	0011nnnnmmmm0011	—	Comparison result	—
CMP/HI	Rm,Rn	When Rn > Rm (unsigned), 1 → T Otherwise, 0 → T	0011nnnnmmmm0110	—	Comparison result	—
CMP/GT	Rm,Rn	When Rn > Rm (signed), 1 → T Otherwise, 0 → T	0011nnnnmmmm0111	—	Comparison result	—
CMP/PZ	Rn	When Rn ≥ 0, 1 → T Otherwise, 0 → T	0100nnnn00010001	—	Comparison result	—
CMP/PL	Rn	When Rn > 0, 1 → T Otherwise, 0 → T	0100nnnn00010101	—	Comparison result	—

Instruction		Operation	Instruction Code	Privileged	T Bit	New
CMP/STR	Rm,Rn	When any bytes are equal, 1 → T Otherwise, 0 → T	0010nnnnmmmm1100	—	Comparison result	—
DIV1	Rm,Rn	1-step division (Rn ÷ Rm)	0011nnnnmmmm0100	—	Calculation result	—
DIV0S	Rm,Rn	MSB of Rn → Q, MSB of Rm → M, M^Q → T	0010nnnnmmmm0111	—	Calculation result	—
DIV0U		0 → M/Q/T	0000000000011001	—	0	—
DMULS.L	Rm,Rn	Signed, Rn × Rm → MAC, 32 × 32 → 64 bits	0011nnnnmmmm1101	—	—	—
DMULU.L	Rm,Rn	Unsigned, Rn × Rm → MAC, 32 × 32 → 64 bits	0011nnnnmmmm0101	—	—	—
DT	Rn	Rn – 1 → Rn; when Rn = 0, 1 → T When Rn ≠ 0, 0 → T	0100nnnn00010000	—	Comparison result	—
EXTS.B	Rm,Rn	Rm sign-extended from byte → Rn	0110nnnnmmmm1110	—	—	—
EXTS.W	Rm,Rn	Rm sign-extended from word → Rn	0110nnnnmmmm1111	—	—	—
EXTU.B	Rm,Rn	Rm zero-extended from byte → Rn	0110nnnnmmmm1100	—	—	—
EXTU.W	Rm,Rn	Rm zero-extended from word → Rn	0110nnnnmmmm1101	—	—	—
MAC.L	@Rm+,@Rn+	Signed, (Rn) × (Rm) + MAC → MAC Rn + 4 → Rn, Rm + 4 → Rm 32 × 32 + 64 → 64 bits	0000nnnnmmmm1111	—	—	—
MAC.W	@Rm+,@Rn+	Signed, (Rn) × (Rm) + MAC → MAC Rn + 2 → Rn, Rm + 2 → Rm 16 × 16 + 64 → 64 bits	0100nnnnmmmm1111	—	—	—
MUL.L	Rm,Rn	Rn × Rm → MACL 32 × 32 → 32 bits	0000nnnnmmmm0111	—	—	—
MULS.W	Rm,Rn	Signed, Rn × Rm → MACL 16 × 16 → 32 bits	0010nnnnmmmm1111	—	—	—

Instruction		Operation	Instruction Code	Privileged	T Bit	New
MULU.W	Rm,Rn	Unsigned, Rn × Rm → MACL 16 × 16 → 32 bits	0010nnnnmmmm1110	—	—	—
NEG	Rm,Rn	0 – Rm → Rn	0110nnnnmmmm1011	—	—	—
NEGC	Rm,Rn	0 – Rm – T → Rn, borrow → T	0110nnnnmmmm1010	—	Borrow	—
SUB	Rm,Rn	Rn – Rm → Rn	0011nnnnmmmm1000	—	—	—
SUBC	Rm,Rn	Rn – Rm – T → Rn, borrow → T	0011nnnnmmmm1010	—	Borrow	—
SUBV	Rm,Rn	Rn – Rm → Rn, underflow → T	0011nnnnmmmm1011	—	Underflow	—

Table 2.11 Logic Operation Instructions

Instruction		Operation	Instruction Code	Privileged	T Bit	New
AND	Rm,Rn	Rn & Rm → Rn	0010nnnnmmmm1001	—	—	—
AND	#imm,R0	R0 & imm → R0	11001001iiiiiiii	—	—	—
AND.B	#imm, @(R0,GBR)	(R0 + GBR) & imm → (R0 + GBR)	11001101iiiiiiii	—	—	—
NOT	Rm,Rn	~Rm → Rn	0110nnnnmmmm0111	—	—	—
OR	Rm,Rn	Rn | Rm → Rn	0010nnnnmmmm1011	—	—	—
OR	#imm,R0	R0 | imm → R0	11001011iiiiiiii	—	—	—
OR.B	#imm, @(R0,GBR)	(R0 + GBR) | imm → (R0 + GBR)	11001111iiiiiiii	—	—	—
TAS.B	@Rn	When (Rn) = 0, 1 → T Otherwise, 0 → T In both cases, 1 → MSB of (Rn)	0100nnnn00011011	—	Test result	—
TST	Rm,Rn	Rn & Rm; when result = 0, 1 → T Otherwise, 0 → T	0010nnnnmmmm1000	—	Test result	—
TST	#imm,R0	R0 & imm; when result = 0, 1 → T Otherwise, 0 → T	11001000iiiiiiii	—	Test result	—
TST.B	#imm, @(R0,GBR)	(R0 + GBR) & imm; when result = 0, 1 → T Otherwise, 0 → T	11001100iiiiiiii	—	Test result	—
XOR	Rm,Rn	Rn ∧ Rm → Rn	0010nnnnmmmm1010	—	—	—

Instruction		Operation	Instruction Code	Privileged	T Bit	New
XOR	#imm,R0	R0 ∧ imm → R0	11001010iiiiiiii	—	—	—
XOR.B	#imm, @(R0,GBR)	(R0 + GBR) ∧ imm → (R0 + GBR)	11001110iiiiiiii	—	—	—

Table 2.12 Shift Instructions

Instruction		Operation	Instruction Code	Privileged	T Bit	New
ROTL	Rn	T ← Rn ← MSB	0100nnnn00000100	—	MSB	—
ROTR	Rn	LSB → Rn → T	0100nnnn00000101	—	LSB	—
ROTCL	Rn	T ← Rn ← T	0100nnnn00100100	—	MSB	—
ROTCR	Rn	T → Rn → T	0100nnnn00100101	—	LSB	—
SHAD	Rm,Rn	When Rm ≥ 0, Rn << Rm → Rn When Rm < 0, Rn >> Rm → [MSB → Rn]	0100nnnnmmmm1100	—	—	—
SHAL	Rn	T ← Rn ← 0	0100nnnn00100000	—	MSB	—
SHAR	Rn	MSB → Rn → T	0100nnnn00100001	—	LSB	—
SHLD	Rm,Rn	When Rm ≥ 0, Rn << Rm → Rn When Rm < 0, Rn >> Rm → [0 → Rn]	0100nnnnmmmm1101	—	—	—
SHLL	Rn	T ← Rn ← 0	0100nnnn00000000	—	MSB	—
SHLR	Rn	0 → Rn → T	0100nnnn00000001	—	LSB	—
SHLL2	Rn	Rn << 2 → Rn	0100nnnn00001000	—	—	—
SHLR2	Rn	Rn >> 2 → Rn	0100nnnn00001001	—	—	—
SHLL8	Rn	Rn << 8 → Rn	0100nnnn00011000	—	—	—
SHLR8	Rn	Rn >> 8 → Rn	0100nnnn00011001	—	—	—
SHLL16	Rn	Rn << 16 → Rn	0100nnnn00101000	—	—	—
SHLR16	Rn	Rn >> 16 → Rn	0100nnnn00101001	—	—	—

Table 2.13 Branch Instructions

Instruction		Operation	Instruction Code	Privileged	T Bit	New
BF	label	When T = 0, disp × 2 + PC + 4 → PC When T = 1, nop	10001011dddddddd	—	—	—
BF/S	label	Delayed branch; when T = 0, disp × 2 + PC + 4 → PC When T = 1, nop	10001111dddddddd	—	—	—
BT	label	When T = 1, disp × 2 + PC + 4 → PC When T = 0, nop	10001001dddddddd	—	—	—
BT/S	label	Delayed branch; when T = 1, disp × 2 + PC + 4 → PC When T = 0, nop	10001101dddddddd	—	—	—
BRA	label	Delayed branch, disp × 2 + PC + 4 → PC	1010dddddddddddd	—	—	—
BRAF	Rn	Delayed branch, Rn + PC + 4 → PC	0000nnnn00100011	—	—	—
BSR	label	Delayed branch, PC + 4 → PR, disp × 2 + PC + 4 → PC	1011dddddddddddd	—	—	—
BSRF	Rn	Delayed branch, PC + 4 → PR, Rn + PC + 4 → PC	0000nnnn00000011	—	—	—
JMP	@Rn	Delayed branch, Rn → PC	0100nnnn00101011	—	—	—
JSR	@Rn	Delayed branch, PC + 4 → PR, Rn → PC	0100nnnn00001011	—	—	—
RTS		Delayed branch, PR → PC	0000000000001011	—	—	—

Table 2.14 System Control Instructions

Instruction		Operation	Instruction Code	Privileged	T Bit	New
CLRMAC		0 → MACH, MACL	0000000000101000	—	—	—
CLRS		0 → S	0000000001001000	—	—	—
CLRT		0 → T	0000000000001000	—	0	—
ICBI	@Rn	Invalidates instruction cache block	0000nnnn11100011	—	—	New
LDC	Rm,SR	Rm → SR	0100mmmm00001110	Privileged	LSB	—
LDC	Rm,GBR	Rm → GBR	0100mmmm00011110	—	—	—
LDC	Rm,VBR	Rm → VBR	0100mmmm00101110	Privileged	—	—
LDC	Rm,SGR	Rm → SGR	0100mmmm00111010	Privileged	—	—

Instruction		Operation	Instruction Code	Privileged	T Bit	New
LDC	Rm,SSR	Rm → SSR	0100mmmm00111110	Privileged	—	—
LDC	Rm,SPC	Rm → SPC	0100mmmm01001110	Privileged	—	—
LDC	Rm,DBR	Rm → DBR	0100mmmm11111010	Privileged	—	—
LDC	Rm,Rn_BANK	Rm → Rn_BANK (n = 0 to 7)	0100mmmm1nnn1110	Privileged	—	—
LDC.L	@Rm+,SR	(Rm) → SR, Rm + 4 → Rm	0100mmmm00000111	Privileged	LSB	—
LDC.L	@Rm+,GBR	(Rm) → GBR, Rm + 4 → Rm	0100mmmm00010111	—	—	—
LDC.L	@Rm+,VBR	(Rm) → VBR, Rm + 4 → Rm	0100mmmm00100111	Privileged	—	—
LDC.L	@Rm+,SGR	(Rm) → SGR, Rm + 4 → Rm	0100mmmm00110110	Privileged	—	—
LDC.L	@Rm+,SSR	(Rm) → SSR, Rm + 4 → Rm	0100mmmm00110111	Privileged	—	—
LDC.L	@Rm+,SPC	(Rm) → SPC, Rm + 4 → Rm	0100mmmm01000111	Privileged	—	—
LDC.L	@Rm+,DBR	(Rm) → DBR, Rm + 4 → Rm	0100mmmm11110110	Privileged	—	—
LDC.L	@Rm+,Rn_BANK	(Rm) → Rn_BANK, Rm + 4 → Rm	0100mmmm1nnn0111	Privileged	—	—
LDS	Rm,MACH	Rm → MACH	0100mmmm00001010	—	—	—
LDS	Rm,MACL	Rm → MACL	0100mmmm00011010	—	—	—
LDS	Rm,PR	Rm → PR	0100mmmm00101010	—	—	—
LDS.L	@Rm+,MACH	(Rm) → MACH, Rm + 4 → Rm	0100mmmm00000110	—	—	—
LDS.L	@Rm+,MACL	(Rm) → MACL, Rm + 4 → Rm	0100mmmm00010110	—	—	—
LDS.L	@Rm+,PR	(Rm) → PR, Rm + 4 → Rm	0100mmmm00100110	—	—	—
LDTLB		PTEH/PTEL (/PTEA) → TLB	0000000000111000	Privileged	—	—
MOVCA.L	R0,@Rn	R0 → (Rn) (without fetching cache block)	0000nnnn11000011	—	—	—
NOP		No operation	0000000000001001	—	—	—
OCBI	@Rn	Invalidates operand cache block	0000nnnn10010011	—	—	—
OCBP	@Rn	Writes back and invalidates operand cache block	0000nnnn10100011	—	—	—
OCBWB	@Rn	Writes back operand cache block	0000nnnn10110011	—	—	—
PREF	@Rn	(Rn) → operand cache	0000nnnn10000011	—	—	—
PREFI	@Rn	Reads 32-byte instruction block into instruction cache	0000nnnn11010011	—	—	New
RTE		Delayed branch, SSR/SPC → SR/PC	0000000000101011	Privileged	—	—

Instruction		Operation	Instruction Code	Privileged	T Bit	New
SETS		1 → S	0000000001011000	—	—	—
SETT		1 → T	0000000000011000	—	1	—
SLEEP		Sleep, software standby or deep standby	0000000000011011	Privileged	—	—
STC	SR,Rn	SR → Rn	0000nnnn00000010	Privileged	—	—
STC	GBR,Rn	GBR → Rn	0000nnnn00010010	—	—	—
STC	VBR,Rn	VBR → Rn	0000nnnn00100010	Privileged	—	—
STC	SSR,Rn	SSR → Rn	0000nnnn00110010	Privileged	—	—
STC	SPC,Rn	SPC → Rn	0000nnnn01000010	Privileged	—	—
STC	SGR,Rn	SGR → Rn	0000nnnn00111010	Privileged	—	—
STC	DBR,Rn	DBR → Rn	0000nnnn11111010	Privileged	—	—
STC	Rm_BANK,Rn	Rm_BANK → Rn (m = 0 to 7)	0000nnnn1mmm0010	Privileged	—	—
STC.L	SR,@-Rn	Rn – 4 → Rn, SR → (Rn)	0100nnnn00000011	Privileged	—	—
STC.L	GBR,@-Rn	Rn – 4 → Rn, GBR → (Rn)	0100nnnn00010011	—	—	—
STC.L	VBR,@-Rn	Rn – 4 → Rn, VBR → (Rn)	0100nnnn00100011	Privileged	—	—
STC.L	SSR,@-Rn	Rn – 4 → Rn, SSR → (Rn)	0100nnnn00110011	Privileged	—	—
STC.L	SPC,@-Rn	Rn – 4 → Rn, SPC → (Rn)	0100nnnn01000011	Privileged	—	—
STC.L	SGR,@-Rn	Rn – 4 → Rn, SGR → (Rn)	0100nnnn00110010	Privileged	—	—
STC.L	DBR,@-Rn	Rn – 4 → Rn, DBR → (Rn)	0100nnnn11110010	Privileged	—	—
STC.L	Rm_BANK,@-Rn	Rn – 4 → Rn, Rm_BANK → (Rn) (m = 0 to 7)	0100nnnn1mmm0011	Privileged	—	—
STS	MACH,Rn	MACH → Rn	0000nnnn00001010	—	—	—
STS	MACL,Rn	MACL → Rn	0000nnnn00011010	—	—	—
STS	PR,Rn	PR → Rn	0000nnnn00101010	—	—	—
STS.L	MACH,@-Rn	Rn – 4 → Rn, MACH → (Rn)	0100nnnn00000010	—	—	—
STS.L	MACL,@-Rn	Rn – 4 → Rn, MACL → (Rn)	0100nnnn00010010	—	—	—
STS.L	PR,@-Rn	Rn – 4 → Rn, PR → (Rn)	0100nnnn00100010	—	—	—

Instruction		Operation	Instruction Code	Privileged	T Bit	New
SYNCO		Data accesses invoked by the following instructions are not executed until execution of data accesses which precede this instruction has been completed.	0000000010101011	—	—	New
TRAPA	#imm	PC + 2 → SPC, SR → SSR, R15 → SGR, 1 → SR.MD/BL/RB, #imm << 2 → TRA, H'160 → EXPEVT, VBR + H'0100 → PC	11000011iiiiiiii	—	—	—

Table 2.15 Floating-Point Single-Precision Instructions

Instruction		Operation	Instruction Code	Privileged	T Bit	New
FLDI0	FRn	H'0000 0000 → FRn	1111nnnn10001101	—	—	—
FLDI1	FRn	H'3F80 0000 → FRn	1111nnnn10011101	—	—	—
FMOV	FRm,FRn	FRm → FRn	1111nnnnmmmm1100	—	—	—
FMOV.S	@Rm,FRn	(Rm) → FRn	1111nnnnmmmm1000	—	—	—
FMOV.S	@(R0,Rm),FRn	(R0 + Rm) → FRn	1111nnnnmmmm0110	—	—	—
FMOV.S	@Rm+,FRn	(Rm) → FRn, Rm + 4 → Rm	1111nnnnmmmm1001	—	—	—
FMOV.S	FRm,@Rn	FRm → (Rn)	1111nnnnmmmm1010	—	—	—
FMOV.S	FRm,@-Rn	Rn-4 → Rn, FRm → (Rn)	1111nnnnmmmm1011	—	—	—
FMOV.S	FRm,@(R0,Rn)	FRm → (R0 + Rn)	1111nnnnmmmm0111	—	—	—
FMOV	DRm,DRn	DRm → DRn	1111nnn0mmm01100	—	—	—
FMOV	@Rm,DRn	(Rm) → DRn	1111nnn0mmmm1000	—	—	—
FMOV	@(R0,Rm),DRn	(R0 + Rm) → DRn	1111nnn0mmmm0110	—	—	—
FMOV	@Rm+,DRn	(Rm) → DRn, Rm + 8 → Rm	1111nnn0mmmm1001	—	—	—
FMOV	DRm,@Rn	DRm → (Rn)	1111nnnnmmm01010	—	—	—
FMOV	DRm,@-Rn	Rn-8 → Rn, DRm → (Rn)	1111nnnnmmm01011	—	—	—
FMOV	DRm,@(R0,Rn)	DRm → (R0 + Rn)	1111nnnnmmm00111	—	—	—
FLDS	FRm,FPUL	FRm → FPUL	1111mmmm00011101	—	—	—
FSTS	FPUL,FRn	FPUL → FRn	1111nnnn00001101	—	—	—
FABS	FRn	FRn & H'7FFF FFFF → FRn	1111nnnn01011101	—	—	—

Instruction		Operation	Instruction Code	Privileged	T Bit	New
FADD	FRm,FRn	FRn + FRm → FRn	1111nnnnmmmm0000	—	—	—
FCMP/EQ	FRm,FRn	When FRn = FRm, 1 → T Otherwise, 0 → T	1111nnnnmmmm0100	—	Comparison result	—
FCMP/GT	FRm,FRn	When FRn > FRm, 1 → T Otherwise, 0 → T	1111nnnnmmmm0101	—	Comparison result	—
FDIV	FRm,FRn	FRn/FRm → FRn	1111nnnnmmmm0011	—	—	—
FLOAT	FPUL,FRn	(float) FPUL → FRn	1111nnnn00101101	—	—	—
FMAC	FR0,FRm,FRn	FR0*FRm + FRn → FRn	1111nnnnmmmm1110	—	—	—
FMUL	FRm,FRn	FRn*FRm → FRn	1111nnnnmmmm0010	—	—	—
FNEG	FRn	FRn ∧ H'8000 0000 → FRn	1111nnnn01001101	—	—	—
FSQRT	FRn	√FRn → FRn	1111nnnn01101101	—	—	—
FSUB	FRm,FRn	FRn – FRm → FRn	1111nnnnmmmm0001	—	—	—
FTRC	FRm,FPUL	(long) FRm → FPUL	1111mmmm00111101	—	—	—

Table 2.16 Floating-Point Double-Precision Instructions

Instruction		Operation	Instruction Code	Privileged	T Bit	New
FABS	DRn	DRn & H'7FFF FFFF FFFF FFFF → DRn	1111nnn001011101	—	—	—
FADD	DRm,DRn	DRn + DRm → DRn	1111nnn0mmm00000	—	—	—
FCMP/EQ	DRm,DRn	When DRn = DRm, 1 → T Otherwise, 0 → T	1111nnn0mmm00100	—	Comparison result	—
FCMP/GT	DRm,DRn	When DRn > DRm, 1 → T Otherwise, 0 → T	1111nnn0mmm00101	—	Comparison result	—
FDIV	DRm,DRn	DRn /DRm → DRn	1111nnn0mmm00011	—	—	—
FCNVDS	DRm,FPUL	double_to_ float(DRm) → FPUL	1111mmm010111101	—	—	—
FCNVSD	FPUL,DRn	float_to_ double (FPUL) → DRn	1111nnn010101101	—	—	—
FLOAT	FPUL,DRn	(float)FPUL → DRn	1111nnn000101101	—	—	—
FMUL	DRm,DRn	DRn *DRm → DRn	1111nnn0mmm00010	—	—	—
FNEG	DRn	DRn ^ H'8000 0000 0000 0000 → DRn	1111nnn001001101	—	—	—
FSQRT	DRn	√DRn → DRn	1111nnn001101101	—	—	—
FSUB	DRm,DRn	DRn – DRm → DRn	1111nnn0mmm00001	—	—	—
FTRC	DRm,FPUL	(long) DRm → FPUL	1111mmm000111101	—	—	—

Table 2.17 Floating-Point Control Instructions

Instruction		Operation	Instruction Code	Privileged	T Bit	New
LDS	Rm,FPSCR	Rm → FPSCR	0100mmmm01101010	—	—	—
LDS	Rm,FPUL	Rm → FPUL	0100mmmm01011010	—	—	—
LDS.L	@Rm+,FPSCR	(Rm) → FPSCR, Rm+4 → Rm	0100mmmm01100110	—	—	—
LDS.L	@Rm+,FPUL	(Rm) → FPUL, Rm+4 → Rm	0100mmmm01010110	—	—	—
STS	FPSCR,Rn	FPSCR → Rn	0000nnnn01101010	—	—	—
STS	FPUL,Rn	FPUL → Rn	0000nnnn01011010	—	—	—
STS.L	FPSCR,@-Rn	Rn – 4 → Rn, FPSCR → (Rn)	0100nnnn01100010	—	—	—
STS.L	FPUL,@-Rn	Rn – 4 → Rn, FPUL → (Rn)	0100nnnn01010010	—	—	—

Table 2.18 Floating-Point Graphics Acceleration Instructions

Instruction		Operation	Instruction Code	Privileged	T Bit	New
FMOV	DRm,XDn	DRm → XDn	1111nnn1mmm01100	—	—	—
FMOV	XDm,DRn	XDm → DRn	1111nnn0mmm11100	—	—	—
FMOV	XDm,XDn	XDm → XDn	1111nnn1mmm11100	—	—	—
FMOV	@Rm,XDn	(Rm) → XDn	1111nnn1mmmm1000	—	—	—
FMOV	@Rm+,XDn	(Rm) → XDn, Rm + 8 → Rm	1111nnn1mmmm1001	—	—	—
FMOV	@(R0,Rm),XDn	(R0 + Rm) → XDn	1111nnn1mmmm0110	—	—	—
FMOV	XDm,@Rn	XDm → (Rn)	1111nnnnmmm11010	—	—	—
FMOV	XDm,@-Rn	Rn – 8 → Rn, XDm → (Rn)	1111nnnnmmm11011	—	—	—
FMOV	XDm,@(R0,Rn)	XDm → (R0 + Rn)	1111nnnnmmm10111	—	—	—
FIPR	FVm,FVn	inner_product (FVm, FVn) → FR[n+3]	1111nnmm11101101	—	—	—
FTRV	XMTRX,FVn	transform_vector (XMTRX, FVn) → FVn	1111nn0111111101	—	—	—
FRCHG		~FPSCR.FR → FPSCR.FR	1111101111111101	—	—	—
FSCHG		~FPSCR.SZ → FPSCR.SZ	1111001111111101	—	—	—
FPCHG		~FPSCR.PR → FPSCR.PR	1111011111111101	—	—	New
FSRRA	FRn	1/sqrt(FRn) → FRn	1111nnnn01111101	—	—	New
FSCA	FPUL,DRn	sin(FPUL) → FRn cos(FPUL) → FR[n + 1]	1111nnn011111101	—	—	New

Note: * sqrt(FRn) is the square root of FRn.

2.4 Pipelining

The SH-4A is a 2-ILP (instruction-level-parallelism) superscalar pipelining microprocessor. Instruction execution is pipelined, and two instructions can be executed in parallel.

2.4.1 Pipelines

Figure 2.10 shows the basic pipelines. Normally, a pipeline consists of eight stages: instruction fetch (I1/I2/I3), decode and register read (ID), execution (E1/E2/E3), and write-back (WB). An instruction is executed as a combination of basic pipelines.

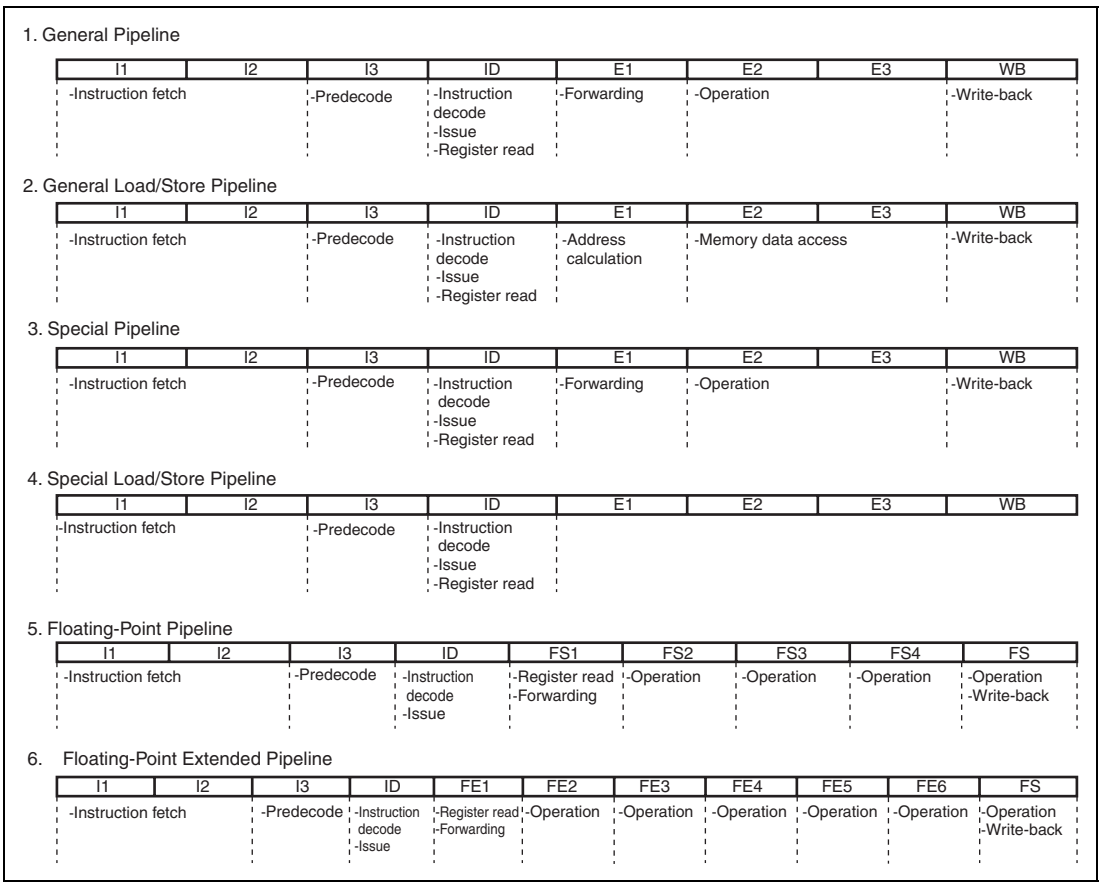


Figure 2.10 Basic Pipelines

Figures 2.11 to 2.19 show the instruction execution patterns. Representations in figures 2.11 to 2.19 and their descriptions are listed in table 2.19.

Table 2.19 Representations of Instruction Execution Patterns

Representation	Description
E1 E2 E3 WB	CPU EX pipe is occupied
S1 S2 S3 WB	CPU LS pipe is occupied (with memory access)
s1 s2 s3 WB	CPU LS pipe is occupied (without memory access)
E1/S1	Either CPU EX pipe or CPU LS pipe is occupied
E1S1 , E1s1	Both CPU EX pipe and CPU LS pipe are occupied
M2 M3 MS	CPU MULT operation unit is occupied
FE1 FE2 FE3 FE4 FE5 FE6 FS	FPU-EX pipe is occupied
FS1 FS2 FS3 FS4 FS	FPU-LS pipe is occupied
ID	ID stage is locked
	Both CPU and FPU pipes are occupied

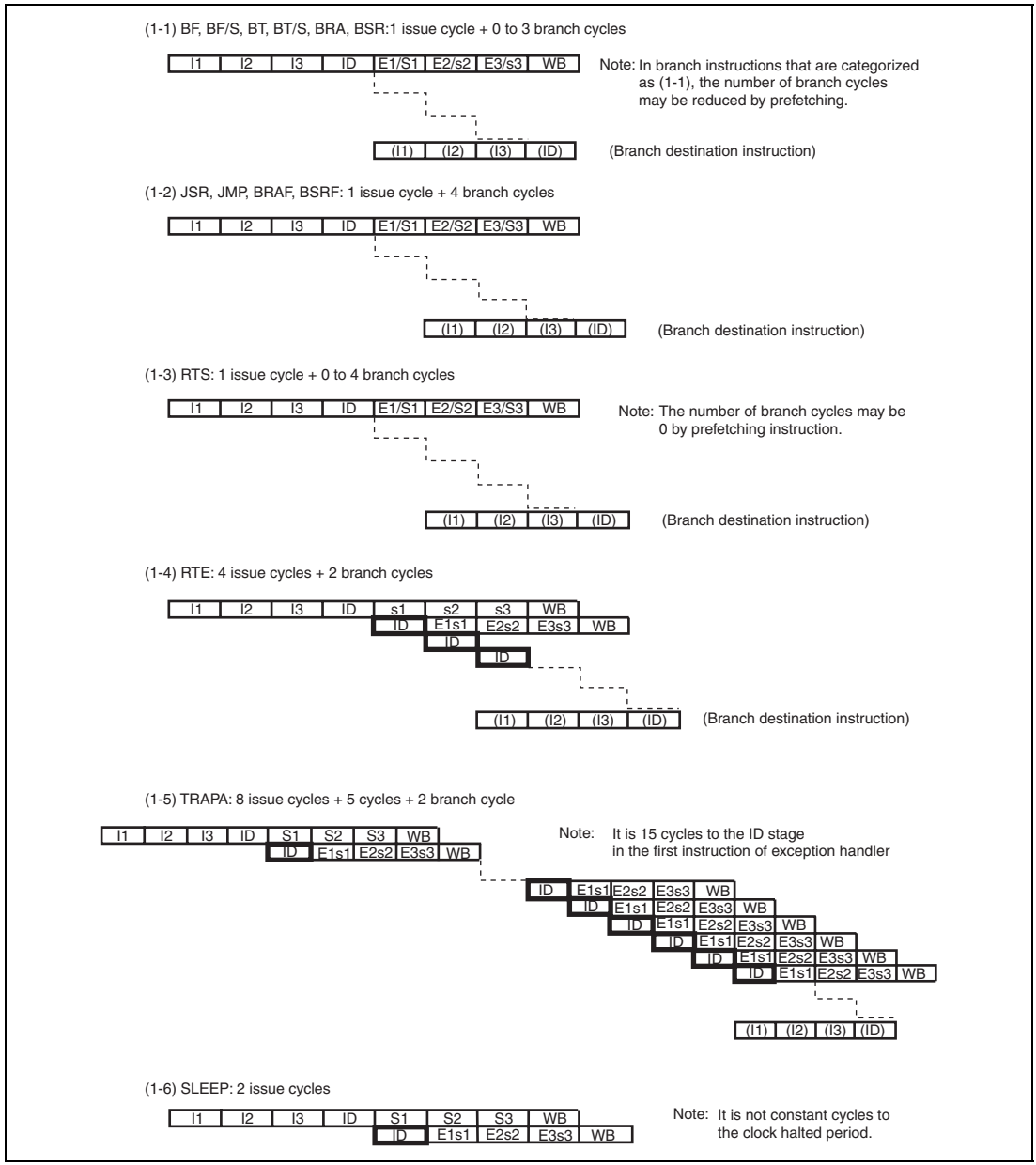


Figure 2.11 Instruction Execution Patterns (1)

(2-1) 1-step operation (EX type): 1 issue cycle

EXT[SU].[BW], MOVT, SWAP, XTRCT, ADD*, CMP*, DIV*, DT, NEG*, SUB*, AND, AND#, NOT, OR, OR#, TST, TST#, XOR, XOR#, ROT*, SHA*, SHL*, CLRS, CLRT, SETS, SETT

Note: Except for AND#, OR#, TST#, and XOR# instructions using GBR relative addressing mode

I1	I2	I3	ID	E1	E2	E3	WB

(2-2) 1-step operation (LS type): 1 issue cycle

MOVA

I1	I2	I3	ID	s1	s2	s3	WB

(2-3) 1-step operation (MT type): 1 issue cycle

MOV#, NOP

I1	I2	I3	ID	E1/S1	E2/s2	E3/s3	WB

(2-4) MOV (MT type): 1 issue cycle

MOV

I1	I2	I3	ID	E1/s1	E2/s2	E3/S3	WB

Figure 2.12 Instruction Execution Patterns (2)

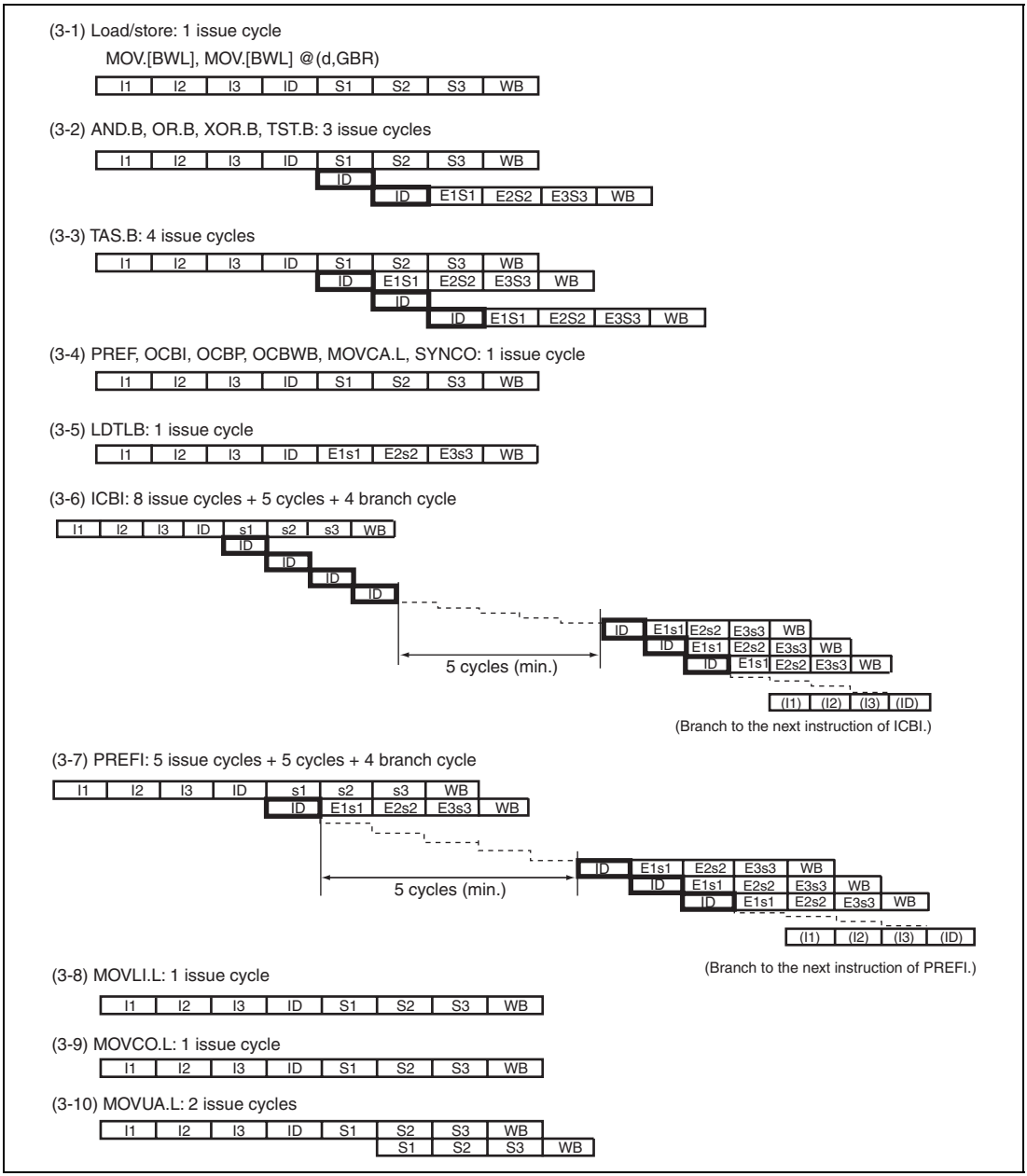


Figure 2.13 Instruction Execution Patterns (3)

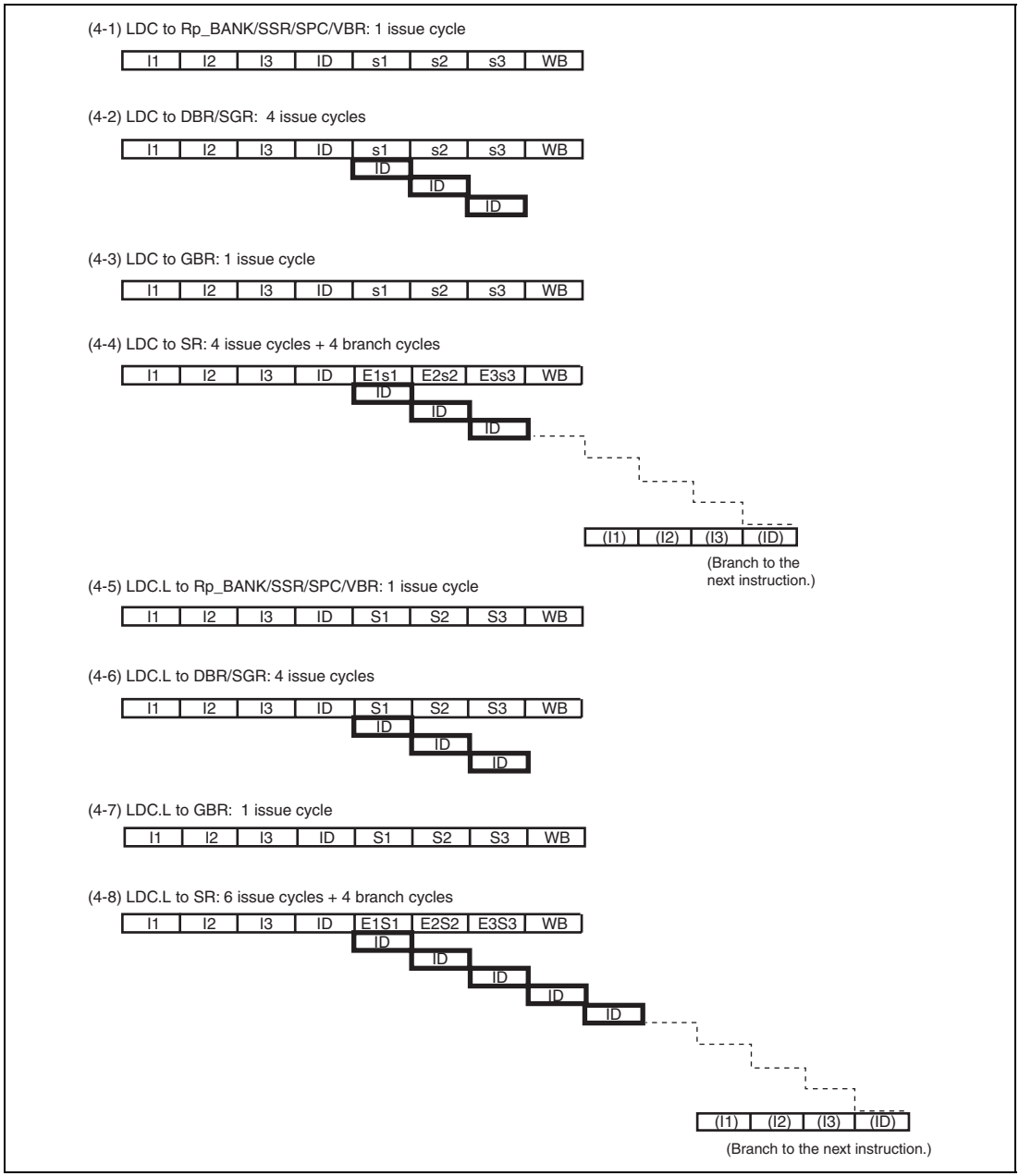


Figure 2.14 Instruction Execution Patterns (4)

(4-9) STC from DBR/GBR/Rp_BANK/SSR/SPC/VBR/SGR: 1 issue cycle

I1	I2	I3	ID	s1	s2	s3	WB

(4-10) STC from SR: 1 issue cycle

I1	I2	I3	ID	E1s1	E2s2	E3s3	WB

(4-11) STC.L from DBR/GBR/Rp_BANK/SSR/SPC/VBR/SGR: 1 issue cycle

I1	I2	I3	ID	S1	S2	S3	WB

(4-12) STC.L from SR: 1 issue cycle

I1	I2	I3	ID	E1S1	E2S2	E3S3	WB

(4-13) LDS to PR: 1 issue cycle

I1	I2	I3	ID	s1	s2	s3	WB

(4-14) LDS.L to PR: 1 issue cycle

I1	I2	I3	ID	S1	S2	S3	WB

(4-15) STS from PR: 1 issue cycle

I1	I2	I3	ID	s1	s2	s3	WB

(4-16) STS.L from PR: 1 issue cycle

I1	I2	I3	ID	S1	S2	S3	WB

(4-17) BSRF, BSR, JSR delay slot instructions (PR set): 0 issue cycle

(I1)	(I2)	(I3)	(ID)	(??1)	(??2)	(??3)	(WB)

Notes: The value of PR is changed in the E3 stage of delay slot instruction.
When the STS and STS.L instructions from PR are used as delay slot instructions, changed PR value is used.

Figure 2.15 Instruction Execution Patterns (5)

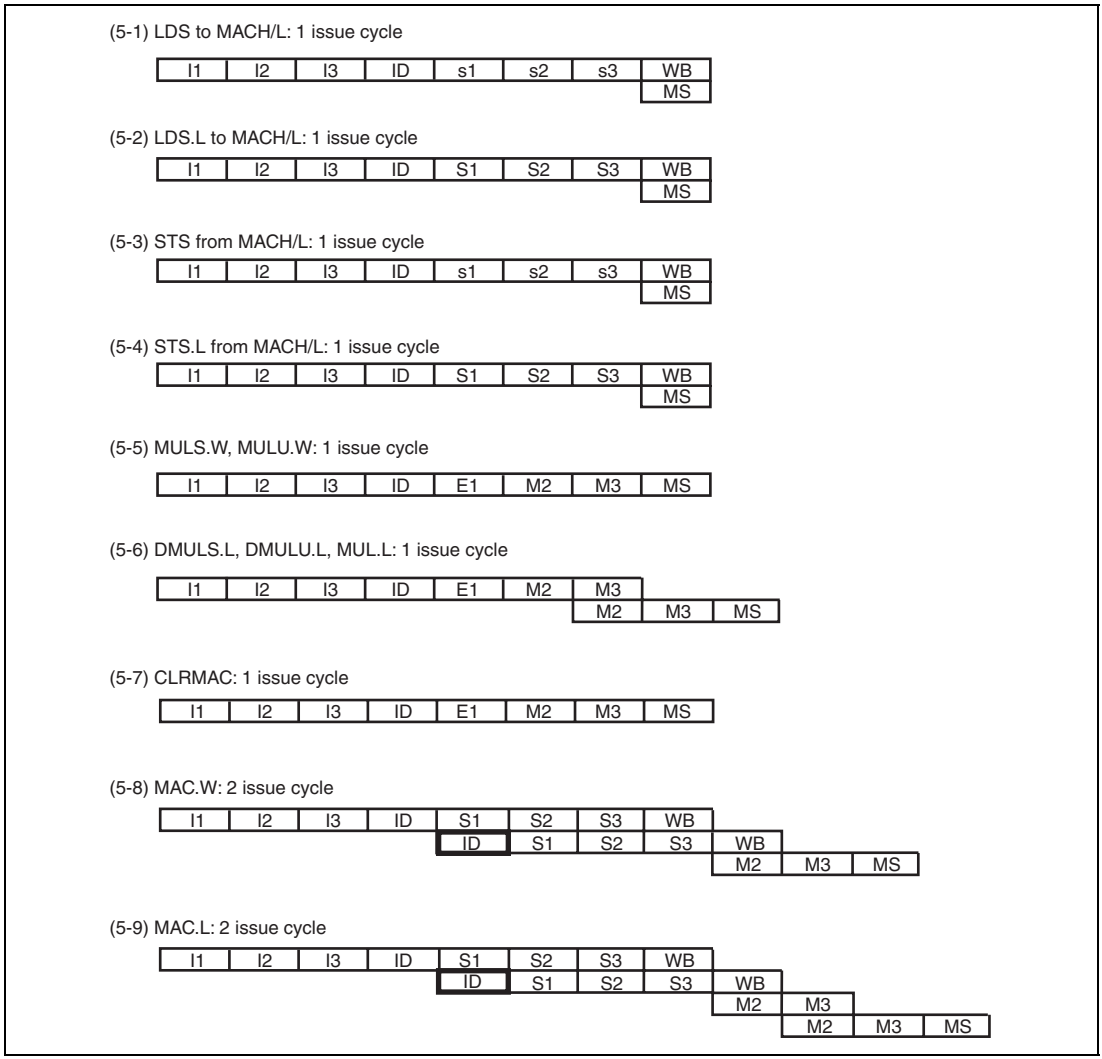


Figure 2.16 Instruction Execution Patterns (6)

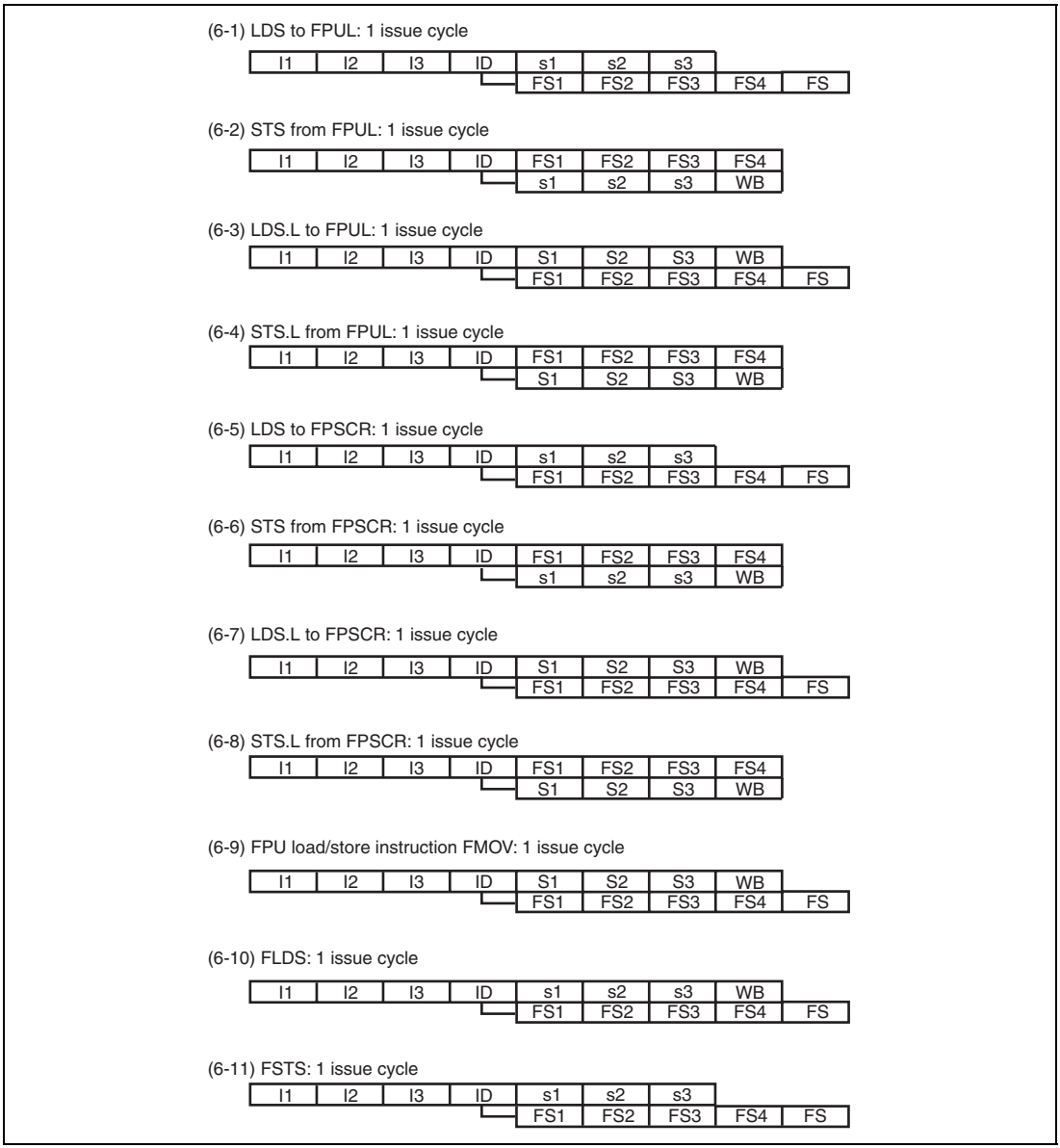


Figure 2.17 Instruction Execution Patterns (7)

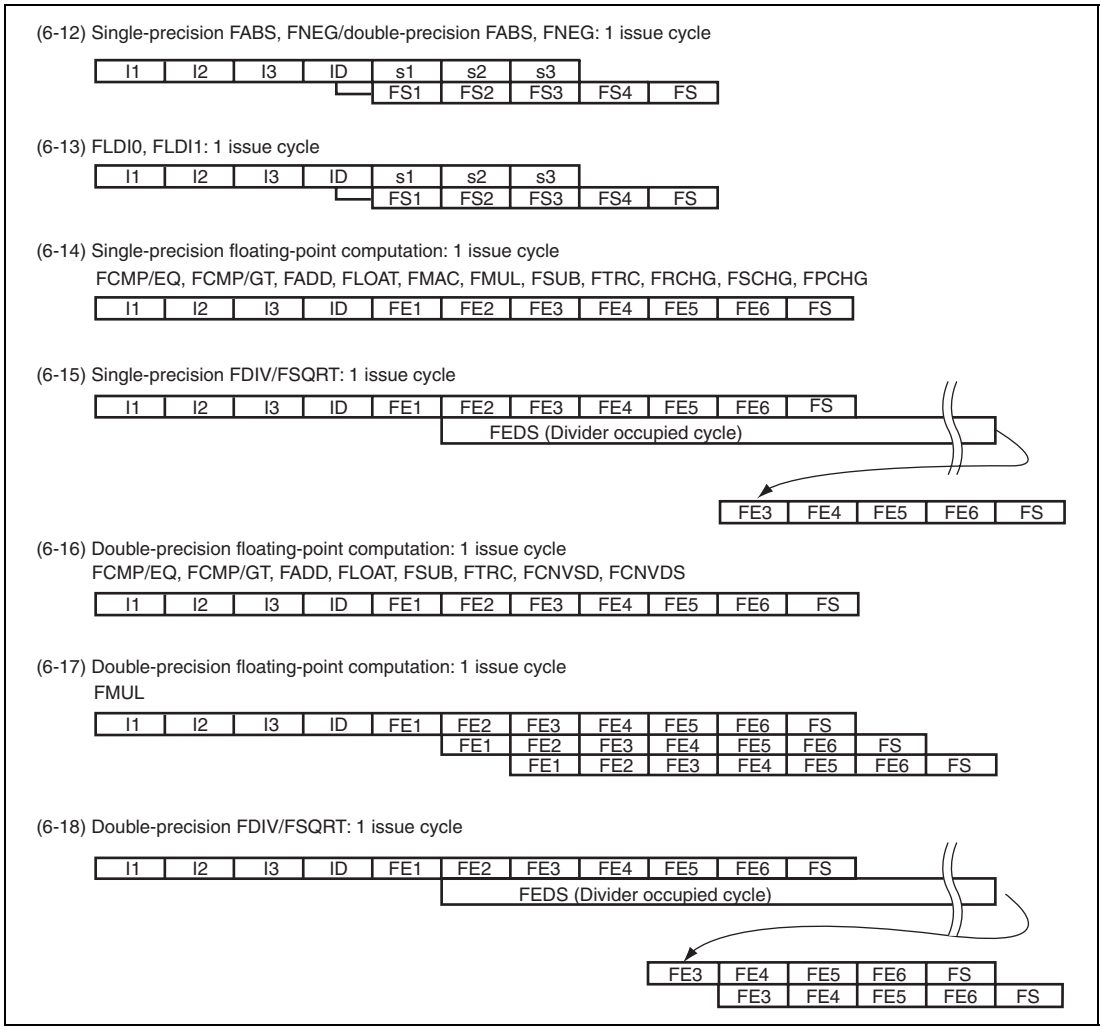


Figure 2.18 Instruction Execution Patterns (8)

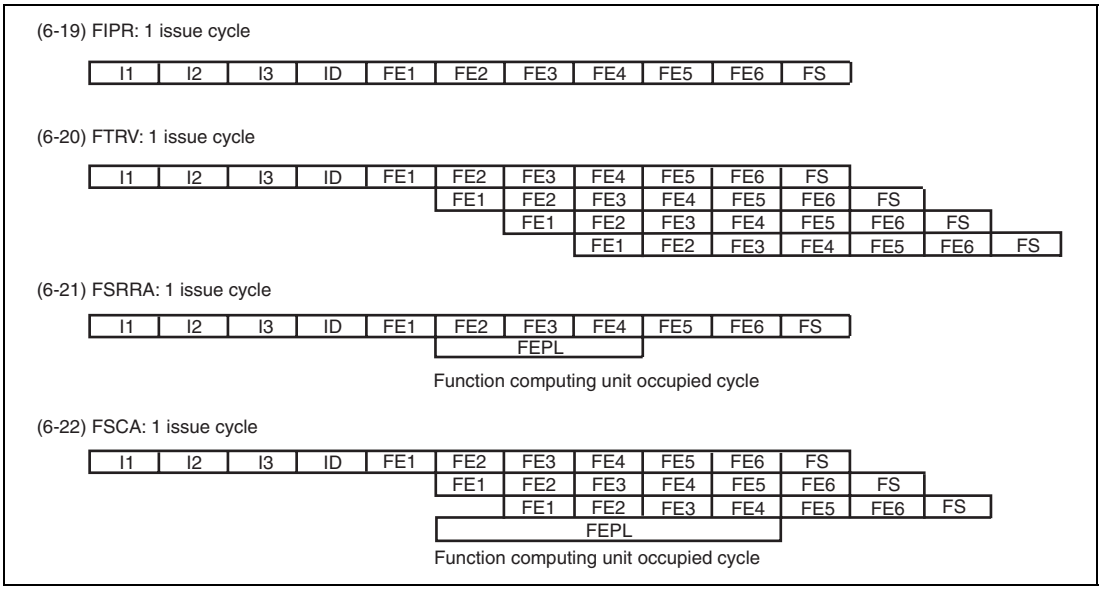


Figure 2.19 Instruction Execution Patterns (9)

2.4.2 Parallel-Executability

Instructions are categorized into six groups according to the internal function blocks used, as shown in table 2.20. Table 2.21 shows the parallel-executability of pairs of instructions in terms of groups. For example, ADD in the EX group and BRA in the BR group can be executed in parallel.

Table 2.20 Instruction Groups

Instruction Group	Instruction			
EX	ADD	DT	ROTL	SHLR8
	ADDC	EXTS	ROTR	SHLR16
	ADDV	EXTU	SETS	SUB
	AND #imm,R0	MOVT	SETT	SUBC
	AND Rm,Rn	MUL.L	SHAD	SUBV
	CLRMAC	MULS.W	SHAL	SWAP
	CLRS	MULU.W	SHAR	TST #imm,R0
	CLRT	NEG	SHLD	TST Rm,Rn
	CMP	NEGC	SHLL	XOR #imm,R0
	DIV0S	NOT	SHLL2	XOR Rm,Rn
	DIV0U	OR #imm,R0	SHLL8	XTRCT
	DIV1	OR Rm,Rn	SHLL16	
	DMUS.L	ROTCL	SHLR	
	DMULU.L	ROTCR	SHLR2	
MT	MOV #imm,Rn	MOV Rm,Rn	NOP	
BR	BF	BRAF	BT	JSR
	BF/S	BSR	BT/S	RTS
	BRA	BSRF	JMP	
LS	FABS	FMOV.S FR,@adr	MOV.[BWL] @adr,R	STC CR2,Rn
	FNEG	FSTS	MOV.[BWL] R,@adr	STC.L CR2,@-Rn
	FLDI0	LDC Rm,CR1	MOVA	STS SR2,Rn
	FLDI1	LDC.L @Rm+,CR1	MOVCA.L	STS.L SR2,@-Rn
	FLDS	LDS Rm,SR1	MOVUA	STS SR1,Rn
	FMOV @adr,FR	LDS Rm,SR2	OCBI	STS.L SR1,@-Rn
	FMOV FR,@adr	LDS.L @adr,SR2	OCBP	
	FMOV FR,FR	LDS.L @Rm+,SR1	OCBWB	
	FMOV.S @adr,FR	LDS.L @Rm+,SR2	PREF	

Instruction Group	Instruction			
FE	FADD	FDIV	FRCHG	FSCA
	FSUB	FIPR	FSCHG	FSRRA
	FCMP (S/D)	FLOAT	FSQRT	FPCHG
	FCNVDS	FMAC	FTRC	
	FCNVSD	FMUL	FTRV	
CO	AND.B #imm,@(R0,GBR)	LDC.L @Rm+,SR	PREFI	TRAPA
	ICBI	LDTLB	RTE	TST.B #imm,@(R0,GBR)
	LDC Rm,DBR	MAC.L	SLEEP	XOR.B #imm,@(R0,GBR)
	LDC Rm, SGR	MAC.W	STC SR,Rn	
	LDC Rm,SR	MOVCO	STC.L SR,@-Rn	
	LDC.L @Rm+,DBR	MOVLI	SYNCO	
	LDC.L @Rm+,SGR	OR.B #imm,@(R0,GBR)	TAS.B	

[Legend]
R: Rm/Rn
@adr: Address
SR1: MACH/MACL/PR
SR2: FPUL/FPSCR
CR1: GBR/Rp_BANK/SPC/SSR/VBR
CR2: CR1/DBR/SGR
FR: FRm/FRn/DRm/DRn/XDm/XDn

The parallel execution of two instructions can be carried out under following conditions.

1. Both addr (preceding instruction) and addr+2 (following instruction) are specified within the minimum page size (1 Kbyte).
2. The execution of these two instructions is supported in table 2.21, Combination of Preceding and Following Instructions.
3. Data used by an instruction of addr does not conflict with data used by a previous instruction
4. Data used by an instruction of addr+2 does not conflict with data used by a previous instruction
5. Both instructions are valid

Table 2.21 Combination of Preceding and Following Instructions

		Preceding Instruction (addr)					
		EX	**MT**	**BR**	**LS**	**FE**	**CO**
Following Instruction (addr+2)	EX	No	Yes	Yes	Yes	Yes	
	MT	Yes	Yes	Yes	Yes	Yes	
	BR	Yes	Yes	No	Yes	Yes	
	LS	Yes	Yes	Yes	No	Yes	
	FE	Yes	Yes	Yes	Yes	No	
	CO						No

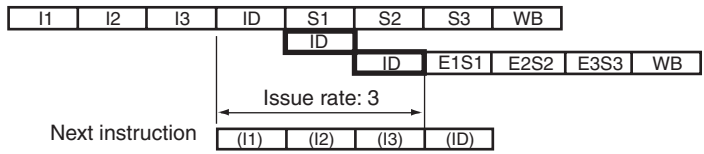
2.4.3 Issue Rates and Execution States

Instruction execution states are summarized in table 2.22. Instruction Group in the table 2.22 corresponds to the category in the table 2.20. Penalty cycles due to a pipeline stall are not considered in the issue rates and execution states in this section.

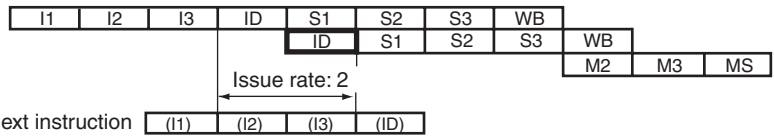
1. Issue Rate

Issue rates indicates the issue period between one instruction and next instruction.

E.g. AND.B instruction



E.g. MAC.W instruction

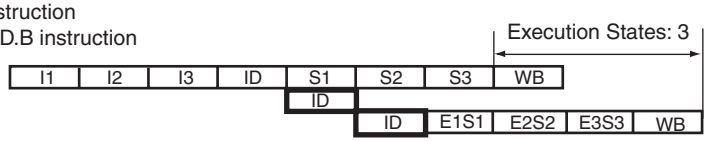


2. Execution States

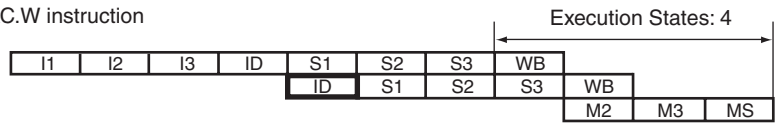
Execution states indicates the cycle counts an instruction occupied the pipeline based on the next rules.

CPU instruction

E.g. AND.B instruction

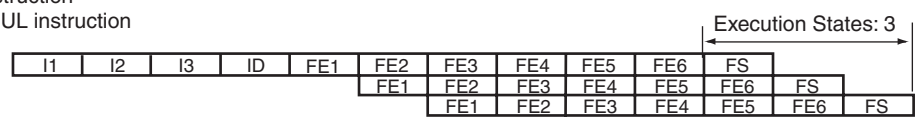


E.g. MAC.W instruction



FPU instruction

E.g. FMUL instruction



E.g. FDIV instruction

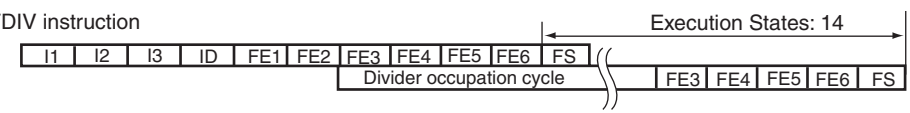


Table 2.22 Issue Rates and Execution States

Functional Category	No.	Instruction		Instruction Group	Issue Rate	Execution States	Execution Pattern
Data transfer instructions	1	EXTS.B	Rm,Rn	EX	1	1	2-1
	2	EXTS.W	Rm,Rn	EX	1	1	2-1
	3	EXTU.B	Rm,Rn	EX	1	1	2-1
	4	EXTU.W	Rm,Rn	EX	1	1	2-1
	5	MOV	Rm,Rn	MT	1	1	2-4
	6	MOV	#imm,Rn	MT	1	1	2-3
	7	MOVA	@(disp,PC),R0	LS	1	1	2-2
	8	MOV.W	@(disp,PC),Rn	LS	1	1	3-1
	9	MOV.L	@(disp,PC),Rn	LS	1	1	3-1
	10	MOV.B	@Rm,Rn	LS	1	1	3-1
	11	MOV.W	@Rm,Rn	LS	1	1	3-1
	12	MOV.L	@Rm,Rn	LS	1	1	3-1
	13	MOV.B	@Rm+,Rn	LS	1	1	3-1
	14	MOV.W	@Rm+,Rn	LS	1	1	3-1
	15	MOV.L	@Rm+,Rn	LS	1	1	3-1
	16	MOV.B	@(disp,Rm),R0	LS	1	1	3-1
	17	MOV.W	@(disp,Rm),R0	LS	1	1	3-1
	18	MOV.L	@(disp,Rm),Rn	LS	1	1	3-1
	19	MOV.B	@(R0,Rm),Rn	LS	1	1	3-1
	20	MOV.W	@(R0,Rm),Rn	LS	1	1	3-1
	21	MOV.L	@(R0,Rm),Rn	LS	1	1	3-1
	22	MOV.B	@(disp,GBR),R0	LS	1	1	3-1
	23	MOV.W	@(disp, GBR),R0	LS	1	1	3-1
	24	MOV.L	@(disp, GBR),R0	LS	1	1	3-1
	25	MOV.B	Rm,@Rn	LS	1	1	3-1
	26	MOV.W	Rm,@Rn	LS	1	1	3-1
	27	MOV.L	Rm,@Rn	LS	1	1	3-1
	28	MOV.B	Rm,@-Rn	LS	1	1	3-1
	29	MOV.W	Rm,@-Rn	LS	1	1	3-1

Functional Category	No.	Instruction		Instruction Group	Issue Rate	Execution States	Execution Pattern
Data transfer instructions	30	MOV.L	Rm,@-Rn	LS	1	1	3-1
	31	MOV.B	R0,@(disp,Rn)	LS	1	1	3-1
	32	MOV.W	R0,@(disp,Rn)	LS	1	1	3-1
	33	MOV.L	Rm,@(disp,Rn)	LS	1	1	3-1
	34	MOV.B	Rm,@(R0,Rn)	LS	1	1	3-1
	35	MOV.W	Rm,@(R0,Rn)	LS	1	1	3-1
	36	MOV.L	Rm,@(R0,Rn)	LS	1	1	3-1
	37	MOV.B	R0,@(disp,GBR)	LS	1	1	3-1
	38	MOV.W	R0,@(disp,GBR)	LS	1	1	3-1
	39	MOV.L	R0,@(disp,GBR)	LS	1	1	3-1
	40	MOVCA.L	R0,@Rn	LS	1	1	3-4
	41	MOVCO.L	R0,@Rn	CO	1	1	3-9
	42	MOVLI.L	@Rm,R0	CO	1	1	3-8
	43	MOVUA.L	@Rm,R0	LS	2	2	3-10
	44	MOVUA.L	@Rm+,R0	LS	2	2	3-10
	45	MOVT	Rn	EX	1	1	2-1
	46	OCBI	@Rn	LS	1	1	3-4
	47	OCBP	@Rn	LS	1	1	3-4
	48	OCBWB	@Rn	LS	1	1	3-4
	49	PREF	@Rn	LS	1	1	3-4
	50	SWAP.B	Rm,Rn	EX	1	1	2-1
	51	SWAP.W	Rm,Rn	EX	1	1	2-1
	52	XTRCT	Rm,Rn	EX	1	1	2-1
Fixed-point arithmetic instructions	53	ADD	Rm,Rn	EX	1	1	2-1
	54	ADD	#imm,Rn	EX	1	1	2-1
	55	ADDC	Rm,Rn	EX	1	1	2-1
	56	ADDV	Rm,Rn	EX	1	1	2-1
	57	CMP/EQ	#imm,R0	EX	1	1	2-1
	58	CMP/EQ	Rm,Rn	EX	1	1	2-1
	59	CMP/GE	Rm,Rn	EX	1	1	2-1

Functional Category	No.	Instruction		Instruction Group	Issue Rate	Execution States	Execution Pattern
Fixed-point arithmetic instructions	60	CMP/GT	Rm,Rn	EX	1	1	2-1
	61	CMP/HI	Rm,Rn	EX	1	1	2-1
	62	CMP/HS	Rm,Rn	EX	1	1	2-1
	63	CMP/PL	Rn	EX	1	1	2-1
	64	CMP/PZ	Rn	EX	1	1	2-1
	65	CMP/STR	Rm,Rn	EX	1	1	2-1
	66	DIV0S	Rm,Rn	EX	1	1	2-1
	67	DIV0U		EX	1	1	2-1
	68	DIV1	Rm,Rn	EX	1	1	2-1
	69	DMULS.L	Rm,Rn	EX	1	2	5-6
	70	DMULU.L	Rm,Rn	EX	1	2	5-6
	71	DT	Rn	EX	1	1	2-1
	72	MAC.L	@Rm+,@Rn+	CO	2	5	5-9
	73	MAC.W	@Rm+,@Rn+	CO	2	4	5-8
	74	MUL.L	Rm,Rn	EX	1	2	5-6
	75	MULS.W	Rm,Rn	EX	1	1	5-5
	76	MULU.W	Rm,Rn	EX	1	1	5-5
	77	NEG	Rm,Rn	EX	1	1	2-1
	78	NEGC	Rm,Rn	EX	1	1	2-1
	79	SUB	Rm,Rn	EX	1	1	2-1
	80	SUBC	Rm,Rn	EX	1	1	2-1
	81	SUBV	Rm,Rn	EX	1	1	2-1
Logical instructions	82	AND	Rm,Rn	EX	1	1	2-1
	83	AND	#imm,R0	EX	1	1	2-1
	84	AND.B	#imm,@(R0,GBR)	CO	3	3	3-2
	85	NOT	Rm,Rn	EX	1	1	2-1
	86	OR	Rm,Rn	EX	1	1	2-1
	87	OR	#imm,R0	EX	1	1	2-1
	88	OR.B	#imm,@(R0,GBR)	CO	3	3	3-2
	89	TAS.B	@Rn	CO	4	4	3-3

Functional Category	No.	Instruction		Instruction Group	Issue Rate	Execution States	Execution Pattern
Logical instructions	90	TST	Rm,Rn	EX	1	1	2-1
	91	TST	#imm,R0	EX	1	1	2-1
	92	TST.B	#imm,@(R0,GBR)	CO	3	3	3-2
	93	XOR	Rm,Rn	EX	1	1	2-1
	94	XOR	#imm,R0	EX	1	1	2-1
	95	XOR.B	#imm,@(R0,GBR)	CO	3	3	3-2
Shift instructions	96	ROTL	Rn	EX	1	1	2-1
	97	ROTR	Rn	EX	1	1	2-1
	98	ROTCL	Rn	EX	1	1	2-1
	99	ROTCR	Rn	EX	1	1	2-1
	100	SHAD	Rm,Rn	EX	1	1	2-1
	101	SHAL	Rn	EX	1	1	2-1
	102	SHAR	Rn	EX	1	1	2-1
	103	SHLD	Rm,Rn	EX	1	1	2-1
	104	SHLL	Rn	EX	1	1	2-1
	105	SHLL2	Rn	EX	1	1	2-1
	106	SHLL8	Rn	EX	1	1	2-1
	107	SHLL16	Rn	EX	1	1	2-1
	108	SHLR	Rn	EX	1	1	2-1
	109	SHLR2	Rn	EX	1	1	2-1
	110	SHLR8	Rn	EX	1	1	2-1
	111	SHLR16	Rn	EX	1	1	2-1
Branch instructions	112	BF	disp	BR	1+0 to 2	1	1-1
	113	BF/S	disp	BR	1+0 to 2	1	1-1
	114	BT	disp	BR	1+0 to 2	1	1-1
	115	BT/S	disp	BR	1+0 to 2	1	1-1
	116	BRA	disp	BR	1+0 to 2	1	1-1
	117	BRAF	Rm	BR	1+3	1	1-2
	118	BSR	disp	BR	1+0 to 2	1	1-1
	119	BSRF	Rm	BR	1+3	1	1-2

Functional Category	No.	Instruction		Instruction Group	Issue Rate	Execution States	Execution Pattern
Branch instructions	120	JMP	@Rn	BR	1+3	1	1-2
	121	JSR	@Rn	BR	1+3	1	1-2
	122	RTS		BR	1+0 to 3	1	1-3
System control instruction	123	NOP		MT	1	1	2-3
	124	CLRMAC		EX	1	1	5-7
	125	CLRS		EX	1	1	2-1
	126	CLRT		EX	1	1	2-1
	127	ICBI	@Rn	CO	8+5+3	13	3-6
	128	SETS		EX	1	1	2-1
	129	SETT		EX	1	1	2-1
	130	PREFI	@Rn	CO	5+5+3	10	3-7
	131	SYNCO		CO	Undefined	Undefined	3-4
	132	TRAPA	#imm	CO	8+5+1	13	1-5
	133	RTE		CO	4+1	4	1-4
	134	SLEEP		CO	Undefined	Undefined	1-6
	135	LDTLB		CO	1	1	3-5
	136	LDC	Rm,DBR	CO	4	4	4-2
	137	LDC	Rm,SGR	CO	4	4	4-2
	138	LDC	Rm,GBR	LS	1	1	4-3
	139	LDC	Rm,Rp_BANK	LS	1	1	4-1
	140	LDC	Rm,SR	CO	4+3	4	4-4
	141	LDC	Rm,SSR	LS	1	1	4-1
	142	LDC	Rm,SPC	LS	1	1	4-1
	143	LDC	Rm,VBR	LS	1	1	4-1
	144	LDC.L	@Rm+,DBR	CO	4	4	4-6
	145	LDC.L	@Rm+,SGR	CO	4	4	4-6
	146	LDC.L	@Rm+,GBR	LS	1	1	4-7
	147	LDC.L	@Rm+,Rp_BANK	LS	1	1	4-5
	148	LDC.L	@Rm+,SR	CO	6+3	4	4-8
	149	LDC.L	@Rm+,SSR	LS	1	1	4-5

Functional Category	No.	Instruction		Instruction Group	Issue Rate	Execution States	Execution Pattern
System control instructions	150	LDC.L	@Rm+,SPC	LS	1	1	4-5
	151	LDC.L	@Rm+,VBR	LS	1	1	4-5
	152	LDS	Rm,MACH	LS	1	1	5-1
	153	LDS	Rm,MACL	LS	1	1	5-1
	154	LDS	Rm,PR	LS	1	1	4-13
	155	LDS.L	@Rm+,MACH	LS	1	1	5-2
	156	LDS.L	@Rm+,MACL	LS	1	1	5-2
	157	LDS.L	@Rm+,PR	LS	1	1	4-14
	158	STC	DBR,Rn	LS	1	1	4-9
	159	STC	SGR,Rn	LS	1	1	4-9
	160	STC	GBR,Rn	LS	1	1	4-9
	161	STC	Rp_BANK,Rn	LS	1	1	4-9
	162	STC	SR,Rn	CO	1	1	4-10
	163	STC	SSR,Rn	LS	1	1	4-9
	164	STC	SPC,Rn	LS	1	1	4-9
	165	STC	VBR,Rn	LS	1	1	4-9
	166	STC.L	DBR,@-Rn	LS	1	1	4-11
	167	STC.L	SGR,@-Rn	LS	1	1	4-11
	168	STC.L	GBR,@-Rn	LS	1	1	4-11
	169	STC.L	Rp_BANK,@-Rn	LS	1	1	4-11
	170	STC.L	SR,@-Rn	CO	1	1	4-12
	171	STC.L	SSR,@-Rn	LS	1	1	4-11
	172	STC.L	SPC,@-Rn	LS	1	1	4-11
	173	STC.L	VBR,@-Rn	LS	1	1	4-11
	174	STS	MACH,Rn	LS	1	1	5-3
	175	STS	MACL,Rn	LS	1	1	5-3
	176	STS	PR,Rn	LS	1	1	4-15
	177	STS.L	MACH,@-Rn	LS	1	1	5-4
	178	STS.L	MACL,@-Rn	LS	1	1	5-4
	179	STS.L	PR,@-Rn	LS	1	1	4-16

Functional Category	No.	Instruction		Instruction Group	Issue Rate	Execution States	Execution Pattern
Single-precision floating-point instructions	180	FLDI0	FRn	LS	1	1	6-13
	181	FLDI1	FRn	LS	1	1	6-13
	182	FMOV	FRm,FRn	LS	1	1	6-9
	183	FMOV.S	@Rm,FRn	LS	1	1	6-9
	184	FMOV.S	@Rm+,FRn	LS	1	1	6-9
	185	FMOV.S	@(R0,Rm),FRn	LS	1	1	6-9
	186	FMOV.S	FRm,@Rn	LS	1	1	6-9
	187	FMOV.S	FRm,@-Rn	LS	1	1	6-9
	188	FMOV.S	FRm,@(R0,Rn)	LS	1	1	6-9
	189	FLDS	FRm,FPUL	LS	1	1	6-10
	190	FSTS	FPUL,FRn	LS	1	1	6-11
	191	FABS	FRn	LS	1	1	6-12
	192	FADD	FRm,FRn	FE	1	1	6-14
	193	FCMP/EQ	FRm,FRn	FE	1	1	6-14
	194	FCMP/GT	FRm,FRn	FE	1	1	6-14
	195	FDIV	FRm,FRn	FE	1	14	6-15
	196	FLOAT	FPUL,FRn	FE	1	1	6-14
	197	FMAC	FR0,FRm,FRn	FE	1	1	6-14
	198	FMUL	FRm,FRn	FE	1	1	6-14
	199	FNEG	FRn	LS	1	1	6-12
	200	FSQRT	FRn	FE	1	30	6-15
	201	FSUB	FRm,FRn	FE	1	1	6-14
	202	FTRC	FRm,FPUL	FE	1	1	6-14
	203	FMOV	DRm,DRn	LS	1	1	6-9
	204	FMOV	@Rm,DRn	LS	1	1	6-9
	205	FMOV	@Rm+,DRn	LS	1	1	6-9
	206	FMOV	@(R0,Rm),DRn	LS	1	1	6-9
	207	FMOV	DRm,@Rn	LS	1	1	6-9
	208	FMOV	DRm,@-Rn	LS	1	1	6-9
	209	FMOV	DRm,@(R0,Rn)	LS	1	1	6-9

Functional Category	No.	Instruction		Instruction Group	Issue Rate	Execution States	Execution Pattern
Double-precision floating-point instructions	210	FABS	DRn	LS	1	1	6-12
	211	FADD	DRm,DRn	FE	1	1	6-16
	212	FCMP/EQ	DRm,DRn	FE	1	1	6-16
	213	FCMP/GT	DRm,DRn	FE	1	1	6-16
	214	FCNVDS	DRm,FPUL	FE	1	1	6-16
	215	FCNVSD	FPUL,DRn	FE	1	1	6-16
	216	FDIV	DRm,DRn	FE	1	14	6-18
	217	FLOAT	FPUL,DRn	FE	1	1	6-16
	218	FMUL	DRm,DRn	FE	1	3	6-17
	219	FNEG	DRn	LS	1	1	6-12
	220	FSQRT	DRn	FE	1	30	6-18
	221	FSUB	DRm,DRn	FE	1	1	6-16
	222	FTRC	DRm,FPUL	FE	1	1	6-16
FPU system control instructions	223	LDS	Rm,FPUL	LS	1	1	6-1
	224	LDS	Rm,FPSCR	LS	1	1	6-5
	225	LDS.L	@Rm+,FPUL	LS	1	1	6-3
	226	LDS.L	@Rm+,FPSCR	LS	1	1	6-7
	227	STS	FPUL,Rn	LS	1	1	6-2
	228	STS	FPSCR,Rn	LS	1	1	6-6
	229	STS.L	FPUL,@-Rn	LS	1	1	6-4
	230	STS.L	FPSCR,@-Rn	LS	1	1	6-8
Graphics acceleration instructions	231	FMOV	DRm,XDn	LS	1	1	6-9
	232	FMOV	XDm,DRn	LS	1	1	6-9
	233	FMOV	XDm,XDn	LS	1	1	6-9
	234	FMOV	@Rm,XDn	LS	1	1	6-9
	235	FMOV	@Rm+,XDn	LS	1	1	6-9
	236	FMOV	@(R0,Rm),XDn	LS	1	1	6-9
	237	FMOV	XDm,@Rn	LS	1	1	6-9
	238	FMOV	XDm,@-Rn	LS	1	1	6-9
	239	FMOV	XDm,@(R0,Rn)	LS	1	1	6-9
	240	FIPR	FVm,FVn	FE	1	1	6-19

Functional Category	No.	Instruction		Instruction Group	Issue Rate	Execution States	Execution Pattern
Graphics acceleration instructions	241	FRCHG		FE	1	1	6-14
	242	FSCHG		FE	1	1	6-14
	243	FPCHG		FE	1	1	6-14
	244	FSRRA	FRn	FE	1	1	6-21
	245	FSCA	FPUL,DRn	FE	1	3	6-22
	246	FTRV	XMTRX,FVn	FE	1	4	6-20

2.5 Exception Handling

2.5.1 Summary of Exception Handling

Exception handling processing is handled by a special routine which is executed by a reset, general exception handling, or interrupt. For example, if the executing instruction ends abnormally, appropriate action must be taken in order to return to the original program sequence, or report the abnormality before terminating the processing. The process of generating an exception handling request in response to abnormal termination, and passing control to a user-written exception handling routine, in order to support such functions, is given the generic name of exception handling.

The exception handling in the SH-4A is of three kinds: resets, general exceptions, and interrupts.

2.5.2 Register Descriptions

Table 2.23 lists the configuration of registers related exception handling.

Table 2.23 Register Configuration

Register Name	Abbr.	R/W	P4 Address*	Area 7 Address*	Access Size
TRAPA exception register	TRA	R/W	H'FF00 0020	H'1F00 0020	32
Exception event register	EXPEVT	R/W	H'FF00 0024	H'1F00 0024	32
Interrupt event register	INTEVT	R/W	H'FF00 0028	H'1F00 0028	32
Non-support detection exception register	EXPMASK	R/W	H'FF2F 0004	H'1F2F 0004	32

Note: * P4 is the address when virtual address space P4 area is used. Area 7 is the address when physical address space area 7 is accessed by using the TLB.

Table 2.24 States of Register in Each Operating Mode

Register Name	Abbr.	Power-on Reset	Manual Reset	Sleep	Software Standby	Deep Standby
TRAPA exception register	TRA	Undefined	Undefined	Retained	Retained	Undefined
Exception event register	EXPEVT	H'0000 0000	H'0000 0020	Retained	Retained	H'0000 0000
Interrupt event register	INTEVT	Undefined	Undefined	Retained	Retained	Undefined
Non-support detection exception register	EXPMASK	Initialized (H'0000 0013)	Initialized (H'0000 0013)	Retained	Retained	Initialized (H'0000 0013)

(1) TRAPA Exception Register (TRA)

The TRAPA exception register (TRA) consists of 8-bit immediate data (imm) for the TRAPA instruction. TRA is set automatically by hardware when a TRAPA instruction is executed. TRA can also be modified by software.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TRACODE								—	—
Initial value:	0	0	0	0	0	0	—	—	—	—	—	—	—	—	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved For details on reading/writing this bit, see General Precautions on Handling of Product.
9 to 2	TRACODE	Undefined	R/W	TRAPA Code 8-bit immediate data of TRAPA instruction is set
1, 0	—	All 0	R	Reserved For details on reading/writing this bit, see General Precautions on Handling of Product.

(2) Exception Event Register (EXPEVT)

The exception event register (EXPEVT) consists of a 12-bit exception code. The exception code set in EXPEVT is that for a reset or general exception event. The exception code is set automatically by hardware when an exception occurs. EXPEVT can also be modified by software.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	EXPCODE											
Initial value:	0	0	0	0	0	0	0	0	0	0	0/1	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved For details on reading/writing this bit, see General Precautions on Handling of Product.
11 to 0	EXPCODE	H'000 or H'020	R/W	Exception Code The exception code for a reset or general exception is set. For details, see table 2.25.

(3) Interrupt Event Register (INTEVT)

The interrupt event register (INTEVT) consists of a 14-bit exception code. The exception code is set automatically by hardware when an exception occurs. INTEVT can also be modified by software.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	INTCODE													
Initial value:	0	0	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved For details on reading/writing this bit, see General Precautions on Handling of Product.
13 to 0	INTCODE	Undefined	R/W	Exception Code The exception code for an interrupt is set. For details, see table 2.25.

(4) Non-Support Detection Exception Register (EXPMASK)

The non-support detection exception register (EXPMASK) is used to enable or disable the generation of exceptions in response to the use of any of functions 1 to 3 listed below. The functions of 1 to 3 are planned not to be supported in the future SuperH-family products. The exception generation functions of EXPMASK can be used in advance of execution; the detection function then checks for the use of these functions in the software. This will ease the transfer of software to the future SuperH-family products that do not support the respective functions.

1. Handling of an instruction other than the NOP instruction in the delay slot of the RTE instruction.
2. Handling of the SLEEP instruction in the delay slot of the branch instruction.
3. Performance of IC/OC memory-mapped associative write operations.

According to the value of EXPMASK, functions 1 and 2 can generate a slot illegal instruction exception, and 3 can generate a data address error exception.

Generation of each exception can be disabled by writing 1 to the corresponding bit in EXPMASK. However, it is recommended that the above functions should not be used when making a program to maintain the compatibility with the future products.

Use the store instruction of the CPU to update EXPMASK. After updating the register and then reading the register once, execute either of the following instructions. Executing either instruction guarantees the operation with the updated register value.

- Execute the RTE instruction.
- Execute the ICBI instruction for any address (including non-cacheable area).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	–	–	–	–	–	–	–	–	–	–	–	MM CAW	–	–	BRDS SLP	RTE DS
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved For details on reading/writing these bits, see General Precautions on Handling of Product.
4	MMCAW	1	R/W	Memory-Mapped Cache Associative Write 0: Memory-mapped cache associative write is disabled. (A data address error exception will occur.) 1: Memory-mapped cache associative write is enabled.
3, 2	—	All 0	R	Reserved For details on reading/writing these bits, see General Precautions on Handling of Product.
1	BRDSSLP	1	R/W	Delay Slot SLEEP Instruction 0: The SLEEP instruction in the delay slot is disabled. (The SLEEP instruction is taken as a slot illegal instruction.) 1: The SLEEP instruction in the delay slot is enabled.
0	RTEDS	1	R/W	RTE Delay Slot 0: An instruction other than the NOP instruction in the delay slot of the RTE instruction is disabled. (An instruction other than the NOP instruction is taken as a slot illegal instruction). 1: An instruction other than the NOP instruction in the delay slot of the RTE instruction is enabled.

2.5.3 Exception Handling Functions

(1) Exception Handling Flow

In exception handling, the contents of the program counter (PC), status register (SR), and R15 are saved in the saved program counter (SPC), saved status register (SSR), and saved general register 15 (SGR), and the CPU starts execution of the appropriate exception handling routine according to the vector address. An exception handling routine is a program written by the user to handle a specific exception. The exception handling routine is terminated and control returned to the original program by executing a return-from-exception instruction (RTE). This instruction restores the PC and SR contents and returns control to the normal processing routine at the point at which the exception occurred. The SGR contents are not written back to R15 with an RTE instruction.

The basic processing flow is as follows. For the meaning of the SR bits, see section 2.2, Programming Model.

1. The PC, SR, and R15 contents are saved in SPC, SSR, and SGR, respectively.
2. The block bit (BL) in SR is set to 1.
3. The mode bit (MD) in SR is set to 1.
4. The register bank bit (RB) in SR is set to 1.
5. In a reset, the FPU disable bit (FD) in SR is cleared to 0.
6. The exception code is written to bits 11 to 0 of the exception event register (EXPEVT) or interrupt event register (INTEVT).
7. When the interrupt mode switch bit (INTMU) in CPUOPM has been 1, the interrupt mask level bit (IMASK) in SR is changed to accepted interrupt level.
8. The CPU branches to the determined exception handling vector address, and the exception handling routine begins.

(2) Exception Handling Vector Addresses

The reset vector address is fixed at H'A0000000. Exception and interrupt vector addresses are determined by adding the offset for the specific event to the vector base address, which is set by software in the vector base register (VBR). In the case of the TLB miss exception, for example, the offset is H'00000400, so if H'9C080000 is set in VBR, the exception handling vector address will be H'9C080400. If a further exception occurs at the exception handling vector address, a duplicate exception will result, and recovery will be difficult; therefore, addresses that are not to be converted (in P1 and P2 areas) should be specified for vector addresses.

2.5.4 Exception Types and Priorities

Table 2.25 shows the types of exceptions, with their relative priorities, vector addresses, and exception/interrupt codes.

Table 2.25 Exceptions

Exception Category	Execution Mode	Exception	Priority Level*2	Priority Order*2	Exception Transition Direction*3		Exception Code*4
					Vector Address	Offset	
Reset	Abort type	Power-on reset	1	1	H'A000 0000	—	H'000
		Manual reset	1	2	H'A000 0000	—	H'020
		H-UDI reset	1	1	H'A000 0000	—	H'000
		Instruction TLB multiple-hit exception	1	2	H'A000 0000	—	H'140
		Data TLB multiple-hit exception	1	3	H'A000 0000	—	H'140
General exception	Re-execution type	User break before instruction execution*	2	0	(VBR/DBR)	H'100/—	H'1E0
		Instruction address error	2	1	(VBR)	H'100	H'0E0
		Instruction TLB miss exception	2	2	(VBR)	H'400	H'040
		Instruction TLB protection violation exception	2	3	(VBR)	H'100	H'0A0
		General illegal instruction exception	2	4	(VBR)	H'100	H'180
		Slot illegal instruction exception	2	4	(VBR)	H'100	H'1A0
		General FPU disable exception	2	4	(VBR)	H'100	H'800
		Slot FPU disable exception	2	4	(VBR)	H'100	H'820
		Data address error (read)	2	5	(VBR)	H'100	H'0E0
		Data address error (write)	2	5	(VBR)	H'100	H'100
		Data TLB miss exception (read)	2	6	(VBR)	H'400	H'040
		Data TLB miss exception (write)	2	6	(VBR)	H'400	H'060
		Data TLB protection violation exception (read)	2	7	(VBR)	H'100	H'0A0
		Data TLB protection violation exception (write)	2	7	(VBR)	H'100	H'0C0
		FPU exception	2	8	(VBR)	H'100	H'120
		Initial page write exception	2	9	(VBR)	H'100	H'080

Exception Category	Execution Mode	Exception	Priority Level*2	Priority Order*2	Exception Transition Direction*3 Vector Address	Exception Transition Direction*3 Offset	Exception Code*4
General exception	Completion type	Unconditional trap (TRAPA)	2	4	(VBR)	H'100	H'160
		User break after instruction execution*	2	10	(VBR/DBR)	H'100/—	H'1E0
Interrupt	Completion type	Nonmaskable interrupt	3	—	(VBR)	H'600	H'1C0
		General interrupt request	4	—	(VBR)	H'600	—

Note: 1. When CBCR.UBDE = 1, PC = DBR. In other cases, PC = VBR + H'100.
2. Priority is first assigned by priority level, then by priority order within each level (the lowest number represents the highest priority).
3. Control passes to H'A000 0000 in a reset, and to [VBR + offset] in other cases.
4. Stored in EXPEVT for a reset or general exception, and in INTEVT for an interrupt.

2.5.5 Exception Flow

(1) Exception Flow

Figure 2.20 shows an outline flowchart of the basic operations in instruction execution and exception handling. For the sake of clarity, the following description assumes that instructions are executed sequentially, one by one. Figure 2.20 shows the relative priority order of the different kinds of exceptions (reset, general exception, and interrupt). Register settings in the event of an exception are shown only for SSR, SPC, SGR, EXPEVT/INTEVT, SR, and PC. However, other registers may be set automatically by hardware, depending on the exception. For details, see section 2.5.6, Description of Exceptions. Also, see section 2.5.6 (4), Priority Order with Multiple Exceptions, for exception handling during execution of a delayed branch instruction and a delay slot instruction, or in the case of instructions in which two data accesses are performed.

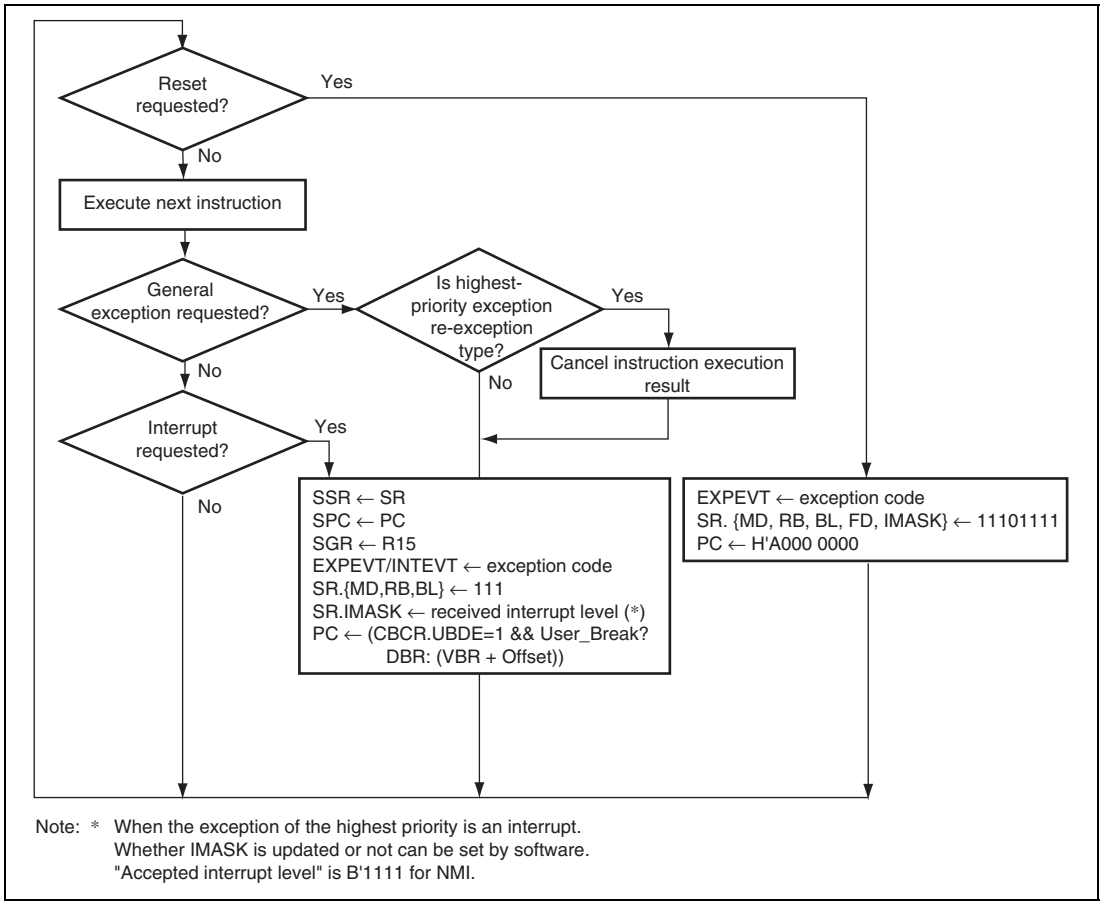


Figure 2.20 Instruction Execution and Exception Handling

(2) Exception Source Acceptance

A priority ranking is provided for all exceptions for use in determining which of two or more simultaneously generated exceptions should be accepted. Five of the general exceptions—general illegal instruction exception, slot illegal instruction exception, general FPU disable exception, slot FPU disable exception, and unconditional trap exception—are detected in the process of instruction decoding, and do not occur simultaneously in the instruction pipeline. These exceptions therefore all have the same priority. General exceptions are detected in the order of instruction execution. However, exception handling is performed in the order of instruction flow (program order). Thus, an exception for an earlier instruction is accepted before that for a later instruction. An example of the order of acceptance for general exceptions is shown in figure 2.21.

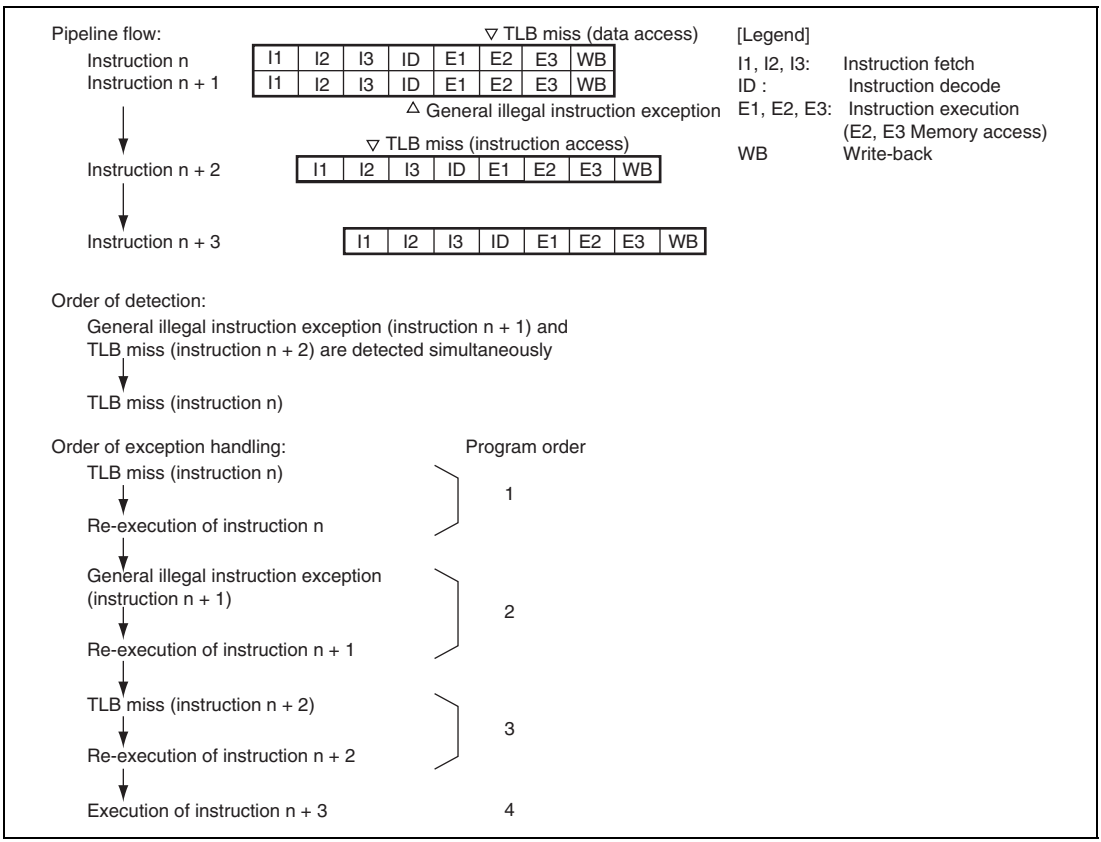


Figure 2.21 Example of General Exception Acceptance Order

(3) Exception Requests and BL Bit

When the BL bit in SR is 0, general exceptions and interrupts are accepted.

When the BL bit in SR is 1 and an general exception other than a user break is generated, the CPU's internal registers and the registers of the other modules are set to their states following a manual reset, and the CPU branches to the same address as in a reset (H'A0000000). For the operation in the event of a user break, see section 38, User Break Controller (UBC). If an ordinary interrupt occurs, the interrupt request is held pending and is accepted after the BL bit has been cleared to 0 by software. If a nonmaskable interrupt (NMI) occurs, it can be held pending or accepted according to the setting made by software.

Thus, normally, SPC and SSR are saved and then the BL bit in SR is cleared to 0, to enable multiple exception state acceptance.

(4) Return from Exception Handling

The RTE instruction is used to return from exception handling. When the RTE instruction is executed, the SPC contents are restored to PC and the SSR contents to SR, and the CPU returns from the exception handling routine by branching to the SPC address. If SPC and SSR were saved to external memory, set the BL bit in SR to 1 before restoring the SPC and SSR contents and issuing the RTE instruction.

2.5.6 Description of Exceptions

The various exception handling operations explained here are exception sources, transition address on the occurrence of exception, and processor operation when a transition is made.

(1) Resets

(a) Power-On Reset

- Condition:
  Power-on reset request
- Operations:
  Exception code H'000 is set in EXPEVT, initialization of the CPU and on-chip peripheral module is carried out, and then a branch is made to the reset vector (H'A0000000). For details, see the register descriptions in the relevant sections. A power-on reset should be executed when power is supplied.

(b) Manual Reset

- Condition:
  Manual reset request
- Operations:
  Exception code H'020 is set in EXPEVT, initialization of the CPU and on-chip peripheral module is carried out, and then a branch is made to the branch vector (H'A0000000). The registers initialized by a power-on reset and manual reset are different. For details, see the register descriptions in the relevant sections.

(c) H-UDI Reset

- Source: SDIR.TI[7:4] = B'0110 (negation) or B'0111 (assertion)
- Transition address: H'A0000000
- Transition operations:
  Exception code H'000 is set in EXPEVT, initialization of VBR and SR is performed, and a branch is made to PC = H'A0000000.
  CPU and on-chip peripheral module initialization is performed. For details, see section 39, User Debugging Interface (H-UDI), and the register descriptions in the relevant sections.

(d) Instruction TLB Multiple Hit Exception

- Source: Multiple ITLB address matches
- Transition address: H'A0000000
- Transition operations:

  The virtual address (32 bits) at which this exception occurred is set in TEA, and the corresponding virtual page number (22 bits) is set in PTEH [31:10]. ASID in PTEH indicates the ASID when this exception occurred.

  Exception code H'140 is set in EXPEVT, initialization of VBR and SR is performed, and a branch is made to PC = H'A0000000.

  CPU and on-chip peripheral module initialization is performed in the same way as in a manual reset. For details, see the register descriptions in the relevant sections.

(e) Data TLB Multiple-Hit Exception

- Source: Multiple UTLB address matches
- Transition address: H'A0000000
- Transition operations:

  The virtual address (32 bits) at which this exception occurred is set in TEA, and the corresponding virtual page number (22 bits) is set in PTEH [31:10]. ASID in PTEH indicates the ASID when this exception occurred.

  Exception code H'140 is set in EXPEVT, initialization of VBR and SR is performed, and a branch is made to PC = H'A0000000.

  CPU and on-chip peripheral module initialization is performed in the same way as in a manual reset. For details, see the register descriptions in the relevant sections.

(2) General Exceptions

(a) Data TLB Miss Exception

- Source: Address mismatch in UTLB address comparison
- Transition address: VBR + H'00000400
- Transition operations:

  The virtual address (32 bits) at which this exception occurred is set in TEA, and the corresponding virtual page number (22 bits) is set in PTEH [31:10]. ASID in PTEH indicates the ASID when this exception occurred.

  The PC and SR contents for the instruction at which this exception occurred are saved in SPC and SSR. The R15 contents at this time are saved in SGR.

  Exception code H'040 (for a read access) or H'060 (for a write access) is set in EXPEVT. The BL, MD, and RB bits are set to 1 in SR, and a branch is made to PC = VBR + H'0400.

  To speed up TLB miss processing, the offset is separate from that of other exceptions.

```
Data_TLB_miss_exception()
{
    TEA = EXCEPTION_ADDRESS;
    PTEH.VPN = PAGE_NUMBER;
    SPC = PC;
    SSR = SR;
    SGR = R15;
    EXPEVT = read_access ? H'0000 0040 : H'0000 0060;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = VBR + H'0000 0400;
}
```

(b) Instruction TLB Miss Exception

- Source: Address mismatch in ITLB address comparison
- Transition address: VBR + H'00000400
- Transition operations:

  The virtual address (32 bits) at which this exception occurred is set in TEA, and the corresponding virtual page number (22 bits) is set in PTEH [31:10]. ASID in PTEH indicates the ASID when this exception occurred.

  The PC and SR contents for the instruction at which this exception occurred are saved in SPC and SSR. The R15 contents at this time are saved in SGR.

  Exception code H'40 is set in EXPEVT. The BL, MD, and RB bits are set to 1 in SR, and a branch is made to PC = VBR + H'0400.

  To speed up TLB miss processing, the offset is separate from that of other exceptions.

```
ITLB_miss_exception()
{
    TEA = EXCEPTION_ADDRESS;
    PTEH.VPN = PAGE_NUMBER;
    SPC = PC;
    SSR = SR;
    SGR = R15;
    EXPEVT = H'0000 0040;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = VBR + H'0000 0400;
}
```

(c) Initial Page Write Exception

- Source: TLB is hit in a store access, but dirty bit D = 0
- Transition address: VBR + H'00000100
- Transition operations:

  The virtual address (32 bits) at which this exception occurred is set in TEA, and the corresponding virtual page number (22 bits) is set in PTEH [31:10]. ASID in PTEH indicates the ASID when this exception occurred.

  The PC and SR contents for the instruction at which this exception occurred are saved in SPC and SSR. The R15 contents at this time are saved in SGR.

  Exception code H'080 is set in EXPEVT. The BL, MD, and RB bits are set to 1 in SR, and a branch is made to PC = VBR + H'0100.

```
Initial_write_exception()
{
    TEA = EXCEPTION_ADDRESS;
    PTEH.VPN = PAGE_NUMBER;
    SPC = PC;
    SSR = SR;
    SGR = R15;
    EXPEVT = H'0000 0080;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = VBR + H'0000 0100;
}
```

(d) Data TLB Protection Violation Exception

- Source: The access does not accord with the UTLB protection information (PR bits or EPR bits) shown in table 2.26 and table 2.27.

Table 2.26 UTLB Protection Information (TLB Compatible Mode)

PR	Privileged Mode	User Mode
00	Only read access possible	Access not possible
01	Read/write access possible	Access not possible
10	Only read access possible	Only read access possible
11	Read/write access possible	Read/write access possible

Table 2.27 UTLB Protection Information (TLB Extended Mode)

EPR [5]	Read Permission in Privileged Mode
0	Read access possible
1	Read access not possible

EPR [4]	Write Permission in Privileged Mode
0	Write access possible
1	Write access not possible

EPR [2]	Read Permission in User Mode
0	Read access possible
1	Read access not possible

EPR [1]	Write Permission in User Mode
0	Write access possible
1	Write access not possible

- Transition address: VBR + H'00000100
- Transition operations:

  The virtual address (32 bits) at which this exception occurred is set in TEA, and the corresponding virtual page number (22 bits) is set in PTEH [31:10]. ASID in PTEH indicates the ASID when this exception occurred.

The PC and SR contents for the instruction at which this exception occurred are saved in SPC and SSR. The R15 contents at this time are saved in SGR.

Exception code H'0A0 (for a read access) or H'0C0 (for a write access) is set in EXPEVT. The BL, MD, and RB bits are set to 1 in SR, and a branch is made to PC = VBR + H'0100.

```
Data_TLB_protection_violation_exception()
{
    TEA = EXCEPTION_ADDRESS;
    PTEH.VPN = PAGE_NUMBER;
    SPC = PC;
    SSR = SR;
    SGR = R15;
    EXPEVT = read_access ? H'0000 00A0 : H'0000 00C0;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = VBR + H'0000 0100;
}
```

(e) Instruction TLB Protection Violation Exception

- Source: The access does not accord with the ITLB protection information (PR bits or EPR bits) shown in table 2.28 and table 2.29.

Table 2.28 ITLB Protection Information (TLB Compatible Mode)

PR	Privileged Mode	User Mode
0	Access possible	Access not possible
1	Access possible	Access possible

Table 2.29 ITLB Protection Information (TLB Extended Mode)

EPR [5], EPR [3]	Execution Permission in Privileged Mode
11, 01	Execution of instructions possible
10	Instruction fetch not possible Execution of Rn access by ICBI possible
00	Execution of instructions not possible

EPR [2], EPR [0]	Execution Permission in User Mode
11, 01	Execution of instructions possible
10	Instruction fetch not possible Execution of Rn access by ICBI possible
00	Execution of instructions not possible

- Transition address: VBR + H'00000100
- Transition operations:

  The virtual address (32 bits) at which this exception occurred is set in TEA, and the corresponding virtual page number (22 bits) is set in PTEH [31:10]. ASID in PTEH indicates the ASID when this exception occurred.

  The PC and SR contents for the instruction at which this exception occurred are saved in SPC and SSR. The R15 contents at this time are saved in SGR.

  Exception code H'0A0 is set in EXPEVT. The BL, MD, and RB bits are set to 1 in SR, and a branch is made to PC = VBR + H'0100.

```
ITLB_protection_violation_exception()
{
    TEA = EXCEPTION_ADDRESS;
    PTEH.VPN = PAGE_NUMBER;
    SPC = PC;
    SSR = SR;
    SGR = R15;
    EXPEVT = H'0000 00A0;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = VBR + H'0000 0100;
}
```

(f) Data Address Error

- Sources:
  - Word data access from other than a word boundary (2n +1)
  - Longword data access from other than a longword data boundary (4n +1, 4n + 2, or 4n +3) (Except MOVLIA)
  - Quadword data access from other than a quadword data boundary (8n +1, 8n + 2, 8n +3, 8n + 4, 8n + 5, 8n + 6, or 8n + 7)
  - Access to area H'80000000 to H'FFFFFFFF in user mode

    Areas H'E0000000 to H'E3FFFFFF and H'E5000000 to H'E5FFFFFF can be accessed in user mode. For details, see section 2.7, Memory Management Unit (MMU) and section 2.9, On-Chip Memory.
  - The MMCAW bit in EXPMASK is 0, and the IC/OC memory mapped associative write is performed.
- Transition address: VBR + H'0000100

- Transition operations:

  The virtual address (32 bits) at which this exception occurred is set in TEA, and the corresponding virtual page number (22 bits) is set in PTEH [31:10]. ASID in PTEH indicates the ASID when this exception occurred.

  The PC and SR contents for the instruction at which this exception occurred are saved in SPC and SSR. The R15 contents at this time are saved in SGR.

  Exception code H'0E0 (for a read access) or H'100 (for a write access) is set in EXPEVT. The BL, MD, and RB bits are set to 1 in SR, and a branch is made to PC = VBR + H'0100. For details, see section 2.7, Memory Management Unit (MMU).

```
Data_address_error()
{
    TEA = EXCEPTION_ADDRESS;
    PTEH.VPN = PAGE_NUMBER;
    SPC = PC;
    SSR = SR;
    SGR = R15;
    EXPEVT = read_access? H'0000 00E0: H'0000 0100;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = VBR + H'0000 0100;
}
```

(g) Instruction Address Error

- Sources:
  - Instruction fetch from other than a word boundary (2n +1)
  - Instruction fetch from area H'80000000 to H'FFFFFFFF in user mode

    Area H'E5000000 to H'E5FFFFFF can be accessed in user mode. For details, see section 2.9, On-Chip Memory.
- Transition address: VBR + H'00000100
- Transition operations:

  The virtual address (32 bits) at which this exception occurred is set in TEA, and the corresponding virtual page number (22 bits) is set in PTEH [31:10]. ASID in PTEH indicates the ASID when this exception occurred.

  The PC and SR contents for the instruction at which this exception occurred are saved in the SPC and SSR. The R15 contents at this time are saved in SGR.

  Exception code H'0E0 is set in EXPEVT. The BL, MD, and RB bits are set to 1 in SR, and a branch is made to PC = VBR + H'0100. For details, see section 2.7, Memory Management Unit (MMU).

```
Instruction_address_error()
{
    TEA = EXCEPTION_ADDRESS;
    PTEH.VPN = PAGE_NUMBER;
    SPC = PC;
    SSR = SR;
    SGR = R15;
    EXPEVT = H'0000 00E0;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = VBR + H'0000 0100;
}
```

(h) Unconditional Trap

- Source: Execution of TRAPA instruction
- Transition address: VBR + H'00000100
- Transition operations:

  As this is a processing-completion-type exception, the PC contents for the instruction following the TRAPA instruction are saved in SPC. The value of SR and R15 when the TRAPA instruction is executed are saved in SSR and SGR. The 8-bit immediate value in the TRAPA instruction is multiplied by 4, and the result is set in TRA [9:0]. Exception code H'160 is set in EXPEVT. The BL, MD, and RB bits are set to 1 in SR, and a branch is made to PC = VBR + H'0100.

```
TRAPA_exception()
{
    SPC = PC + 2;
    SSR = SR;
    SGR = R15;
    TRA = imm << 2;
    EXPEVT = H'0000 0160;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = VBR + H'0000 0100;
}
```

(i) General Illegal Instruction Exception

- Sources:
  - Decoding of an undefined instruction not in a delay slot
    Delayed branch instructions: JMP, JSR, BRA, BRAF, BSR, BSRF, RTS, RTE, BT/S, BF/S
    Undefined instruction: H'FFFD
  - Decoding in user mode of a privileged instruction not in a delay slot
    Privileged instructions: LDC, STC, RTE, LDTLB, SLEEP, but excluding LDC/STC instructions that access GBR
- Transition address: VBR + H'00000100
- Transition operations:
  The PC and SR contents for the instruction at which this exception occurred are saved in SPC and SSR. The R15 contents at this time are saved in SGR.
  Exception code H'180 is set in EXPEVT. The BL, MD, and RB bits are set to 1 in SR, and a branch is made to PC = VBR + H'0100. Operation is not guaranteed if an undefined code other than H'FFFD is decoded.

```
General_illegal_instruction_exception()
{
    SPC = PC;
    SSR = SR;
    SGR = R15;
    EXPEVT = H'0000 0180;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = VBR + H'0000 0100;
}
```

(j) Slot Illegal Instruction Exception

- Sources:
  - — Decoding of an undefined instruction in a delay slot
    Delayed branch instructions: JMP, JSR, BRA, BRAF, BSR, BSRF, RTS, RTE, BT/S, BF/S
    Undefined instruction: H'FFFD
  - — Decoding of an instruction that modifies PC in a delay slot
    Instructions that modify PC: JMP, JSR, BRA, BRAF, BSR, BSRF, RTS, RTE, BT, BF, BT/S, BF/S, TRAPA, LDC Rm,SR, LDC.L @Rm+,SR, ICBI, PREFI
  - — Decoding in user mode of a privileged instruction in a delay slot
    Privileged instructions: LDC, STC, RTE, LDTLB, SLEEP, but excluding LDC/STC instructions that access GBR
  - — Decoding of a PC-relative MOV instruction or MOVA instruction in a delay slot
  - — The BRDSSLP bit in EXPMASK is 0, and the SLEEP instruction in the delay slot is executed.
  - — The RTEDS bit in EXPMASK is 0, and an instruction other than the NOP instruction in the delay slot is executed.
- Transition address: VBR + H'000 0100
- Transition operations:
  The PC contents for the preceding delayed branch instruction are saved in SPC. The SR and R15 contents when this exception occurred are saved in SSR and SGR.
  Exception code H'1A0 is set in EXPEVT. The BL, MD, and RB bits are set to 1 in SR, and a branch is made to PC = VBR + H'0100. Operation is not guaranteed if an undefined code other than H'FFFD is decoded.

```
Slot_illegal_instruction_exception()
{
    SPC = PC - 2;
    SSR = SR;
    SGR = R15;
    EXPEVT = H'0000 01A0;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = VBR + H'0000 0100;
}
```

(k) General FPU Disable Exception

- Source: Decoding of an FPU instruction* not in a delay slot with SR.FD = 1
- Transition address: VBR + H'00000100
- Transition operations:

  The PC and SR contents for the instruction at which this exception occurred are saved in SPC and SSR. The R15 contents at this time are saved in SGR.

  Exception code H'800 is set in EXPEVT. The BL, MD, and RB bits are set to 1 in SR, and a branch is made to PC = VBR + H'0100.

Note: * FPU instructions are instructions in which the first 4 bits of the instruction code are F (but excluding undefined instruction H'FFFD), and the LDS, STS, LDS.L, and STS.L instructions corresponding to FPUL and FPSCR.

```
General_fpu_disable_exception()
{
    SPC = PC;
    SSR = SR;
    SGR = R15;
    EXPEVT = H'0000 0800;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = VBR + H'0000 0100;
}
```

(l) Slot FPU Disable Exception

- Source: Decoding of an FPU instruction in a delay slot with SR.FD =1
- Transition address: VBR + H'00000100
- Transition operations:

  The PC contents for the preceding delayed branch instruction are saved in SPC. The SR and R15 contents when this exception occurred are saved in SSR and SGR.

  Exception code H'820 is set in EXPEVT. The BL, MD, and RB bits are set to 1 in SR, and a branch is made to PC = VBR + H'0100.

```
Slot_fpu_disable_exception()
{
    SPC = PC - 2;
    SSR = SR;
    SGR = R15;
    EXPEVT = H'0000 0820;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = VBR + H'0000 0100;
}
```

(m) Pre-Execution User Break/Post-Execution User Break

- Source: Fulfilling of a break condition set in the user break controller
- Transition address: VBR + H'00000100, or DBR
- Transition operations:

  In the case of a post-execution break, the PC contents for the instruction following the instruction at which the breakpoint is set are set in SPC. In the case of a pre-execution break, the PC contents for the instruction at which the breakpoint is set are set in SPC.

  The SR and R15 contents when the break occurred are saved in SSR and SGR. Exception code H'1E0 is set in EXPEVT.

  The BL, MD, and RB bits are set to 1 in SR, and a branch is made to PC = VBR + H'0100. It is also possible to branch to PC = DBR.

  For details of PC, etc., when a data break is set, see section 38, User Break Controller (UBC).

```
User_break_exception()
{
    SPC = (pre_execution break? PC : PC + 2);
    SSR = SR;
    SGR = R15;
    EXPEVT = H'0000 01E0;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = (CBCR.UBDE==1 ? DBR : VBR + H’0000 0100);
}
```

(n) FPU Exception

- Source: Exception due to execution of a floating-point operation
- Transition address: VBR + H'00000100
- Transition operations:

  The PC and SR contents for the instruction at which this exception occurred are saved in SPC and SSR . The R15 contents at this time are saved in SGR. Exception code H'120 is set in EXPEVT. The BL, MD, and RB bits are set to 1 in SR, and a branch is made to PC = VBR + H'0100.

```
FPU_exception()
{
    SPC = PC;
    SSR = SR;
    SGR = R15;
    EXPEVT = H'0000 0120;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = VBR + H'0000 0100;
}
```

(3) Interrupts

(a) NMI (Nonmaskable Interrupt)

- Source: NMI pin edge detection
- Transition address: VBR + H'00000600
- Transition operations:

  The PC and SR contents for the instruction immediately after this exception is accepted are saved in SPC and SSR. The R15 contents at this time are saved in SGR.

  Exception code H'1C0 is set in INTEVT. The BL, MD, and RB bits are set to 1 in SR, and a branch is made to PC = VBR + H'0600. When the BL bit in SR is 0, this interrupt is not masked by the interrupt mask bits in SR, and is accepted at the highest priority level. When the BL bit in SR is 1, a software setting can specify whether this interrupt is to be masked or accepted. When the INTMU bit in CPUOPM is 1 and the NMI interrupt is accessed, B'1111 is set to IMASK bit in SR. For details, see section 7, INTC/INTC2.

```
NMI()
{
    SPC = PC;
    SSR = SR;
    SGR = R15;
    INTEVT = H'0000 01C0;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    If (cond) SR.IMASK = B'1111;
    PC = VBR + H'0000 0600;
}
```

(b) General Interrupt Request

- Source: The interrupt mask level bits setting in SR is smaller than the interrupt level of interrupt request, and the BL bit in SR is 0 (accepted at instruction boundary).
- Transition address: VBR + H'00000600
- Transition operations:

  The PC contents immediately after the instruction at which the interrupt is accepted are set in SPC. The SR and R15 contents at the time of acceptance are set in SSR and SGR.

The code corresponding to the each interrupt source is set in INTEVT. The BL, MD, and RB bits are set to 1 in SR, and a branch is made to VBR + H'0600. When the INTMU bit in CPUOPM is 1, IMASK bit in SR is changed to accepted interrupt level. For details, see section 7, INTC/INTC2.

```
Module_interruption()
{
    SPC = PC;
    SSR = SR;
    SGR = R15;
    INTEVT = H'0000 0400 ~ H'0000 3FE0;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    if (cond) SR.IMASK = level_of accepted_interrupt ();
    PC = VBR + H'0000 0600;
}
```

(4) Priority Order with Multiple Exceptions

With some instructions, such as instructions that make two accesses to memory, and the indivisible pair comprising a delayed branch instruction and delay slot instruction, multiple exceptions occur. Care is required in these cases, as the exception priority order differs from the normal order.

(a) Instructions that Make Two Accesses to Memory

With MAC instructions, memory-to-memory arithmetic/logic instructions, TAS instructions, and MOVUA instructions, two data transfers are performed by a single instruction, and an exception will be detected for each of these data transfers. In these cases, therefore, the following order is used to determine priority.

1. Data address error in first data transfer
2. TLB miss in first data transfer
3. TLB protection violation in first data transfer
4. Initial page write exception in first data transfer
5. Data address error in second data transfer
6. TLB miss in second data transfer
7. TLB protection violation in second data transfer

8. Initial page write exception in second data transfer

(b) Indivisible Delayed Branch Instruction and Delay Slot Instruction

As a delayed branch instruction and its associated delay slot instruction are indivisible, they are treated as a single instruction. Consequently, the priority order for exceptions that occur in these instructions differs from the usual priority order. The priority order shown below is for the case where the delay slot instruction has only one data transfer.

1. A check is performed for the interrupt type and re-execution type exceptions of priority levels 1 and 2 in the delayed branch instruction.
2. A check is performed for the interrupt type and re-execution type exceptions of priority levels 1 and 2 in the delay slot instruction.
3. A check is performed for the completion type exception of priority level 2 in the delayed branch instruction.
4. A check is performed for the completion type exception of priority level 2 in the delay slot instruction.
5. A check is performed for priority level 3 in the delayed branch instruction and priority level 3 in the delay slot instruction. (There is no priority ranking between these two.)
6. A check is performed for priority level 4 in the delayed branch instruction and priority level 4 in the delay slot instruction. (There is no priority ranking between these two.)

If the delay slot instruction has a second data transfer, two checks are performed in step 2, as in the above case (Instructions that make two accesses to memory).

If the accepted exception (the highest-priority exception) is a delay slot instruction re-execution type exception, the branch instruction PR register write operation (PC → PR operation performed in a BSR, BSRF, or JSR instruction) is not disabled. Note that in this case, the contents of PR register are not guaranteed.

2.5.7 Usage Notes

(1) Return from Exception Handling

A. Check the BL bit in SR with software. If SPC and SSR have been saved to memory, set the BL bit in SR to 1 before restoring them.

B. Issue an RTE instruction. When RTE is executed, the SPC contents are saved in PC, the SSR contents are saved in SR, and branch is made to the SPC address to return from the exception handling routine.

(2) If a General Exception or Interrupt Occurs When BL Bit in SR = 1

A. General exception

When a general exception other than a user break occurs, the PC value for the instruction at which the exception occurred in SPC, and a manual reset is executed. The value in EXPEVT at this time is H'00000020; the SSR contents are undefined.

B. Interrupt

If an ordinary interrupt occurs, the interrupt request is held pending and is accepted after the BL bit in SR has been cleared to 0 by software. If a nonmaskable interrupt (NMI) occurs, it can be held pending or accepted according to the setting made by software.

In sleep mode, however, an interrupt is accepted even if the BL bit in SR is set to 1.

(3) SPC when an Exception Occurs

A. Re-execution type general exception

The PC value for the instruction at which the exception occurred is set in SPC, and the instruction is re-executed after returning from the exception handling routine. If an exception occurs in a delay slot instruction, however, the PC value for the delayed branch instruction is saved in SPC regardless of whether or not the preceding delay slot instruction condition is satisfied.

B. Completion type general exception or interrupt

The PC value for the instruction following that at which the exception occurred is set in SPC. If an exception occurs in a branch instruction with delay slot, however, the PC value for the branch destination is saved in SPC.

(4) RTE Instruction Delay Slot

A. The instruction in the delay slot of the RTE instruction is executed only after the value saved in SSR has been restored to SR. The acceptance of the exception related to the instruction access is determined depending on SR before restoring, while the acceptance of other exceptions is determined depending on the processing mode by SR after restoring or the BL bit. The completion type exception is accepted before branching to the destination of RTE instruction. However, if the re-execution type exception is occurred, the operation cannot be guaranteed.

B. The user break is not accepted by the instruction in the delay slot of the RTE instruction.

(5) Changing the SR Register Value and Accepting Exception

A. When the MD or BL bit in the SR register is changed by the LDC instruction, the acceptance of the exception is determined by the changed SR value, starting from the next instruction.* In the completion type exception, an exception is accepted after the next instruction has been executed. However, an interrupt of completion type exception is accepted before the next instruction is executed.

Note: * When the LDC instruction for SR is executed, following instructions are fetched again and the instruction fetch exception is evaluated again by the changed SR.

2.6 Floating-Point Unit (FPU)

2.6.1 Features

The FPU has the following features.

- Conforms to IEEE754 standard
- 32 single-precision floating-point registers (can also be referenced as 16 double-precision registers)
- Two rounding modes: Round to Nearest and Round to Zero
- Two denormalization modes: Flush to Zero and Treat Denormalized Number
- Six exception sources: FPU Error, Invalid Operation, Divide By Zero, Overflow, Underflow, and Inexact
- Comprehensive instructions: Single-precision, double-precision, graphics support, and system control
- In the SH-4A, the following three instructions are added on to the instruction set of the SH-4 FSRRA, FSCA, and FPCHG

When the FD bit in SR is set to 1, the FPU cannot be used, and an attempt to execute an FPU instruction will cause an FPU disable exception (general FPU disable exception or slot FPU disable exception).

2.6.2 Data Formats

(1) Floating-Point Format

A floating-point number consists of the following three fields:

- Sign bit (s)
- Exponent field (e)
- Fraction field (f)

The SH-4A can handle single-precision and double-precision floating-point numbers, using the formats shown in figures 2.22 and 2.23.

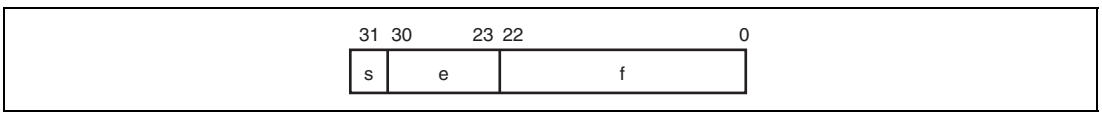


Figure 2.22 Format of Single-Precision Floating-Point Number

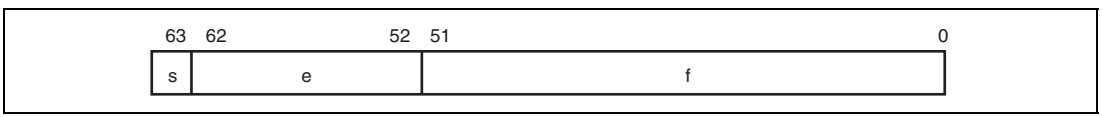


Figure 2.23 Format of Double-Precision Floating-Point Number

The exponent is expressed in biased form, as follows:

$$e = E + bias$$

The range of unbiased exponent E is $E_{min} - 1$ to $E_{max} + 1$. The two values $E_{min} - 1$ and $E_{max} + 1$ are distinguished as follows. $E_{min} - 1$ indicates zero (both positive and negative sign) and a denormalized number, and $E_{max} + 1$ indicates positive or negative infinity or a non-number (NaN). Table 2.30 shows floating-point formats and parameters.

Table 2.30 Floating-Point Number Formats and Parameters

Parameter	Single-Precision	Double-Precision
Total bit width	32 bits	64 bits
Sign bit	1 bit	1 bit
Exponent field	8 bits	11 bits
Fraction field	23 bits	52 bits
Precision	24 bits	53 bits
Bias	+127	+1023
E_{max}	+127	+1023
E_{min}	–126	–1022

Floating-point number value v is determined as follows:

If $E = E_{max} + 1$ and $f \neq 0$, v is a non-number (NaN) irrespective of sign s
If $E = E_{max} + 1$ and $f = 0$, $v = (-1)^s$ (infinity) [positive or negative infinity]
If $E_{min} \leq E \leq E_{max}$, $v = (-1)^s 2^E$ (1.f) [normalized number]
If $E = E_{min} - 1$ and $f \neq 0$, $v = (-1)^s 2^{Emin}$ (0.f) [denormalized number]
If $E = E_{min} - 1$ and $f = 0$, $v = (-1)^s 0$ [positive or negative zero]

Table 2.31 shows the ranges of the various numbers in hexadecimal notation. For the signaling non-number and quiet non-number, see section 2.6.2 (2), Non-Numbers (NaN). For the denormalized number, see section 2.6.2 (3), Denormalized Numbers.

Table 2.31 Floating-Point Ranges

Type	Single-Precision	Double-Precision
Signaling non-number	H'7FFF FFFF to H'7FC0 0000	H'7FFF FFFF FFFF FFFF to H'7FF8 0000 0000 0000
Quiet non-number	H'7FBF FFFF to H'7F80 0001	H'7FF7 FFFF FFFF FFFF to H'7FF0 0000 0000 0001
Positive infinity	H'7F80 0000	H'7FF0 0000 0000 0000
Positive normalized number	H'7F7F FFFF to H'0080 0000	H'7FEF FFFF FFFF FFFF to H'0010 0000 0000 0000
Positive denormalized number	H'007F FFFF to H'0000 0001	H'000F FFFF FFFF FFFF to H'0000 0000 0000 0001
Positive zero	H'0000 0000	H'0000 0000 0000 0000
Negative zero	H'8000 0000	H'8000 0000 0000 0000
Negative denormalized number	H'8000 0001 to H'807F FFFF	H'8000 0000 0000 0001 to H'800F FFFF FFFF FFFF
Negative normalized number	H'8080 0000 to H'FF7F FFFF	H'8010 0000 0000 0000 to H'FFEF FFFF FFFF FFFF
Negative infinity	H'FF80 0000	H'FFF0 0000 0000 0000
Quiet non-number	H'FF80 0001 to H'FFBF FFFF	H'FFF0 0000 0000 0001 to H'FFF7 FFFF FFFF FFFF
Signaling non-number	H'FFC0 0000 to H'FFFF FFFF	H'FFF8 0000 0000 0000 to H'FFFF FFFF FFFF FFFF

(2) Non-Numbers (NaN)

Figure 2.24 shows the bit pattern of a non-number (NaN). A value is NaN in the following case:

- Sign bit: Don't care
- Exponent field: All bits are 1
- Fraction field: At least one bit is 1

The NaN is a signaling NaN (sNaN) if the MSB of the fraction field is 1, and a quiet NaN (qNaN) if the MSB is 0.

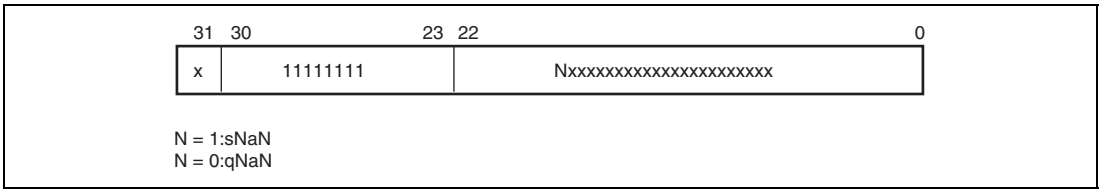


Figure 2.24 Single-Precision NaN Bit Pattern

An sNaN is assumed to be the input data in an operation, except the transfer instructions between registers, FABS, and FNEG, that generates a floating-point value.

- When the EN.V bit in FPSCR is 0, the operation result (output) is a qNaN.
- When the EN.V bit in FPSCR is 1, an invalid operation exception will be generated. In this case, the contents of the operation destination register are unchanged.

Following three instructions are used as transfer instructions between registers.

- FMOV FRm,FRn
- FLDS FRm,FPUL
- FSTS FPUL,FRn

If a qNaN is input in an operation that generates a floating-point value, and an sNaN has not been input in that operation, the output will always be a qNaN irrespective of the setting of the EN.V bit in FPSCR. An exception will not be generated in this case.

The qNAN values as operation results are as follows:

- Single-precision qNaN: H'7FBF FFFF
- Double-precision qNaN: H'7FF7 FFFF FFFF FFFF

See section 11, Instruction Descriptions of the SH-4A Extended Functions Software Manual for details of floating-point operations when a non-number (NaN) is input.

(3) Denormalized Numbers

For a denormalized number floating-point value, the exponent field is expressed as 0, and the fraction field as a non-zero value.

When the DN bit in FPSCR of the FPU is 1, a denormalized number (source operand or operation result) is always positive or negative zero in a floating-point operation that generates a value (an operation other than transfer instructions between registers, FNEG, or FABS).

When the DN bit in FPSCR is 0, a denormalized number (source operand or operation result) is processed as it is. See section 11, Instruction Descriptions of the SH-4A Extended Functions Software Manual for details of floating-point operations when a denormalized number is input.

2.6.3 Register Descriptions

(1) Floating-Point Registers

Figure 2.25 shows the floating-point register configuration. There are thirty-two 32-bit floating-point registers comprised with two banks: FPR0_BANK0 to FPR15_BANK0, and FPR0_BANK1 to FPR15_BANK1. These thirty-two registers are referenced as FR0 to FR15, DR0/2/4/6/8/10/12/14, FV0/4/8/12, XF0 to XF15, XD0/2/4/6/8/10/12/14, and XMTRX. Corresponding registers to FPR0_BANK0 to FPR15_BANK0, and FPR0_BANK1 to FPR15_BANK1 are determined according to the FR bit of FPSCR.

1. Floating-point registers, FPRi_BANKj (32 registers)
   FPR0_BANK0 to FPR15_BANK0
   FPR0_BANK1 to FPR15_BANK1

2. Single-precision floating-point registers, FRi (16 registers)
   When FPSCR.FR = 0, FR0 to FR15 are allocated to FPR0_BANK0 to FPR15_BANK0;
   when FPSCR.FR = 1, FR0 to FR15 are allocated to FPR0_BANK1 to FPR15_BANK1.

3. Double-precision floating-point registers, DRi (8 registers): A DR register comprises two FR registers.
   DR0 = {FR0, FR1}, DR2 = {FR2, FR3}, DR4 = {FR4, FR5}, DR6 = {FR6, FR7},
   DR8 = {FR8, FR9}, DR10 = {FR10, FR11}, DR12 = {FR12, FR13}, DR14 = {FR14, FR15}

4. Single-precision floating-point vector registers, FVi (4 registers): An FV register comprises four FR registers.
   FV0 = {FR0, FR1, FR2, FR3}, FV4 = {FR4, FR5, FR6, FR7},
   FV8 = {FR8, FR9, FR10, FR11}, FV12 = {FR12, FR13, FR14, FR15}

5. Single-precision floating-point extended registers, XFi (16 registers)
   When FPSCR.FR = 0, XF0 to XF15 are allocated to FPR0_BANK1 to FPR15_BANK1;
   when FPSCR.FR = 1, XF0 to XF15 are allocated to FPR0_BANK0 to FPR15_BANK0.

6. Double-precision floating-point extended registers, XDi (8 registers): An XD register comprises two XF registers.
   XD0 = {XF0, XF1}, XD2 = {XF2, XF3}, XD4 = {XF4, XF5}, XD6 = {XF6, XF7},
   XD8 = {XF8, XF9}, XD10 = {XF10, XF11}, XD12 = {XF12, XF13}, XD14 = {XF14, XF15}

7. Single-precision floating-point extended register matrix, XMTRX: XMTRX comprises all 16 XF registers.

$$XMTRX = \begin{bmatrix} XF0 & XF4 & XF8 & XF12 \\ XF1 & XF5 & XF9 & XF13 \\ XF2 & XF6 & XF10 & XF14 \\ XF3 & XF7 & XF11 & XF15 \end{bmatrix}$$

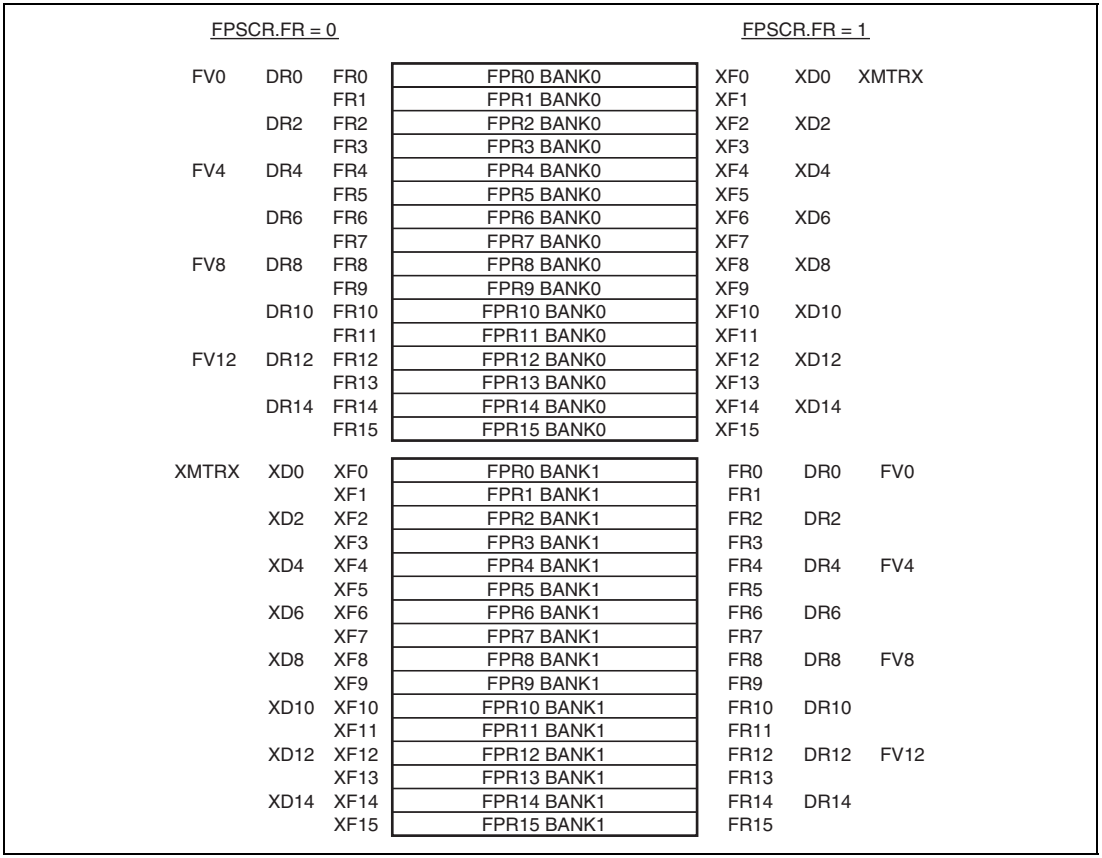


Figure 2.25 Floating-Point Registers

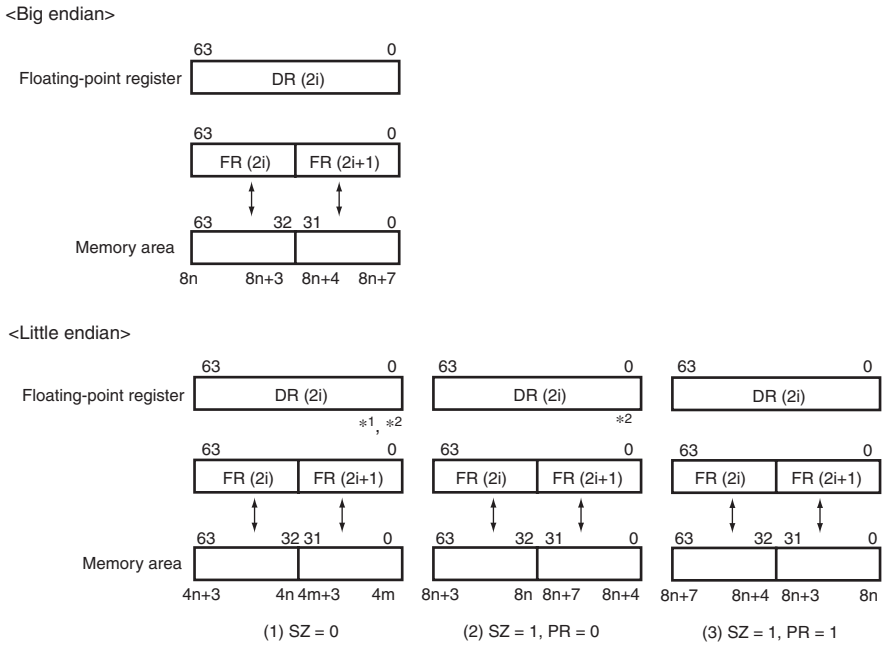
(2) Floating-Point Status/Control Register (FPSCR)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	FR	SZ	PR	DN	Cause	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Cause				Enable (EN)					Flag					RM	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21	FR	0	R/W	Floating-Point Register Bank 0: FPR0_BANK0 to FPR15_BANK0 are assigned to FR0 to FR15 and FPR0_BANK1 to FPR15_BANK1 are assigned to XF0 to XF15 1: FPR0_BANK0 to FPR15_BANK0 are assigned to XF0 to XF15 and FPR0_BANK1 to FPR15_BANK1 are assigned to FR0 to FR15
20	SZ	0	R/W	Transfer Size Mode 0: Data size of FMOV instruction is 32-bits 1: Data size of FMOV instruction is a 32-bit register pair (64 bits) For relations between endian and the SZ and PR bits, see figure 2.26.
19	PR	0	R/W	Precision Mode 0: Floating-point instructions are executed as single-precision operations 1: Floating-point instructions are executed as double-precision operations (graphics support instructions are undefined) For relations between endian and the SZ and PR bits, see figure 2.26.
18	DN	1	R/W	Denormalization Mode 0: Denormalized number is treated as such 1: Denormalized number is treated as zero

Bit	Bit Name	Initial Value	R/W	Description
17 to 12	Cause	000000	R/W	FPU Exception Cause Field
11 to 7	Enable (EN)	00000	R/W	FPU Exception Enable Field
6 to 2	Flag	00000	R/W	FPU Exception Flag Field Each time an FPU operation instruction is executed, the FPU exception cause field is cleared to 0. When an FPU exception occurs, the bits corresponding to FPU exception cause field and flag field are set to 1. The FPU exception flag field remains set to 1 until it is cleared to 0 by software. For bit allocations of each field, see table 2.32.
1, 0	RM	01	R/W	Rounding Mode These bits select the rounding mode. 00: Round to Nearest 01: Round to Zero 10: Reserved (setting prohibited) 11: Reserved (setting prohibited)



Notes: 1. In the case of SZ = 0 and PR = 0, DR register can not be used.

2. The bit-location of DR register is used for double precision format when PR = 1.
(In the case of (2), it is used when PR is changed from 0 to 1.)

Figure 2.26 Relation between SZ Bit and Endian

Table 2.32 Bit Allocation for FPU Exception Handling

Field Name		FPU Error (E)	Invalid Operation (V)	Division by Zero (Z)	Overflow (O)	Underflow (U)	Inexact (I)
Cause	FPU exception cause field	Bit 17	Bit 16	Bit 15	Bit 14	Bit 13	Bit 12
Enable	FPU exception enable field	None	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7
Flag	FPU exception flag field	None	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2

(3) Floating-Point Communication Register (FPUL)

Information is transferred between the FPU and CPU via FPUL. FPUL is a 32-bit system register that is accessed from the CPU side by means of LDS and STS instructions. For example, to convert the integer stored in general register R1 to a single-precision floating-point number, the processing flow is as follows:

R1 → (LDS instruction) → FPUL → (single-precision FLOAT instruction) → FR1

2.6.4 Rounding

In a floating-point instruction, rounding is performed when generating the final operation result from the intermediate result. Therefore, the result of combination instructions such as FMAC, FTRV, and FIPR will differ from the result when using a basic instruction such as FADD, FSUB, or FMUL. Rounding is performed once in FMAC, but twice in FADD, FSUB, and FMUL.

Which of the two rounding methods is to be used is determined by the RM bits in FPSCR.

FPSCR.RM[1:0] = 00: Round to Nearest
FPSCR.RM[1:0] = 01: Round to Zero

(1) Round to Nearest

The operation result is rounded to the nearest expressible value. If there are two nearest expressible values, the one with an LSB of 0 is selected.

If the unrounded value is $2^{Emax} (2 - 2^{-P})$ or more, the result will be infinity with the same sign as the unrounded value. The values of Emax and P, respectively, are 127 and 24 for single-precision, and 1023 and 53 for double-precision.

(2) Round to Zero

The digits below the round bit of the unrounded value are discarded.

If the unrounded value is larger than the maximum expressible absolute value, the value will become the maximum expressible absolute value with the same sign as unrounded value.

2.6.5 Floating-Point Exceptions

(1) General FPU Disable Exceptions and Slot FPU Disable Exceptions

FPU-related exceptions are occurred when an FPU instruction is executed with SR.FD set to 1. When the FPU instruction is in other than delayed slot, the general FPU disable exception is occurred. When the FPU instruction is in the delay slot, the slot FPU disable exception is occurred.

(2) FPU Exception Sources

The exception sources are as follows:

- FPU error (E): When FPSCR.DN = 0 and a denormalized number is input
- Invalid operation (V): In case of an invalid operation, such as NaN input
- Division by zero (Z): Division with a zero divisor
- Overflow (O): When the operation result overflows
- Underflow (U): When the operation result underflows
- Inexact exception (I): When overflow, underflow, or rounding occurs

The FPU exception cause field in FPSCR contains bits corresponding to all of above sources E, V, Z, O, U, and I, and the FPU exception flag and enable fields in FPSCR contain bits corresponding to sources V, Z, O, U, and I, but not E. Thus, FPU errors cannot be disabled.

When an FPU exception occurs, the corresponding bit in the FPU exception cause field is set to 1, and 1 is added to the corresponding bit in the FPU exception flag field. When an FPU exception does not occur, the corresponding bit in the FPU exception cause field is cleared to 0, but the corresponding bit in the FPU exception flag field remains unchanged.

(3) FPU Exception Handling

FPU exception handling is initiated in the following cases:

- FPU error (E): FPSCR.DN = 0 and a denormalized number is input
- Invalid operation (V): FPSCR.Enable.V = 1 and (instruction = FTRV or invalid operation)
- Division by zero (Z): FPSCR.Enable.Z = 1 and division with a zero divisor or the input of FSRRA is zero
- Overflow (O): FPSCR.Enable.O = 1 and possibility of operation result overflow
- Underflow (U): FPSCR.Enable.U = 1 and possibility of operation result underflow
- Inexact exception (I): FPSCR.Enable.I = 1 and instruction with possibility of inexact operation result

Please refer to section 11, Instruction Descriptions of the SH-4A Extended Functions Software Manual about the FPU exception case in detail.

All exception events that originate in the FPU are assigned as the same exception event. The meaning of an exception is determined by software by reading from FPSCR and interpreting the information it contains. Also, the destination register is not changed by any FPU exception handling operation.

If the FPU exception sources except for above are generated, the bit corresponding to source V, Z, O, U, or I is set to 1, and a default value is generated as the operation result.

- Invalid operation (V): qNaN is generated as the result.
- Division by zero (Z): Infinity with the same sign as the unrounded value is generated.
- Overflow (O):
  When rounding mode = RZ, the maximum normalized number, with the same sign as the unrounded value, is generated.
  When rounding mode = RN, infinity with the same sign as the unrounded value is generated.
- Underflow (U):
  When FPSCR.DN = 0, a denormalized number with the same sign as the unrounded value, or zero with the same sign as the unrounded value, is generated.
  When FPSCR.DN = 1, zero with the same sign as the unrounded value, is generated.
- Inexact exception (I): An inexact result is generated.

2.6.6 Graphics Support Functions

The SH-4A supports two kinds of graphics functions: new instructions for geometric operations, and pair single-precision transfer instructions that enable high-speed data transfer.

(1) Geometric Operation Instructions

Geometric operation instructions perform approximate-value computations. To enable high-speed computation with a minimum of hardware, the SH-4A ignores comparatively small values in the partial computation results of four multiplications. Consequently, the error shown below is produced in the result of the computation:

$$\text{Maximum error} = \text{MAX (individual multiplication result} \times 2^{-\text{MIN (number of multiplier significant digits}-1,\ \text{number of multiplicand significant digits}-1)}) + \text{MAX (result value} \times 2^{-23},\ 2^{-149})$$

The number of significant digits is 24 for a normalized number and 23 for a denormalized number (number of leading zeros in the fractional part).

In a future version of the SH Series, the above error is guaranteed, but the same result between different processor cores is not guaranteed.

(a) FIPR FVm, FVn (m, n: 0, 4, 8, 12)

This instruction is basically used for the following purposes:

- Inner product ($m \neq n$):
  This operation is generally used for surface/rear surface determination for polygon surfaces.
- Sum of square of elements ($m = n$):
  This operation is generally used to find the length of a vector.

Since an inexact exception is not detected by an FIPR instruction, the inexact exception (I) bit in both the FPU exception cause field and flag field are always set to 1 when an FIPR instruction is executed. Therefore, if the I bit is set in the FPU exception enable field, FPU exception handling will be executed.

(b) FTRV XMTRX, FVn (n: 0, 4, 8, 12)

This instruction is basically used for the following purposes:

- Matrix $(4 \times 4) \cdot$ vector (4):
  This operation is generally used for viewpoint changes, angle changes, or movements called vector transformations (4-dimensional). Since affine transformation processing for angle + parallel movement basically requires a 4×4 matrix, the SH-4A supports 4-dimensional operations.
- Matrix $(4 \times 4) \times$ matrix (4×4):
  This operation requires the execution of four FTRV instructions.

Since an inexact exception is not detected by an FIRV instruction, the inexact exception (I) bit in both the FPU exception cause field and flag field are always set to 1 when an FTRV instruction is executed. Therefore, if the I bit is set in the FPU exception enable field, FPU exception handling will be executed. It is not possible to check all data types in the registers beforehand when executing an FTRV instruction. If the V bit is set in the FPU exception enable field, FPU exception handling will be executed.

(c) FRCHG

This instruction modifies banked registers. For example, when the FTRV instruction is executed, matrix elements must be set in an array in the background bank. However, to create the actual elements of a translation matrix, it is easier to use registers in the foreground bank. When the LDS instruction is used on FPSCR, this instruction takes four to five cycles in order to maintain the FPU state. With the FRCHG instruction, the FR bit in FPSCR can be changed in one cycle.

(2) Pair Single-Precision Data Transfer

In addition to the powerful new geometric operation instructions, the SH-4A also supports high-speed data transfer instructions.

When the SZ bit is 1, the SH-4A can perform data transfer by means of pair single-precision data transfer instructions.

- FMOV DRm/XDm, DRn/XDRn (m, n: 0, 2, 4, 6, 8, 10, 12, 14)
- FMOV DRm/XDm, @Rn (m: 0, 2, 4, 6, 8, 10, 12, 14; n: 0 to 15)

These instructions enable two single-precision (2×32-bit) data items to be transferred; that is, the transfer performance of these instructions is doubled.

- FSCHG

This instruction changes the value of the SZ bit in FPSCR, enabling fast switching between use and non-use of pair single-precision data transfer.

2.7 Memory Management Unit (MMU)

The SH-4A supports an 8-bit address space identifier, a 32-bit virtual address space, and a 29-bit or 32-bit physical address space. Address translation from virtual addresses to physical addresses is enabled by the memory management unit (MMU) in the SH-4A. The MMU performs high-speed address translation by caching user-created address translation table information in an address translation buffer (translation lookaside buffer: TLB).

The SH-4A has four instruction TLB (ITLB) entries and 64 unified TLB (UTLB) entries. UTLB copies are stored in the ITLB by hardware. A paging system is used for address translation. It is possible to set the virtual address space access right and implement memory protection independently for privileged mode and user mode.

The MMU of the SH-4A runs in several operating modes. In view of physical address mapping ranges, 29-bit address mode and 32-bit address extended mode are provided. In view of flag functions of the MMU, TLB compatible mode (four paging sizes with four protection bits) and TLB extended mode (eight paging sizes with six protection bits) are provided.

Selection between TLB compatible mode and TLB extended mode is made by setting the relevant control register (bit ME in the MMUCR register) by software. The range of physical address mapping is explained through sections 2.7.1, Overview of MMU, to 2.7.7, Memory-Mapped TLB Configuration, for the case of 29-bit address mode, which is followed by section 2.7.8, 32-Bit Address Extended Mode, where differences from 29-bit address mode are explained.

The flag functions of the MMU are explained in parallel for both TLB compatible mode and TLB extended mode.

2.7.1 Overview of MMU

The MMU was conceived as a means of making efficient use of physical memory. As shown in (0) in figure 2.27, when a process is smaller in size than the physical memory, the entire process can be mapped onto physical memory, but if the process increases in size to the point where it does not fit into physical memory, it becomes necessary to divide the process into smaller parts, and map the parts requiring execution onto physical memory as occasion arises ((1) in figure 2.27). Having this mapping onto physical memory executed consciously by the process itself imposes a heavy burden on the process. The virtual memory system was devised as a means of handling all physical memory mapping to reduce this burden ((2) in figure 2.27). With a virtual memory system, the size of the available virtual memory is much larger than the actual physical memory, and processes are mapped onto this virtual memory. Thus processes only have to consider their operation in virtual memory, and mapping from virtual memory to physical memory is handled by the MMU. The MMU is normally managed by the OS, and physical memory switching is carried out so as to enable the virtual memory required by a process to be mapped smoothly onto physical memory. Physical memory switching is performed via secondary storage, etc.

The virtual memory system that came into being in this way works to best effect in a time sharing system (TSS) that allows a number of processes to run simultaneously ((3) in figure 2.27). Running a number of processes in a TSS did not increase efficiency since each process had to take account of physical memory mapping. Efficiency is improved and the load on each process reduced by the use of a virtual memory system ((4) in figure 2.27). In this virtual memory system, virtual memory is allocated to each process. The task of the MMU is to map a number of virtual memory areas onto physical memory in an efficient manner. It is also provided with memory protection functions to prevent a process from inadvertently accessing another process's physical memory.

When address translation from virtual memory to physical memory is performed using the MMU, it may happen that the translation information has not been recorded in the MMU, or the virtual memory of a different process is accessed by mistake. In such cases, the MMU will generate an exception, change the physical memory mapping, and record the new address translation information.

Although the functions of the MMU could be implemented by software alone, having address translation performed by software each time a process accessed physical memory would be very inefficient. For this reason, a buffer for address translation (the translation lookaside buffer: TLB) is provided by hardware, and frequently used address translation information is placed here. The TLB can be described as a cache for address translation information. However, unlike a cache, if address translation fails—that is, if an exception occurs—switching of the address translation

information is normally performed by software. Thus memory management can be performed in a flexible manner by software.

There are two methods by which the MMU can perform mapping from virtual memory to physical memory: the paging method, using fixed-length address translation, and the segment method, using variable-length address translation. With the paging method, the unit of translation is a fixed-size address space called a page.

In the following descriptions, the address space in virtual memory in the SH-4A is referred to as virtual address space, and the address space in physical memory as physical address space.

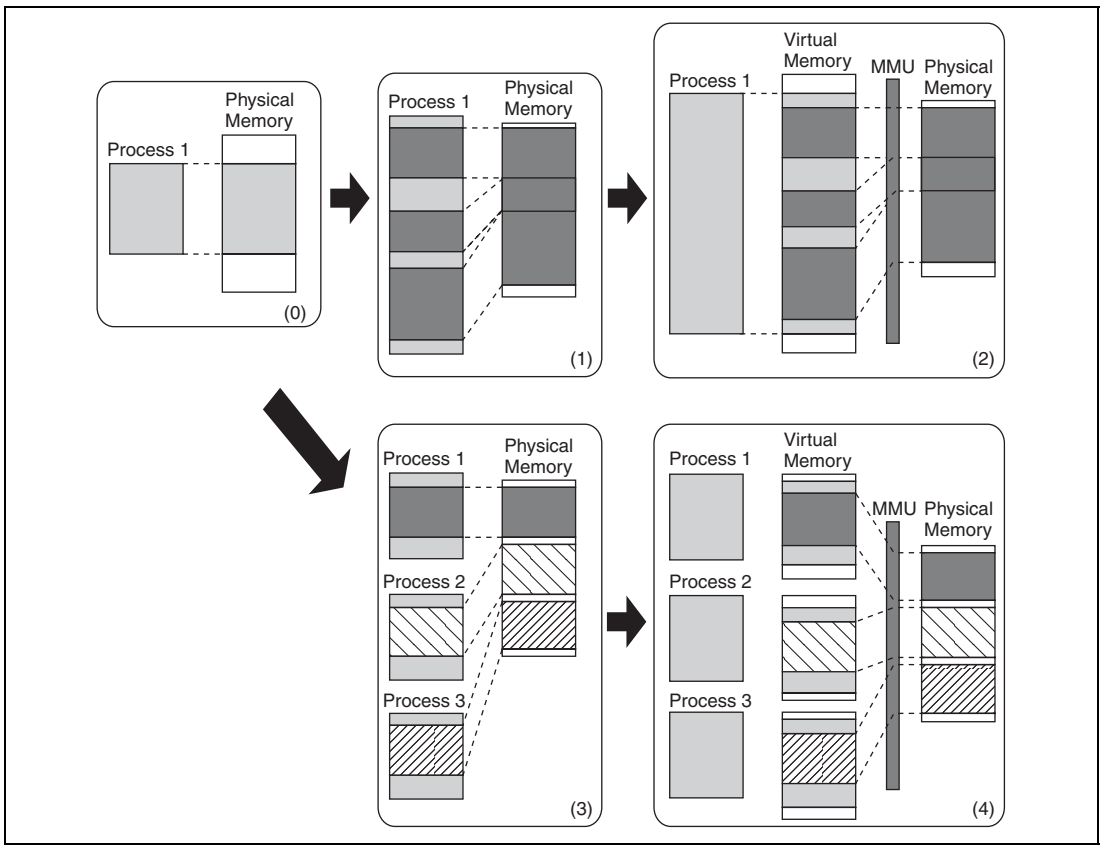


Figure 2.27 Role of MMU

(1) Address Spaces

(a) Virtual Address Space

The SH-4A supports a 32-bit virtual address space, and can access a 4-Gbyte address space. The virtual address space is divided into a number of areas, as shown in figures 2.28 and 2.29. In privileged mode, the 4-Gbyte space from the P0 area to the P4 area can be accessed. In user mode, a 2-Gbyte space in the U0 area can be accessed. When the SQMD bit in the MMU control register (MMUCR) is 0, a 64-Mbyte space in the store queue area can be accessed. When the RMD bit in the on-chip memory control register (RAMCR) is 1, a 16-Mbyte space in on-chip memory area can be accessed. Accessing areas other than the U0 area, store queue area, and on-chip memory area in user mode will cause an address error.

When the AT bit in MMUCR is set to 1 and the MMU is enabled, the P0, P3, and U0 areas can be mapped onto any physical address space in 1-, 4-, 64-Kbyte, or 1-Mbyte page units in TLB compatible mode and in 1-, 4-, 8-, 64-, 256-Kbyte, 1-, 4-, or 64-Mbyte page units in TLB extended mode. By using an 8-bit address space identifier, the P0, P3, and U0 areas can be increased to a maximum of 256. Mapping from the virtual address space to the 29-bit physical address space is carried out using the TLB.

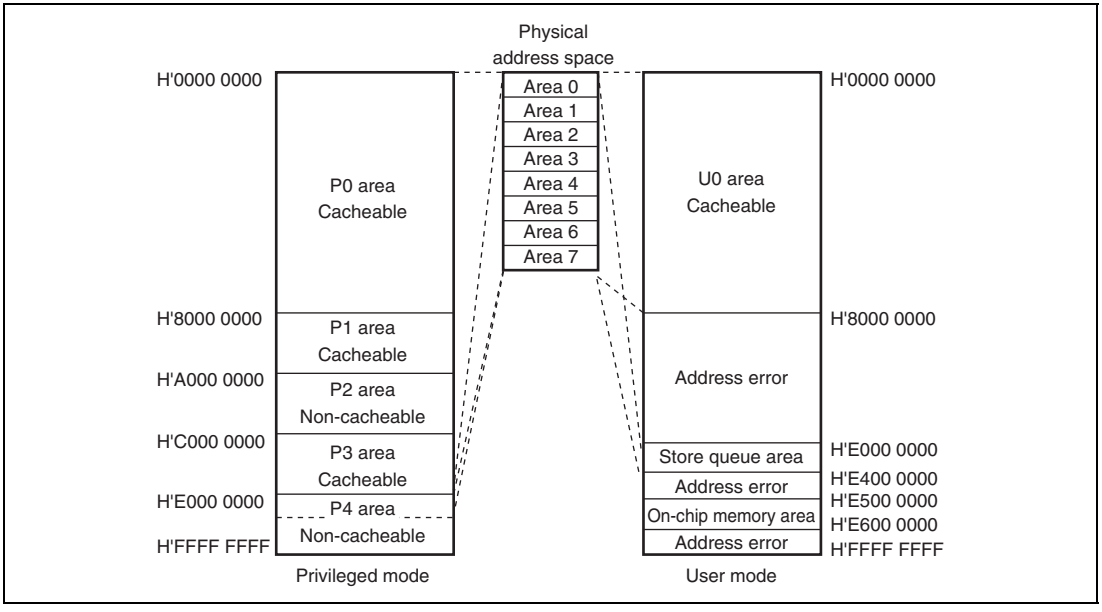


Figure 2.28 Virtual Address Space (AT in MMUCR= 0)

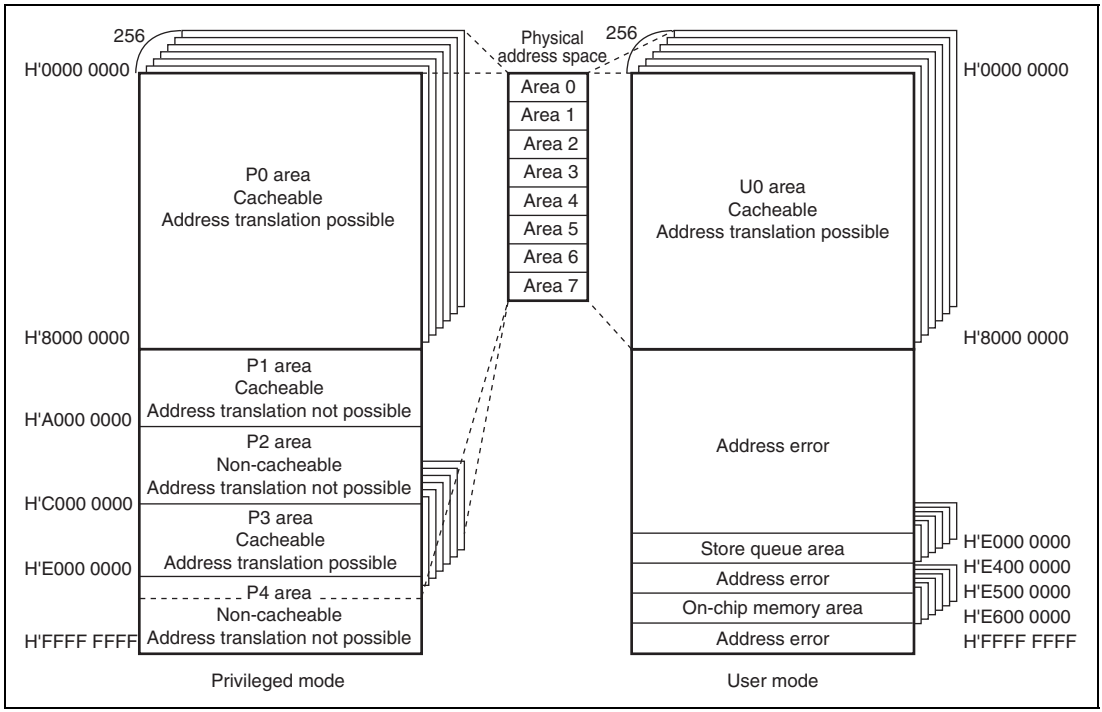


Figure 2.29 Virtual Address Space (AT in MMUCR= 1)

1. P0, P3, and U0 Areas

   The P0, P3, and U0 areas allow address translation using the TLB and access using the cache.

   When the MMU is disabled, replacing the upper 3 bits of an address with 0s gives the corresponding physical address. Whether or not the cache is used is determined by the CCR setting. When the cache is used, switching between the copy-back method and the write-through method for write accesses is specified by the WT bit in CCR.

   When the MMU is enabled, these areas can be mapped onto any physical address space in 1-, 4-, 64-Kbyte, or 1-Mbyte page units in TLB compatible mode and in 1-, 4-, 8-, 64, 256-Kbyte, 1-, 4-, or 64-Mbyte page units in TLB extended mode using the TLB. When CCR is in the cache enabled state and the C bit for the corresponding page of the TLB entry is 1, accesses can be performed using the cache. When the cache is used, switching between the copy-back method and the write-through method for write accesses is specified by the WT bit of the TLB entry.

   When the P0, P3, and U0 areas are mapped onto the control register area which is allocated in the area 7 in physical address space by means of the TLB, the C bit for the corresponding page must be cleared to 0.

2. P1 Area

   The P1 area does not allow address translation using the TLB but can be accessed using the cache.

   Regardless of whether the MMU is enabled or disabled, clearing the upper 3 bits of an address to 0 gives the corresponding physical address. Whether or not the cache is used is determined by the CCR setting. When the cache is used, switching between the copy-back method and the write-through method for write accesses is specified by the CB bit in CCR.

3. P2 Area

   The P2 area does not allow address translation using the TLB and access using the cache.

   Regardless of whether the MMU is enabled or disabled, clearing the upper 3 bits of an address to 0 gives the corresponding physical address.

4. P4 Area

   The P4 area is mapped onto the internal resource of the SH-4A. This area except the store queue and on-chip memory areas does not allow address translation using the TLB. This area cannot be accessed using the cache. The P4 area is shown in detail in figure 2.30.

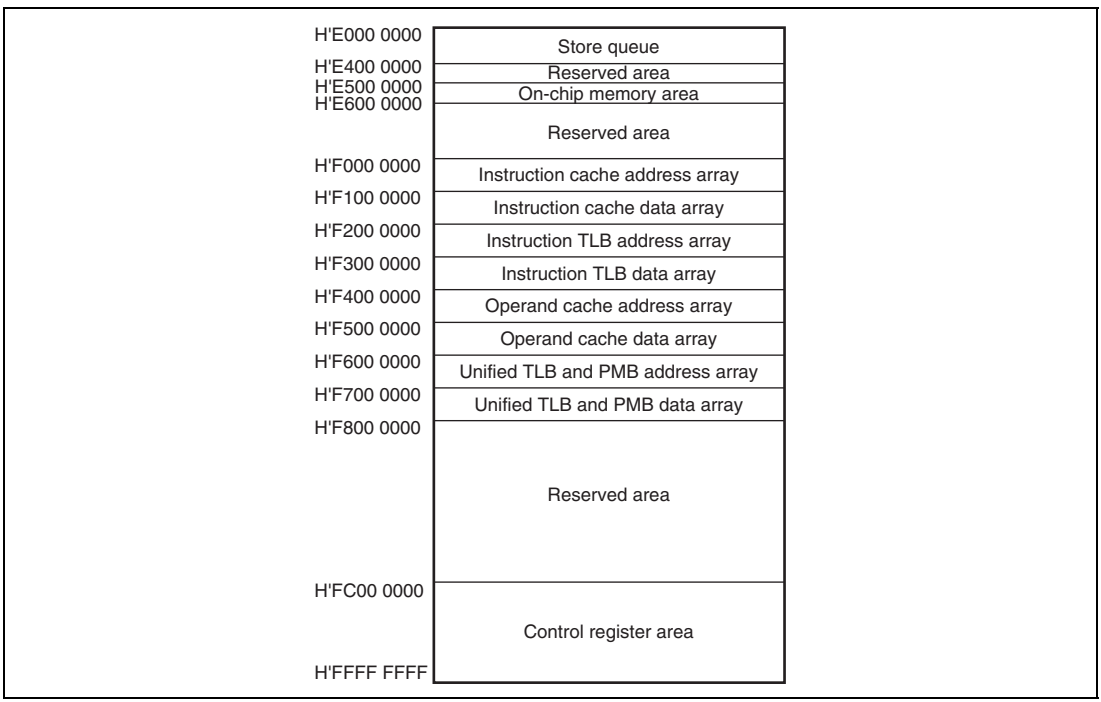


Figure 2.30 P4 Area

The area from H'E000 0000 to H'E3FF FFFF comprises addresses for accessing the store queues (SQs). In user mode, the access right is specified by the SQMD bit in MMUCR. For details, see section 2.8.7, Store Queues.

The area from H'E500 0000 to H'E5FF FFFF comprises addresses for accessing the on-chip memory. In user mode, the access right is specified by the RMD bit in RAMCR. For details, see section 2.9, On-Chip Memory.

The area from H'F000 0000 to H'F0FF FFFF is used for direct access to the instruction cache address array. For details, see section 2.8.6 (1), IC Address Array.

The area from H'F100 0000 to H'F1FF FFFF is used for direct access to the instruction cache data array. For details, see section 2.8.6 (2), IC Data Array.

The area from H'F200 0000 to H'F2FF FFFF is used for direct access to the instruction TLB address array. For details, see section 2.7.7 (1), ITLB Address Array.

The area from H'F300 0000 to H'F37F FFFF is used for direct access to instruction TLB data array. For details, see section 2.7.7 (2), ITLB Data Array (TLB Compatible Mode) and section 2.7.7 (3), ITLB Data Array (TLB Extended Mode).

The area from H'F400 0000 to H'F4FF FFFF is used for direct access to the operand cache address array. For details, see section 2.8.6 (3), OC Address Array.

The area from H'F500 0000 to H'F5FF FFFF is used for direct access to the operand cache data array. For details, see section 2.8.6 (4), OC Data Array.

The area from H'F600 0000 to H'F60F FFFF is used for direct access to the unified TLB address array. For details, see section 2.7.7 (4), UTLB Address Array.

The area from H'F610 0000 to H'F61F FFFF is used for direct access to the PMB address array. For details, see section 2.7.8 (5), Memory-Mapped PMB Configuration.

The area from H'F700 0000 to H'F70F FFFF is used for direct access to unified TLB data array. For details, see section 2.7.7 (5), UTLB Data Array (TLB Compatible Mode) and section 2.7.7 (6), UTLB Data Array (TLB Extended Mode).

The area from H'F710 0000 to H'F71F FFFF is used for direct access to the PMB data array. For details, see section 2.7.8 (5), Memory-Mapped PMB Configuration.

The area from H'FC00 0000 to H'FFFF FFFF is the on-chip peripheral module control register area. For details, see register descriptions in each section.

(b) Physical Address Space

The SH-4A supports a 29-bit physical address space. The physical address space is divided into eight areas as shown in figure 2.31. Area 7 is a reserved area. For details, see section 1, Overview.

Only when area 7 in the physical address space is accessed using the TLB, addresses H'1C00 0000 to H'1FFF FFFF of area 7 are not designated as a reserved area, but are equivalent to the control register area in the P4 area, in the virtual address space.

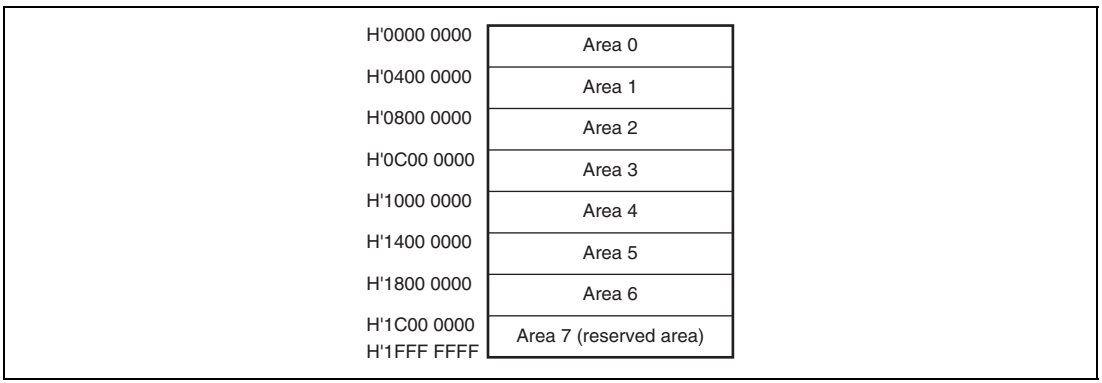


Figure 2.31 Physical Address Space

(c) Address Translation

When the MMU is used, the virtual address space is divided into units called pages, and translation to physical addresses is carried out in these page units. The address translation table in external memory contains the physical addresses corresponding to virtual addresses and additional information such as memory protection codes. Fast address translation is achieved by caching the contents of the address translation table located in external memory into the TLB. In the SH-4A, basically, the ITLB is used for instruction accesses and the UTLB for data accesses. In the event of an access to an area other than the P4 area, the accessed virtual address is translated to a physical address. If the virtual address belongs to the P1 or P2 area, the physical address is uniquely determined without accessing the TLB. If the virtual address belongs to the P0, U0, or P3 area, the TLB is searched using the virtual address, and if the virtual address is recorded in the TLB, a TLB hit is made and the corresponding physical address is read from the TLB. If the accessed virtual address is not recorded in the TLB, a TLB miss exception is generated and processing switches to the TLB miss exception handling routine. In the TLB miss exception handling routine, the address translation table in external memory is searched, and the corresponding physical address and page management information are recorded in the TLB. After the return from the exception handling routine, the instruction which caused the TLB miss exception is re-executed.

(d) Single Virtual Memory Mode and Multiple Virtual Memory Mode

There are two virtual memory systems, single virtual memory and multiple virtual memory, either of which can be selected with the SV bit in MMUCR. In the single virtual memory system, a number of processes run simultaneously, using virtual address space on an exclusive basis, and the physical address corresponding to a particular virtual address is uniquely determined. In the multiple virtual memory system, a number of processes run while sharing the virtual address space, and particular virtual addresses may be translated into different physical addresses depending on the process. The only difference between the single virtual memory and multiple virtual memory systems in terms of operation is in the TLB address comparison method (see section 2.7.3 (3), Address Translation Method).

(e) Address Space Identifier (ASID)

In multiple virtual memory mode, an 8-bit address space identifier (ASID) is used to distinguish between multiple processes running simultaneously while sharing the virtual address space. Software can set the 8-bit ASID of the currently executing process in PTEH in the MMU. The TLB does not have to be purged when processes are switched by means of ASID.

In single virtual memory mode, ASID is used to provide memory protection for multiple processes running simultaneously while using the virtual address space on an exclusive basis.

Note: Two or more entries with the same virtual page number (VPN) but different ASID must not be set in the TLB simultaneously in single virtual memory mode.

2.7.2 Register Descriptions

The following registers are related to MMU processing.

Table 2.33 Register Configuration

Register Name	Abbreviation	R/W	P4 Address*	Area 7 Address*	Size
Page table entry high register	PTEH	R/W	H'FF00 0000	H'1F00 0000	32
Page table entry low register	PTEL	R/W	H'FF00 0004	H'1F00 0004	32
Translation table base register	TTB	R/W	H'FF00 0008	H'1F00 0008	32
TLB exception address register	TEA	R/W	H'FF00 000C	H'1F00 000C	32
MMU control register	MMUCR	R/W	H'FF00 0010	H'1F00 0010	32
Page table entry assistance register	PTEA	R/W	H'FF00 0034	H'1F00 0034	32
Physical address space control register	PASCR	R/W	H'FF00 0070	H'1F00 0070	32
Instruction re-fetch inhibit control register	IRMCR	R/W	H'FF00 0078	H'1F00 0078	32

Note: * These P4 addresses are for the P4 area in the virtual address space. These area 7 addresses are accessed from area 7 in the physical address space by means of the TLB.

Table 2.34 Register States in Each Processing State

Register Name	Abbreviation	Power-on Reset	Manual Reset	Sleep	Software Standby	Deep Standby
Page table entry high register	PTEH	Undefined	Undefined	Retained	Retained	Undefined
Page table entry low register	PTEL	Undefined	Undefined	Retained	Retained	Undefined
Translation table base register	TTB	Undefined	Undefined	Retained	Retained	Undefined
TLB exception address register	TEA	Undefined	Retained	Retained	Retained	Undefined
MMU control register	MMUCR	H'0000 0000	H'0000 0000	Retained	Retained	H'0000 0000
Page table entry assistance register	PTEA	H'0000 xxx0	H'0000 xxx0	Retained	Retained	H'0000 xxx0
Physical address space control register	PASCR	H'0000 0000	H'0000 0000	Retained	Retained	H'0000 0000
Instruction re-fetch inhibit control register	IRMCR	H'0000 0000	H'0000 0000	Retained	Retained	H'0000 0000

(1) Page Table Entry High Register (PTEH)

PTEH consists of the virtual page number (VPN) and address space identifier (ASID). When an MMU exception or address error exception occurs, the VPN of the virtual address at which the exception occurred is set in the VPN bit by hardware. VPN varies according to the page size, but the VPN set by hardware when an exception occurs consists of the upper 22 bits of the virtual address which caused the exception. VPN setting can also be carried out by software. The number of the currently executing process is set in the ASID bit by software. ASID is not updated by hardware. VPN and ASID are recorded in the UTLB by means of the LDTLB instruction.

After the ASID field in PTEH has been updated, execute one of the following three methods before an access (including an instruction fetch) to the P0, P3, or U0 area that uses the updated ASID value is performed.

1. Execute a branch using the RTE instruction. In this case, the branch destination may be the P0, P3, or U0 area.
2. Execute the ICBI instruction for any address (including non-cacheable area).
3. If the R2 bit in IRMCR is 0 (initial value) before updating the ASID field, the specific instruction does not need to be executed. However, note that the CPU processing performance will be lowered because the instruction fetch is performed again for the next instruction after the ASID field has been updated.

Note that the method 3 may not be guaranteed in the future SuperH Series. Therefore, it is recommended that the method 1 or 2 should be used for being compatible with the future SuperH Series.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	VPN															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VPN						—	—	ASID							
Initial value:	—	—	—	—	—	—	0	0	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	VPN	Undefined	R/W	Virtual Page Number
9, 8	—	All 0	R	Reserved For details on reading from or writing to these bits, see description in General Precautions on Handling of Product.
7 to 0	ASID	Undefined	R/W	Address Space Identifier

(2) Page Table Entry Low Register (PTEL)

PTEL is used to hold the physical page number and page management information to be recorded in the UTLB by means of the LDTLB instruction. The contents of this register are not changed unless a software directive is issued.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	PPN												
Initial value:	0	0	0	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PPN						—	V	SZ1	PR1	PR0	SZ0	C	D	SH	WT
Initial value:	—	—	—	—	—	—	0	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved For details on reading from or writing to these bits, see description in General Precautions on Handling of Product.
28 to 10	PPN	Undefined	R/W	Physical Page Number
9	—	0	R	Reserved For details on reading from or writing to this bit, see description in General Precautions on Handling of Product.

Bit	Bit Name	Initial Value	R/W	Description
8	V	Undefined	R/W	Page Management Information
7	SZ1	Undefined	R/W	The meaning of each bit is same as that of corresponding bit in Common TLB (UTLB). For details, see section 2.7.3, TLB Functions (TLB Compatible Mode; MMUCR.ME = 0) and section 2.7.4, TLB Functions (TLB Extended Mode; MMUCR.ME = 1). Note: SZ1, PR1, SZ0, and PR0 bits are valid only in TLB compatible mode.
6	PR1	Undefined	R/W	
5	PR0	Undefined	R/W	
4	SZ0	Undefined	R/W	
3	C	Undefined	R/W	
2	D	Undefined	R/W	
1	SH	Undefined	R/W	
0	WT	Undefined	R/W	

(3) Translation Table Base Register (TTB)

TTB is used to store the base address of the currently used page table, and so on. The contents of TTB are not changed unless a software directive is issued. This register can be used freely by software.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TTB															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TTB															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

(4) TLB Exception Address Register (TEA)

After an MMU exception or address error exception occurs, the virtual address at which the exception occurred is stored. The contents of this register can be changed by software.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TEA								Virtual address at which MMU exception or address error occurred							
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TEA								Virtual address at which MMU exception or address error occurred							
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

(5) MMU Control Register (MMUCR)

The individual bits perform MMU settings as shown below. Therefore, MMUCR rewriting should be performed by a program in the P1 or P2 area.

After MMUCR has been updated, execute one of the following three methods before an access (including an instruction fetch) to the P0, P3, U0, or store queue area is performed.

1. Execute a branch using the RTE instruction. In this case, the branch destination may be the P0, P3, or U0 area.
2. Execute the ICBI instruction for any address (including non-cacheable area).
3. If the R2 bit in IRMCR is 0 (initial value) before updating MMUCR, the specific instruction does not need to be executed. However, note that the CPU processing performance will be lowered because the instruction fetch is performed again for the next instruction after MMUCR has been updated.

Note that the method 3 may not be guaranteed in the future SuperH Series. Therefore, it is recommended that the method 1 or 2 should be used for being compatible with the future SuperH Series.

MMUCR contents can be changed by software. However, the LRUI and URC bits may also be updated by hardware.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LRUI						—	—	URB						—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	URC						SQMD	SV	ME	—	—	—	—	TI	—	AT
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	LRUI	000000	R/W	Least Recently Used ITLB These bits indicate the ITLB entry to be replaced. The LRU (least recently used) method is used to decide the ITLB entry to be replaced in the event of an ITLB miss. The entry to be purged from the ITLB can be confirmed using the LRUI bits. LRUI is updated by means of the algorithm shown below. x means that updating is not performed. 000xxx: ITLB entry 0 is used 1xx00x: ITLB entry 1 is used x1x1x0: ITLB entry 2 is used xx1x11: ITLB entry 3 is used xxxxxx: Other than above When the LRUI bit settings are as shown below, the corresponding ITLB entry is updated by an ITLB miss. Ensure that values for which "Setting prohibited" is indicated below are not set at the discretion of software. After a power-on or manual reset, the LRUI bits are initialized to 0, and therefore a prohibited setting is never made by a hardware update. x means "don't care". 111xxx: ITLB entry 0 is updated 0xx11x: ITLB entry 1 is updated x0x0x1: ITLB entry 2 is updated xx0x00: ITLB entry 3 is updated Other than above: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
25, 24	—	All 0	R	Reserved For details on reading from or writing to these bits, see description in General Precautions on Handling of Product.
23 to 18	URB	000000	R/W	UTLB Replace Boundary These bits indicate the UTLB entry boundary at which replacement is to be performed. Valid only when URB ≠ 0.
17, 16	—	All 0	R	Reserved For details on reading from or writing to these bits, see description in General Precautions on Handling of Product.
15 to 10	URC	000000	R/W	UTLB Replace Counter These bits serve as a random counter for indicating the UTLB entry for which replacement is to be performed with an LDTLB instruction. This bit is incremented each time the UTLB is accessed. If URB > 0, URC is cleared to 0 when the condition URC = URB is satisfied. Also note that if a value is written to URC by software which results in the condition of URC > URB, incrementing is first performed in excess of URB until URC = H'3F. URC is not incremented by an LDTLB instruction.
9	SQMD	0	R/W	Store Queue Mode Specifies the right of access to the store queues. 0: User/privileged access possible 1: Privileged access possible (address error exception in case of user access)
8	SV	0	R/W	Single Virtual Memory Mode/Multiple Virtual Memory Mode Switching When this bit is changed, ensure that 1 is also written to the TI bit. 0: Multiple virtual memory mode 1: Single virtual memory mode

Bit	Bit Name	Initial Value	R/W	Description
7	ME	0	R/W	TLB Extended Mode Switching 0: TLB compatible mode 1: TLB extended mode For modifying the ME bit value, always set the TI bit to 1 to invalidate the contents of ITLB and UTLB. The selection of TLB operating mode made by the ME bit does not affect the functionality or operation of the PMB.
6 to 3	—	All 0	R	Reserved For details on reading from or writing to these bits, see description in General Precautions on Handling of Product.
2	TI	0	R/W	TLB Invalidate Bit Writing 1 to this bit invalidates (clears to 0) all valid UTLB/ITLB bits. This bit is always read as 0.
1	—	0	R	Reserved For details on reading from or writing to this bit, see description in General Precautions on Handling of Product.
0	AT	0	R/W	Address Translation Enable Bit These bits enable or disable the MMU. 0: MMU disabled 1: MMU enabled MMU exceptions are not generated when the AT bit is 0. In the case of software that does not use the MMU, the AT bit should be cleared to 0.

(6) Page Table Entry Assistance Register (PTEA)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	–	–	EPR						ESZ				–	–	–	–
Initial value:	0	0	–	–	–	–	–	–	–	–	–	–	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved For details on reading/writing these bits, see General Precautions on Handling of Product.
13 to 8	EPR	Undefined	R/W	Page Control Information
7 to 4	ESZ	Undefined	R/W	Each bit has the same function as the corresponding bit of the unified TLB (UTLB). For details, see section 2.7.4, TLB Functions (TLB Extended Mode; MMUCR.ME = 1)
3 to 0	—	All 0	R	Reserved For details on reading/writing these bits, see General Precautions on Handling of Product.

(7) Physical Address Space Control Register (PASCR)

PASCR controls the operation in the physical address space.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	UB							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved For details on reading from or writing to these bits, see description in General Precautions on Handling of Product.
7 to 0	UB	H'00	R/W	Buffered Write Control for Each Area (64 Mbytes) When writing is performed without using the cache or in the cache write-through mode, these bits specify whether the next bus access from the CPU waits for the end of writing for each area. 0 : The CPU does not wait for the end of writing bus access and starts the next bus access 1 : The CPU waits for the end of writing bus access and starts the next bus access UB[7]: Corresponding to the control register area UB[6]: Corresponding to area 6 UB[5]: Corresponding to area 5 UB[4]: Corresponding to area 4 UB[3]: Corresponding to area 3 UB[2]: Corresponding to area 2 UB[1]: Corresponding to area 1 UB[0]: Corresponding to area 0

(8) Instruction Re-Fetch Inhibit Control Register (IRMCR)

When the specific resource is changed, IRMCR controls whether the instruction fetch is performed again for the next instruction. The specific resource means the part of control registers, TLB, and cache.

In the initial state, the instruction fetch is performed again for the next instruction after changing the resource. However, the CPU processing performance will be lowered because the instruction fetch is performed again for the next instruction every time the resource is changed. Therefore, it is recommended that each bit in IRMCR is set to 1 and the specific instruction should be executed after all necessary resources have been changed prior to execution of the program which uses changed resources.

For details on the specific sequence, see descriptions in each resource.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	R2	R1	LT	MT	MC
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved For details on reading from or writing to these bits, see description in General Precautions on Handling of Product.
4	R2	0	R/W	Re-Fetch Inhibit 2 after Register Change When MMUCR, PASCR, CCR, PTEH, or RAMCR is changed, this bit controls whether re-fetch is performed for the next instruction. 0: Re-fetch is performed 1: Re-fetch is not performed

Bit	Bit Name	Initial Value	R/W	Description
3	R1	0	R/W	Re-Fetch Inhibit 1 after Register Change When a register allocated in addresses H'FF200000 to H'FF2FFFFF is changed, this bit controls whether re-fetch is performed for the next instruction. 0: Re-fetch is performed 1: Re-fetch is not performed
2	LT	0	R/W	Re-Fetch Inhibit after LDTLB Execution This bit controls whether re-fetch is performed for the next instruction after the LDTLB instruction has been executed. 0: Re-fetch is performed 1: Re-fetch is not performed
1	MT	0	R/W	Re-Fetch Inhibit after Writing Memory-Mapped TLB This bit controls whether re-fetch is performed for the next instruction after writing memory-mapped ITLB/UTLB while the AT bit in MMUCR is set to 1. 0: Re-fetch is performed 1: Re-fetch is not performed
0	MC	0	R/W	Re-Fetch Inhibit after Writing Memory-Mapped IC This bit controls whether re-fetch is performed for the next instruction after writing memory-mapped IC while the ICE bit in CCR is set to 1. 0: Re-fetch is performed 1: Re-fetch is not performed

2.7.3 TLB Functions (TLB Compatible Mode; MMUCR.ME = 0)

(1) Unified TLB (UTLB) Configuration

The UTLB is used for the following two purposes:

1. To translate a virtual address to a physical address in a data access
2. As a table of address translation information to be recorded in the ITLB in the event of an ITLB miss

The UTLB is so called because of its use for the above two purposes. Information in the address translation table located in external memory is cached into the UTLB. The address translation table contains virtual page numbers and address space identifiers, and corresponding physical page numbers and page management information. Figure 2.32 shows the UTLB configuration. The UTLB consists of 64 fully-associative type entries. Figure 2.33 shows the relationship between the page size and address format.

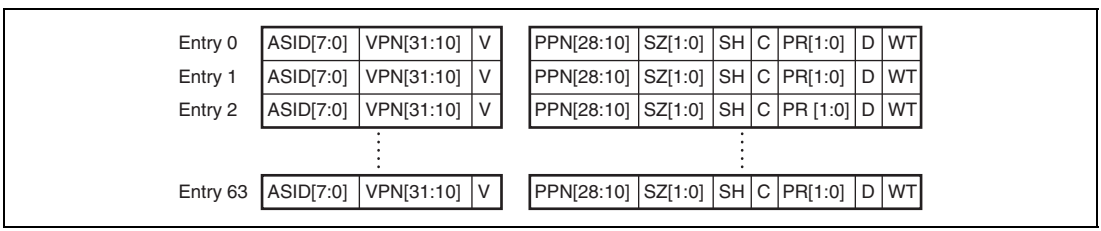


Figure 2.32 UTLB Configuration (TLB Compatible Mode)

[Legend]

- VPN: Virtual page number
  For 1-Kbyte page: Upper 22 bits of virtual address
  For 4-Kbyte page: Upper 20 bits of virtual address
  For 64-Kbyte page: Upper 16 bits of virtual address
  For 1-Mbyte page: Upper 12 bits of virtual address

- ASID: Address space identifier
  Indicates the process that can access a virtual page.
  In single virtual memory mode and user mode, or in multiple virtual memory mode, if the SH bit is 0, this identifier is compared with the ASID in PTEH when address comparison is performed.

- SH: Share status bit
  When 0, pages are not shared by processes.
  When 1, pages are shared by processes.

- SZ[1:0]: Page size bits
  Specify the page size.
  00: 1-Kbyte page
  01: 4-Kbyte page
  10: 64-Kbyte page
  11: 1-Mbyte page

- V: Validity bit
  Indicates whether the entry is valid.
  0: Invalid
  1: Valid
  Cleared to 0 by a power-on reset.
  Not affected by a manual reset.

- PPN: Physical page number
  Upper 22 bits of the physical address of the physical page number.
  With a 1-Kbyte page, PPN[28:10] are valid.
  With a 4-Kbyte page, PPN[28:12] are valid.
  With a 64-Kbyte page, PPN[28:16] are valid.
  With a 1-Mbyte page, PPN[28:20] are valid.
  The synonym problem must be taken into account when setting the PPN (see section 2.7.5 (5), Avoiding Synonym Problems).

- PR[1:0]: Protection key data
  2-bit data expressing the page access right as a code.
  00: Can be read from only in privileged mode
  01: Can be read from and written to in privileged mode
  10: Can be read from only in privileged or user mode
  11: Can be read from and written to in privileged mode or user mode

- C: Cacheability bit
  Indicates whether a page is cacheable.
  0: Not cacheable

1: Cacheable

When the control register area is mapped, this bit must be cleared to 0.

- D: Dirty bit

  Indicates whether a write has been performed to a page.

  0: Write has not been performed

  1: Write has been performed

- WT: Write-through bit

  Specifies the cache write mode.

  0: Copy-back mode

  1: Write-through mode

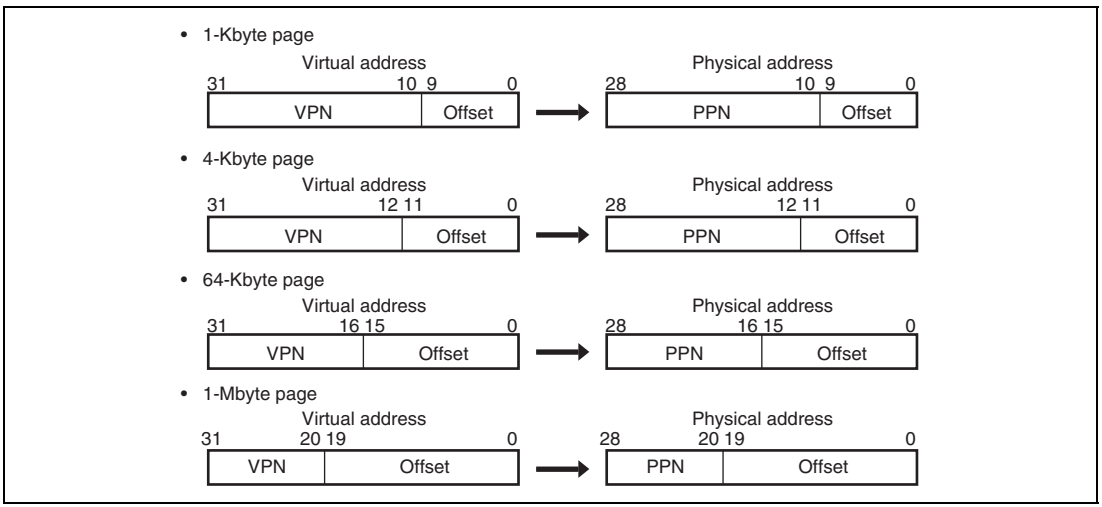


Figure 2.33 Relationship between Page Size and Address Format (TLB Compatible Mode)

(2) Instruction TLB (ITLB) Configuration

The ITLB is used to translate a virtual address to a physical address in an instruction access. Information in the address translation table located in the UTLB is cached into the ITLB. Figure 2.34 shows the ITLB configuration. The ITLB consists of four fully-associative type entries.

Entry 0	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	SZ[1:0]	SH	C	PR
Entry 1	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	SZ[1:0]	SH	C	PR
Entry 2	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	SZ[1:0]	SH	C	PR
Entry 3	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	SZ[1:0]	SH	C	PR

Notes: 1. The D and WT bits are not supported.
2. There is only one PR bit, corresponding to the upper bit of the PR bits in the UTLB.

Figure 2.34 ITLB Configuration (TLB Compatible Mode)

(3) Address Translation Method

Figure 2.35 shows a flowchart of a memory access using the UTLB.

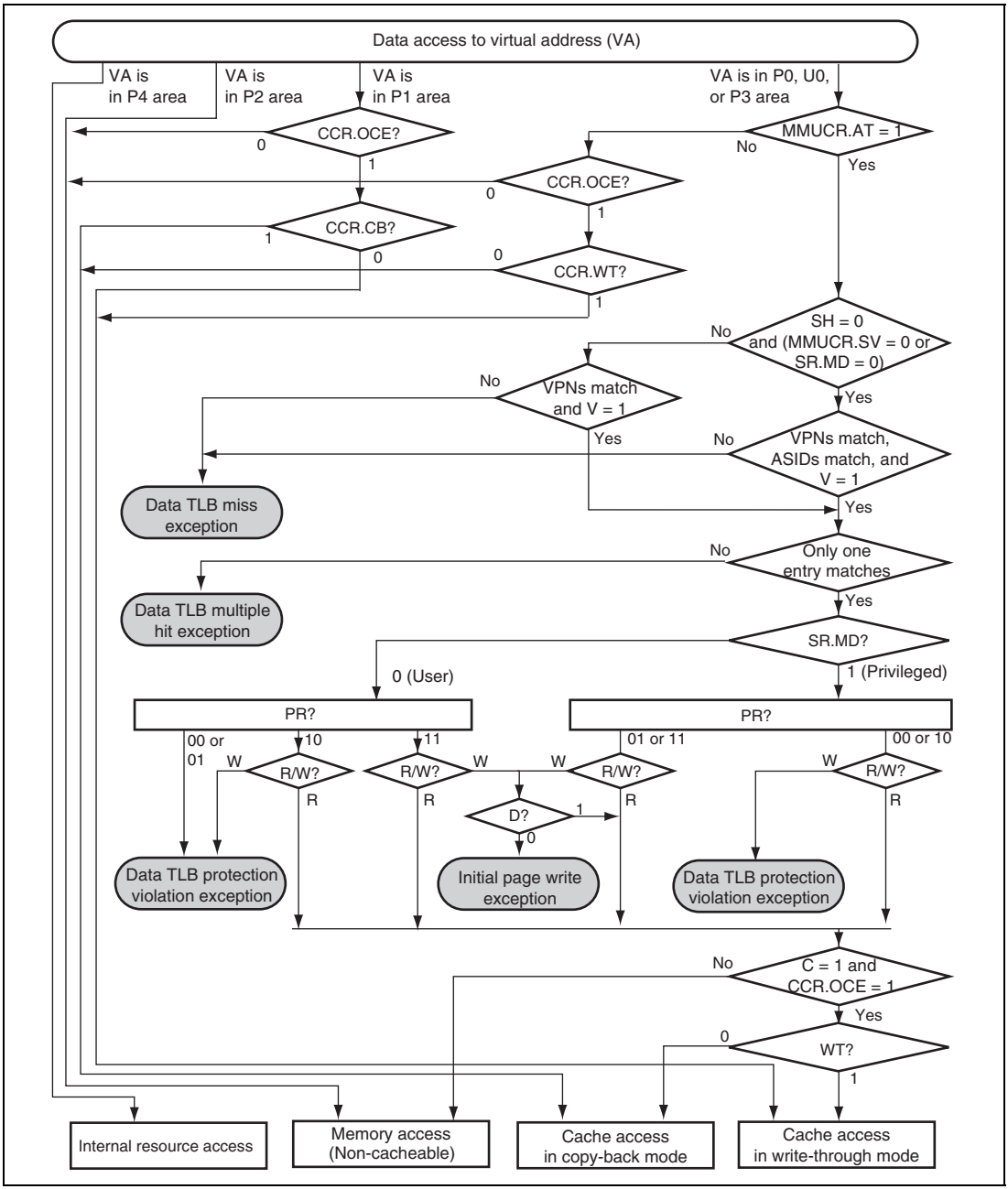


Figure 2.35 Flowchart of Memory Access Using UTLB (TLB Compatible Mode)

Figure 2.36 shows a flowchart of a memory access using the ITLB.

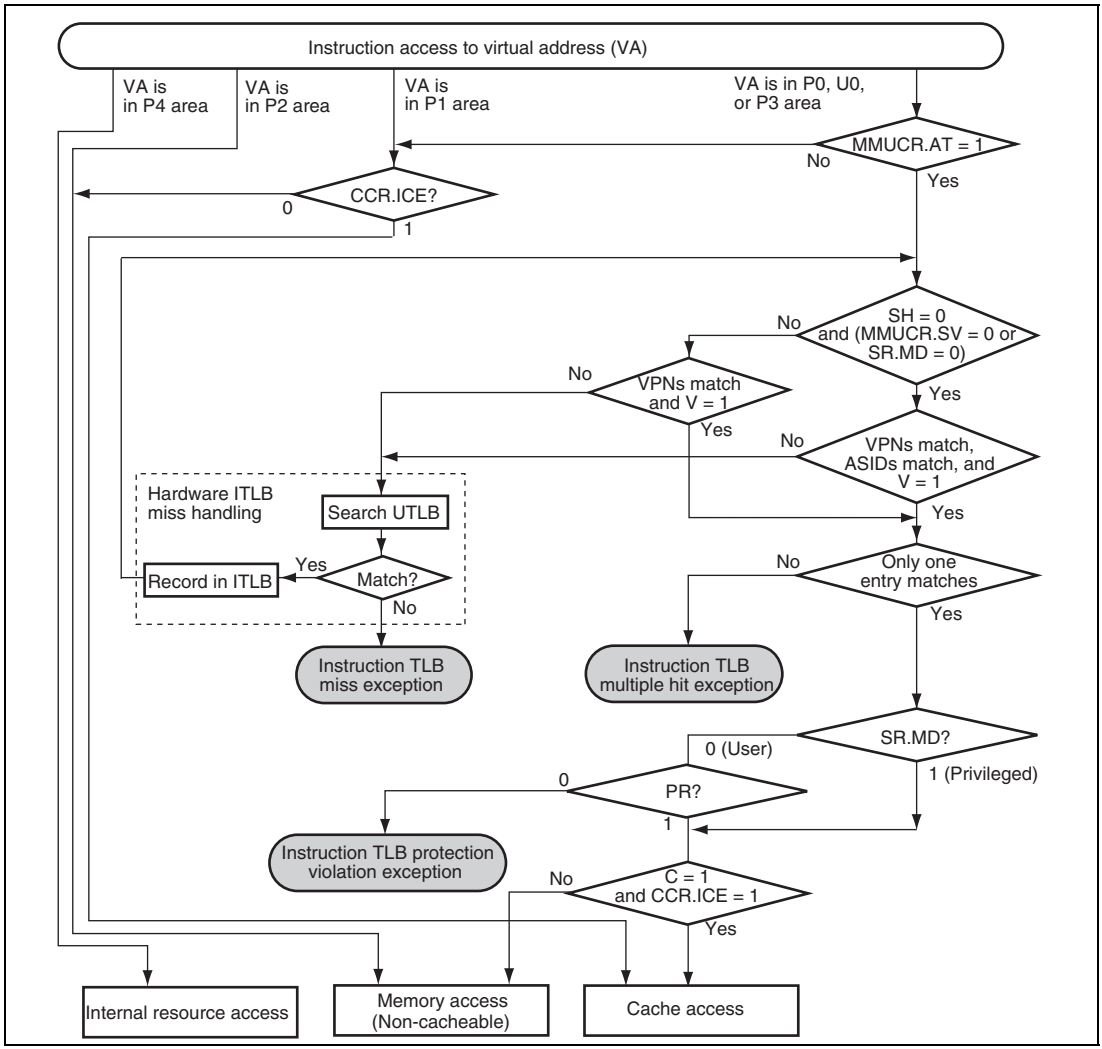


Figure 2.36 Flowchart of Memory Access Using ITLB (TLB Compatible Mode)

2.7.4 TLB Functions (TLB Extended Mode; MMUCR.ME = 1)

(1) Unified TLB (UTLB) Configuration

Figure 2.37 shows the configuration of the UTLB in TLB extended mode. Figure 2.38 shows the relationship between the page size and address format.

Entry 0	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	ESZ[3:0]	SH	C	EPR[5:0]	D	WT
Entry 1	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	ESZ[3:0]	SH	C	EPR[5:0]	D	WT
Entry 2	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	ESZ[3:0]	SH	C	EPR[5:0]	D	WT
⋮				⋮						
Entry 63	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	ESZ[3:0]	SH	C	EPR[5:0]	D	WT

Figure 2.37 UTLB Configuration (TLB Extended Mode)

[Legend]

- VPN: Virtual page number
  For 1-Kbyte page: Upper 22 bits of virtual address
  For 4-Kbyte page: Upper 20 bits of virtual address
  For 8-Kbyte page: Upper 19 bits of virtual address
  For 64-Kbyte page: Upper 16 bits of virtual address
  For 256-Kbyte page: Upper 14 bits of virtual address
  For 1-Mbyte page: Upper 12 bits of virtual address
  For 4-Mbyte page: Upper 10 bits of virtual address
  For 64-Mbyte page: Upper 6 bits of virtual address
- ASID: Address space identifier
  Indicates the process that can access a virtual page.
  In single virtual memory mode and user mode, or in multiple virtual memory mode, if the SH bit is 0, this identifier is compared with the ASID in PTEH when address comparison is performed.
- SH: Share status bit
  When 0, pages are not shared by processes.
  When 1, pages are shared by processes.
- ESZ: Page size bits
  Specify the page size.

0000: 1-Kbyte page
0001: 4-Kbyte page
0010: 8-Kbyte page
0100: 64-Kbyte page
0101: 256-Kbyte page
0111: 1-Mbyte page
1000: 4-Mbyte page
1100: 64-Mbyte page

Note: When a value other than those listed above is recorded, operation is not guaranteed.

- V: Validity bit

  Indicates whether the entry is valid.

  0: Invalid

  1: Valid

  Cleared to 0 by a power-on reset.

  Not affected by a manual reset.

- PPN: Physical page number

  Upper 19 bits of the physical address.

  With a 1-Kbyte page, PPN[28:10] are valid.

  With a 4-Kbyte page, PPN[28:12] are valid.

  With a 8-Kbyte page, PPN[28:13] are valid.

  With a 64-Kbyte page, PPN[28:16] are valid.

  With a 256-Kbyte page, PPN[28:18] are valid.

  With a 1-Mbyte page, PPN[28:20] are valid.

  With a 4-Mbyte page, PPN[28:22] are valid.

  With a 64-Mbyte page, PPN[28:26] are valid.

  The synonym problem must be taken into account when setting the PPN (see section 2.7.5 (5), Avoiding Synonym Problems).

- EPR: Protection key data

  6-bit data expressing the page access right as a code.

  Reading, writing, and execution (instruction fetch) in privileged mode and reading, writing, and execution (instruction fetch) in user mode can be set independently. Each bit is disabled by 0 and enabled by 1.

  EPR[5]: Reading in privileged mode

  EPR[4]: Writing in privileged mode

  EPR[3]: Execution in privileged mode (instruction fetch)

EPR[2]: Reading in user mode

EPR[1]: Writing in user mode

EPR[0]: Execution in user mode (instruction fetch)

- C: Cacheability bit

  Indicates whether a page is cacheable.

  0: Not cacheable

  1: Cacheable

  When the control register area is mapped, this bit must be cleared to 0.

- D: Dirty bit

  Indicates whether a write has been performed to a page.

  0: Write has not been performed.

  1: Write has been performed.

- WT: Write-through bit

  Specifies the cache write mode.

  0: Copy-back mode

  1: Write-through mode

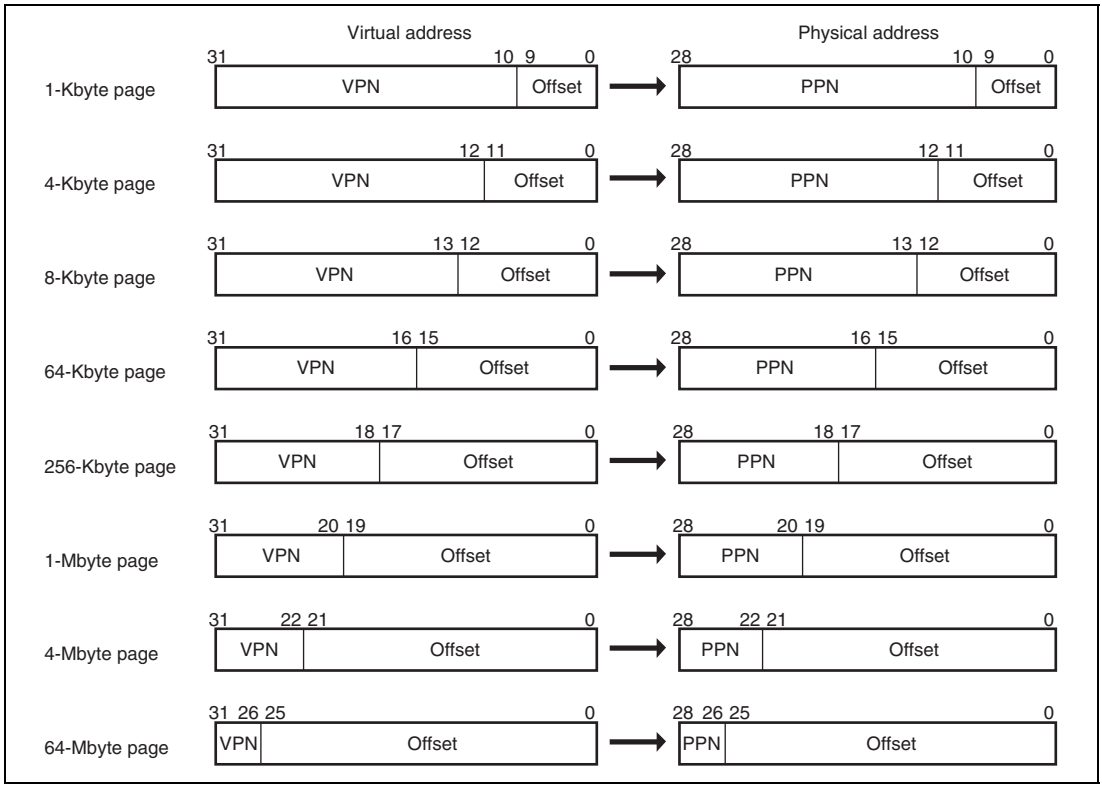


Figure 2.38 Relationship between Page Size and Address Format (TLB Extended Mode)

(2) Instruction TLB (ITLB) Configuration

Figure 2.39 shows the configuration of the ITLB in TLB extended mode.

Entry 0	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	ESZ[3:0]	SH	C	EPR[5]	EPR[3]	EPR[2]	EPR[0]
Entry 1	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	ESZ[3:0]	SH	C	EPR[5]	EPR[3]	EPR[2]	EPR[0]
Entry 2	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	ESZ[3:0]	SH	C	EPR[5]	EPR[3]	EPR[2]	EPR[0]
Entry 3	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	ESZ[3:0]	SH	C	EPR[5]	EPR[3]	EPR[2]	EPR[0]

Note: Bits EPR[4], EPR[1], D, and WT are not supported.

Figure 2.39 ITLB Configuration (TLB Extended Mode)

(3) Address Translation Method

Figure 2.40 is a flowchart of memory access using the UTLB in TLB extended mode.

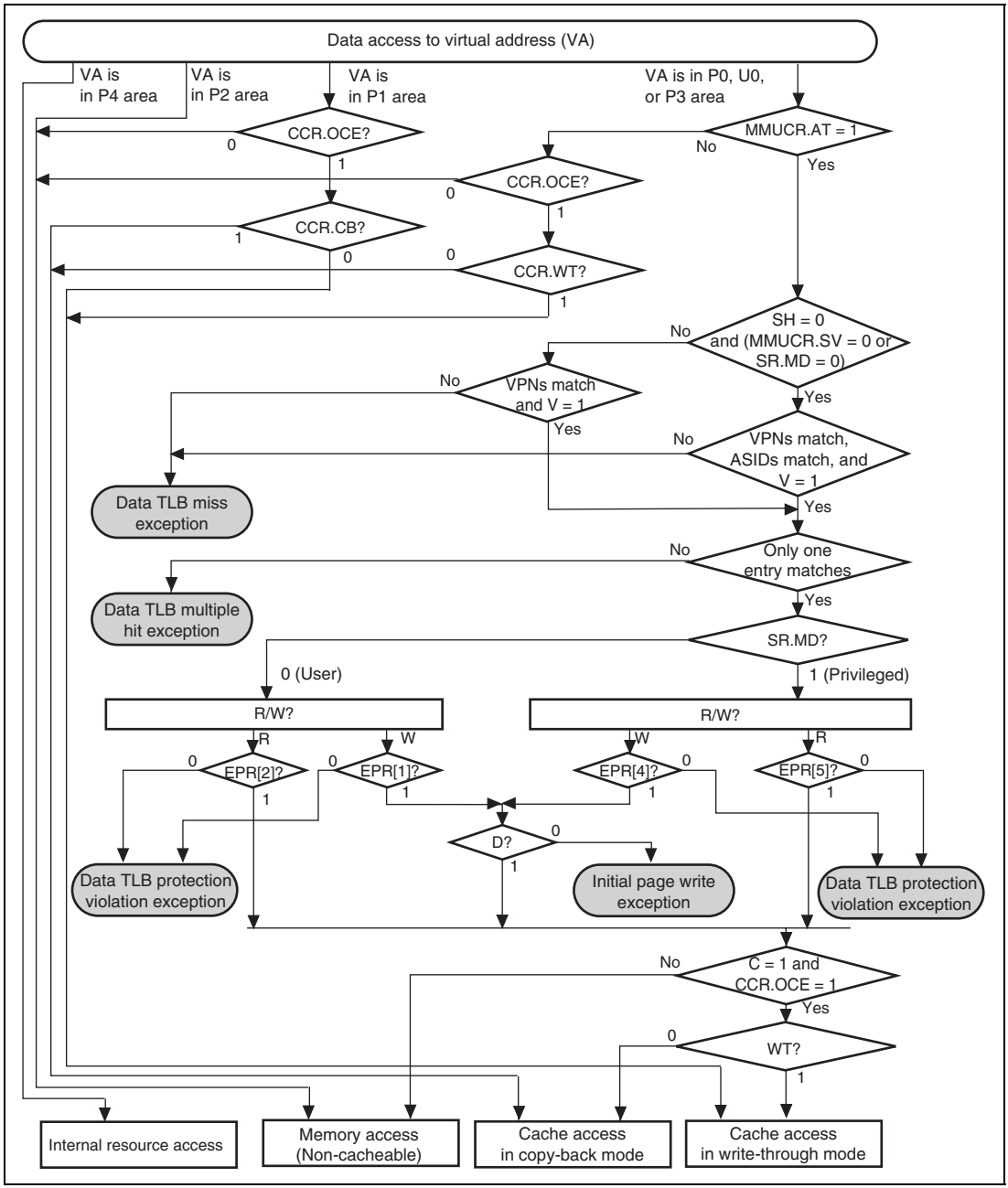


Figure 2.40 Flowchart of Memory Access Using UTLB (TLB Extended Mode)

Figure 2.41 is a flowchart of memory access using the ITLB in TLB extended mode.

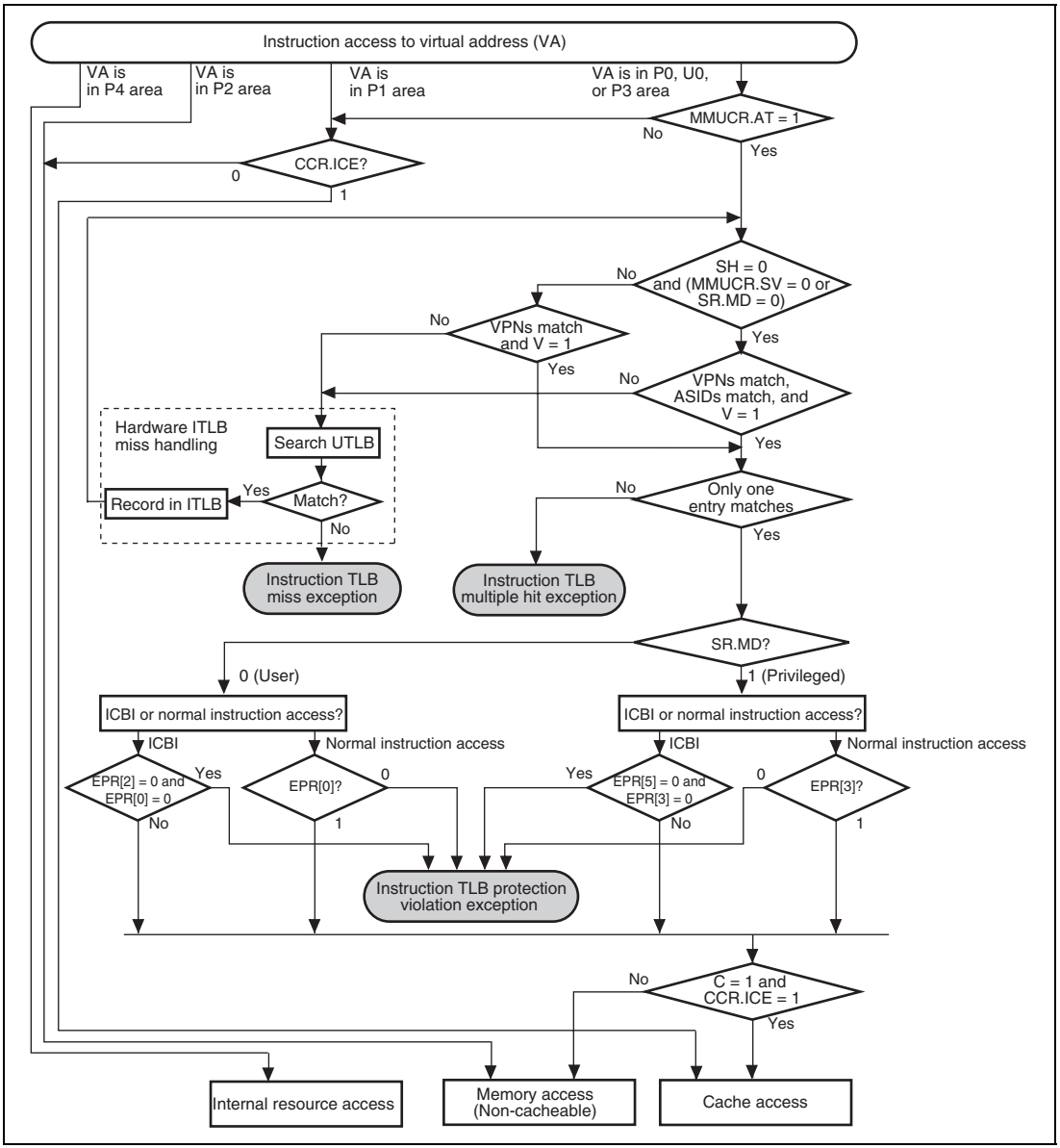


Figure 2.41 Flowchart of Memory Access Using ITLB (TLB Extended Mode)

2.7.5 MMU Functions

(1) MMU Hardware Management

The SH-4A supports the following MMU functions.

1. The MMU decodes the virtual address to be accessed by software, and performs address translation by controlling the UTLB/ITLB in accordance with the MMUCR settings.
2. The MMU determines the cache access status on the basis of the page management information read during address translation (C and WT bits).
3. If address translation cannot be performed normally in a data access or instruction access, the MMU notifies software by means of an MMU exception.
4. If address translation information is not recorded in the ITLB in an instruction access, the MMU searches the UTLB. If the necessary address translation information is recorded in the UTLB, the MMU copies this information into the ITLB in accordance with the LRUI bit setting in MMUCR.

(2) MMU Software Management

Software processing for the MMU consists of the following:

1. Setting of MMU-related registers. Some registers are also partially updated by hardware automatically.
2. Recording, deletion, and reading of TLB entries. There are two methods of recording UTLB entries: by using the LDTLB instruction, or by writing directly to the memory-mapped UTLB. ITLB entries can only be recorded by writing directly to the memory-mapped ITLB. Deleting or reading UTLB/ITLB entries is enabled by accessing the memory-mapped UTLB/ITLB.
3. MMU exception handling. When an MMU exception occurs, processing is performed based on information set by hardware.

(3) MMU Instruction (LDTLB)

A TLB load instruction (LDTLB) is provided for recording UTLB entries. When an LDTLB instruction is issued, the SH-4A copies the contents of PTEH and PTEL (also the contents of PTEA in TLB extended mode) to the UTLB entry indicated by the URC bit in MMUCR. ITLB entries are not updated by the LDTLB instruction, and therefore address translation information purged from the UTLB entry may still remain in the ITLB entry. As the LDTLB instruction changes address translation information, ensure that it is issued by a program in the P1 or P2 area.

After the LDTLB instruction has been executed, execute one of the following three methods before an access (include an instruction fetch) the area where TLB is used to translate the address is performed.

1. Execute a branch using the RTE instruction. In this case, the branch destination may be the area where TLB is used to translate the address.
2. Execute the ICBI instruction for any address (including non-cacheable area).
3. If the LT bit in IRMCR is 0 (initial value) before executing the LDTLB instruction, the specific instruction does not need to be executed. However, note that the CPU processing performance will be lowered because the instruction fetch is performed again for the next instruction after MMUCR has been updated.

Note that the method 3 may not be guaranteed in the future SuperH Series. Therefore, it is recommended that the method 1 or 2 should be used for being compatible with the future SuperH Series.

The operation of the LDTLB instruction is shown in figures 2.42 and 2.43.

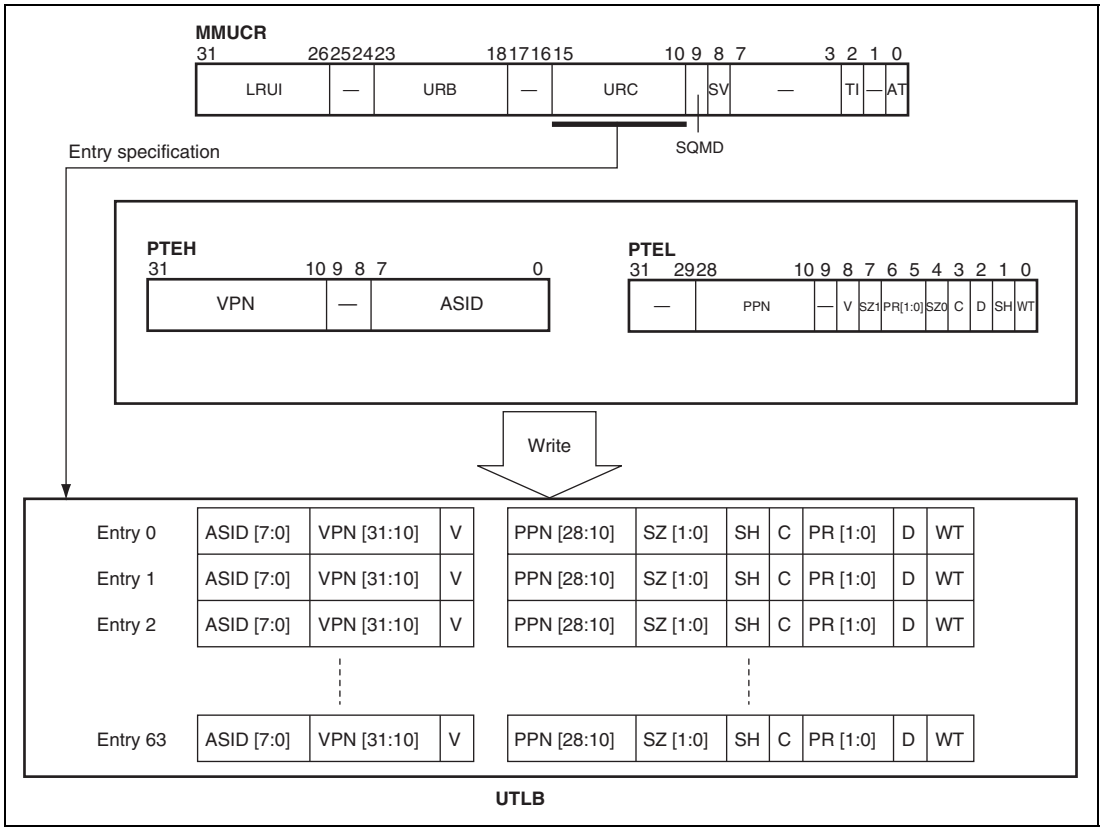


Figure 2.42 Operation of LDTLB Instruction (TLB Compatible Mode)

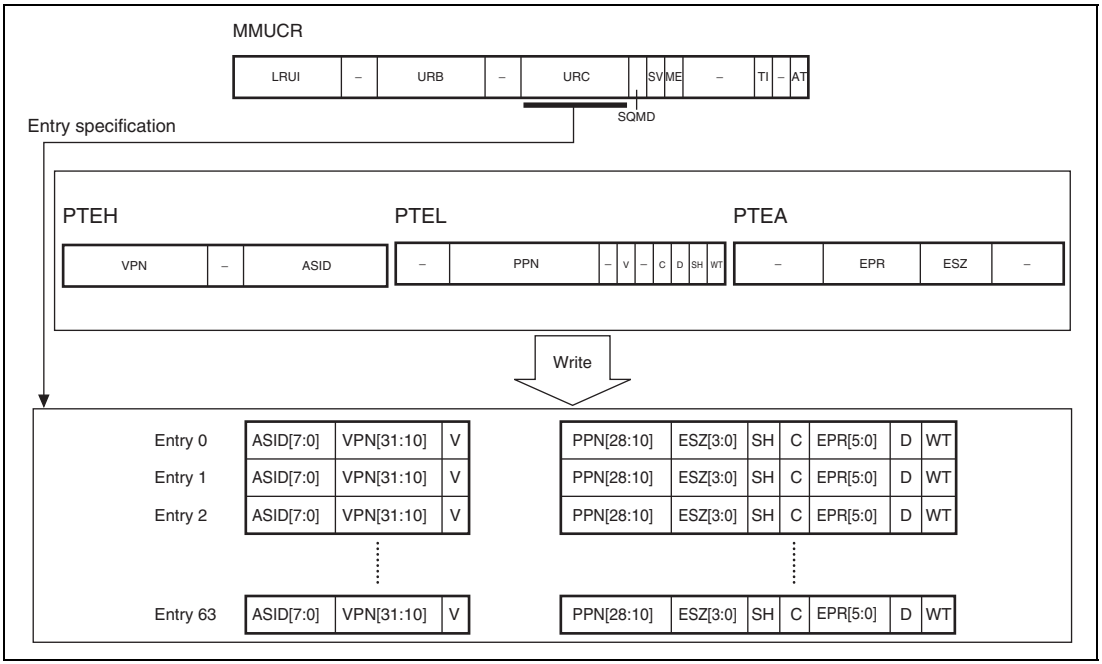


Figure 2.43 Operation of LDTLB Instruction (TLB Extended Mode)

(4) Hardware ITLB Miss Handling

In an instruction access, the SH-4A searches the ITLB. If it cannot find the necessary address translation information (ITLB miss occurred), the UTLB is searched by hardware, and if the necessary address translation information is present, it is recorded in the ITLB. This procedure is known as hardware ITLB miss handling. If the necessary address translation information is not found in the UTLB search, an instruction TLB miss exception is generated and processing passes to software.

(5) Avoiding Synonym Problems

When information on 1- or 4-Kbyte pages is written as TLB entries, a synonym problem may arise. The problem is that, when a number of virtual addresses are mapped onto a single physical address, the same physical address data is written to a number of cache entries, and it becomes impossible to guarantee data integrity. This problem does not occur with the instruction TLB and instruction cache because only data is read in these cases. In this LSI, entry specification is performed using bits 12 to 5 of the virtual address in order to achieve fast operand cache operation. However, bits 12 to 10 of the virtual address in the case of a 1-Kbyte page, and bit 12 of the virtual address in the case of a 4-Kbyte page, are subject to address translation. As a result, bits 12 to 10 of the physical address after translation may differ from bits 12 to 10 of the virtual address.

Consequently, the following restrictions apply to the writing of address translation information as UTLB entries.

- When address translation information whereby a number of 1-Kbyte page UTLB entries are translated into the same physical address is written to the UTLB, ensure that the VPN[12:10] values are the same.
- When address translation information whereby a number of 4-Kbyte page UTLB entries are translated into the same physical address is written to the UTLB, ensure that the VPN[12] value is the same.
- Do not use 1-Kbyte page UTLB entry physical addresses with UTLB entries of a different page size.
- Do not use 4-Kbyte page UTLB entry physical addresses with UTLB entries of a different page size.

The above restrictions apply only when performing accesses using the cache.

2.7.6 MMU Exceptions

There are seven MMU exceptions: instruction TLB multiple hit exception, instruction TLB miss exception, instruction TLB protection violation exception, data TLB multiple hit exception, data TLB miss exception, data TLB protection violation exception, and initial page write exception. Refer to figures 2.35, 2.36, 2.40, 2.41.

(1) Instruction TLB Multiple Hit Exception

An instruction TLB multiple hit exception occurs when more than one ITLB entry matches the virtual address to which an instruction access has been made. If multiple hits occur when the UTLB is searched by hardware in hardware ITLB miss handling, an instruction TLB multiple hit exception will result.

When an instruction TLB multiple hit exception occurs, a reset is executed and cache coherency is not guaranteed.

(a) Hardware Processing

In the event of an instruction TLB multiple hit exception, hardware carries out the following processing:

1. Sets the virtual address at which the exception occurred in TEA.
2. Sets exception code H'140 in EXPEVT.
3. Branches to the reset handling routine (H'A000 0000).

(b) Software Processing (Reset Routine)

The ITLB entries which caused the multiple hit exception are checked in the reset handling routine. This exception is intended for use in program debugging, and should not normally be generated.

(2) Instruction TLB Miss Exception

An instruction TLB miss exception occurs when address translation information for the virtual address to which an instruction access is made is not found in the UTLB entries by the hardware ITLB miss handling routine. The instruction TLB miss exception processing carried out by hardware and software is shown below. This is the same as the processing for a data TLB miss exception.

(a) Hardware Processing

In the event of an instruction TLB miss exception, hardware carries out the following processing:

1. Sets the VPN of the virtual address at which the exception occurred in PTEH.
2. Sets the virtual address at which the exception occurred in TEA.
3. Sets exception code H'040 in EXPEVT.
4. Sets the PC value indicating the address of the instruction at which the exception occurred in SPC. If the exception occurred at a delay slot, sets the PC value indicating the address of the delayed branch instruction in SPC.
5. Sets the SR contents at the time of the exception in SSR. The R15 contents at this time are saved in SGR.
6. Sets the MD bit in SR to 1, and switches to privileged mode.
7. Sets the BL bit in SR to 1, and masks subsequent exception requests.
8. Sets the RB bit in SR to 1.
9. Branches to the address obtained by adding offset H'0000 0400 to the contents of VBR, and starts the instruction TLB miss exception handling routine.

(b) Software Processing (Instruction TLB Miss Exception Handling Routine)

Software is responsible for searching the external memory page table and assigning the necessary page table entry. Software should carry out the following processing in order to find and assign the necessary page table entry.

1. In TLB compatible mode, write to PTEL the values of the PPN, PR, SZ, C, D, SH, V, and WT bits in the page table entry stored in the address translation table for external memory.
   In TLB extended mode, write to PTEL and PTEA the values of the PPN, EPR, ESZ, C, D, SH, V, and WT bits in the page table entry stored in the address translation table for external memory.
2. When the entry to be replaced in entry replacement is specified by software, write the value to the URC bits in MMUCR. If URC is greater than URB at this time, the value should be changed to an appropriate value after issuing an LDTLB instruction.

3. In TLB compatible mode, execute the LDTLB instruction and write the contents of PTEH and PTEL to the TLB.
   In TLB extended mode, execute the LDTLB instruction and write the contents of PTEH, PTEL, PTEA to the UTLB.
4. Finally, execute the exception handling return instruction (RTE) to terminate the exception handling routine and return control to the normal flow. The RTE instruction should be issued at least one instruction after the LDTLB instruction.

(3) Instruction TLB Protection Violation Exception

An instruction TLB protection violation exception occurs when, even though an ITLB entry contains address translation information matching the virtual address to which an instruction access is made, the actual access type is not permitted by the access right specified by the PR or EPR bit. The instruction TLB protection violation exception processing carried out by hardware and software is shown below.

(a) Hardware Processing

In the event of an instruction TLB protection violation exception, hardware carries out the following processing:

1. Sets the VPN of the virtual address at which the exception occurred in PTEH.
2. Sets the virtual address at which the exception occurred in TEA.
3. Sets exception code H'0A0 in EXPEVT.
4. Sets the PC value indicating the address of the instruction at which the exception occurred in SPC. If the exception occurred at a delay slot, sets the PC value indicating the address of the delayed branch instruction in SPC.
5. Sets the SR contents at the time of the exception in SSR. The R15 contents at this time are saved in SGR.
6. Sets the MD bit in SR to 1, and switches to privileged mode.
7. Sets the BL bit in SR to 1, and masks subsequent exception requests.
8. Sets the RB bit in SR to 1.
9. Branches to the address obtained by adding offset H'0000 0100 to the contents of VBR, and starts the instruction TLB protection violation exception handling routine.

(b) Software Processing (Instruction TLB Protection Violation Exception Handling Routine)

Resolve the instruction TLB protection violation, execute the exception handling return instruction (RTE), terminate the exception handling routine, and return control to the normal flow. The RTE instruction should be issued at least one instruction after the LDTLB instruction.

(4) Data TLB Multiple Hit Exception

A data TLB multiple hit exception occurs when more than one UTLB entry matches the virtual address to which a data access has been made.

When a data TLB multiple hit exception occurs, a reset is executed, and cache coherency is not guaranteed. The contents of PPN in the UTLB prior to the exception may also be corrupted.

(a) Hardware Processing

In the event of a data TLB multiple hit exception, hardware carries out the following processing:

1. Sets the virtual address at which the exception occurred in TEA.
2. Sets exception code H'140 in EXPEVT.
3. Branches to the reset handling routine (H'A000 0000).

(b) Software Processing (Reset Routine)

The UTLB entries which caused the multiple hit exception are checked in the reset handling routine. This exception is intended for use in program debugging, and should not normally be generated.

(5) Data TLB Miss Exception

A data TLB miss exception occurs when address translation information for the virtual address to which a data access is made is not found in the UTLB entries. The data TLB miss exception processing carried out by hardware and software is shown below.

(a) Hardware Processing

In the event of a data TLB miss exception, hardware carries out the following processing:

1. Sets the VPN of the virtual address at which the exception occurred in PTEH.
2. Sets the virtual address at which the exception occurred in TEA.
3. Sets exception code H'040 in the case of a read, or H'060 in the case of a write in EXPEVT (OCBP, OCBWB: read; OCBI, MOVCA.L: write).
4. Sets the PC value indicating the address of the instruction at which the exception occurred in SPC. If the exception occurred at a delay slot, sets the PC value indicating the address of the delayed branch instruction in SPC.
5. Sets the SR contents at the time of the exception in SSR. The R15 contents at this time are saved in SGR.
6. Sets the MD bit in SR to 1, and switches to privileged mode.

7. Sets the BL bit in SR to 1, and masks subsequent exception requests.
8. Sets the RB bit in SR to 1.
9. Branches to the address obtained by adding offset H'0000 0400 to the contents of VBR, and starts the data TLB miss exception handling routine.

(b) Software Processing (Data TLB Miss Exception Handling Routine)

Software is responsible for searching the external memory page table and assigning the necessary page table entry. Software should carry out the following processing in order to find and assign the necessary page table entry.

1. In TLB compatible mode, write to PTEL the values of the PPN, PR, SZ, C, D, SH, V, and WT bits in the page table entry stored in the address translation table for external memory.
   In TLB extended mode, write to PTEL and PTEA the values of the PPN, EPR, ESZ, C, D, SH, V, and WT bits in the page table entry stored in the address translation table for external memory.
2. When the entry to be replaced in entry replacement is specified by software, write the value to the URC bits in MMUCR. If URC is greater than URB at this time, the value should be changed to an appropriate value after issuing an LDTLB instruction.
3. In TLB compatible mode, execute the LDTLB instruction and write the contents of PTEH and PTEL to the TLB.
   In TLB extended mode, execute the LDTLB instruction and write the contents of PTEH, PTEL, PTEA to the UTLB.
4. Finally, execute the exception handling return instruction (RTE), terminate the exception handling routine, and return control to the normal flow. The RTE instruction should be issued at least one instruction after the LDTLB instruction.

(6) Data TLB Protection Violation Exception

A data TLB protection violation exception occurs when, even though a UTLB entry contains address translation information matching the virtual address to which a data access is made, the actual access type is not permitted by the access right specified by the PR or EPR bit. The data TLB protection violation exception processing carried out by hardware and software is shown below.

(a) Hardware Processing

In the event of a data TLB protection violation exception, hardware carries out the following processing:

1. Sets the VPN of the virtual address at which the exception occurred in PTEH.
2. Sets the virtual address at which the exception occurred in TEA.
3. Sets exception code H'0A0 in the case of a read, or H'0C0 in the case of a write in EXPEVT (OCBP, OCBWB: read; OCBI, MOVCA.L: write).
4. Sets the PC value indicating the address of the instruction at which the exception occurred in SPC. If the exception occurred at a delay slot, sets the PC value indicating the address of the delayed branch instruction in SPC.
5. Sets the SR contents at the time of the exception in SSR. The R15 contents at this time are saved in SGR.
6. Sets the MD bit in SR to 1, and switches to privileged mode.
7. Sets the BL bit in SR to 1, and masks subsequent exception requests.
8. Sets the RB bit in SR to 1.
9. Branches to the address obtained by adding offset H'0000 0100 to the contents of VBR, and starts the data TLB protection violation exception handling routine.

(b) Software Processing (Data TLB Protection Violation Exception Handling Routine)

Resolve the data TLB protection violation, execute the exception handling return instruction (RTE), terminate the exception handling routine, and return control to the normal flow. The RTE instruction should be issued at least one instruction after the LDTLB instruction.

(7) Initial Page Write Exception

An initial page write exception occurs when the D bit is 0 even though a UTLB entry contains address translation information matching the virtual address to which a data access (write) is made, and the access is permitted. The initial page write exception processing carried out by hardware and software is shown below.

(a) Hardware Processing

In the event of an initial page write exception, hardware carries out the following processing:

1. Sets the VPN of the virtual address at which the exception occurred in PTEH.
2. Sets the virtual address at which the exception occurred in TEA.
3. Sets exception code H'080 in EXPEVT.
4. Sets the PC value indicating the address of the instruction at which the exception occurred in SPC. If the exception occurred at a delay slot, sets the PC value indicating the address of the delayed branch instruction in SPC.
5. Sets the SR contents at the time of the exception in SSR. The R15 contents at this time are saved in SGR.
6. Sets the MD bit in SR to 1, and switches to privileged mode.
7. Sets the BL bit in SR to 1, and masks subsequent exception requests.
8. Sets the RB bit in SR to 1.
9. Branches to the address obtained by adding offset H'0000 0100 to the contents of VBR, and starts the initial page write exception handling routine.

(b) Software Processing (Initial Page Write Exception Handling Routine)

Software is responsible for the following processing:

1. Retrieve the necessary page table entry from external memory.
2. Write 1 to the D bit in the external memory page table entry.
3. In TLB compatible mode, write to PTEL the values of the PPN, PR, SZ, C, D, SH, V, and WT bits in the page table entry stored in the address translation table for external memory.
   In TLB extended mode, write to PTEL and PTEA the values of the PPN, EPR, ESZ, C, D, SH, V, and WT bits in the page table entry stored in the address translation table for external memory.
4. When the entry to be replaced in entry replacement is specified by software, write that value to the URC bits in MMUCR. If URC is greater than URB at this time, the value should be changed to an appropriate value after issuing an LDTLB instruction.
5. In TLB compatible mode, execute the LDTLB instruction and write the contents of PTEH and PTEL to the TLB.
   In TLB extended mode, execute the LDTLB instruction and write the contents of PTEH, PTEL, PTEA to the UTLB.
6. Finally, execute the exception handling return instruction (RTE), terminate the exception handling routine, and return control to the normal flow. The RTE instruction should be issued at least one instruction after the LDTLB instruction.

2.7.7 Memory-Mapped TLB Configuration

To enable the ITLB and UTLB to be managed by software, their contents are allowed to be read from and written to by a program in the P1/P2 area with a MOV instruction in privileged mode. Operation is not guaranteed if access is made from a program in another area.

After the memory-mapped TLB has been accessed, execute one of the following three methods before an access (including an instruction fetch) to an area other than the P1/P2 area is performed.

1. Execute a branch using the RTE instruction. In this case, the branch destination may be an area other than the P1/P2 area.
2. Execute the ICBI instruction for any address (including non-cacheable area).
3. If the MT bit in IRMCR is 0 (initial value) before accessing the memory-mapped TLB, the specific instruction does not need to be executed. However, note that the CPU processing performance will be lowered because the instruction fetch is performed again for the next instruction after MMUCR has been updated.

Note that the method 3 may not be guaranteed in the future SuperH Series. Therefore, it is recommended that the method 1 or 2 should be used for being compatible with the future SuperH Series.

The ITLB and UTLB are allocated to the P4 area in the virtual address space.

In TLB compatible mode, VPN, V, and ASID in the ITLB can be accessed as an address array, PPN, V, SZ, PR, C, and SH as a data array. VPN, D, V, and ASID in the UTLB can be accessed as an address array, PPN, V, SZ, PR, C, D, WT, and SH as a data array. V and D can be accessed from both the address array side and the data array side.

In TLB extended mode, VPN, V, and ASID in the ITLB can be accessed as an address array, PPN, V, ESZ, EPR, C, and SH as a data array. VPN, D, V, and ASID in the UTLB can be accessed as an address array, PPN, V, ESZ, EPR, C, D, WT, and SH as a data array. V and D can be accessed from both the address array side and the data array side.

In both TLB compatible mode and TLB extended mode, only longword access is possible. Instruction fetches cannot be performed in these areas. For reserved bits, a write value of 0 should be specified; their read value is undefined.

(1) ITLB Address Array

The ITLB address array is allocated to addresses H'F200 0000 to H'F2FF FFFF in the P4 area. An address array access requires a 32-bit address field specification (when reading or writing) and a 32-bit data field specification (when writing). Information for selecting the entry to be accessed is specified in the address field, and VPN, V, and ASID to be written to the address array are specified in the data field.

In the address field, bits [31:24] have the value H'F2 indicating the ITLB address array and the entry is specified by bits [9:8]. As only longword access is used, 0 should be specified for address field bits [1:0].

In the data field, bits [31:10] indicate VPN, bit [8] indicates V, and bits [7:0] indicate ASID.

The following two kinds of operation can be used on the ITLB address array:

1. ITLB address array read

   VPN, V, and ASID are read into the data field from the ITLB entry corresponding to the entry set in the address field.
2. ITLB address array write

   VPN, V, and ASID specified in the data field are written to the ITLB entry corresponding to the entry set in the address field.

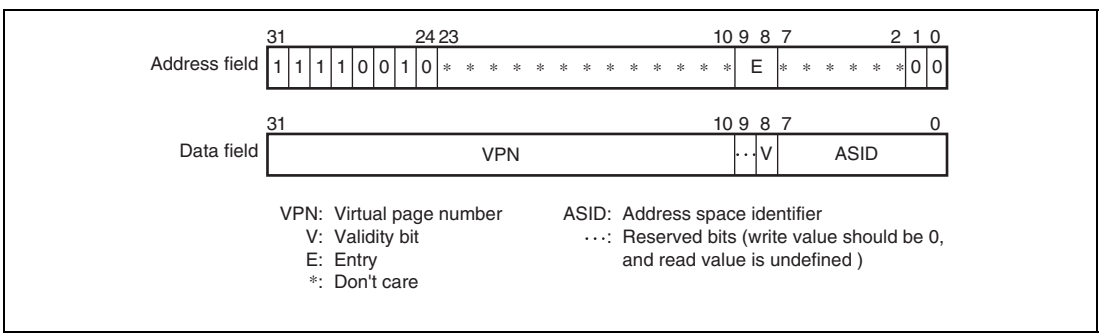


Figure 2.44 Memory-Mapped ITLB Address Array

(2) ITLB Data Array (TLB Compatible Mode)

The ITLB data array is allocated to addresses H'F300 0000 to H'F37F FFFF in the P4 area. A data array access requires a 32-bit address field specification (when reading or writing) and a 32-bit data field specification (when writing). Information for selecting the entry to be accessed is specified in the address field, and PPN, V, SZ, PR, C, and SH to be written to the data array are specified in the data field.

In the address field, bits [31:23] have the value H'F30 indicating ITLB data array and the entry is specified by bits [9:8].

In the data field, bits [28:10] indicate PPN, bit [8] indicates V, bits [7] and [4] indicate SZ, bit [6] indicates PR, bit [3] indicates C, and bit [1] indicates SH.

The following two kinds of operation can be used on ITLB data array:

1. ITLB data array read
   PPN, V, SZ, PR, C, and SH are read into the data field from the ITLB entry corresponding to the entry set in the address field.
2. ITLB data array write
   PPN, V, SZ, PR, C, and SH specified in the data field are written to the ITLB entry corresponding to the entry set in the address field.

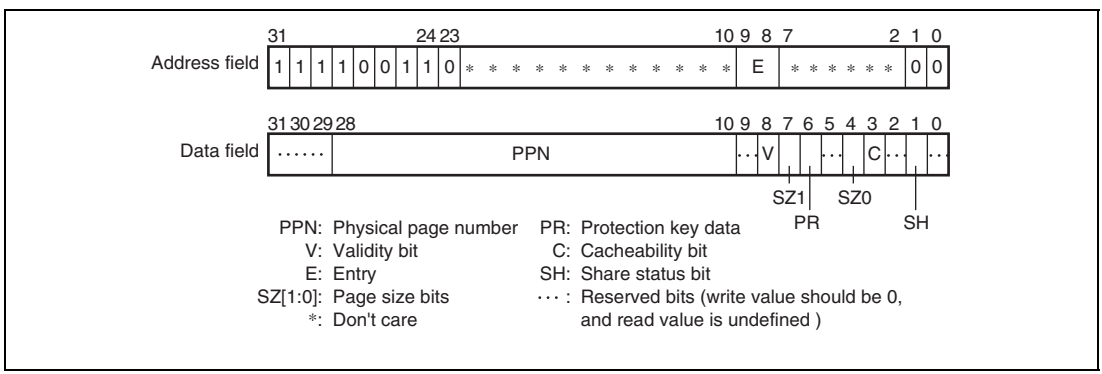


Figure 2.45 Memory-Mapped ITLB Data Array (TLB Compatible Mode)

(3) ITLB Data Array (TLB Extended Mode)

In TLB extended mode the names of the data arrays have been changed from ITLB data array to ITLB data array 1, ITLB data array 2 is added, and the EPR and ESZ bits are accessible. In TLB extended mode, the PR and SZ bits of ITLB data array 1 are reserved and 0 should be specified as the write value for these bits. In addition, when a write to ITLB data array 1 is performed, a write to ITLB data array 2 of the same entry should always be performed.

In TLB compatible mode (MMUCR.ME = 0), ITLB data array 2 cannot be accessed. Operation if they are accessed is not guaranteed.

(a) ITLB Data Array 1

In TLB extended mode, bits 7, 6, and 4 in the data field, which correspond to the PR and SZ bits in compatible mode, are reserved. Specify 0 as the write value for these bits.

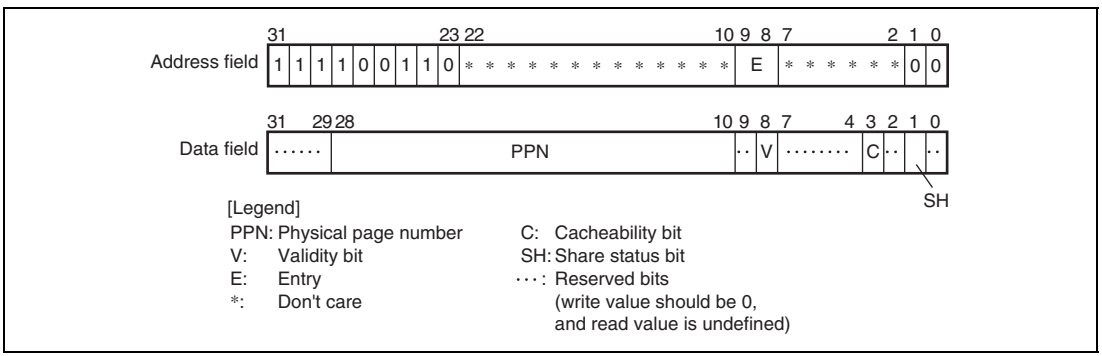


Figure 2.46 Memory-Mapped ITLB Data Array 1 (TLB Extended Mode)

(b) ITLB Data Array 2

The ITLB data array is allocated to addresses H'F380 0000 to H'F3FF FFFF in the P4 area. Access to data array 2 requires a 32-bit address field specification (when reading or writing) and a 32-bit data field specification (when writing). Information for selecting the entry to be accessed is specified in the address field, and EPR and ESZ to be written to data array 2 are specified in the data field.

In the address field, bits [31:23] have the value H'F38 indicating ITLB data array 2 and the entry is specified by bits [9:8].

In the data field, bits [13], [11], [10], and [8] indicate EPR[5], [3], [2], and [0], and bits [7:4] indicate ESZ, respectively.

The following two kinds of operation can be applied to ITLB data array 2:

1. ITLB data array 2 read

   EPR and ESZ are read into the data field from the ITLB entry corresponding to the entry set in the address field.
2. ITLB data array 2 write

   EPR and ESZ specified in the data field are written to the ITLB entry corresponding to the entry set in the address field.

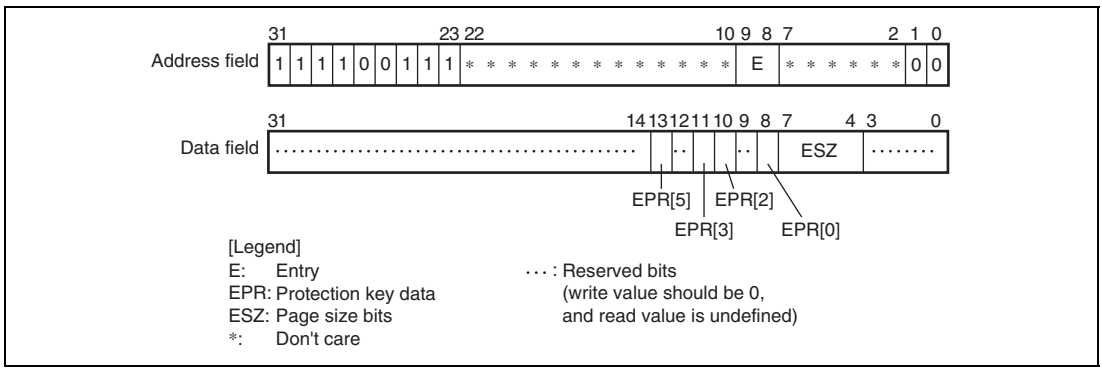


Figure 2.47 Memory-Mapped ITLB Data Array 2 (TLB Extended Mode)

(4) UTLB Address Array

The UTLB address array is allocated to addresses H'F600 0000 to H'F60F FFFF in the P4 area. An address array access requires a 32-bit address field specification (when reading or writing) and a 32-bit data field specification (when writing). Information for selecting the entry to be accessed is specified in the address field, and VPN, D, V, and ASID to be written to the address array are specified in the data field.

In the address field, bits [31:20] have the value H'F60 indicating the UTLB address array and the entry is specified by bits [13:8]. Bit [7] that is the association bit (A bit) in the address field specifies whether address comparison is performed in a write to the UTLB address array.

In the data field, bits [31:10] indicate VPN, bit [9] indicates D, bit [8] indicates V, and bits [7:0] indicate ASID.

The following three kinds of operation can be used on the UTLB address array:

1. UTLB address array read

   VPN, D, V, and ASID are read into the data field from the UTLB entry corresponding to the entry set in the address field. In a read, associative operation is not performed regardless of whether the association bit specified in the address field is 1 or 0.
2. UTLB address array write (non-associative)

   VPN, D, V, and ASID specified in the data field are written to the UTLB entry corresponding to the entry set in the address field. The A bit in the address field should be cleared to 0.
3. UTLB address array write (associative)

   When a write is performed with the A bit in the address field set to 1, comparison of all the UTLB entries is carried out using the VPN specified in the data field and ASID in PTEH. The usual address comparison rules are followed, but if a UTLB miss occurs, the result is no operation, and an exception is not generated. If the comparison identifies a UTLB entry corresponding to the VPN specified in the data field, D and V specified in the data field are written to that entry. This associative operation is simultaneously carried out on the ITLB, and if a matching entry is found in the ITLB, V is written to that entry. Even if the UTLB comparison results in no operation, a write to the ITLB is performed as long as a matching entry is found in the ITLB. If there is a match in both the UTLB and ITLB, the UTLB information is also written to the ITLB.

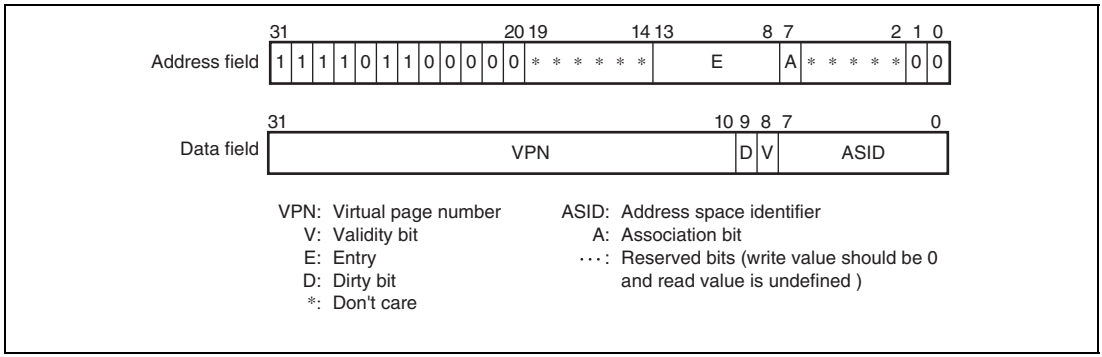


Figure 2.48 Memory-Mapped UTLB Address Array

(5) UTLB Data Array (TLB Compatible Mode)

The UTLB data array is allocated to addresses H'F700 0000 to H'F70F FFFF in the P4 area. A data array access requires a 32-bit address field specification (when reading or writing) and a 32-bit data field specification (when writing). Information for selecting the entry to be accessed is specified in the address field, and PPN, V, SZ, PR, C, D, SH, and WT to be written to data array are specified in the data field.

In the address field, bits [31:20] have the value H'F70 indicating UTLB data array and the entry is specified by bits [13:8].

In the data field, bits [28:10] indicate PPN, bit [8] indicates V, bits [7] and [4] indicate SZ, bits [6:5] indicate PR, bit [3] indicates C, bit [2] indicates D, bit [1] indicates SH, and bit [0] indicates WT.

The following two kinds of operation can be used on UTLB data array:

1. UTLB data array read

   PPN, V, SZ, PR, C, D, SH, and WT are read into the data field from the UTLB entry corresponding to the entry set in the address field.
2. UTLB data array write

   PPN, V, SZ, PR, C, D, SH, and WT specified in the data field are written to the UTLB entry corresponding to the entry set in the address field.

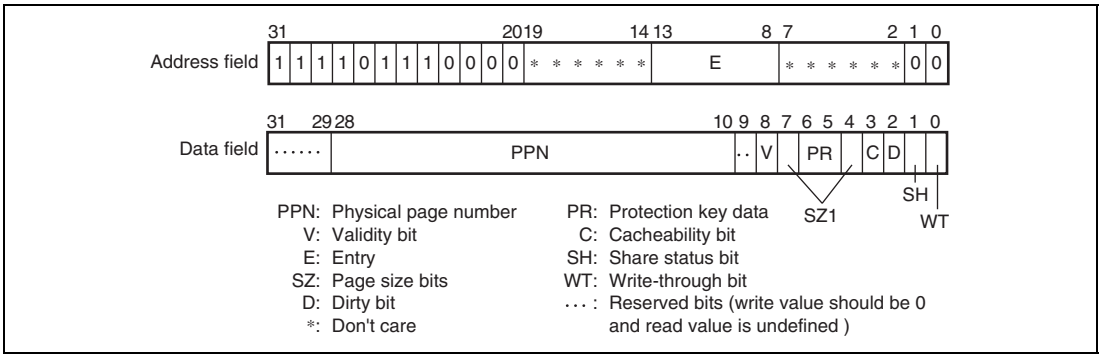


Figure 2.49 Memory-Mapped UTLB Data Array (TLB Compatible Mode)

(6) UTLB Data Array (TLB Extended Mode)

In TLB extended mode, the names of the data arrays have been changed from UTLB data array to UTLB data array 1, UTLB data array 2 is added, and the EPR and ESZ bits are accessible. In TLB extended mode, the PR and SZ bits of UTLB data array 1 are reserved and 0 should be specified as the write value for these bits. In addition, when a write to UTLB data array 1 is performed, a write to UTLB data array 2 of the same entry should always be performed after that.

In TLB compatible mode (MMUCR.ME = 0), UTLB data array 2 cannot be accessed. Operation if they are accessed is not guaranteed.

(a) UTLB Data Array 1

In TLB extended mode, bits 7 to 4 in the data field, which correspond to the PR and SZ bits in compatible mode, are reserved. Specify 0 as the write value for these bits.

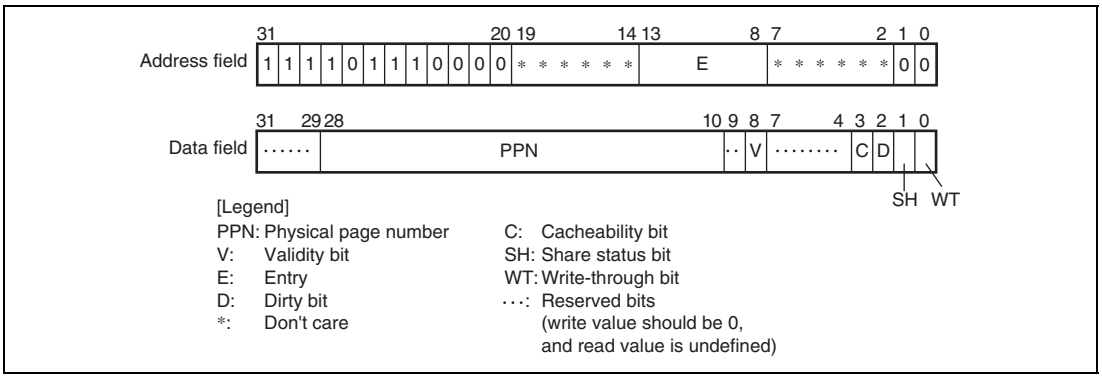


Figure 2.50 Memory-Mapped UTLB Data Array 1 (TLB Extended Mode)

(b) UTLB Data Array 2

The UTLB data array is allocated to addresses H'F780 0000 to H'F78F FFFF in the P4 area. Access to data array 2 requires a 32-bit address field specification (when reading or writing) and a 32-bit data field specification (when writing). Information for selecting the entry to be accessed is specified in the address field, and EPR and ESZ to be written to data array 2 are specified in the data field.

In the address field, bits [31:20] have the value H'F78 indicating UTLB data array 2 and the entry is specified by bits [13:8].

In the data field, bits [13:8] indicate EPR, and bits [7:4] indicate ESZ, respectively.

The following two kinds of operation can be applied to UTLB data array 2:

1. UTLB data array 2 read

   EPR and ESZ are read into the data field from the UTLB entry corresponding to the entry set in the address field.
2. UTLB data array 2 write

   EPR and ESZ specified in the data field are written to the UTLB entry corresponding to the entry set in the address field.

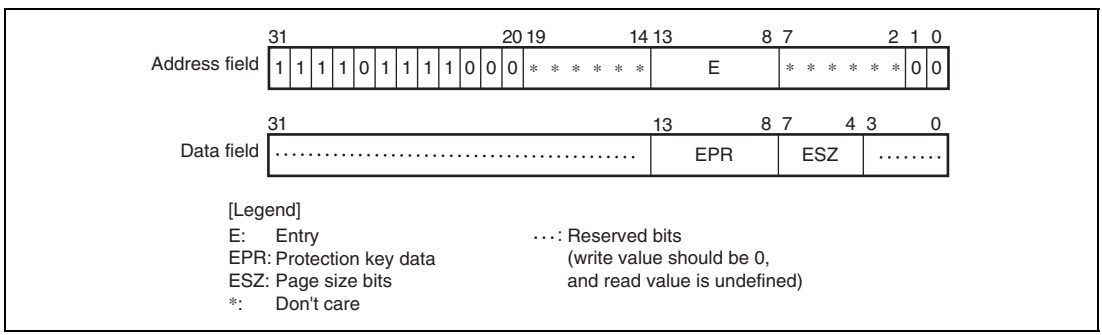


Figure 2.51 Memory-Mapped UTLB Data Array 2 (TLB Extended Mode)

2.7.8 32-Bit Address Extended Mode

Setting the SE bit in PASCR to 1 changes mode from 29-bit address mode which handles the 29-bit physical address space to 32-bit address extended mode which handles the 32-bit physical address space.

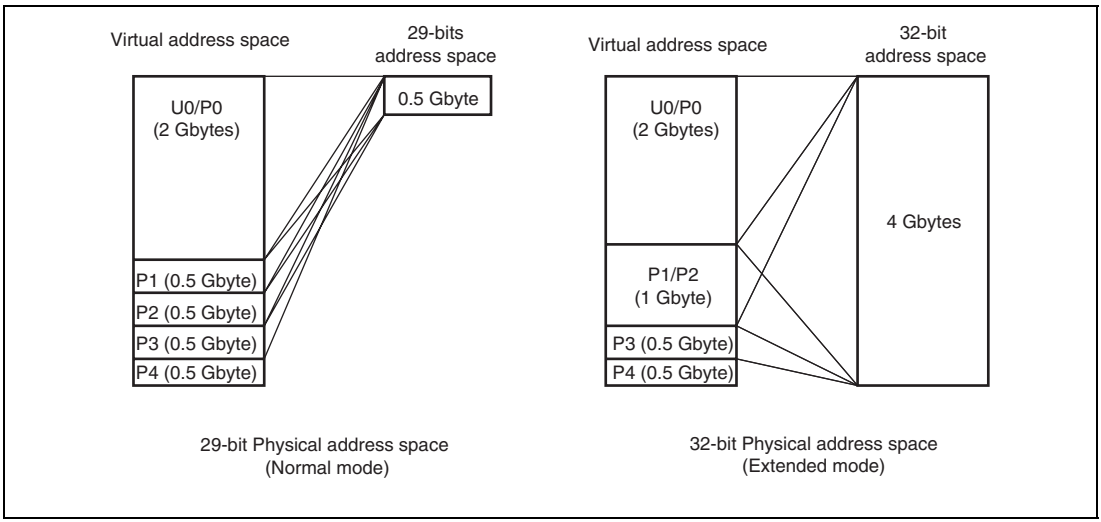


Figure 2.52 Physical Address Space (32-Bit Address Extended Mode)

(1) Overview of 32-Bit Address Extended Mode

In 32-bit address extended mode, the privileged space mapping buffer (PMB) is introduced. The PMB maps virtual addresses in the P1 or P2 area which are not translated in 29-bit address mode to the 32-bit physical address space. In areas which are target for address translation of the TLB (UTLB/ITLB), upper three bits in the PPN field of the UTLB or ITLB are extended and then addresses after the TLB translation can handle the 32-bit physical addresses.

As for the cache operation, P1 area is cacheable and P2 area is non-cacheable in the case of 29-bit address mode, but the cache operation of both P1 and P2 area are determined by the C bit and WT bit in the PMB in the case of 32-bit address mode.

(2) Transition to 32-Bit Address Extended Mode

The SH-4A enters 29-bit address mode after a power-on reset. Transition is made to 32-bit address extended mode by setting the SE bit in PASCR to 1. In 32-bit address extended mode, the MMU operates as follows.

1. When the AT bit in MMUCR is 0, virtual addresses in the U0, P0, or P3 area become 32-bit physical addresses. Addresses in the P1 or P2 area are translated according to the PMB mapping information. B'10 should be set to the upper 2 bits of virtual page number (VPN[31:30]) in the PMB in order to indicate P1 or P2 area. The operation is not guaranteed when the value except B'10 is set to these bits.
2. When the AT bit in MMUCR is 1, virtual addresses in the U0, P0, or P3 area are translated to 32-bit physical addresses according to the TLB conversion information. Addresses in the P1 or P2 area are translated according to the PMB mapping information. B'10 should be set to the upper 2 bits of virtual page number (VPN[31:30]) in the PMB in order to indicate P1 or P2 area. The operation is not guaranteed when the value except B'10 is set to these bits.
3. Regardless of the setting of the AT bit in MMUCR, bits 31 to 29 in physical addresses become B'111 in the control register area (addresses H'FC00 0000 to H'FFFF FFFF). When the control register area is recorded in the UTLB and accessed, B'111 should be set to PPN[31:29].

(3) Privileged Space Mapping Buffer (PMB) Configuration

In 32-bit address extended mode, virtual addresses in the P1 or P2 area are translated according to the PMB mapping information. The PMB has 16 entries and configuration of each entry is as follows.

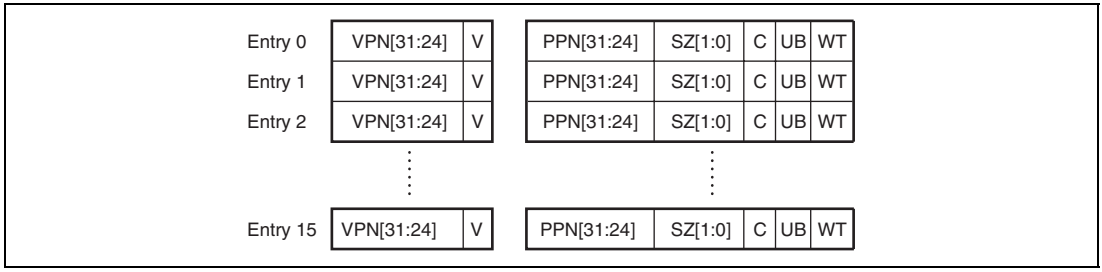


Figure 2.53 PMB Configuration

[Legend]

- VPN: Virtual page number
  For 16-Mbyte page: Upper 8 bits of virtual address
  For 64-Mbyte page: Upper 6 bits of virtual address
  For 128-Mbyte page: Upper 5 bits of virtual address
  For 512-Mbyte page: Upper 3 bits of virtual address

  Note: B'10 should be set to the upper 2 bits of VPN in order to indicate P1 or P2 area.

- SZ: Page size bits
  Specify the page size.
  00: 16-Mbyte page
  01: 64-Mbyte page
  10: 128-Mbyte page
  11: 512-Mbyte page

- V: Validity bit
  Indicates whether the entry is valid.
  0: Invalid
  1: Valid
  Cleared to 0 by a power-on reset.
  Not affected by a manual reset.

- PPN: Physical page number
  Upper 8 bits of the physical address of the physical page number.
  With a 16-Mbyte page, PPN[31:24] are valid.
  With a 64-Mbyte page, PPN[31:26] are valid.
  With a 128-Mbyte page, PPN[31:27] are valid.
  With a 512-Mbyte page, PPN[31:29] are valid.

- C: Cacheability bit
  Indicates whether a page is cacheable.
  0: Not cacheable
  1: Cacheable

- WT: Write-through bit
  Specifies the cache write mode.

0: Copy-back mode

1: Write-through mode

- UB: Buffered write bit

  Specifies whether a buffered write is performed.

  0: Buffered write (Data access of subsequent processing proceeds without waiting for the write to complete.)

  1: Unbuffered write (Data access of subsequent processing is stalled until the write has completed.)

(4) PMB Function

The SH-4A supports the following PMB functions.

1. Only memory-mapped write can be used for writing to the PMB. The LDTLB instruction cannot be used to write to the PMB.
2. Software must ensure that every accessed P1 or P2 address has a corresponding PMB entry before the access occurs. When an access to an address in the P1 or P2 area which is not recorded in the PMB is made, the SH-4A is reset by the TLB. In this case, the accessed address in the P1 or P2 area which causes the TLB reset is stored in the TEA and code H′140 in the EXPEVT.
3. The SH-4A does not guarantee the operation when multiple hit occurs in the PMB. Special care should be taken when the PMB mapping information is recorded by software.
4. The PMB does not have an associative write function.
5. Since there is no PR field in the PMB, read/write protection cannot be preformed. The address translation target of the PMB is the P1 or P2 address. In user mode access, an address error exception occurs.
6. Both entries from the UTLB and PMB are mixed and recorded in the ITLB by means of the hardware ITLB miss handling. However, these entries can be identified by checking whether VPN[31:30] is 10 or not. When an entry from the PMB is recorded in the ITLB, H′00, 01, and 1 are recorded in the ASID, PR, and SH fields which do not exist in the PMB, respectively.

(5) Memory-Mapped PMB Configuration

To enable the PMB to be managed by software, its contents are allowed to be read from and written to by a P1 or P2 area program with a MOV instruction in privileged mode. The PMB address array is allocated to addresses H'F610 0000 to H'F61F FFFF in the P4 area and the PMB data array to addresses H'F710 0000 to H'F71F FFFF in the P4 area. VPN and V in the PMB can be accessed as an address array, PPN, V, SZ, C, WT, and UB as a data array. V can be accessed from both the address array side and the data array side. A program which executes a PMB memory-mapped access should be placed in the page area at which the C bit in PMB is cleared to 0.

1. PMB address array read

   When memory reading is performed while bits 31 to 20 in the address field are specified as H'F61 which indicates the PMB address array and bits 11 to 8 in the address field as an entry, bits 31 to 24 in the data field are read as VPN and bit 8 in the data field as V.

2. PMB address array write

   When memory writing is performed while bits 31 to 20 in the address field are specified as H'F61 which indicates the PMB address array and bits 11 to 8 in the address field as an entry, and bits 31 to 24 in the data field are specified as VPN and bit 8 in the data field as V, data is written to the specified entry.

3. PMB data array read

   When memory reading is performed while bits 31 to 20 in the address field are specified as H'F71 which indicates the PMB data array and bits 11 to 8 in the address field as an entry, bits 31 to 24 in the data field are read as PPN, bit 9 in the data field as UB, bit 8 in the data field as V, bits 7 and 4 in the data field as SZ, bit 3 in the data field as C, and bit 0 in the data field as WT.

4. PMB data array write

   When memory writing is performed while bits 31 to 20 in the address field are specified as H'F71 which indicates the PMB data array and bits 11 to 8 in the address field as an entry, and bits 31 to 24 in the data field are specified as PPN, bit 9 in the data field as UB, bit 8 in the data field as V, bits 7 and 4 in the data field as SZ, bit 3 in the data field as C, and bit 0 in the data field as WT, data is written to the specified entry.

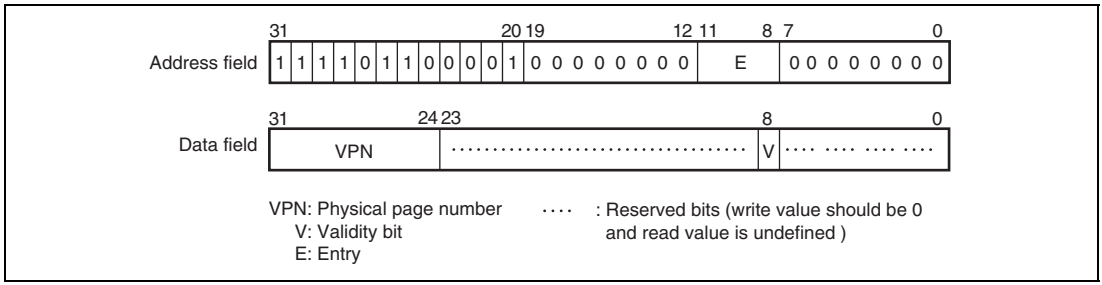


Figure 2.54 Memory-Mapped PMB Address Array

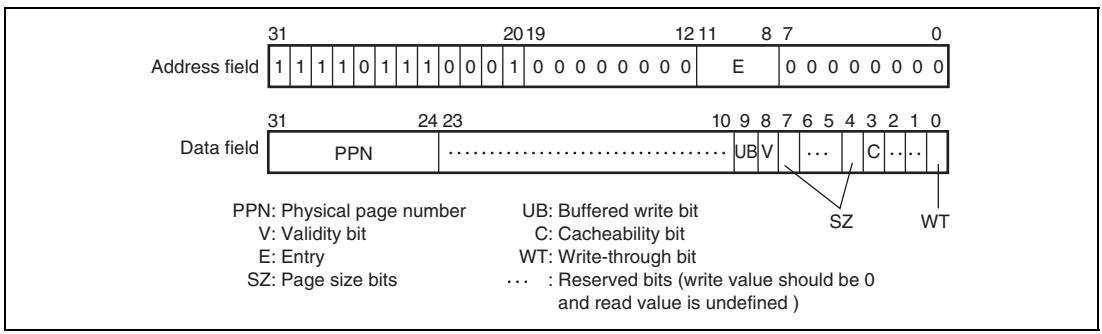


Figure 2.55 Memory-Mapped PMB Data Array

(6) Notes on Using 32-Bit Address Extended Mode

When using 32-bit address extended mode, note that the items described in this section are extended or changed as follows.

(a) PASCR.SE

The SE bit is added in bit 31 in the control register (PASCR). The bits 6 to 0 of the UB in the PASCR are invalid (Note that the bit 7 of the UB is still valid). When writing to the P1 or P2 area, the UB bit in the PMB controls whether a buffered write is performed or not. When the MMU is enabled, the UB bit in the TLB controls writing to the P0, P3, or U0 area. When the MMU is disabled, writing to the P0, P3, or U0 area is always performed as a buffered write.

Bit	Bit Name	Initial Value	R/W	Description
31	SE	0	R/W	0: 29-bit address mode 1: 32-bit address extended mode

Bit	Bit Name	Initial Value	R/W	Description
30 to 8	—	All 0	R	Reserved For details on reading from or writing to these bits, see description in General Precautions on Handling of Product.
7 to 0	UB	All 0	R/W	Buffered Write Control for Each Area (64 Mbytes) When writing is performed without using the cache or in the cache write-through mode, these bits specify whether the next bus access from the CPU waits for the end of writing for each area. 0: The CPU does not wait for the end of writing bus access and starts the next bus access 1: The CPU waits for the end of writing bus access and starts the next bus access UB[7]: Buffered write control for the control register area UB[6:0]: Buffered write control for each area (64 Mbytes; these bits are invalid in 32-bit address extended mode).

(b) ITLB

The PPN field in the ITLB is extended to bits 31 to 10.

(c) UTLB

The PPN field in the UTLB is extended to bits 31 to 10. The same UB bit as that in the PMB is added in each entry of the UTLB.

- UB: Buffered write bit

  Specifies whether a buffered write is performed.

  0: Buffered write (Data access of subsequent processing proceeds without waiting for the write to complete.)

  1: Unbuffered write (Data access of subsequent processing is stalled until the write has completed.)

In a memory-mapped TLB access, the UB bit can be read from or written to by bit 9 in the data array.

(d) PTEL

The same UB bit as that in the PMB is added in bit 9 in PTEL. This UB bit is written to the UB bit in the UTLB by the LDTLB instruction. The PPN field is extended to bits 31 to 10.

(e) CCR.CB

The CB bit in CCR is invalid. Whether a cacheable write for the P1 area is performed in copy-back mode or write-though mode is determined by the WT bit in the PMB.

(f) IRMCR.MT

The MT bit in IRMCR is valid for a memory-mapped PMB write.

(g) QACR0, QACR1

AREA0[4:2]/AREA1[4:2] fields of QACR0/QACR1 are extended to AREA0[7:2]/AREA1[7:2] corresponding to physical address [31:26].

(h) LSA0, LSA1, LDA0, LDA1

L0SADR, L1SADR, L0DADR, and L1DADR fields are extended to bits 31 to 10.

When using 32-bit extended address mode, the following notes should be applied to software.

1. For the SE bit switching, switching from 0 to 1 is only supported in a boot routine which is allocated in an area where caching and TLB-based address translation are not allowed and runs after a power-on reset or manual reset.
2. After switching the SE bit, an area in which the program is allocated becomes the target of the PMB address translation. Therefore, the area should be recorded in the PMB before switching the SE bit. An address which may be accessed in the P1 or P2 area such as the exception handler should also be recorded in the PMB.
3. When an external memory access occurs by an operand memory access located before the MOV.L instruction which switches the SE bit, external memory space addresses accessed in both address modes should be the same.
4. Note that the V bit is mapped to both address array and data array in PMB registration. That is, first write 0 to the V bit in one of arrays and then write 1 to the V bit in another array.

2.7.9 32-Bit Boot Function

The address mode of the SH-4A after a power-on reset or manual reset can be switched between 29-bit address mode and 32-bit address extended mode by specifying external pins. The following changes apply when the SH-4A is booted up in 32-bit address extended mode.

(1) Initial Entries to PMB

When 32-bit address extended mode is specified by external pins, the following initial entries are recorded in the PMB after a power-on reset or manual reset, and the SE bit in the PASCR register is initialized to 1. For entries 2 to 15, only the V bit is initialized to 0.

Entry	VPN[31:24]	PPN[31:24]	V	SZ[1:0]	C	UB	WT
0	10000000	00000000	1	11	1	0	1
1	10100000	00000000	1	11	0	0	0

(2) Notes on 32-Bit Boot

Immediately after a power-on or manual reset, the P1 or P2 area is mapped to the PMB. Therefore, when an area other than that indicated by the initial entry needs to be mapped, follow the procedures below to modify the PMB, taking care not to generate PMB misses and multiple PMB hits. The procedure should be set up within the boot routine and should be executed before activation of the caches and TLB (CCR.ICE = 1, CCR.OCE = 1, and MMUCR.AT = 1). Do not use routines other than the boot routine to change the value recorded in the PMB.

(a) When the Program Modifying the PMB is in the P1 or P2 Area

1. Read the initial entry, change only the SZ bits to reduce the page size, and save the new value over the previous entry. The program that changes the PMB should be allocated within 1 Mbyte of the top of the page with the reduced size.
2. Invalidate the entry remaining in the ITLB that corresponds to the PMB by writing 1 to the TI bit in the MMUCR register.
3. In the memory-mapped PMB, record PMB entries to fill the P1 or P2 area in which the PMB translation information is evicted by step 1.
4. Execute one of the following steps, A, B, and C. Do not execute a branch or operand access for the P1 or P2 area in which the PMB translation information is evicted by step 1.
   A. Perform a branch using the RTE instruction.
   B. Execute the ICBI instruction for any address (including non-cacheable area).
   C. If the MT bit in IRMCR is set to 0 (initial value) before accessing the memory-mapped PMB, no specific sequence is required.

However, correct operation with method C may no longer be guaranteed in future SuperH-family products. Selection of step A or B is recommended to ensure compatibility with future SuperH-family products.

(b) When the Program Modifying the PMB is in Areas Other than the P1 or P2 Area

1. Invalidate the entry remaining in the ITLB by writing 1 to the TI bit in MMUCR.
2. In the memory-mapped PMB, change PMB entries.
3. Execute one of the following steps, A, B, and C. Do not execute a branch or operand access for the P1 or P2 area before this execution.
   A. Perform a branch using the RTE instruction.
   B. Execute the ICBI instruction for any address (including non-cacheable area).
   C. If the MT bit in IRMCR is set to 0 (initial value) before accessing the memory-mapped PMB, no specific sequence is required.

   However, correct operation with method C may no longer be guaranteed in future SuperH-family products. Selection of step A or B is recommended to ensure compatibility with future SuperH-family products.

2.7.10 Usage Notes

(1) Note on Using LDTLB Instruction

When using an LDTLB instruction instead of software to a value to the MMUCR. URC, execute 1 or 2 below.

1. In 29-bit address mode, follow A. and D. below. In 32-bit address mode, follow A. through D. below.
   A. Place the TLB miss exception handling routine*[1] only in the P1, P2 area ,or the on-chip memory so that all the instruction accesses*[3] in the TLB miss exception handling routine should occur solely in the P1, P2 area, or the on-chip memory. Clear the RP bit in the RAMCR register to 0 (initial value), when the TLB miss exception handling routine is placed in the on-chip memory.
   B. In 32-bit address mode, use only one page of the PMB for instruction accesses*[3] in the TLB miss exception handling routine*[1]. Do not place them in the last 64 bytes of a page of the PMB.
   C. In 32-bit address mode, obey 1 and 2 below when recording information in the UTLB in the MMU-related exception*[2] handling routine.
      a. When the TLB miss exception occurs, and recording the information of a page with the access right in the UTLB, do not record the page, in which the exception has occurred, in the UTLB using the following two operations.
         - Specifies the protection key data that causes a protection violation exception upon re-execution of the instruction that has caused the TLB miss exception and records the page, in which the TLB miss exception has occurred, in the UTLB.
         - Specifies the protection key data that does not cause a protection violation exception in the protection violation exception handling routine to record the page in the UTLB and re-executes the instruction that has caused the protection violation exception.
      b. When an initial page write exception occurs and the TLB entry in the UTLB of which the dirty bit is 1 is replaced, before the write instruction for the page corresponding to this replaced TLB entry is completed, register the TLB entry of which the dirty bit is 1.
   D. Do not make an attempt to execute the FDIV or FSQRT instruction in the TLB miss exception handling routine.
2. If a TLB miss exception occurs, add 1 to MMUCR.URC before executing an LDTLB instruction.

Notes: 1. An exception handling routine is an entire set of instructions that are executed from the address (VBR + offset) upon occurrence of an exception to the RTE for returning to the original program or to the RTE delay slot.

2. MMU-related exceptions are: instruction TLB miss exception, instruction TLB miss protection violation exception, data TLB miss exception, data TLB protection violation exception, and initial page write exception.
3. Instruction accesses include the PREFI and ICBI instructions.

2.8 Caches

This LSI has an on-chip 32-Kbyte instruction cache (IC) for instructions and an on-chip 32-Kbyte operand cache (OC) for data.

2.8.1 Features

The features of the cache are given in table 2.35.

The SH-4A supports two 32-byte store queues (SQs) to perform high-speed writes to external memory. The features of the store queues are given in table 2.36.

Table 2.35 Cache Features

Item	Instruction Cache	Operand Cache
Capacity	32-Kbyte cache	32-Kbyte cache
Type	4-way set-associative, virtual address index/physical address tag	4-way set-associative, virtual address index/physical address tag
Line size	32 bytes	32 bytes
Entries	256 entries/way	256 entries/way
Write method	—	Copy-back/write-through selectable
Replacement method	LRU (least-recently-used) algorithm	LRU (least-recently-used) algorithm

Table 2.36 Store Queue Features

Item	Store Queues
Capacity	32 bytes × 2
Addresses	H'E000 0000 to H'E3FF FFFF
Write	Store instruction (1-cycle write)
Write-back	Prefetch instruction (PREF instruction)
Access right	When MMU is disabled: Determined by SQMD bit in MMUCR When MMU is enabled: Determined by PR for each page

The operand cache of the SH-4A is 4-way set associative, each may comprising 256 cache lines. Figure 2.56 shows the configuration of the operand cache.

The instruction cache is 4-way set-associative, each way comprising 256 cache lines. Figure 2.57 shows the configuration of the instruction cache.

The SH-4A has an IC way prediction scheme to reduce power consumption. In addition, memory-mapped associative writing, which is detectable as an exception, can be enabled by using the non-support detection exception register (EXPMASK). For details, see section 2.5, Exception Handling.

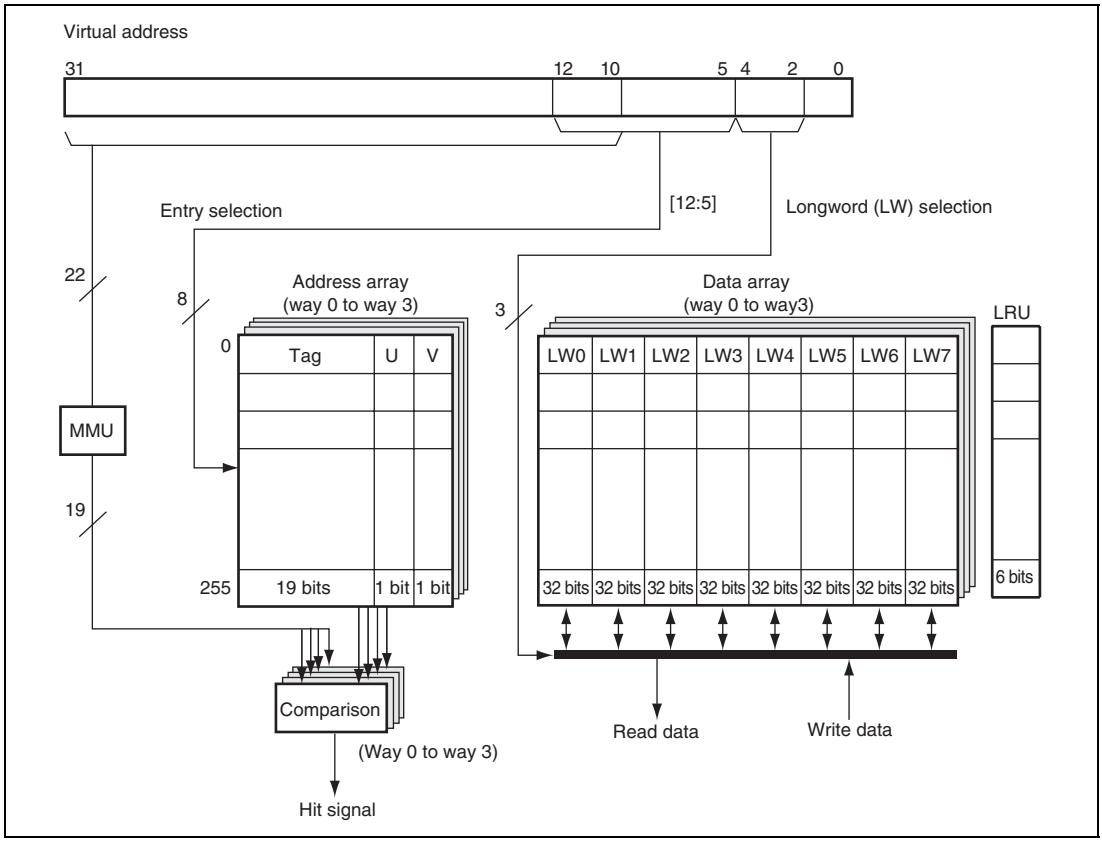


Figure 2.56 Configuration of Operand Cache (Cache size = 32 Kbytes)

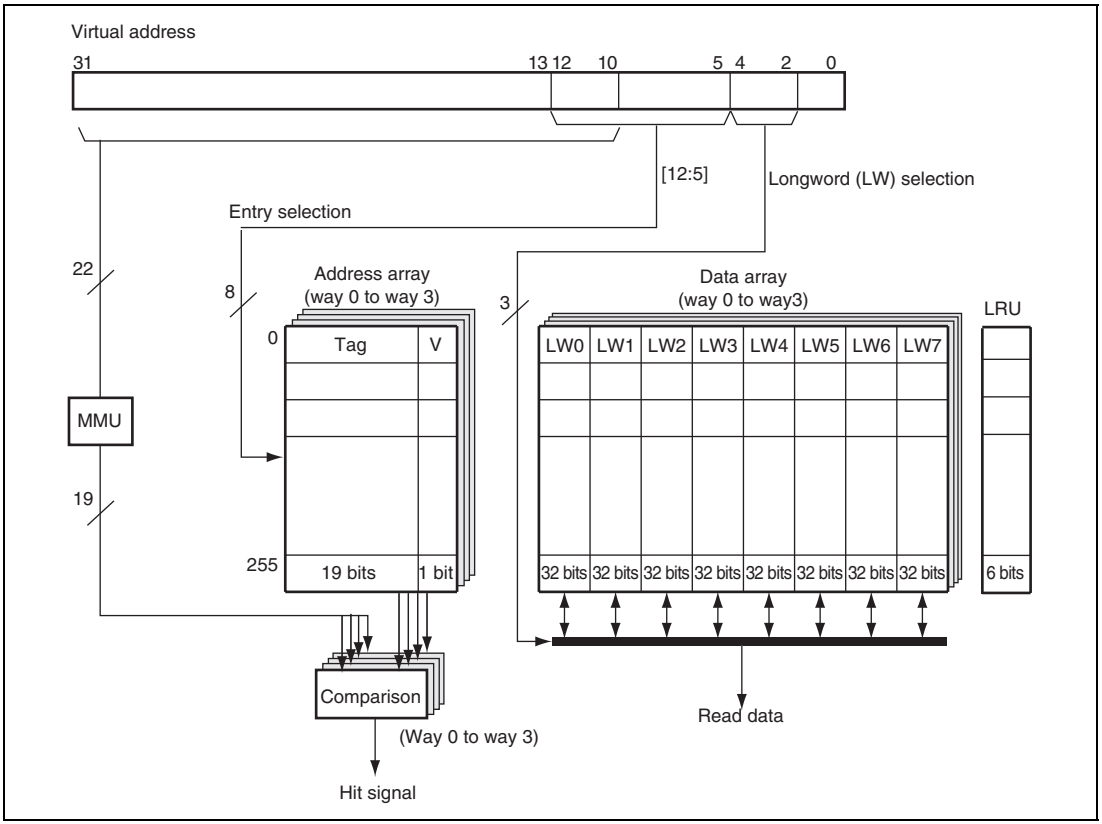


Figure 2.57 Configuration of Instruction Cache (Cache size = 32 Kbytes)

- Tag

  Stores the upper 19 bits of the 29-bit physical address of the data line to be cached. The tag is not initialized by a power-on or manual reset.

- V bit (validity bit)

  Indicates that valid data is stored in the cache line. When this bit is 1, the cache line data is valid. The V bit is initialized to 0 by a power-on reset, but retains its value in a manual reset.

- U bit (dirty bit)

  The U bit is set to 1 if data is written to the cache line while the cache is being used in copy-back mode. That is, the U bit indicates a mismatch between the data in the cache line and the data in external memory. The U bit is never set to 1 while the cache is being used in write-through mode, unless it is modified by accessing the memory-mapped cache (see section 2.8.6, Memory-Mapped Cache Configuration). The U bit is initialized to 0 by a power-on reset, but retains its value in a manual reset.

- Data array

  The data field holds 32 bytes (256 bits) of data per cache line. The data array is not initialized by a power-on or manual reset.

- LRU

  In a 4-way set-associative method, up to 4 items of data can be registered in the cache at each entry address. When an entry is registered, the LRU bit indicates which of the 4 ways it is to be registered in. The LRU mechanism uses 6 bits of each entry, and its usage is controlled by hardware. The LRU (least-recently-used) algorithm is used for way selection, and selects the less recently accessed way. The LRU bits are initialized to 0 by a power-on reset but not by a manual reset. The LRU bits cannot be read from or written to by software.

2.8.2 Register Descriptions

The following registers are related to cache.

Table 2.37 Register Configuration

Register Name	Abbreviation	R/W	P4 Address*	Area 7 Address*	Size
Cache control register	CCR	R/W	H'FF00 001C	H'1F00 001C	32
Queue address control register 0	QACR0	R/W	H'FF00 0038	H'1F00 0038	32
Queue address control register 1	QACR1	R/W	H'FF00 003C	H'1F00 003C	32
On-chip memory control register	RAMCR	R/W	H'FF00 0074	H'1F00 0074	32

Note: * These P4 addresses are for the P4 area in the virtual address space. These area 7 addresses are accessed from area 7 in the physical address space by means of the TLB.

Table 2.38 Register States in Each Processing State

Register Name	Abbreviation	Power-on Reset	Manual Reset	Sleep	Software Standby	Deep Standby
Cache control register	CCR	H'0000 0000	H'0000 0000	Retained	Retained	H'0000 0000
Queue address control register 0	QACR0	Undefined	Undefined	Retained	Retained	Undefined
Queue address control register 1	QACR1	Undefined	Undefined	Retained	Retained	Undefined
On-chip memory control register	RAMCR	H'0000 0000	H'0000 0000	Retained	Retained	H'0000 0000

(1) Cache Control Register (CCR)

CCR controls the cache operating mode, the cache write mode, and invalidation of all cache entries.

CCR modifications must only be made by a program in the non-cacheable P2 area or IL memory. After CCR has been updated, execute one of the following three methods before an access (including an instruction fetch) to the cacheable area is performed.

1. Execute a branch using the RTE instruction. In this case, the branch destination may be the cacheable area.
2. Execute the ICBI instruction for any address (including non-cacheable area).
3. If the R2 bit in IRMCR is 0 (initial value) before updating CCR, the specific instruction does not need to be executed. However, note that the CPU processing performance will be lowered because the instruction fetch is performed again for the next instruction after CCR has been updated.

Note that the method 3 may not be guaranteed in the future SuperH Series. Therefore, it is recommended that the method 1 or 2 should be used for being compatible with the future SuperH Series.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	ICI	—	—	ICE	—	—	—	—	OCI	CB	WT	OCE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R	R	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved For details on reading from or writing to these bits, see description in General Precautions on Handling of Product.
11	ICI	0	R/W	IC Invalidation Bit When 1 is written to this bit, the V bits of all IC entries are cleared to 0. This bit is always read as 0.

Bit	Bit Name	Initial Value	R/W	Description
10, 9	—	All 0	R	Reserved For details on reading from or writing to these bits, see description in General Precautions on Handling of Product.
8	ICE	0	R/W	IC Enable Bit Selects whether the IC is used. Note however when address translation is performed, the IC cannot be used unless the C bit in the page management information is also 1. 0: IC not used 1: IC used
7 to 4	—	All 0	R	Reserved For details on reading from or writing to these bits, see description in General Precautions on Handling of Product.
3	OCI	0	R/W	OC Invalidation Bit When 1 is written to this bit, the V and U bits of all OC entries are cleared to 0. This bit is always read as 0.
2	CB	0	R/W	Copy-Back Bit Indicates the P1 area cache write mode. 0: Write-through mode 1: Copy-back mode
1	WT	0	R/W	Write-Through Mode Indicates the P0, U0, and P3 area cache write mode. When address translation is performed, the value of the WT bit in the page management information has priority. 0: Copy-back mode 1: Write-through mode
0	OCE	0	R/W	OC Enable Bit Selects whether the OC is used. Note however when address translation is performed, the OC cannot be used unless the C bit in the page management information is also 1. 0: OC not used 1: OC used

(2) Queue Address Control Register 0 (QACR0)

QACR0 specifies the area onto which store queue 0 (SQ0) is mapped when the MMU is disabled.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	AREA0			—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	—	—	—	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved For details on reading from or writing to these bits, see description in General Precautions on Handling of Product.
4 to 2	AREA0	Undefined	R/W	When the MMU is disabled, these bits generate physical address bits [28:26] for SQ0.
1, 0	—	All 0	R	Reserved For details on reading from or writing to these bits, see description in General Precautions on Handling of Product.

(3) Queue Address Control Register 1 (QACR1)

QACR1 specifies the area onto which store queue 1 (SQ1) is mapped when the MMU is disabled.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	AREA1			—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	—	—	—	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved For details on reading from or writing to these bits, see description in General Precautions on Handling of Product.
4 to 2	AREA1	Undefined	R/W	When the MMU is disabled, these bits generate physical address bits [28:26] for SQ1.
1, 0	—	All 0	R	Reserved For details on reading from or writing to these bits, see description in General Precautions on Handling of Product.

(4) On-Chip Memory Control Register (RAMCR)

RAMCR controls the number of ways in the IC and OC and prediction of the IC way.

RAMCR modifications must only be made by a program in the non-cacheable P2 area. After RAMCR has been updated, execute one of the following three methods before an access (including an instruction fetch) to the cacheable area or the IL memory area is performed.

1. Execute a branch using the RTE instruction. In this case, the branch destination may be the cacheable area or the IL memory area.
2. Execute the ICBI instruction for any address (including non-cacheable area).
3. If the R2 bit in IRMCR is 0 (initial value) before updating RAMCR, the specific instruction does not need to be executed. However, note that the CPU processing performance will be lowered because the instruction fetch is performed again for the next instruction after RAMCR has been updated.

Note that the method 3 may not be guaranteed in the future SuperH Series. Therefore, it is recommended that the method 1 or 2 should be used for being compatible with the future SuperH Series.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RMD	RP	IC2W	OC2W	ICWPW	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved For details on reading from or writing to these bits, see description in General Precautions on Handling of Product.
9	RMD	0	R/W	On-Chip Memory Access Mode Bit For details, see section 2.9.4, On-Chip Memory Protective Functions.

Bit	Bit Name	Initial Value	R/W	Description
8	RP	0	R/W	On-Chip Memory Protection Enable Bit For details, see section 2.9.4, On-Chip Memory Protective Functions.
7	IC2W	0	R/W	IC Two-Way Mode bit 0: IC is a four-way operation 1: IC is a two-way operation For details, see section 2.8.4 (3), IC Two-Way Mode.
6	OC2W	0	R/W	OC Two-Way Mode bit 0: OC is a four-way operation 1: OC is a two-way operation For details, see section 2.8.3 (6), OC Two-Way Mode.
5	ICWPD	0	R/W	IC Way Prediction Stop Selects whether the IC way prediction is used. 0: Instruction cache performs way prediction. 1: Instruction cache does not perform way prediction.
4 to 0	—	All 0	R	Reserved For details on reading from or writing to these bits, see description in General Precautions on Handling of Product.

2.8.3 Operand Cache Operation

(1) Read Operation

When the Operand Cache (OC) is enabled (OCE = 1 in CCR) and data is read from a cacheable area, the cache operates as follows:

1. The tag, V bit, U bit, and LRU bits on each way are read from the cache line indexed by virtual address bits [12:5].
2. The tags read from the each way is compared with bits [28:10] of the physical address resulting from virtual address translation by the MMU:

- If there is a way whose tag matches and its V bit is 1, see No. 3.
- If there is no way whose tag matches and its V bit is 1 and the U bit of the way which is selected to replace using the LRU bits is 0, see No. 4.
- If there is no way whose tag matches and its V bit is 1 and the U bit of the way which is selected to replace using the LRU bits is 1, see No. 5.

3. Cache hit

   The data indexed by virtual address bits [4:0] is read from the data field of the cache line on the hit way in accordance with the access size. Then the LRU bits are updated to indicate the hit way is the latest one.

4. Cache miss (no write-back)

   Data is read into the cache line on the way, which is selected to replace, from the physical address space corresponding to the virtual address. Data reading is performed, using the wraparound method, in order from the quad-word data(8 bytes) including the cache-missed data. When the corresponding data arrives in the cache, the read data is returned to the CPU. While the remaining data on the cache line is being read, the CPU can execute the next processing. When reading of one line of data is completed, the tag corresponding to the physical address is recorded in the cache, 1 is written to the V bit and 0 is written to the U bit on the way. Then the LRU bit is updated to indicate the way is latest one.

5. Cache miss (with write-back)

   The tag and data field of the cache line on the way which is selected to replace are saved in the write-back buffer. Then data is read into the cache line on the way which is selected to replace from the physical address space corresponding to the virtual address. Data reading is performed, using the wraparound method, in order from the quad-word data (8 bytes) including the cache-missed data, and when the corresponding data arrives in the cache, the read data is returned to the CPU. While the remaining one cache line of data is being read, the CPU can execute the next processing. When reading of one line of data is completed, the tag corresponding to the physical address is recorded in the cache, 1 is written to the V bit, and 0 to the U bit. And the LRU bits are updated to indicate the way is latest one. The data in the

write-back buffer is then written back to external memory.

(2) Prefetch Operation

When the Operand Cache (OC) is enabled (OCE = 1 in CCR) and data is prefetched from a cacheable area, the cache operates as follows:

1. The tag, V bit, U bit, and LRU bits on each way are read from the cache line indexed by virtual address bits [12:5].
2. The tag, read from each way, is compared with bits [28:10] of the physical address resulting from virtual address translation by the MMU:

- If there is a way whose tag matches and its V bit is 1, see No. 3.
- If there is no way whose tag matches and the V bit is 1, and the U bit of the way which is selected to replace using the LRU bits is 0, see No. 4.
- If there is no way whose tag matches and the V bit is 1, and the U bit of the way which is selected to replace using the LRU bits is 1, see No. 5.

3. Cache hit

   Then the LRU bits are updated to indicate the hit way is the latest one.

4. Cache miss (no write-back)

   Data is read into the cache line on the way, which is selected to replace, from the physical address space corresponding to the virtual address. Data reading is performed, using the wraparound method, in order from the quad-word data (8 bytes) including the cache-missed data. In the prefetch operation the CPU doesn't wait the data arrives. While the one cache line of data is being read, the CPU can execute the next processing. When reading of one line of data is completed, the tag corresponding to the physical address is recorded in the cache, 1 is written to the V bit and 0 is written to the U bit on the way. And the LRU bit is updated to indicate the way is latest one.

5. Cache miss (with write-back)

   The tag and data field of the cache line on the way which is selected to replace are saved in the write-back buffer. Then data is read into the cache line on the way which is selected to replace from the physical address space corresponding to the virtual address. Data reading is performed, using the wraparound method, in order from the quad-word data (8 bytes) including the cache-missed data. In the prefetch operation the CPU doesn't wait the data arrives. While the one cache line of data is being read, the CPU can execute the next processing. And the LRU bits are updated to indicate the way is latest one. The data in the write-back buffer is then written back to external memory.

(3) Write Operation

When the Operand Cache (OC) is enabled (OCE = 1 in CCR) and data is written to a cacheable area, the cache operates as follows:

1. The tag, V bit, U bit, and LRU bits on each way are read from the cache line indexed by virtual address bits [12:5].
2. The tag, read from each way, is compared with bits [28:10] of the physical address resulting from virtual address translation by the MMU:

- If there is a way whose tag matches and its V bit is 1, see No. 3 for copy-back and No. 4 for write-through.
- I If there is no way whose tag matches and its V bit is 1 and the U bit of the way which is selected to replace using the LRU bits is 0, see No. 5 for copy-back and No. 7 for write-through.
- If there is no way whose tag matches and its V bit is 1 and the U bit of the way which is selected to replace using the LRU bits is 1, see No. 6 for copy-back and No. 7 for write-through.

3. Cache hit (copy-back)

   A data write in accordance with the access size is performed for the data field on the hit way which is indexed by virtual address bits [4:0]. Then 1 is written to the U bit. The LRU bits are updated to indicate the way is the latest one.

4. Cache hit (write-through)

   A data write in accordance with the access size is performed for the data field on the hit way which is indexed by virtual address bits [4:0]. A write is also performed to external memory corresponding to the virtual address. Then the LRU bits are updated to indicate the way is the latest one. In this case, the U bit isn't updated.

5. Cache miss (copy-back, no write-back)

   A data write in accordance with the access size is performed for the data field on the hit way which is indexed by virtual address bits [4:0]. Then, the data, excluding the cache-missed data which is written already, is read into the cache line on the way which is selected to replace from the physical address space corresponding to the virtual address.

   Data reading is performed, using the wraparound method, in order from the quad-word data (8 bytes) including the cache-missed data. While the remaining data on the cache line is being read, the CPU can execute the next processing. When reading of one line of data is completed, the tag corresponding to the physical address is recorded in the cache, 1 is written to the V bit and the U bit on the way. Then the LRU bit is updated to indicate the way is latest one.

6. Cache miss (copy-back, with write-back)

   The tag and data field of the cache line on the way which is selected to replace are saved in the write-back buffer. Then a data write in accordance with the access size is performed for the data field on the hit way which is indexed by virtual address bits [4:0]. Then, the data, excluding the cache-missed data which is written already, is read into the cache line on the way which is selected to replace from the physical address space corresponding to the virtual address. Data reading is performed, using the wraparound method, in order from the quad-word data (8 bytes) including the cache-missed data. While the remaining data on the cache line is being read, the CPU can execute the next processing. When reading of one line of data is completed, the tag corresponding to the physical address is recorded in the cache, 1 is written to the V bit and the U bit on the way. Then the LRU bit is updated to indicate the way is latest one. Then the data in the write-back buffer is then written back to external memory.
7. Cache miss (write-through)

   A write of the specified access size is performed to the external memory corresponding to the virtual address. In this case, a write to cache is not performed.

(4) Write-Back Buffer

In order to give priority to data reads to the cache and improve performance, the SH-4A has a write-back buffer which holds the relevant cache entry when it becomes necessary to purge a dirty cache entry into external memory as the result of a cache miss. The write-back buffer contains one cache line of data and the physical address of the purge destination.

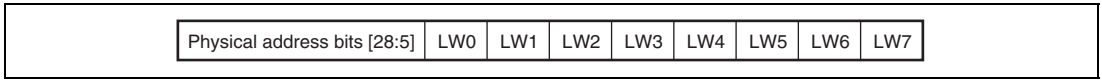


Figure 2.58 Configuration of Write-Back Buffer

(5) Write-Through Buffer

The SH-4A has a 64-bit buffer for holding write data when writing data in write-through mode or writing to a non-cacheable area. This allows the CPU to proceed to the next operation as soon as the write to the write-through buffer is completed, without waiting for completion of the write to external memory.

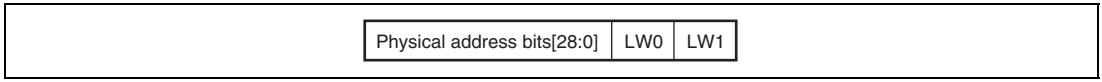


Figure 2.59 Configuration of Write-Through Buffer

(6) OC Two-Way Mode

When the OC2W bit in RAMCR is set to 1, OC two-way mode which only uses way 0 and way 1 in the OC is entered. Thus, power consumption can be reduced. In this mode, only way 0 and way 1 are used even if a memory-mapped OC access is made.

The OC2W bit should be modified by a program in the P2 area. At that time, if the valid line has already been recorded in the OC, data should be written back by software, if necessary, 1 should be written to the OCI bit in CCR, and all entries in the OC should be invalid before modifying the OC2W bit.

2.8.4 Instruction Cache Operation

(1) Read Operation

When the IC is enabled (ICE = 1 in CCR) and instruction fetches are performed from a cacheable area, the instruction cache operates as follows:

1. The tag, V bit, U bit and LRU bits on each way are read from the cache line indexed by virtual address bits [12:5].
2. The tag, read from each way, is compared with bits [28:10] of the physical address resulting from virtual address translation by the MMU:

- If there is a way whose tag matches and the V bit is 1, see No. 3.
- If there is no way whose tag matches and the V bit is 1, see No. 4.

3. Cache hit

   The data indexed by virtual address bits [4:2] is read as an instruction from the data field on the hit way. The LRU bits are updated to indicate the way is the latest one.

4. Cache miss

   Data is read into the cache line on the way which selected using LRU bits to replace from the physical address space corresponding to the virtual address. Data reading is performed, using the wraparound method, in order from the quad-word data (8 bytes) including the cache-missed data, and when the corresponding data arrives in the cache, the read data is returned to the CPU as an instruction. While the remaining one cache line of data is being read, the CPU can execute the next processing. When reading of one line of data is completed, the tag corresponding to the physical address is recorded in the cache, and 1 is written to the V bit, the LRU bits are updated to indicate the way is the latest one.

(2) Prefetch Operation

When the IC is enabled (ICE = 1 in CCR) and instruction prefetches are performed from a cacheable area, the instruction cache operates as follows:

1. The tag, V bit, Ubit and LRU bits on each way are read from the cache line indexed by virtual address bits [12:5].
2. The tag, read from each way, is compared with bits [28:10] of the physical address resulting from virtual address translation by the MMU:

- If there is a way whose tag matches and the V bit is 1, see No. 3.
- If there is no way whose tag matches and the V bit is 1, see No. 4.

3. Cache hit

   The LRU bits is updated to indicate the way is the latest one.

4. Cache miss

   Data is read into the cache line on a way which selected using the LRU bits to replace from the physical address space corresponding to the virtual address. Data reading is performed, using the wraparound method, in order from the quad-word data (8 bytes) including the cache-missed data. In the prefetch operation, the CPU doesn't wait the data arrived. While the one cache line of data is being read, the CPU can execute the next processing. When reading of one line of data is completed, the tag corresponding to the physical address is recorded in the cache, and 1 is written to the V bit, the LRU bits is updated to indicate the way is the latest one.

(3) IC Two-Way Mode

When the IC2W bit in RAMCR is set to 1, IC two-way mode which only uses way 0 and way 1 in the IC is entered. Thus, power consumption can be reduced. In this mode, only way 0 and way 1 are used even if a memory-mapped IC access is made.

The IC2W bit should be modified by a program in the P2 area. At that time, if the valid line has already been recorded in the IC, 1 should be written to the ICI bit in CCR and all entries in the IC should be invalid before modifying the IC2W bit.

(4) Instruction Cache Way Prediction Operation

The SH-4A incorporates an instruction cache (IC) way prediction scheme to reduce power consumption. This is achieved by activating only the data array that corresponds to a predicted way. When way prediction misses occur, data must be re-read from the right way, which may lead to lower performance in instruction fetching. Setting the ICWPD bit to 1 disables the IC way prediction scheme. Since way prediction misses do not occur in this mode, there is no loss of performance in instruction fetching but the IC consumes more power. The ICWPD bit should be modified by a program in the non-cacheable P2 area. If a valid line has already been recorded in the IC at this time, invalidate all entries in the IC by writing 1 to the ICI bit in CCR before modifying the ICWPD bit.

2.8.5 Cache Operation Instruction

(1) Coherency between Cache and External Memory

(a) Cache Operation Instruction

Coherency between cache and external memory should be assured by software. In the SH-4A, the following six instructions are supported for cache operations. Details of these instructions are given in section 11, Instruction Descriptions of the SH-4A Extended Functions Software Manual.

- Operand cache invalidate instruction: OCBI @Rn
  Operand cache invalidation (no write-back)
- Operand cache purge instruction: OCBP @Rn
  Operand cache invalidation (with write-back)
- Operand cache write-back instruction: OCBWB @Rn
  Operand cache write-back
- Operand cache allocate instruction: MOVCA.L R0,@Rn
  Operand cache allocation
- Instruction cache invalidate instruction: ICBI @Rn
  Instruction cache invalidation
- Operand access synchronization instruction: SYNCO
  Wait for data transfer completion

(b) Coherency Control

The operand cache can receive "PURGE" and "FLUSH" transaction from SuperHyway bus to control the cache coherency. Since the address used by the PURGE and FLUSH transaction is a physical address, do not use the 1 Kbyte page size to avoid cache synonym problem in MMU enable mode.

- PURGE transaction

  When the operand cache is enabled, the PURGE transaction checks the operand cache and invalidates the hit entry. If the invalidated entry is dirty, the data is written back to the external memory. If the transaction is not hit to the cache, it is no-operation.
- FLUSH transaction

  When the operand cache is enabled, the FLUSH transaction checks the operand cache and if the hit line is dirty, then the data is written back to the external memory. If the transaction is not hit to the cache or the hit entry is not dirty, it is no-operation.

(c) Changes in Instruction Specifications Regarding Coherency Control

Of the operand cache operating instructions, the coherency control-related specifications of OCBI, OCBP, and OCBWB have been changed from those of the SH-4A with H'20-valued VER bits in the processor version register (PVR).

- Changes in the invalidate instruction OCBI@Rn

  When Rn is designating an address in a non-cacheable area, this instruction is executed as NOP in the SH-4A with H'20-valued VER bits in the processor version register (PVR). In the SH-4A with extended functions, this instruction invalidates the operand cache line designated by way = Rn[14:13] and entry = Rn[12:5] provided that Rn[31:24] = H'F4 (OC address array area). In this process, writing back of the line does not take place even if the line to be invalidated is dirty. This operation is only executable in privileged mode, and an address error exception occurs in user mode. TLB-related exceptions do not occur.

  Do not execute this instruction to invalidate the memory-mapped array areas and control register areas for which Rn[31:24] is not H'F4, and their reserved areas (H'F0 to H'F3, H'F5 to H'FF).

- Changes in the purge instruction OCBP@Rn

  When Rn is designating an address in a non-cacheable area, this instruction is executed as NOP in the SH-4A with H'20-valued VER bits in the processor version register (PVR). In the SH-4A with extended functions, this instruction invalidates the operand cache line designated by way = Rn[14:13] and entry = Rn[12:5] provided that Rn[31:24] = H'F4 (OC address array area). In this process, writing back of the line takes place when the line to be invalidated is dirty. This operation is only executable in privileged mode, and an address error exception occurs in user mode. TLB-related exceptions do not occur.

  Do not execute this instruction to invalidate the memory-mapped array areas and control register areas for which Rn[31:24] is not H'F4, and their reserved areas (H'F0 to H'F3, H'F5 to H'FF).

- Changes in the write-back instruction OCBWB@Rn

  When Rn is designating an address in a non-cacheable area, this instruction is executed as NOP in the SH-4A with H'20-valued VER bits in the processor version register (PVR). In the SH-4A with extended functions, provided that Rn[31:24] = H'F4 (OC address array area), this instruction writes back the operand cache line designated by way = Rn[14:13] and entry = Rn[12:5] if it is dirty and clears the dirty bit to 0. This operation is only executable in privileged mode, and an address error exception occurs in user mode. TLB-related exceptions do not occur.

  Do not execute this instruction to invalidate the memory-mapped array areas and control register areas for which Rn[31:24] is not H'F4, and their reserved areas (H'F0 to H'F3, H'F5 to H'FF).

(2) Prefetch Operation

The SH-4A supports a prefetch instruction to reduce the cache fill penalty incurred as the result of a cache miss. If it is known that a cache miss will result from a read or write operation, it is possible to fill the cache with data beforehand by means of the prefetch instruction to prevent a cache miss due to the read or write operation, and so improve software performance. If a prefetch instruction is executed for data already held in the cache, or if the prefetch address results in a UTLB miss or a protection violation, the result is no operation, and an exception is not generated. Details of the prefetch instruction are given in section 11, Instruction Descriptions of the SH-4A Extended Functions Software Manual.

- Prefetch instruction (OC) : PREF @Rn
- Prefetch instruction (IC) : PREFI @Rn

2.8.6 Memory-Mapped Cache Configuration

The IC and OC can be managed by software. The contents of IC data array can be read from or written to by a program in the P2 area by means of a MOV instruction in privileged mode. The contents of IC address array can also be read from or written to in privileged mode by a program in the P2 area or the IL memory area by means of a MOV instruction. Operation is not guaranteed if access is made from a program in another area. In this case, execute one of the following three methods for executing a branch to the P0, U0, P1, or P3 area.

1. Execute a branch using the RTE instruction.
2. Execute a branch to the P0, U0, P1, or P3 area after executing the ICBI instruction for any address (including non-cacheable area).
3. If the MC bit in IRMCR is 0 (initial value) before making an access to the memory-mapped IC, the specific instruction does not need to be executed. However, note that the CPU processing performance will be lowered because the instruction fetch is performed again for the next instruction after making an access to the memory-mapped IC.

Note that the method 3 may not be guaranteed in the future SuperH Series. Therefore, it is recommended that the method 1 or 2 should be used for being compatible with the future SuperH Series.

In privileged mode, the OC contents can be read from or written to by a program in the P1 or P2 area by means of a MOV instruction. Operation is not guaranteed if access is made from a program in another area. The IC and OC are allocated to the P4 area in the virtual address space. Only data accesses can be used on both the IC address array and data array and the OC address array and data array, and accesses are always longword-size. Instruction fetches cannot be performed in these areas. For reserved bits, a write value of 0 should be specified and the read value is undefined.

(1) IC Address Array

The IC address array is allocated to addresses H'F000 0000 to H'F0FF FFFF in the P4 area. An address array access requires a 32-bit address field specification (when reading or writing) and a 32-bit data field specification. The way and entry to be accessed are specified in the address field, and the write tag and V bit are specified in the data field.

In the address field, bits [31:24] have the value H'F0 indicating the IC address array, and the way is specified by bits [14:13] and the entry by bits [12:5]. The association bit (A bit) [3] in the address field specifies whether or not association is performed when writing to the IC address array. As only longword access is used, 0 should be specified for address field bits [1:0].

In the data field, the tag is indicated by bits [31:10], and the V bit by bit [0]. As the IC address array tag is 19 bits in length, data field bits [31:29] are not used in the case of a write in which association is not performed. Data field bits [31:29] are used for the virtual address specification only in the case of a write in which association is performed.

The following three kinds of operation can be used on the IC address array:

1. IC address array read

   The tag and V bit are read into the data field from the IC entry corresponding to the way and entry set in the address field. In a read, associative operation is not performed regardless of whether the association bit specified in the address field is 1 or 0.
2. IC address array write (non-associative)

   The tag and V bit specified in the data field are written to the IC entry corresponding to the way and entry set in the address field. The A bit in the address field should be cleared to 0.
3. IC address array write (associative)

   When a write is performed with the A bit in the address field set to 1, the tag in each way stored in the entry specified in the address field is compared with the tag specified in the data field. The way numbers of bits [14:13] in the address field are not used. If the MMU is enabled at this time, comparison is performed after the virtual address specified by data field bits [31:10] has been translated to a physical address using the ITLB. If the addresses match and the V bit in the way is 1, the V bit specified in the data field is written into the IC entry. In other cases, no operation is performed. This operation is used to invalidate a specific IC entry. If an ITLB miss occurs during address translation, or the comparison shows a mismatch, an exception is not generated, no operation is performed, and the write is not executed.

Note: IC address array associative writing function may not be supported in the future SuperH Series. Therefore, it is recommended that the ICBI instruction should be used to operate the IC definitely by handling ITLB miss and reporting ITLB miss exception.

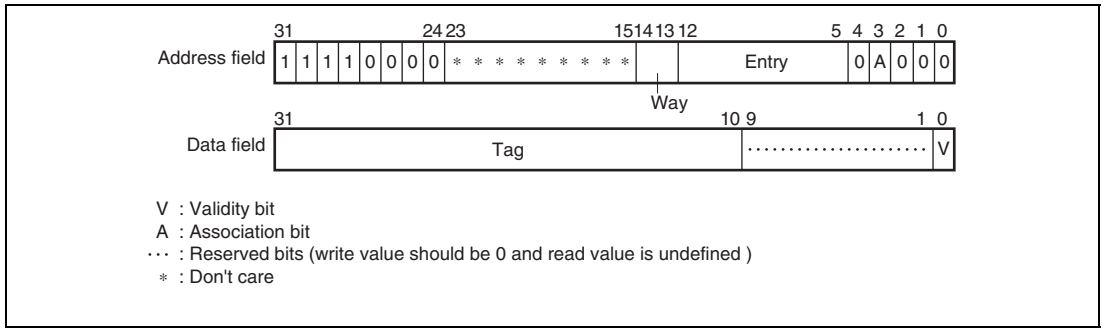


Figure 2.60 Memory-Mapped IC Address Array (Cache size = 32 Kbytes)

(2) IC Data Array

The IC data array is allocated to addresses H'F100 0000 to H'F1FF FFFF in the P4 area. A data array access requires a 32-bit address field specification (when reading or writing) and a 32-bit data field specification. The way and entry to be accessed are specified in the address field, and the longword data to be written is specified in the data field.

In the address field, bits [31:24] have the value H'F1 indicating the IC data array, and the way is specified by bits [14:13] and the entry by bits [12:5]. Address field bits [4:2] are used for the longword data specification in the entry. As only longword access is used, 0 should be specified for address field bits [1:0].

The data field is used for the longword data specification.

The following two kinds of operation can be used on the IC data array:

1. IC data array read

   Longword data is read into the data field from the data specified by the longword specification bits in the address field in the IC entry corresponding to the way and entry set in the address field.
2. IC data array write

   The longword data specified in the data field is written for the data specified by the longword specification bits in the address field in the IC entry corresponding to the way and entry set in the address field.

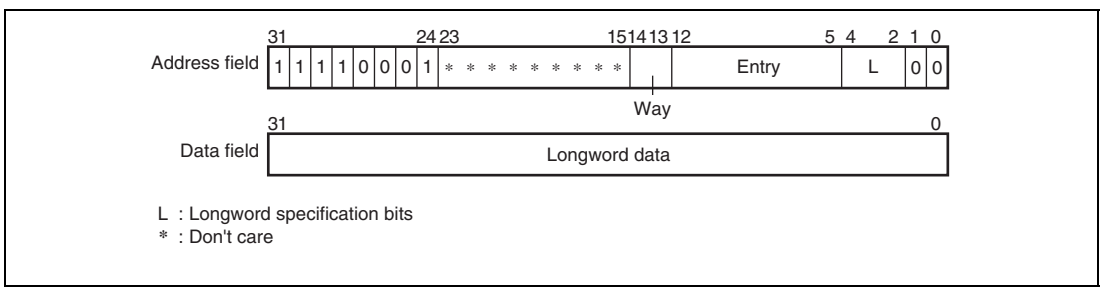


Figure 2.61 Memory-Mapped IC Data Array (Cache size = 32 Kbytes)

(3) OC Address Array

The OC address array is allocated to addresses H'F400 0000 to H'F4FF FFFF in the P4 area. An address array access requires a 32-bit address field specification (when reading or writing) and a 32-bit data field specification. The way and entry to be accessed are specified in the address field, and the write tag, U bit, and V bit are specified in the data field.

In the address field, bits [31:24] have the value H'F4 indicating the OC address array, and the way is specified by bits [14:13] and the entry by bits [12:5]. The association bit (A bit) [3] in the address field specifies whether or not association is performed when writing to the OC address array. As only longword access is used, 0 should be specified for address field bits [1:0].

In the data field, the tag is indicated by bits [31:10], the U bit by bit [1], and the V bit by bit [0]. As the OC address array tag is 19 bits in length, data field bits [31:29] are not used in the case of a write in which association is not performed. Data field bits [31:29] are used for the virtual address specification only in the case of a write in which association is performed.

The following three kinds of operation can be used on the OC address array:

1. OC address array read

   The tag, U bit, and V bit are read into the data field from the OC entry corresponding to the way and entry set in the address field. In a read, associative operation is not performed regardless of whether the association bit specified in the address field is 1 or 0.
2. OC address array write (non-associative)

   The tag, U bit, and V bit specified in the data field are written to the OC entry corresponding to the way and entry set in the address field. The A bit in the address field should be cleared to 0.

   When a write is performed to a cache line for which the U bit and V bit are both 1, after write-back of that cache line, the tag, U bit, and V bit specified in the data field are written.
3. OC address array write (associative)

   When a write is performed with the A bit in the address field set to 1, the tag in each way stored in the entry specified in the address field is compared with the tag specified in the data field. The way numbers of bits [14:13] in the address field are not used. If the MMU is enabled at this time, comparison is performed after the virtual address specified by data field bits [31:10] has been translated to a physical address using the UTLB. If the addresses match and the V bit in the way is 1, the U bit and V bit specified in the data field are written into the OC entry. In other cases, no operation is performed. This operation is used to invalidate a specific OC entry. If the OC entry U bit is 1, and 0 is written to the V bit or to the U bit, write-back is performed. If a UTLB miss occurs during address translation, or the comparison shows a mismatch, an exception is not generated, no operation is performed, and the write is not executed.

Note: OC address array associative writing function may not be supported in the future SuperH Series. Therefore, it is recommended that the OCBI, OCBP, or OCBWB instruction should be used to operate the OC definitely by reporting data TLB miss exception.

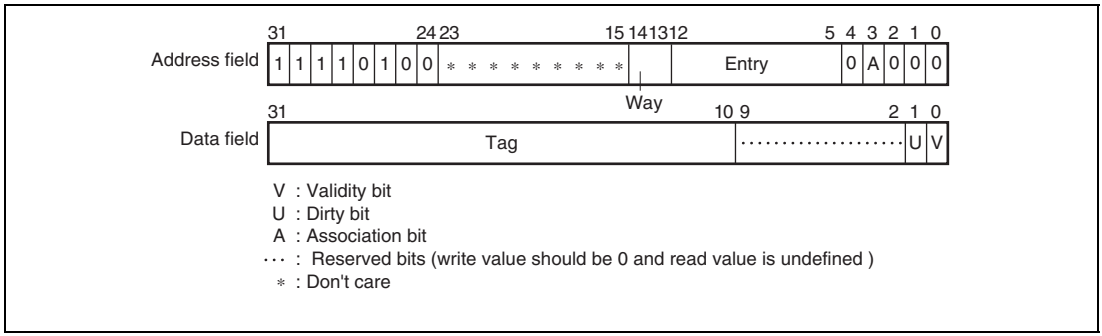


Figure 2.62 Memory-Mapped OC Address Array (Cache size = 32 Kbytes)

(4) OC Data Array

The OC data array is allocated to addresses H'F500 0000 to H'F5FF FFFF in the P4 area. A data array access requires a 32-bit address field specification (when reading or writing) and a 32-bit data field specification. The way and entry to be accessed are specified in the address field, and the longword data to be written is specified in the data field.

In the address field, bits [31:24] have the value H'F5 indicating the OC data array, and the way is specified by bits [14:13] and the entry by bits [12:5]. Address field bits [4:2] are used for the longword data specification in the entry. As only longword access is used, 0 should be specified for address field bits [1:0].

The data field is used for the longword data specification.

The following two kinds of operation can be used on the OC data array:

1. OC data array read

   Longword data is read into the data field from the data specified by the longword specification bits in the address field in the OC entry corresponding to the way and entry set in the address field.
2. OC data array write

   The longword data specified in the data field is written for the data specified by the longword specification bits in the address field in the OC entry corresponding to the way and entry set in the address field. This write does not set the U bit to 1 on the address array side.

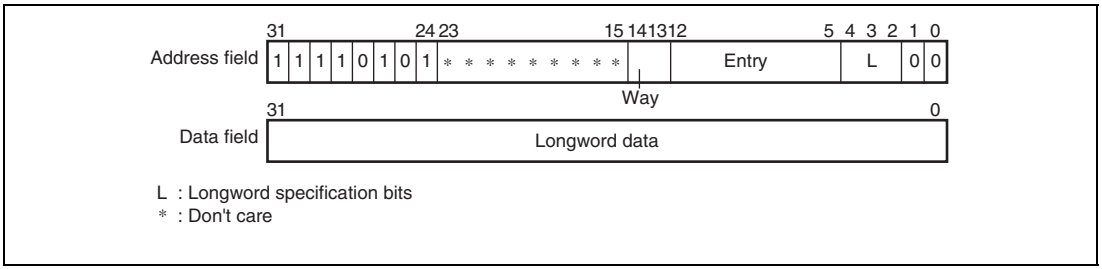


Figure 2.63 Memory-Mapped OC Data Array (Cache size = 32 Kbytes)

(5) Memory-Mapped Cache Associative Write Operation

Associative writing to the IC and OC address arrays may not be supported in future SuperH-family products. The use of instructions ICBI, OCBI, OCBP, and OCBWB is recommended. These instructions handle ITLB misses, and notify instruction TLB miss exceptions and data TLB miss exceptions, thus providing a sure way of controlling the IC and OC. As a transitional measure, the SH-4A generates address errors when this function is used. If compatibility with previous products is a crucial consideration, on the other hand, the MMCAW bit in EXPMASK (H'FF2F 0004) can be set to 1 to enable this function. However, instructions ICBI, OCBI, OCBP, and OCBWB should be used to guarantee compatibility with future SuperH-family products.

2.8.7 Store Queues

The SH-4A supports two 32-byte store queues (SQs) to perform high-speed writes to external memory.

(1) SQ Configuration

There are two 32-byte store queues, SQ0 and SQ1, as shown in figure 2.64. These two store queues can be set independently.

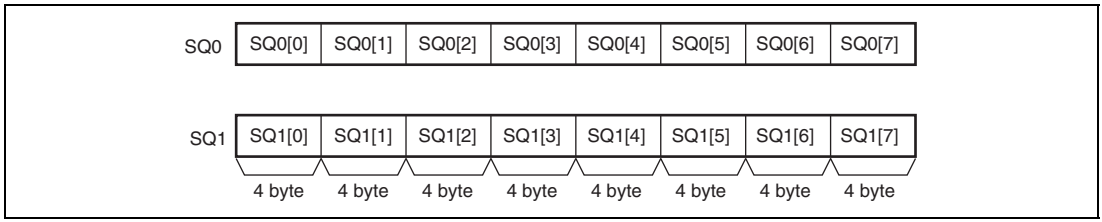


Figure 2.64 Store Queue Configuration

(2) Writing to SQ

A write to the SQs can be performed using a store instruction for addresses H'E000 0000 to H'E3FF FFFC in the P4 area. A longword or quadword access size can be used. The meanings of the address bits are as follows:

[31:26]	: 111000	Store queue specification
[25:6]	: Don't care	Used for external memory transfer/access right
[5]	: 0/1	0: SQ0 specification
		1: SQ1 specification
[4:2]	: LW specification	Specifies longword position in SQ0/SQ1
[1:0]	: 00	Fixed at 0

(3) Transfer to External Memory

Transfer from the SQs to external memory can be performed with a prefetch instruction (PREF). Issuing a PREF instruction for addresses H'E000 0000 to H'E3FF FFFC in the P4 area starts a transfer from the SQs to external memory. The transfer length is fixed at 32 bytes, and the start address is always at a 32-byte boundary. While the contents of one SQ are being transferred to external memory, the other SQ can be written to without a penalty cycle. However, writing to the SQ involved in the transfer to external memory is kept waiting until the transfer is completed.

The physical address bits [28:0] of the SQ transfer destination are specified as shown below, according to whether the MMU is enabled or disabled.

- When MMU is enabled (AT = 1 in MMUCR)

  The SQ area (H'E000 0000 to H'E3FF FFFF) is set in VPN of the UTLB, and the transfer destination physical address in PPN. The ASID, V, SZ, SH, PR, and D bits have the same meaning as for normal address translation, but the C and WT bits have no meaning with regard to this page. When a prefetch instruction is issued for the SQ area, address translation is performed and physical address bits [28:10] are generated in accordance with the SZ bit specification. For physical address bits [9:5], the address prior to address translation is generated in the same way as when the MMU is disabled. Physical address bits [4:0] are fixed at 0. Transfer from the SQs to external memory is performed to this address.
- When MMU is disabled (AT = 0 in MMUCR)

  The SQ area (H'E000 0000 to H'E3FF FFFF) is specified as the address at which a PREF instruction is issued. The meanings of address bits [31:0] are as follows:

[31:26]	: 111000	Store queue specification
[25:6]	: Address	Transfer destination physical address bits [25:6]
[5]	: 0/1	0: SQ0 specification 1: SQ1 specification and transfer destination physical address bit [5]
[4:2]	: Don't care	No meaning in a prefetch
[1:0]	: 00	Fixed at 0

  Physical address bits [28:26], which cannot be generated from the above address, are generated from QACR0 and QACR1.

  QACR0[4:2] : Physical address bits [28:26] corresponding to SQ0
  QACR1[4:2] : Physical address bits [28:26] corresponding to SQ1

Physical address bits [4:0] are always fixed at 0 since burst transfer starts at a 32-byte boundary.

(4) Determination of SQ Access Exception

Determination of an exception in a write to an SQ or transfer to external memory (PREF instruction) is performed as follows according to whether the MMU is enabled or disabled. If an exception occurs during a write to an SQ, the SQ contents before the write are retained. If an exception occurs in a data transfer from an SQ to external memory, the transfer to external memory will be aborted.

- When MMU is enabled (AT = 1 in MMUCR)

  Operation is in accordance with the address translation information recorded in the UTLB, and the SQMD bit in MMUCR. Write type exception judgment is performed for writes to the SQs, and read type exception judgment for transfer from the SQs to external memory (using a PREF instruction). As a result, a TLB miss exception or protection violation exception is generated as required. However, if SQ access is enabled in privileged mode only by the SQMD bit in MMUCR, an address error will occur even if address translation is successful in user mode.

- When MMU is disabled (AT = 0 in MMUCR)

  Operation is in accordance with the SQMD bit in MMUCR.

  0: Privileged/user mode access possible

  1: Privileged mode access possible

  If the SQ area is accessed in user mode when the SQMD bit in MMUCR is set to 1, an address error will occur.

(5) Reading from SQ

In privileged mode in the SH-4A, reading the contents of the SQs may be performed by means of a load instruction for addresses H'FF00 1000 to H'FF00 103C in the P4 area. Only longword access is possible.

[31:6]	: H'FF00 1000	Store queue specification
[5]	: 0/1	0: SQ0 specification
		1: SQ1 specification
[4:2]	: LW specification	Specifies longword position in SQ0/SQ1
[1:0]	: 00	Fixed at 0

2.8.8 Notes on Using 32-Bit Address Extended Mode

In 32-bit address extended mode, the items described in this section are extended as follows.

1. The tag bits [28:10] (19 bits) in the IC and OC are extended to bits [31:10] (22 bits).
2. An instruction which operates the IC (a memory-mapped IC access and writing to the ICI bit in CCR) should be located in the P1 or P2 area. The cacheable bit (C bit) in the corresponding entry in the PMB should be 0.
3. Bits [4:2] (3 bits) for the AREA0 bit in QACR0 and the AREA1 bit in QACR1 are extended to bits [7:2] (6 bits).

2.9 On-Chip Memory

This LSI includes two types of memory modules for storage of instructions and data: OL memory and IL memory. The OL memory is suitable for data storage while the IL memory is suitable for instruction storage.

2.9.1 Features

(1) OL Memory

- Capacity
  The OL memory size is 16 Kbytes.
- Page
  The OL memory is divided into four pages (pages 0A, 0B, 1A and 1B).
- Memory map
  The OL memory is allocated in the addresses shown in table 2.39 in both the virtual address space and the physical address space.

Table 2.39 OL memory Addresses

	Memory Size
Page	**16 Kbytes**
Page 0A	H'E500 E000 to H'E500 EFFF
Page 0B	H'E500 F000 to H'E500 FFFF
Page 1A	H'E501 0000 to H'E501 0FFF
Page 1B	H'E501 1000 to H'E501 1FFF

- Ports
  Each page has three independent read/write ports and is connected to the SuperHyway bus, the cache/RAM internal bus, and operand bus. The operand bus is used when the OL memory is accessed through operand access. The cache/RAM internal bus is used when the OL memory is accessed through instruction fetch. The SuperHyway bus is used for OL memory access from the SuperHyway bus master module.
- Priority
  In the event of simultaneous accesses to the same page from different buses, the access requests are processed according to priority. The priority order is: SuperHyway bus > Cache/RAM internal bus > operand bus.

(2) IL Memory

- Capacity

  The IL memory size is 16 Kbytes.
- Page

  The IL memory is divided into four pages (pages 0, 1, 2, and 3).
- Memory map

  The IL memory is allocated to the addresses shown in table 2.40 in both the virtual address space and the physical address space.

Table 2.40 IL Memory Addresses

	Memory Size
Page	**16 Kbytes**
Page 0	H'E520 0000 to H'E520 0FFF
Page 1	H'E520 1000 to H'E520 1FFF
Page 2	H'E520 2000 to H'E520 2FFF
Page 3	H'E520 3000 to H'E520 3FFF

- Ports

  The page has three independent read/write ports and is connected to the SuperHyway bus, the cache/RAM internal bus, and the instruction bus. The instruction bus is used when the IL memory is accessed through instruction fetch. The cache/RAM internal bus is used when the IL memory is accessed through operand access. The SuperHyway bus is used for IL memory access from the SuperHyway bus master module.
- Priority

  In the event of simultaneous accesses to the same page from different buses, the access requests are processed according to priority. The priority order is: SuperHyway bus > cache/RAM internal bus > instruction bus.

2.9.2 Register Descriptions

The following registers are related to the on-chip memory.

Table 2.41 Register Configuration

Name	Abbreviation	R/W	P4 Address*	Area 7 Address*	Access Size
On-chip memory control register	RAMCR	R/W	H'FF00 0074	H'1F00 0074	32
OL memory transfer source address register 0	LSA0	R/W	H'FF00 0050	H'1F00 0050	32
OL memory transfer source address register 1	LSA1	R/W	H'FF00 0054	H'1F00 0054	32
OL memory transfer destination address register 0	LDA0	R/W	H'FF00 0058	H'1F00 0058	32
OL memory transfer destination address register 1	LDA1	R/W	H'FF00 005C	H'1F00 005C	32

Note: * The P4 address is the address used when using P4 area in the virtual address space. The area 7 address is the address used when accessing from area 7 in the physical address space using the TLB.

Table 2.42 Register States in Each Processing Mode

Name	Abbreviation	Power-On Reset	Manual Reset	Sleep	Software Standby	Deep Standby
On-chip memory control register	RAMCR	H'0000 0000	H'0000 0000	Retained	Retained	H'0000 0000
OL memory transfer source address register 0	LSA0	Undefined	Undefined	Retained	Retained	Undefined
OL memory transfer source address register 1	LSA1	Undefined	Undefined	Retained	Retained	Undefined
OL memory transfer destination address register 0	LDA0	Undefined	Undefined	Retained	Retained	Undefined
OL memory transfer destination address register 1	LDA1	Undefined	Undefined	Retained	Retained	Undefined

(1) On-Chip Memory Control Register (RAMCR)

RAMCR controls the protective functions in the on-chip memory.

When updating RAMCR, please follow limitation described at section 8.2.4, On-Chip Memory Control Register (RAMCR).

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RMD	RP	IC2W	OC2W	ICWPD	—	—	—	—	—
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved For read/write in these bits, refer to General Precautions on Handling of Product.
9	RMD	0	R/W	On-Chip Memory Access Mode Specifies the right of access to the on-chip memory from the virtual address space. 0: An access in privileged mode is allowed. (An address error exception occurs in user mode.) 1: An access in user/ privileged mode is allowed.
8	RP	0	R/W	On-Chip Memory Protection Enable Selects whether or not to use the protective functions using ITLB and UTLB for accessing the on-chip memory from the virtual address space. 0: Protective functions are not used. 1: Protective functions are used. For further details, refer to section 2.9.4, On-Chip Memory Protective Functions.
7	IC2W	0	R/W	IC Two-Way Mode For further details, refer to section 2.8.4 (3), IC Two-Way Mode.

Bit	Bit Name	Initial Value	R/W	Description
6	OC2W	0	R/W	OC Two-Way Mode For further details, refer to section 2.8.3 (6), OC Two-Way Mode.
5	ICWPD	0	R/W	IC Way Prediction Disable For further details, refer to section 2.8.4 (4), Instruction Cache Way Prediction Operation.
4 to 0	—	All 0	R	Reserved For read/write in these bits, refer to General Precautions on Handling of Product.

(2) OL memory Transfer Source Address Register 0 (LSA0)

When MMUCR.AT = 0 or RAMCR.RP = 0, the LSA0 specifies the transfer source physical address for block transfer to page 0A or 0B of the OL memory.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	L0SADR												
Initial value :	0	0	0	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

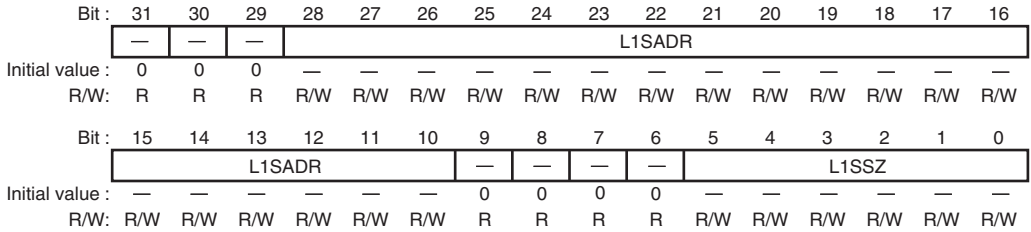
Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	L0SADR						—	—	—	—	L0SSZ					
Initial value :	—	—	—	—	—	—	0	0	0	0	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved For read/write in these bits, refer to General Precautions on Handling of Product.
28 to 10	L0SADR	Undefined	R/W	OL memory Page 0 Block Transfer Source Address When MMUCR.AT = 0 or RAMCR.RP = 0, these bits specify the transfer source physical address for block transfer to page 0A or 0B in the OL memory.
9 to 6	—	All 0	R	Reserved For read/write in these bits, refer to General Precautions on Handling of Product.

Bit	Bit Name	Initial Value	R/W	Description
5 to 0	L0SSZ	Undefined	R/W	OL memory Page 0 Block Transfer Source Address Select When MMUCR.AT = 0 or RAMCR.RP = 0, these bits select whether the operand addresses or L0SADR values are used as bits 15 to 10 of the transfer source physical address for block transfer to page 0A or 0B in the OL memory. L0SSZ[5:0] correspond to the transfer source physical addresses [15:10]. 0: The operand address is used as the transfer source physical address. 1: The L0SADR value is used as the transfer source physical address. Settable values: 111111: Transfer source physical address is specified in 1-Kbyte units. 111110: Transfer source physical address is specified in 2-Kbyte units. 111100: Transfer source physical address is specified in 4-Kbyte units. 111000: Transfer source physical address is specified in 8-Kbyte units. 110000: Transfer source physical address is specified in 16-Kbyte units. 100000: Transfer source physical address is specified in 32-Kbyte units. 000000: Transfer source physical address is specified in 64-Kbyte units. Settings other than the ones given above are prohibited.

(3) OL memory Transfer Source Address Register 1 (LSA1)

When MMUCR.AT = 0 or RAMCR.RP = 0, the LSA1 specifies the transfer source physical address for block transfer to page 1A or 1B in the OL memory.



Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved For read/write in these bits, refer to General Precautions on Handling of Product.
28 to 10	L1SADR	Undefined	R/W	OL memory Page 1 Block Transfer Source Address When MMUCR.AT = 0 or RAMCR.RP = 0, these bits specify transfer source physical address for block transfer to page 1A or 1B in the OL memory.
9 to 6	—	All 0	R	Reserved For read/write in these bits, refer to General Precautions on Handling of Product.

Bit	Bit Name	Initial Value	R/W	Description
5 to 0	L1SSZ	Undefined	R/W	OL memory Page 1 Block Transfer Source Address Select When MMUCR.AT = 0 or RAMCR.RP = 0, these bits select whether the operand addresses or L1SADR values are used as bits 15 to 10 of the transfer source physical address for block transfer to page 1A or 1B in the OL memory. L1SSZ bits [5:0] correspond to the transfer source physical addresses [15:10]. 0: The operand address is used as the transfer source physical address. 1: The L1SADR value is used as the transfer source physical address. Settable values: 111111: Transfer source physical address is specified in 1-Kbyte units. 111110: Transfer source physical address is specified in 2-Kbyte units. 111100: Transfer source physical address is specified in 4-Kbyte units. 111000: Transfer source physical address is specified in 8-Kbyte units. 110000: Transfer source physical address is specified in 16-Kbyte units. 100000: Transfer source physical address is specified in 32-Kbyte units. 000000: Transfer source physical address is specified in 64-Kbyte units. Settings other than the ones given above are prohibited.

(4) OL memory Transfer Destination Address Register 0 (LDA0)

When MMUCR.AT = 0 or RAMCR.RP = 0, LDA0 specifies the transfer destination physical address for block transfer to page 0A or 0B of the OL memory.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	L0DADR												
Initial value :	0	0	0	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	L0DADR						—	—	—	—	L0DSZ					
Initial value :	—	—	—	—	—	—	0	0	0	0	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved For read/write in these bits, refer to General Precautions on Handling of Product.
28 to 10	L0DADR	Undefined	R/W	OL memory Page 0 Block Transfer Destination Address When MMUCR.AT = 0 or RAMCR.RP = 0, these bits specify transfer destination physical address for block transfer to page 0A or 0B in the OL memory.
9 to 6	—	All 0	R	Reserved For read/write in these bits, refer to General Precautions on Handling of Product.

Bit	Bit Name	Initial Value	R/W	Description
5 to 0	L0DSZ	Undefined	R/W	OL memory Page 0 Block Transfer Destination Address Select When MMUCR.AT = 0 or RAMCR.RP = 0, these bits select whether the operand addresses or L0DADR values are used as bits 15 to 10 of the transfer destination physical address for block transfer to page 0A or 0B in the OL memory. L0DSZ bits [5:0] correspond to the transfer destination physical address bits [15:10]. 0: The operand address is used as the transfer destination physical address. 1: The L0DADR value is used as the transfer destination physical address. Settable values: 111111: Transfer destination physical address is specified in 1-Kbyte units. 111110: Transfer destination physical address is specified in 2-Kbyte units. 111100: Transfer destination physical address is specified in 4-Kbyte units. 111000: Transfer destination physical address is specified in 8-Kbyte units. 110000: Transfer destination physical address is specified in 16-Kbyte units. 100000: Transfer destination physical address is specified in 32-Kbyte units. 000000: Transfer destination physical address is specified in 64-Kbyte units. Settings other than the ones given above are prohibited.

(5) OL memory Transfer Destination Address Register 1 (LDA1)

When MMUCR.AT = 0 or RAMCR.RP = 0, LDA1 specifies the transfer destination physical address for block transfer to page 1A or 1B in the OL memory.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	L1DADR												
Initial value :	0	0	0	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	L1DADR						—	—	—	—	L1DSZ					
Initial value :	—	—	—	—	—	—	0	0	0	0	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved For read/write in these bits, refer to General Precautions on Handling of Product.
28 to 10	L1DADR	Undefined	R/W	OL memory Page 1 Block Transfer Destination Address When MMUCR.AT = 0 or RAMCR.RP = 0, these bits specify transfer destination physical address for block transfer to page 1A or 1B in the OL memory.
9 to 6	—	All 0	R	Reserved For read/write in these bits, refer to General Precautions on Handling of Product.

Bit	Bit Name	Initial Value	R/W	Description
5 to 0	L1DSZ	Undefined	R/W	OL memory Page 1 Block Transfer Destination Address Select When MMUCR.AT = 0 or RAMCR.RP = 0, these bits select whether the operand addresses or L1DADR values are used as bits 15 to 10 of the transfer destination physical address for block transfer to page 1A or 1B in the OL memory. L1DSZ bits [5:0] correspond to the transfer destination physical addresses [15:10]. 0: The operand address is used as the transfer destination physical address. 1: The L1DADR value is used as the transfer destination physical address. Settable values: 111111: Transfer destination physical address is specified in 1-Kbyte units. 111110: Transfer destination physical address is specified in 2-Kbyte units. 111100: Transfer destination physical address is specified in 4-Kbyte units. 111000: Transfer destination physical address is specified in 8-Kbyte units. 110000: Transfer destination physical address is specified in 16-Kbyte units. 100000: Transfer destination physical address is specified in 32-Kbyte units. 000000: Transfer destination physical address is specified in 64-Kbyte units. Settings other than the ones given above are prohibited.

2.9.3 Operation

(1) Instruction Fetch Access from the CPU

(a) IL Memory

Instruction fetch access from the CPU is performed directly via the instruction bus for a given virtual address. In the case of successive accesses to the same page of IL memory and as long as no page conflict occurs, the access takes one cycle.

(b) OL Memory

Instruction fetch access from the CPU is performed via the cache/RAM internal bus. This access takes more than one cycle.

(2) Operand Access from the CPU and Access from the FPU

(a) IL Memory

Operand access from the CPU and access from the FPU are performed via the cache/RAM internal bus. Access via the cache/RAM internal bus takes more than one cycle.

(b) OL Memory

Access from the CPU or FPU is performed via the operand bus for a given virtual address. Read access from the operand bus by virtual address takes one cycle if the access is made successively to the same page of OL memory and as long as no page conflict occurs. Write access from the operand bus by virtual address takes one cycle as long as no page conflict occurs.

(3) Access from the SuperHyway Bus Master Module

On-chip memory is always accessed by the SuperHyway bus master module, such as DMAC, via the SuperHyway bus which is a physical address bus. The same addresses as for the virtual addresses must be used.

(4) OL Memory Block Transfer

High-speed data transfer can be performed through block transfer between the OL memory and external memory without cache utilization.

Data can be transferred from the external memory to the OL memory through a prefetch instruction (PREF). Block transfer from the external memory to the OL memory begins when the PREF instruction is issued to the address in the OL memory area in the virtual address space.

Data can be transferred from the OL memory to the external memory through a write-back instruction (OCBWB). Block transfer from the OL memory to the external memory begins when the OCBWB instruction is issued to the address in the OL memory area in the virtual address space.

In either case, transfer rate is fixed to 32 bytes. Since the start address is always limited to a 32-byte boundary, the lower five bits of the address indicated by Rn are ignored, and are always dealt with as all 0s. In either case, other pages and cache can be accessed during block transfer, but the CPU will stall if the page which is being transferred is accessed before data transfer ends.

The physical addresses [28:0] of the external memory performing data transfers with the OL memory are specified as follows according to whether the MMU is enabled or disabled.

(a) When MMU is Enabled (MMUCR.AT = 1) and RAMCR.RP = 1

An address of the OL memory area is specified to the UTLB VPN field, and to the physical address of the transfer source (in the case of the PREF instruction) or the transfer destination (in the case of the OCBWB instruction) to the PPN field. The ASID, V, SZ, SH, PR, and D bits have the same meaning as normal address conversion; however, the C and WT bits have no meaning in this page.

When the PREF instruction is issued to the OL memory area, address conversion is performed in order to generate the physical address bits [28:10] in accordance with the SZ bit specification. The physical address bits [9:5] are generated from the virtual address prior to address conversion. The physical address bits [4:0] are fixed to 0. Block transfer is performed to the OL memory from the external memory which is specified by these physical addresses.

When the OCBWB instruction is issued to the OL memory area, address conversion is performed in order to generate the physical address bits [28:10] in accordance with the SZ bit specification. The physical address bits [9:5] are generated from the virtual address prior to address conversion. The physical address bits [4:0] are fixed to 0. Block transfer is performed from the OL memory to the external memory specified by these physical addresses.

In PREF or OCBWB instruction execution, an MMU exception is checked as read type. After the MMU execution check, a TLB miss exception or protection error exception occurs if necessary. If an exception occurs, the block transfer is inhibited.

(b) When MMU is Disabled (MMUCR.AT = 0) or RAMCR.RP = 0

The transfer source physical address in block transfer to page 0A or 0B in the OL memory is set in the L0SADR bits of the LSA0 register. And the L0SSZ bits in the LSA0 register choose either the virtual addresses specified through the PRFF instruction or the L0SADR values as bits 15 to 10 of the transfer source physical address. In other words, the transfer source area can be specified in units of 1 Kbyte to 64 Kbytes.

The transfer destination physical address in block transfer from page 0A or 0B in the OL memory is set in the L0DADR bits of the LDA0 register. And the L0DSZ bits in the LDA0 register choose either the virtual addresses specified through the OCBWB instruction or the L0DADR values as bits 15 to 10 of the transfer destination physical address. In other words, the transfer source area can be specified in units of 1 Kbyte to 64 Kbytes.

Block transfer to page 1A or 1B in the OL memory is set to LSA1 and LDA1 as with page 0A or 0B in the OL memory.

When the PREF instruction is issued to the OL memory area, the physical address bits [28:10] are generated in accordance with the LSA0 or LSA1 specification. The physical address bits [9:5] are generated from the virtual address. The physical address bits [4:0] are fixed to 0. Block transfer is performed from the external memory specified by these physical addresses to the OL memory.

When the OCBWB instruction is issued to the OL memory area, the physical address bits [28:10] are generated in accordance with the LDA0 or LDA1 specification. The physical address bits [9:5] are generated from the virtual address. The physical address bits [4:0] are fixed to 0. Block transfer is performed from the OL memory to the external memory specified by these physical addresses.

2.9.4 On-Chip Memory Protective Functions

The SH-4A implements the following protective functions to the on-chip memory by using the on-chip memory access mode bit (RMD) and the on-chip memory protection enable bit (RP) in the on-chip memory control register (RAMCR).

- Protective functions for access from the CPU and FPU

  When RAMCR.RMD = 0, and the on-chip memory is accessed in user mode, it is determined to be an address error exception.

  When MMUCR.AT = 1 and RAMCR.RP = 1, MMU exception and address error exception are checked in the on-chip memory area which is a part of area P4 as with the area P0/P3/U0.

The above descriptions are summarized in table 2.43.

Table 2.43 Protective Function Exceptions to Access On-Chip Memory

MMUCR.AT	RAMCR.RP	SR.MD	RAMCR. RMD	Always Occurring Exceptions	Possibly Occurring Exceptions
0	x	0	0	Address error exception	—
			1	—	—
		1	x	—	—
1	0	0	0	Address error exception	—
			1	—	—
		1	x	—	—
	1	0	0	Address error exception	—
			1	—	MMU exception
		1	x	—	MMU exception

[Legend] x: Don't care

2.9.5 Usage Notes

(1) Page Conflict

In the event of simultaneous access to the same page from different buses, page conflict occurs. Although each access is completed correctly, this kind of conflict tends to lower OL memory accessibility. Therefore it is advisable to provide all possible preventative software measures. For example, conflicts will not occur if each bus accesses different pages.

(2) Access Across Different Pages

(a) OL Memory

Read access from the operand bus is performed in one cycle when the access is made successively to the same page but takes multiple cycles (a maximum of two wait cycles may be required) when the access is made across pages or the previous access was made to memory other than OL memory. For this reason, from the viewpoint of performance optimization, it is recommended to design the software such that the page corresponding to the address for read access from the operand bus does not change so often.

(b) IL Memory

Access from the instruction bus is performed in one cycle when the access is made successively to the same page but takes multiple cycles (a maximum of two wait cycles may be required) when the access is made across pages or the previous access was made to memory other than IL memory. For this reason, from the viewpoint of performance optimization, it is recommended to design the software such that the target page does not change so often in access from the instruction bus. For example, allocating a separate program for each page will deliver better efficiency.

(3) IL Memory Coherency

In order to allocate instructions in the IL memory, write an instruction to the IL memory, execute the following sequence, then branch to the rewritten instruction.

- SYNCO
- ICBI @Rn

In this case, the target for the ICBI instruction can be any address (IL memory address may be possible) within the range where no address error exception occurs, and cache hit/miss is possible.

(4) Sleep Mode

The SuperHyway bus master module, such as DMAC, cannot access OL memory and IL memory in sleep mode.

2.9.6 Note on Using 32-Bit Address Extended Mode

In 32-bit address extended mode, L0SADR fields in LSA0, L1SADR fields in LSA1, L0DADR fields in LDA0, and L1DADR fields in LDA1 are extended from 19-bit [28:10] to 22-bit [31:10].

2.10 General Precautions on Handling of Product

2.10.1 Prohibition of Access to Undefined or Reserved Addresses

Access to undefined or reserved addresses is prohibited.

The undefined or reserved addresses may be used to expand functions, or test registers may be allocated to these addresses. Do not access these registers; the system's operation is not guaranteed if they are accessed.

2.10.2 Reading from/Writing to Reserved Bit of Each Register

Treat the reserved bit of register used in each module as follows except in cases where the specifications for values which are read from or written to the bit are provided in the description.

The bit is always read as 0. The write value should be 0 or one, which has been read immediately before writing.

Writing the value, which has been read immediately before writing has the advantage of preventing the bit from being affected on its extended function when the function is assigned.

2.11 Appendix

2.11.1 CPU Operation Mode Register (CPUOPM)

The CPUOPM is used to control the CPU operation mode. This register can be read from or written to the address H'FF2F0000 in P4 area or H'1F2F0000 in area 7 as 32-bit size.

The write value to the reserved bits should be the initial value.

The operation is not guaranteed if the write value is not the initial value.

The CPUOPM register should be updated by the CPU MOV instruction not the access from SuperHyway bus master except CPU.

After the CPUOPM is updated, read CPUOPM once, and execute one of the following two methods.

1. Execute a branch using the RTE instruction.
2. Execute the ICBI instruction for any address (including non-cacheable area).

After one of these methods is executed, it is guaranteed that the CPU runs under the updated CPUOPM value.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	RABD	—	INTMU	—	—	—
Initial value:	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved The write value must be the initial value.
9 to 6	—	All 1	R	Reserved The write value must be the initial value.
5	RABD	0	R/W	Speculative execution bit for subroutine return 0: Instruction fetch for subroutine return is issued speculatively. When this bit is set to 0, refer to section 2.11.3, Speculative Execution for Subroutine Return. 1: Instruction fetch for subroutine return is not issued speculatively.
4	—	0	R	Reserved The write value must be the initial value.
3	INTMU	0	R/W	Interrupt mode switch bit 0: SR.IMASK is not changed when an interrupt is accepted. 1: SR.IMASK is changed to the accepted interrupt level.
2 to 0	—	All 0	R	Reserved The write value must be the initial value.

2.11.2 Instruction Prefetching and Its Side Effects

This LSI is provided with an internal buffer for holding pre-read instructions, and always performs pre-reading. Therefore, program code must not be located in the last 64-byte area of any memory space. If program code is located in these areas, a bus access for instruction prefetch may occur exceeding the memory areas boundary. A case in which this is a problem is shown below.

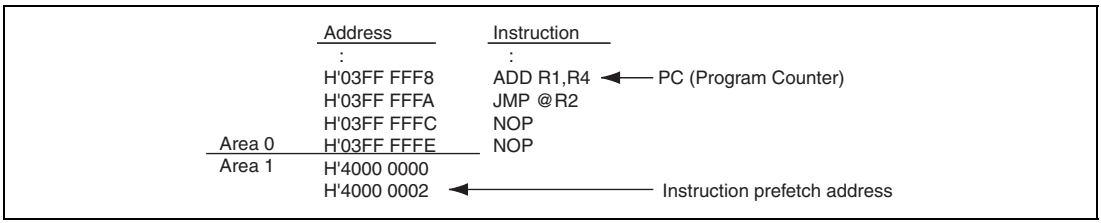


Figure 2.65 Instruction Prefetch

Figure 2.65 presupposes a case in which the instruction (ADD) indicated by the program counter (PC) and the address H'04000002 instruction prefetch are executed simultaneously. It is also assumed that the program branches to an area other than area 1 after executing the following JMP instruction and delay slot instruction.

In this case, a bus access (instruction prefetch) to area 1 may unintentionally occur from the programming flow.

(1) Instruction Prefetch Side Effects

- It is possible that an external bus access caused by an instruction prefetch may result in misoperation of an external device, such as a FIFO, connected to the area concerned.
- If there is no device to reply to an external bus request caused by an instruction prefetch, hang-up will occur.

(2) Remedies

- These illegal instruction fetches can be avoided by using the MMU.
- The problem can be avoided by not locating program code in the last 64 bytes of any area.

2.11.3 Speculative Execution for Subroutine Return

The SH-4A has the mechanism to issue an instruction fetch speculatively when returning from subroutine. By issuing an instruction fetch speculatively, the execution cycles to return from subroutine may be shortened.

This function is enabled by setting 0 to the bit 5 (RABD) of CPU Operation Mode register (CPUOPM). But this speculative instruction fetch may issue the access to the address that should not be accessed from the program. Therefore, a bus access to an unexpected area or an internal instruction address error may cause a problem. As for the effect of this bus access to unexpected memory area, refer to section 2.11.2 (1), Instruction Prefetch Side Effects.

Usage Condition: When the speculative execution for subroutine return is enabled, the RTS instruction should be used to return to the address set in PR by the JSR, BSR, or BSRF instructions. It can prevent the access to unexpected address and avoid the problem.

2.11.4 Version Registers (PVR, PRR)

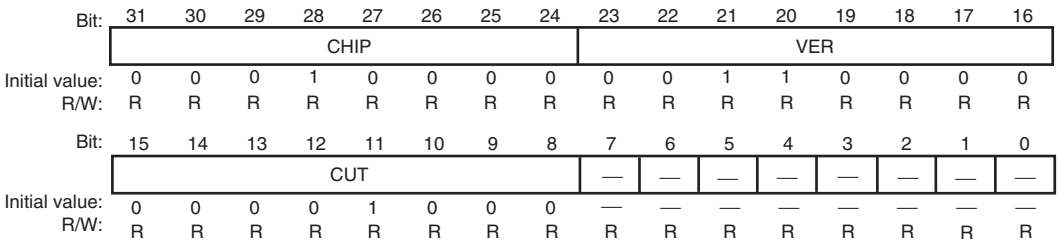
The SH-4A has the read-only registers which show the version of a processor core, and the version of a product. By using the value of these registers, it becomes possible to be able to distinguish the version and product of a processor from software, and to realize the scalability of the high system.

Note: The bit 7 to bit 0 of PVR register and the bit 3 to bit 0 of PRR register should be masked by the software.

Table 2.44 Register Configuration

Register Name	Abbr.	R/W	P4 Address	Area 7 Address	Size
Processor version register	PVR	R	H'FF000030	H'1F000030	32
Product register	PRR	R	H'FF000044	H'1F000044	32

Processor Version Register (PVR):



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	CHIP	H'10	R	Processor Family
23 to 16	VER	H'30	R	Processor Version
15 to 8	CUT	H'0B	R	Processor Version
7 to 0	—	Undefined	R	This value is undefined. It should be masked by software when using it.

Product Register (PRR):

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Product								CUT				—	—	—	—
Initial value:	0	0	1	0	1	1	1	0	0	0	0	0	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	All bits are fixed to 0.
15 to 8	Product	H'2E	R	Product Type H'2E: SH7734
7 to 4	CUT	H'0	R	Product Version 1st Cut: 0000
3 to 0	—	Undefined	R	This value is undefined. It should be masked by software when using it.

Section 3 Resets and Watchdog Timer (RESET/WDT)

The reset and watchdog timer module (WDT) comprises a reset control unit and a watchdog timer control unit, and controls the power-on reset sequence and the reset of the LSI itself and external peripheral devices. The WDT is a single-channel timer that can be used either as a watchdog timer or interval timer.

3.1 Overview

- The watchdog timer unit monitors for system runaway using a timer counting at regular time intervals.
- The watchdog timer unit has two operating modes:
  - Watchdog timer mode means that a power-on reset or manual reset for the internal modules of the chip is initiated on counter overflow.
  - Interval timer mode means that interval timer interrupts are generated on counter overflow.
- In order to prevent accidental writing to some of the WDT-related registers, writing to them is only possible when a certain code is set in the uppermost eight bits in the data for writing.
- The maximum time until counter overflow occurs is approximately 1374 seconds (at 50-MHz peripheral clock (clkp)).
- $\overline{\text{PRESET}}$ pin control
  - Delay the negation of the $\overline{\text{PRESET}}$ pin using a filter cell.
- Mode pin control
  - Latch mode pin input at the rising edge of $\overline{\text{PRESET}}$ pin.
  - Set the mode pin input for clock mode select to mode signal in the chip during reset.
  - Include a register for reading latched mode signals.
- RESET/WDT functions
  - Select one of the following events to be generated upon a WDT overflow:
    - Power-on reset request
    - Manual reset request
    - Interrupt request
  - Set increment intervals in units of one peripheral-clock cycle (clkp).
  - Set two overflow flags: watchdog timer overflow flag (set on a reset request) and interval timer overflow flag (set on an interrupt request).

Figure 3.1 shows a block diagram of the RESET/WDT.

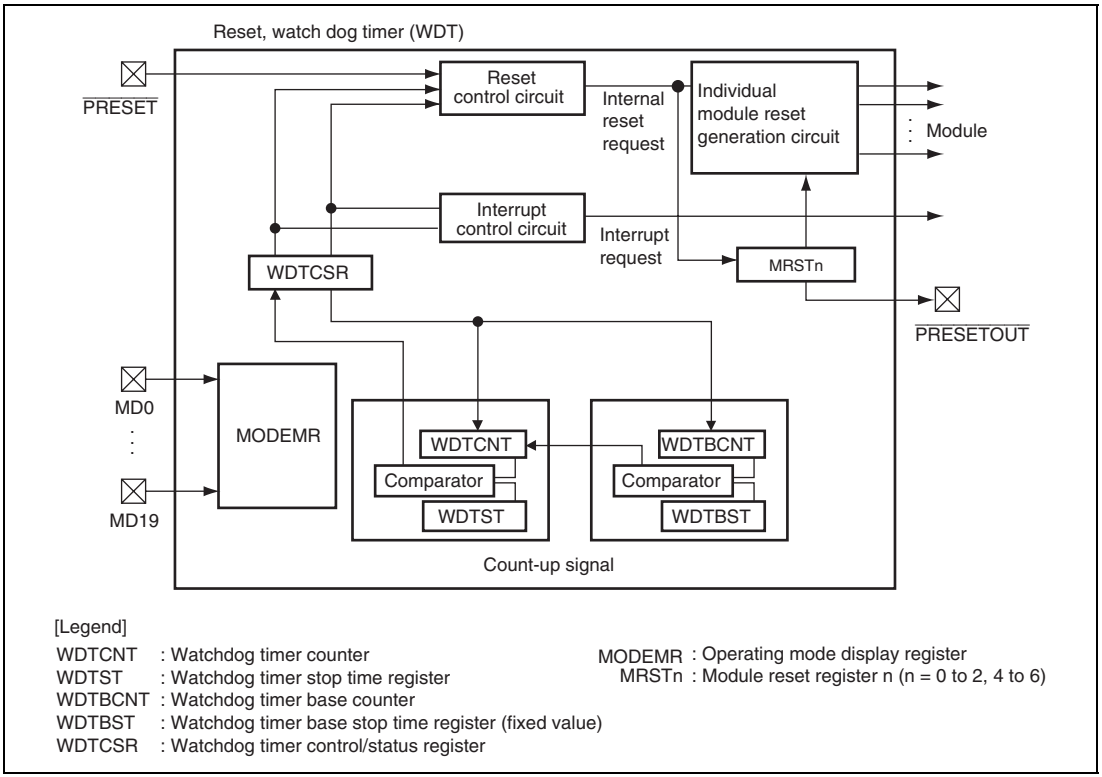


Figure 3.1 Block Diagram of the Reset and Watchdog Timers

3.2 Input/Output Pins

Table 3.1 lists the configuration and functions of the RESET/WDT pins.

Table 3.1 Input/Output Pins

Pin name	I/O	Description
MD0	Input	Mode signals For MD0 to 4, 10, 11, 12, and 15, see section 8, Clock Pulse Generator (CPG). For MD5, 6, 7, 8, 9, 13, and 14, see section 6B, LBSC within Bus Bridge.
MD1	Input	
MD2	Input	
MD3	Input	
MD4	Input	
MD5	Input	
MD6	Input	
MD7	Input	
MD8	Input	
MD9	Input	
MD10	Input	
MD11	Input	
MD12	Input	
MD13	Input	
MD14	Input	
MD15	Input	
MD16	Input	
MD17	Input	
MD18	Input	
MD19	Input	
MPMD	Input	See section 39, User Debugging Interface (H-UDI).
$\overline{\text{PRESET}}$	Input	Power-on reset signal Low level input causes the LSI to be power-on reset.
$\overline{\text{PRESETOUT}}$	Output	Reset output signal The setting of the register for the $\overline{\text{PRESETOUT}}$ pin (MRST2) and all of the sources stipulated in sections 3.4.1 (1), Power-On Reset, and 3.4.1 (2), Manual Reset, of the RESET/WDT specification act as sources for the assertion of a reset via the $\overline{\text{PRESETOUT}}$ pin. For sources other than the MRST2 setting, the flash ROM reset time and flash ROM holding time are secured. The initial value of the register selects the high level.

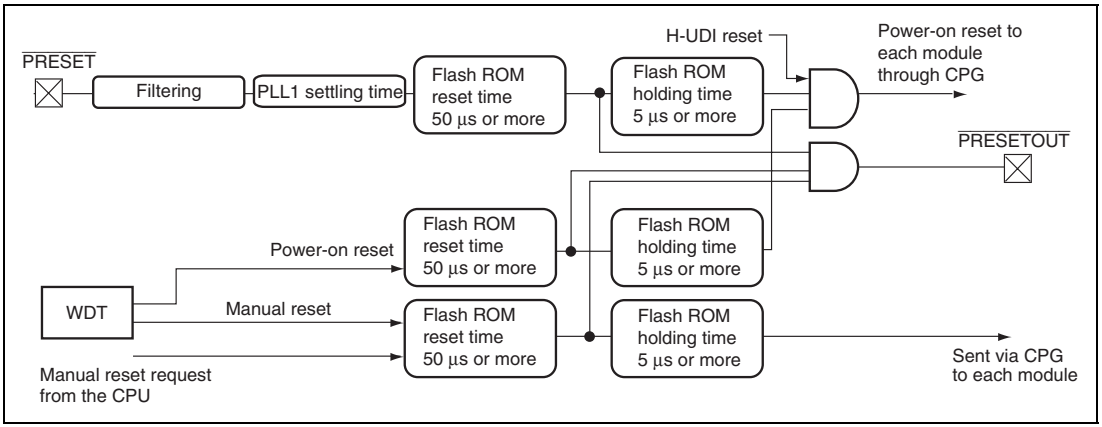


Figure 3.2 $\overline{\text{PRESETOUT}}$ Configuration

3.3 Register Descriptions

Table 3.2 lists the RESET/WDT registers. Table 3.3 lists the register status in each of the processing modes.

Table 3.2 List of Registers

Name	Abbr.	R/W	P4 Address	Area 7 Address	Access Size
Watchdog timer stop time register	WDTST	R/W	H'FFCC 0000	H'1FCC 0000	32
Watchdog timer control/status register	WDTCSR	R/W	H'FFCC 0004	H'1FCC 0004	32
Watchdog timer base stop time register	WDTBST	R/W	H'FFCC 0008	H'1FCC 0008	32
Watchdog timer counter	WDTCNT	R	H'FFCC 0010	H'1FCC 0010	32
Watchdog timer base counter	WDTBCNT	R	H'FFCC 0018	H'1FCC 0018	32
Operating mode display register	MODEMR	R	H'FFCC 0020	H'1FCC 0020	32

Table 3.3 Register Status in Each of the Processing Modes

		Power-on Reset		Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
Name	Abbr.	$\overline{\text{PRESET}}$ Pin	Due to watchdog timer overflow/ H-UDI	Due to watchdog timer overflow/ multiexception	SLEEP Instruction	SLEEP Instruction	—	SLEEP Instruction
Watchdog timer stop time register	WDTST	H'0000 0000	Retained	Retained	Retained	Retained	—	H'0000 0000
Watchdog timer control/status register	WDTCSR	H'0000 0000	Retained	Retained	Retained	Retained	—	H'0000 0000
Watchdog timer base stop time register	WDTBST	H'0000 0000	Retained	Retained	Retained	Retained	—	H'0000 0000
Watchdog timer counter	WDTCNT	H'0000 0000	H'0000 0000	Retained*	Retained	Retained	—	H'0000 0000
Watchdog timer base counter	WDTBCNT	H'0000 0000	H'0000 0000	Retained*	Retained	Retained	—	H'0000 0000
Operating mode display register	MODEMR	H'000- ----	Retained	Retained	Retained	Retained	—	H'000- ----

Notes: Do not write to addresses other than those listed above, otherwise normal operation cannot be guaranteed. Values read from other addresses are undefined.

* The watchdog timer counter and watchdog timer base counter registers are cleared to the initial value (H'0000 0000) upon a WDT overflow.

3.3.1 Watchdog Timer Stop Time Register (WDTST)

WDTST is a 32-bit readable/writable register that sets the time until WDTCNT overflows. The time is minimum when H'5A00_0001 has been written to the register and the time is maximum when H'5A00_0000 has been written to the register.

WDTST should be written in longword. The upper byte of the write value should always be the code value H'5A. If the code value is read, it is always read as 0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Code value (H'5A)								—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	WDTST											
Initial value:	—	—	—	—	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	Code value	H'00	R/W	Code Value (H'5A) These bits are always read as 0.
23 to 12	—	—	R	Reserved
11 to 0	WDTST	H'000	R/W	WDTCNT Overflow Time H'001: Minimum H'000: Maximum

3.3.2 Watchdog Timer Control/Status Register (WDTCSR)

WDTCSR is a 32-bit readable/writable register that consists of the timer mode select bits and overflow flags.

WDTCSR should be written in longword. The upper byte of the write value should always be the code value H'A5. If the code value is read, it is always read as 0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Code value (H'A5)								—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

BIt:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TME	WT/IT	RSTS	WOVF	IOVF	—	—	—
Initial value:	—	—	—	—	—	—	—	—	0	0	0	0	0	—	—	—
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	Code value	H'00	R/W	Code Value (H'A5) These bits are always read as 0.
23 to 8	—	—	R	Reserved
7	TME	0	R/W	Timer Enable 0: Stops incrementing. 1: Starts incrementing.
6	WT/IT	0	R/W	Timer Mode Select 0: Interval timer mode 1: Watchdog timer mode
5	RSTS	0	R/W	Reset Select This bit is valid in watchdog timer mode. 0: Power-on reset 1: Manual reset

Bit	Bit Name	Initial Value	R/W	Description
4	WOVF	0	R/W	Watchdog Timer Overflow This bit is valid in watchdog timer mode. 0: No overflow occurred. 1: WDTCNT overflowed.
3	IOVF	0	R/W	Interval Timer Overflow This bit is valid in interval timer mode. 0: No overflow occurred. 1: WDTCNT overflowed.
2 to 0	—	—	R	Reserved

3.3.3 Watchdog Timer Base Stop Time Register (WDTBST)

WDTBST is a 32-bit readable/writable register that sets the time until WDTBCNT overflows. The time is minimum when H'5500_0001 has been written to the register and the time is maximum when H'5500_0000 has been written to the register.

WDTBST should be written in longword. The upper byte of the write value should always be the code value H'55. If the code value is read, it is always read as 0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Code value (H'55)								WDTBST							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WDTBST															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	Code value	H'00	R/W	Code Value (H'55) These bits are always read as 0.
23 to 0	WDTBST	H'000000	R/W	WDTBCNT Overflow Time H'000001: Minimum H'000000: Maximum

3.3.4 Watchdog Timer Counter (WDTCNT)

WDTCNT is a 32-bit read-only register that is incremented by a WDTBCNT overflow. When WDTCNT overflows, the specified reset is generated in watchdog timer mode or an interrupt is generated in interval timer mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

BIt:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	WDTCNT											
Initial value:	—	—	—	—	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	—	R	Reserved
11 to 0	WDTCNT	H'000	R	Counter Value Incremented by a WDTBCNT overflow. When an overflow occurs, the overflow flag is set and a reset or an interrupt is requested.

3.3.5 Watchdog Timer Base Counter (WDTBCNT)

WDTBCNT is a 32-bit read-only register that is incremented by the peripheral clock (clkp). When WDTBCNT overflows, WDTCNT is incremented and WDTBCNT is cleared to H'0000_0000.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	WDTBCNT							
Initial value:	—	—	—	—	—	—	—	—	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Blt:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WDTBCNT															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	—	R	Reserved
23 to 0	WDTBCNT	H'000000	R	Counter Value When an overflow occurs, WDTCNT is incremented.

3.3.6 Operating Mode Display Register (MODEMR)

The mode settings that have been made at the time of a power-on reset are read.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	MPMD	MODE 19	MODE 18	MODE 17	MODE 16
Initial value:	—	—	—	—	—	—	—	—	—	—	—	*	*	*	*	*
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MODE 15	MODE 14	MODE 13	MODE 12	MODE 11	MODE 10	MODE 09	MODE 08	MODE 07	MODE 06	MODE 05	MODE 04	MODE 03	MODE 02	MODE 01	MODE 00
Initial value:	0	*	*	*	*	*	*	*	*	*	*	0	0	*	*	*
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

*: Depending on the mode

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	—	R	Reserved
20	MPMD	Depends on the mode	R	Value latched from the MPMD Pin. 0: H-UDI mode 1: Boundary scan mode
19	MODE19	Depends on the mode	R	Values latched from the MODE19 to MODE0 Pins Mode bit 19 to 16: Boot mode Mode bit 15: Reserved 0 Mode bit 14: Boot mode switching mode 1: Selects MMC boot mode. 0: Selects normal boot mode. Mode bit 13: 29/32 bit mode 1: 32-bit mode 0: 29-bit mode Mode bit 12: PLL multiplication rate setting (refer to section 8, Clock Pulse Generator (CPG)) Mode bit 11: PLL multiplication rate setting (refer to section 8, Clock Pulse Generator (CPG)) Mode bit 10: XTAL input mode (refer to section 8, Clock Pulse Generator (CPG))
18	MODE18			
17	MODE17			
16	MODE16			
15	MODE15			
14	MODE14			
13	MODE13			
12	MODE12			
11	MODE11			
10	MODE10			

Bit	Bit Name	Initial Value	R/W	Description
9	MODE09	Depending on the mode	R	Mode bit 9 and 7: Area Division
8	MODE08			00: Area 0 64 Mbytes
7	MODE07			Areas 2 to 5, DDR mode
6	MODE06			01: Area 0 128 Mbytes
5	MODE05			Areas 2 to 5, DDR mode
4	MODE04			10: Area 0 64 Mbytes
3	MODE03			Areas 1 to 6, DDR mode
2	MODE02			11: Setting is prohibited.
1	MODE01			Mode bit 8: Big/little endian
0	MODE00			1: Little endian
				0: Big endian
				Mode bit 6: Data width of ExBus
				10: 16-bit bus
				01: 8-bit bus
				Mode bit 5: Data width of ExBus
				10: 16-bit bus
				01: 8-bit bus
				Mode bit 4: Reserved 0
				Mode bit 3: Reserved 0
				Mode bit 2: ExBus side operation frequency (refer to section 8, Clock Pulse Generator (CPG))
				Mode bit 1: ExBus side operation frequency (refer to section 8, Clock Pulse Generator (CPG))
				Mode bit 0: Free-running/step up mode
				1: Step up mode
				0: Free-running mode

3.4 Operation

3.4.1 Reset Requests

(1) Power-On Reset

- Requesting sources
  - A low level input on the $\overline{\text{PRESET}}$ pin
  - WDTCNT overflow when the WT/IT bit is 1 and the RSTS bit is 0 in WDTCSR

- Branch address: H'A000 0000
- Operation until branching

  The exception code H'000 is set in EXPEVT. After initializing VBR and SR, the processing branches by setting PC = H'A000 0000.

  During initialization, VBR is reset to H'0000 0000. SR is initialized such that the MD, RB, and BL bits are set to 1, the FD bit is cleared to 0, and the interrupt mask level bits (IMASK3 to IMASK0) are set to B'1111.

  Then, the CPU and peripheral modules are initialized. For details, refer to the register descriptions in the corresponding sections.

  At power-on, ensure that a low level is input to the $\overline{\text{PRESET}}$ pin. A low level input is also needed on the $\overline{\text{TRST}}$ pin to initialize the H-UDI.

(2) Manual Reset

- Requesting sources
  - A general exception other than a user break when the BL bit in SR is 1.
  - WDTCNT overflows when both the WT/IT and RSTS bits in WDTCSR are 1.

- Branch destination address: address set to H'A000 0000
- Operation until branching

  The exception code H'020 is set in EXPEVT. After initializing VBR and SR, the processing is made to branch by setting PC = H'A000 0000.

  During initialization, VBR is reset to H'0000 0000. SR is initialized such that the MD, RB, and BL bits are set to 1, the FD bit is cleared to 0, and the interrupt mask level bits (IMASK3 to IMASK0) are set to B'1111.

  Then, the CPU and peripheral modules are initialized. For details, refer to the register descriptions in the corresponding sections.

(3) H-UDI Reset

- Requesting source
  — An H-UDI reset. For details, refer to descriptions of User Debug Interface (H-UDI).
- Branch address: H'A000 0000
- Operation until branching
  The processing is the same as that of a power-on reset, except that $\overline{\text{PRESETOUT}}$ is not asserted by an H-UDI reset.

3.4.2 Usage of Watchdog Timer Mode

1. Set the WDTCNT overflow time in the WDTST bits.
2. Set the WT/IT bit in WDTCSR to 1 and select a reset type using the RSTS bit.
3. Set the TME bits in WDTCSR to 1 to start the WDT counter.
4. In watchdog timer mode, avoid overflowing of WDTCNT by periodically clearing WDTCNT or WDTBCNT. For the clearing method, see section 3.4.5, Clearing WDT (CPU0 and CPU1) Counters.
5. If WDTCNT overflows, WOVF in WDTCSR will be set to 1 and a power-on reset or manual reset, respectively, will be generated. After the reset is cleared, WDTCNT or WDTBCNT resume counting.

3.4.3 Usage of Interval Timer Mode in WDT

In interval timer mode, an interval timer interrupt is generated each time the counter overflows, which enables periodic interrupt generation.

1. Set the WDTCNT overflow time in the WDTST bit.
2. Clear the WT/IT bit in WDTCSR to 0.
3. Set the TME bit in WDTCSR to 1 to start the WDT counter.

If WDTCNT overflows, IOVF in WDTCSR will be set to 1 and an interval timer interrupt request will be generated. At this time, WDTCNT and WDTBCNT continue to increment the values.

3.4.4 Time Until WDT Overflows

Figure 3.3 shows relationships between WDTCNT and WDTBCNT in interval timer mode. In interval timer mode, increment is continued after a WDTCNT overflow. In watchdog timer mode, increment is temporarily stopped and resumed after WDTCNT and WDTBCNT are cleared to 0 following the reset cancellation.

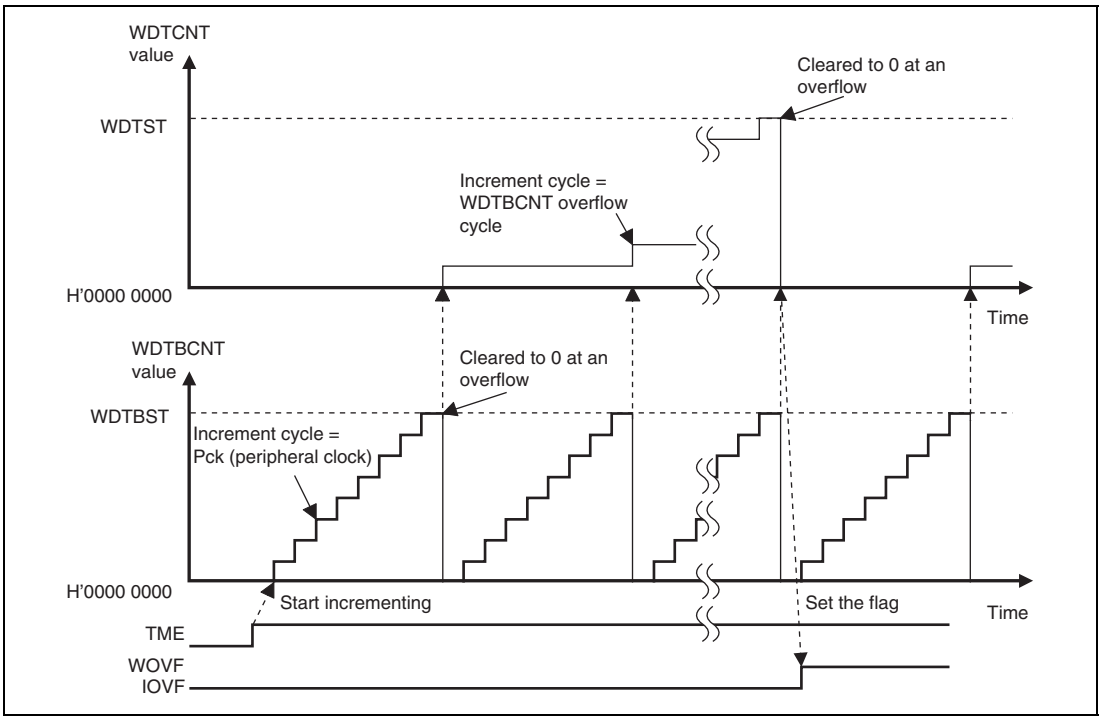


Figure 3.3 WDT Increment Operation (in Interval Timer Mode)

WDTBCNT is an 24-bit counter that is incremented by the peripheral clock. When the frequency of the peripheral clock clkp is tPck (ns), the WDTBCNT overflow cycle is obtained by:

$$2^{24}[\text{bit}] \times tPck[\text{ns}] = 16.78 \times tPck\ [\text{ms}]$$

WDTCNT is a 12-bit counter that is incremented by a WDTBCNT overflow. The interval between WDTCNT overflows is at its maximum when H'5A00_0000 has been written to the WDTST register. When the frequency of the peripheral clock clkp is tPck (ns), the maximum WDTCNT overflow cycle is obtained by:

$$2^{12}\ [\text{bit}] \times (16.78 \times tPck)\ [\text{ms}] = 68.73 \times tPck\ [\text{s}]$$

The interval between WDTCNT overflows is at its minimum when H'5A00_0001 has been written to the WDTST register. At this time, the WDTCNT overflow cycle is the same as the WDTBCNT overflow cycle.

For example, when the peripheral clock is running at 50 MHz so that tPck is 20 ns:

WDTBCNT overflow cycle: $16.78 \times 20 = 335.60$ [ms]

Maximum WDTCNT overflow cycle: $68.73 \times 20 = 1374.6$ [s]

3.4.5 Clearing WDT (CPU0 and CPU1) Counters

- WDTCNT is cleared by re-setting WDTST to the same value as was previously set in WDTST.
- WDTBCNT is cleared by re-setting WDTBST to the same values as is previously set in WDTBST.

An overflow will not occur by the above setting operation. However, if the above setting and an overflow are generated simultaneously, an overflow takes priority. At this time, the counter is cleared by the overflow.

3.5 Reset Timing

3.5.1 Power-On Reset by $\overline{\text{PRESET}}$ Pin

Since the PLL circuit is initialized when the LSI enters the power-on reset state, the power-on oscillation settling time needs to be secured. This means that a high level must not be input to the $\overline{\text{PRESET}}$ pin during the power-on oscillation settling time.

After the state on the $\overline{\text{PRESET}}$ pin input is changed from a low level to high level, the internal reset state continues until the reset holding time elapses. The reset holding time is equal to or more than 40 cycles of the peripheral clock (clkp) after the $\overline{\text{PRESET}}$ pin is negated.

(1) When the Power is Turned On

When the power is turned on, ensure that a low level is input to the $\overline{\text{PRESET}}$ pin. A low level input is also needed on the $\overline{\text{TRST}}$ pin to initialize the H-UDI.

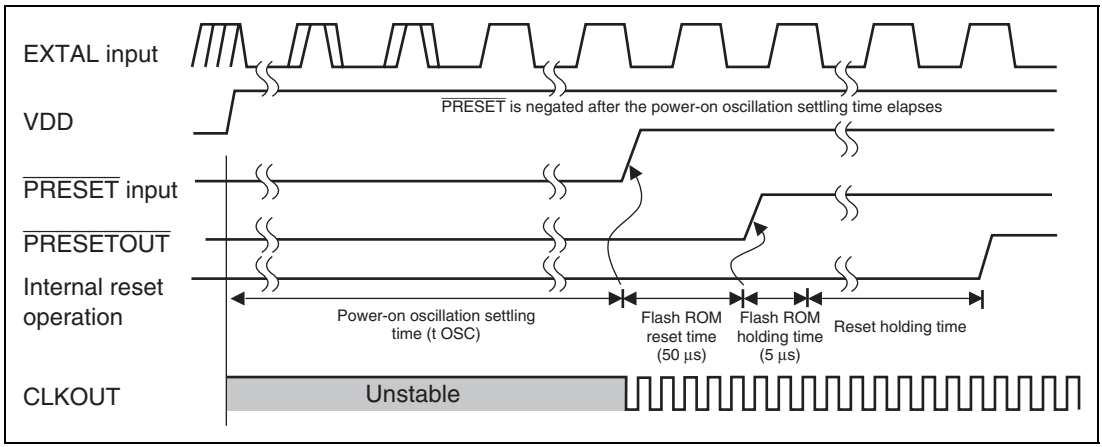


Figure 3.4 Free-Running Mode

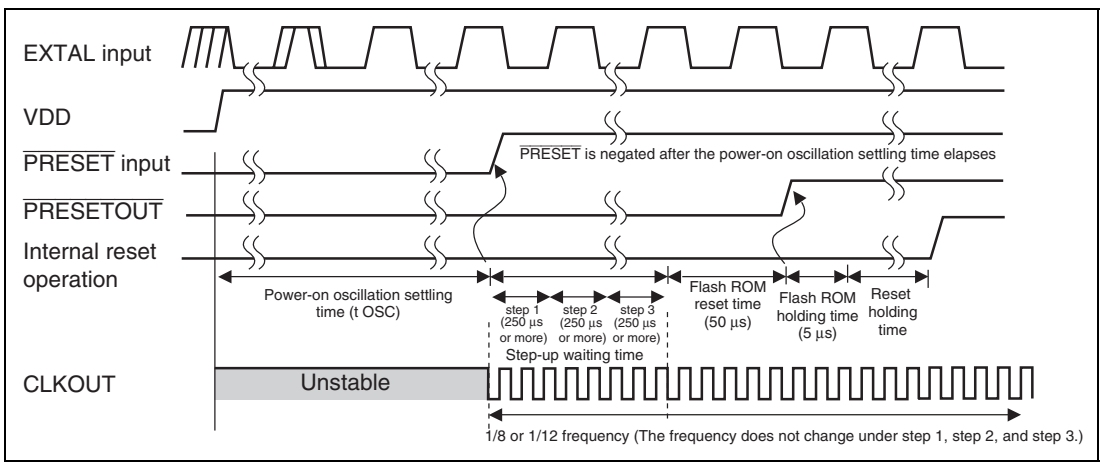


Figure 3.5 Step-Up Mode

Note: For frequency transition during step-up mode, see tables 8.9, 8.10 and 8.11, Frequency Change in Step-Up Mode, in section 8, Clock Pulse Generator (CPG).

(2) Timing of Reset by Internal Reset Source

Figure 3.6 shows the timing of a reset generated by an LSI internal reset source*. $\overline{\text{PRESETOUT}}$ is pulled low for 50 µs.

Note: The LSI is internally initialized by a power-on reset caused by a WDT overflow or a power-on reset, or by a manual reset caused by a WDT overflow or a CPU internal source.

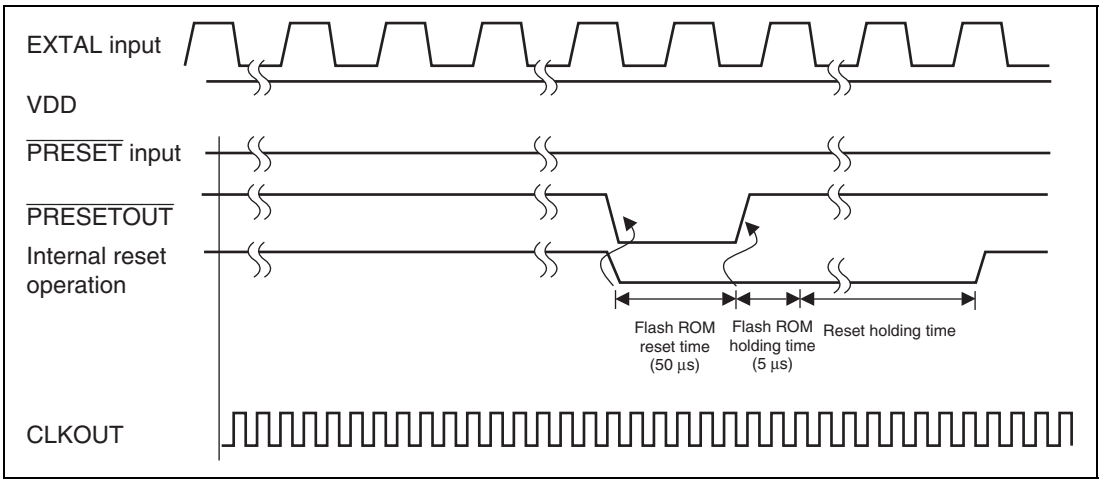


Figure 3.6 Reset by Internal Reset Source

(3) Timing of Reset by H-UDI Reset Source

Figure 3.7 shows the timing of a reset generated by an H-UDI reset source.

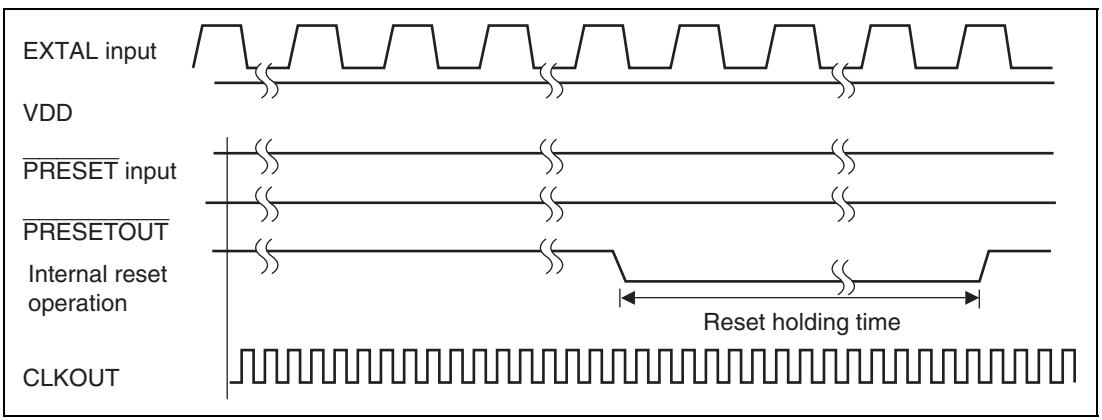


Figure 3.7 Reset by H-UDI Reset Source

Section 4 Memory Controller (DBSC3)

The external memory controller (DBSC3) supports various SDRAMs including DDR2-SDRAM and DDR3-SDRAM (referred to as SDRAM, hereinafter).

4.1 Features

The DBSC3 enables the maximum use of SDRAM bus bandwidth using the following functions:

- Multibank operation that improves the page hit rate.
- DDR2-SDRAM operation with burst length of 4 or 8 and DDR3-SDRAM operation with burst length of 8 that reduce the number of SDRAM command issues.
- Preceding execution of the bank precharge and activate commands for subsequent requests.

Table 4.1 shows the main functions of the DBSC3.

Table 4.1 DBSC3 Functions (common to all the SDRAM types)

Item	Function
Multibank supported	Supports 8-bank multibank operation.
Number of banks	Supports 4 banks (DDR2-SDRAM) and 8 banks (DDR2-SDRAM and DDR3-SDRAM)
External data bus width	16 bits
Preceding precharge/ activate functions	Determines the content of subsequent requests in a request queue and performs preceding precharge/activate processing for the bank to be accessed during an empty command cycle upon page-miss.
Operating modes	Burst length: DDR2-SDRAM: 8 (fixed) or 4 (fixed) DDR3-SDRAM: 8 (fixed) Burst type: Sequential (fixed) DLL OFF mode of DDR3-SDRAM is not supported.
Power-down mode	Supports self-refresh mode, power-down mode, SDRAM power buck-up mode, and deep standby mode. Does not support the partial self-refresh mode.

Item	Function
Timing setting	The following timing settings can be specified by the registers CAS latency, CAS write latency, ACT-READ/WRITE minimum period, PRE period, ACT-ACT/REF minimum period, ACT-PRE minimum period, ACT(A)-ACT(B) minimum period, 4-active-window minimum period, READ-PRE minimum period, write recovery period, READ-WRITE minimum period, WRITE-READ minimum period, REF-ACT/REF minimum period, CKE-Hi minimum period, CKE-Low minimum period, short calibration period. Only 0 is supported for additive latency (AL).
Address order (address translation function)	The addresses arranged in descending order are: row address, bank address, and column address. (Register settings can be used to determine how to divide the bank address.)
Memory to be connected	DDR3-SDRAM compliant with JEDEC. (The controller supports the connection of memory devices having capacities from 512 Mbits to 2 Gbits, with two devices connectable if the data bus width is 8 bits and one device connectable if the data bus width is 16 bits. SDRAM with a data bus width of 4 bits is not supported. The controller does not support write leveling.) DDR2-SDRAM compliant with JEDEC. (The controller supports the connection of memory devices having capacities from 256 Mbits to 2 Gbits, with two devices connectable if the data bus width is 8 bits and one device connectable if the data bus width is 16 bits. SDRAM with a data bus width of 4 bits is not supported.)
Refreshes	Average interval and the maximum post count are set by the register. If an empty cycle for request is found, preceding refresh can be performed.
Power-down operation	Two types of power-down modes. • Auto power-down mode in which a power-down mode is automatically entered when there are no accesses for a specific number of cycles. See section 4.2.40, Power-Down Configuration Register (DBPDNCNF). • Manual power-down mode in which a power-down mode is entered at the desired timing. See section 4.2.6, Manual Command-Issuing Register (DBCMD).

Item	Function
DDR3-SDRAM calibration operation	Two types of calibrations during operation (DDR3-SDRAM only). • Automatic calibration in which calibration is executed at auto-refresh. See section 4.2.37, DDR3-SDRAM Calibration Configuration Register (DBCALCNF), and section 4.2.38, DDR3-SDRAM Calibration Timing Register (DBCALTR). • Manual calibration in which calibration is executed at the desired timing. See section 4.2.6, Manual Command-Issuing Register (DBCMD).

4.1.1 External Pins

Table 4.2 shows the pin configuration of the DBSC3.

Table 4.2 Pin Configuration of the DBSC3

Pin Name*1	Function	I/O	Description
MCK0	Clock	Output	Clock output
$\overline{\text{MCK0}}$	Clock	Output	Clock output, MCK0 inverted clock output
MCKE	Clock enable	Output	CKE output signal
$\overline{\text{MCS}}$	Chip select	Output	Chip select output signal
$\overline{\text{MWE}}$	Write enable	Output	Write enable output signal
$\overline{\text{MRAS}}$	Row address strobe	Output	Row address strobe output signal
$\overline{\text{MCAS}}$	Column address strobe	Output	Column address strobe output signal
MA13 to MA0	Address	Output	Address output signals
MBA2, MBA1, MBA0	Bank address	Output	Bank address output signal
MDQ15 to MDQ0	Data	I/O	Data I/O signals
MDQS1, MDQS0	I/O data strobe	I/O	Data strobe I/O signals
$\overline{\text{MDQS1}}$, $\overline{\text{MDQS0}}$	I/O data strobe	I/O	Data strobe I/O signals or MDQS1 and MDQS0 inverted signals
MDM1, MDM0	Data mask	Output	Data mask output signals
MODT	ODT enable	Output	Enable output signal for ODT in SDRAM
MZQ	Calibration	I/O	Pin for calibration *2
$\overline{\text{MRESET}}$	DDR3-SDRAM reset	Output	Reset output for DDR3-SDRAM. Leave this pin open when DDR2-SDRAM is used.
$\overline{\text{MBKPRST}}$	Power-supply backup reset	Input	When this pin is brought low, the CKE pin is also fixed to low level. *3
SDBUP	Power-supply backup monitor	Input	When this pin is brought low, the $\overline{\text{MRESET}}$ pin is also fixed to low level. The value of BKUP bit in the status register reflects the level of signal. *4
SDSELF	DDR self-refresh notification	Output	Provides notification of the transition to DDR self-refresh mode*5

Pin Name*1	Function	I/O	Description
MVREFCA	Reference voltage input (for address and command pins)	Input	Input reference voltage. Should be fixed at GND.
MVREFDQ	Reference voltage input (for data pins)	Input	Input reference voltage. The VDD_DDR/2 voltage should be applied.

Notes: 1. Signal notation: An overbar is added to an active-low signal: $\overline{xxxx}$

2. MZQ: Connect this to VSS via a 120 Ω resistor (tolerance of 1% or better).
3. $\overline{MBKPRST}$: The power-supply backup reset pin does not have the pull-up function. Pull up the pin to the VDD_DDR power on the board. When neither the power-supply backup function nor deep standby mode is used, the pin can be directly connected to the VDD_DDR power supply.
4. SDBUP: The power-supply backup monitor pin does not have the pull-up function. Pull up the pin to the VDD_DDR power on the board when DDR3 is used. When DDR3 is used but neither the power-supply backup function nor deep standby mode is used, the pin can be directly connected to the VDD_DDR power supply. When DDR2 is used, it should be fixed at either the low or high level.
5. SDSELF: The DDR self-refresh notification pin is pulled up by the pin function controller (PFC) at reset. In power-supply backup state, the signal level is undefined. In deep standby mode, it retains the level that was output immediately before deep standby state is entered. For electrical characteristics such as output voltage, refer to table 41.5, DC Characteristics (3.3-V IO), in section 41, Electrical Characteristics.

4.1.2 Register Configuration

Table 4.3 (1) shows the DBSC3 register mapping, and table 4.3 (2) shows the register states in each operating mode. Register addresses are the sum of the DBSC3 register start address (DB_ADDR) and an offset (hex.) from that start address.

Registers other than particular ones are initialized by power-on reset.

Access the DBSC3 registers in units of 32 bits. Otherwise, a correct operation cannot be guaranteed.

The value of DB_ADDR differs depending on products. The value is H'FE80 0000 for this product.

Table 4.3 (1) DBSC3 Register Configuration

Register Name	Abbreviation	Address	Power-on Reset	Access Size (Bits)
SHwy version control register 0	DBSVCR0	DB_ADDR+H'000	H'7F08 4823	32
SHwy version control register 1	DBSVCR1	DB_ADDR+H'004	H'0000 0000	32
DBSC3 status register	DBSTATE	DB_ADDR+H'00C	H'0000 000x	32
SDRAM access enable register	DBACEN	DB_ADDR+H'010	H'0000 0000	32
Auto-refresh enable register	DBRFEN	DB_ADDR+H'014	H'0000 0000	32
Manual command-issuing register	DBCMD	DB_ADDR+H'018	H'0000 0000	32
Operation completion waiting register	DBWAIT	DB_ADDR+H'01C	H'0000 0000	32
SDRAM kind setting register	DBKIND	DB_ADDR+H'020	H'0000 0000	32
SDRAM configuration setting register	DBCONF	DB_ADDR+H'024	H'0000 0000	32
SDRAM timing register 0	DBTR0	DB_ADDR+H'040	H'0000 0000	32
SDRAM timing register 1	DBTR1	DB_ADDR+H'044	H'0000 0000	32
SDRAM timing register 2	DBTR2	DB_ADDR+H'048	H'0000 0000	32
SDRAM timing register 3	DBTR3	DB_ADDR+H'050	H'0000 0000	32
SDRAM timing register 4	DBTR4	DB_ADDR+H'054	H'0000 0000	32
SDRAM timing register 5	DBTR5	DB_ADDR+H'058	H'0000 0000	32
SDRAM timing register 6	DBTR6	DB_ADDR+H'05C	H'0000 0000	32
SDRAM timing register 7	DBTR7	DB_ADDR+H'060	H'0000 0000	32
SDRAM timing register 8	DBTR8	DB_ADDR+H'064	H'0000 0000	32
SDRAM timing register 9	DBTR9	DB_ADDR+H'068	H'0000 0000	32

Register Name	Abbreviation	Address	Power-on Reset	Access Size (Bits)
SDRAM timing register 10	DBTR10	DB_ADDR+H'06C	H'0000 0000	32
SDRAM timing register 11	DBTR11	DB_ADDR+H'070	H'0000 0000	32
SDRAM timing register 12	DBTR12	DB_ADDR+H'074	H'0000 0000	32
SDRAM timing register 13	DBTR13	DB_ADDR+H'078	H'0000 0000	32
SDRAM timing register 14	DBTR14	DB_ADDR+H'07C	H'0000 0000	32
SDRAM timing register 15	DBTR15	DB_ADDR+H'080	H'0000 0000	32
SDRAM timing register 16	DBTR16	DB_ADDR+H'084	H'0000 0000	32
SDRAM timing register 17	DBTR17	DB_ADDR+H'088	H'0000 0000	32
SDRAM timing register 18	DBTR18	DB_ADDR+H'08C	H'0000 0000	32
SDRAM timing register 19	DBTR19	DB_ADDR+H'090	H'0000 0000	32
SDRAM operation setting register	DBBL	DB_ADDR+H'0B0	H'0000 0000	32
DBSC3 operation adjustment register 0	DBADJ0	DB_ADDR+H'0C0	H'0000 0000	32
DBSC3 operation adjustment register 1	DBADJ1	DB_ADDR+H'0C4	H'0000 0000	32
DBSC3 operation adjustment register 2	DBADJ2	DB_ADDR+H'0C8	H'0000 0000	32
Refresh configuration register 0	DBRFCNF0	DB_ADDR+H'0E0	H'0000 0000	32
Refresh configuration register 1	DBRFCNF1	DB_ADDR+H'0E4	H'0000 0000	32
Refresh configuration register 2	DBRFCNF2	DB_ADDR+H'0E8	H'0000 0000	32
DDR3-SDRAM calibration configuration register	DBCALCNF	DB_ADDR+H'0F4	H'0000 0000	32
DDR3-SDRAM calibration timing register	DBCALTR	DB_ADDR+H'0F8	H'0000 0000	32
ODT operation setting register	DBRNK0	DB_ADDR+H'100	H'0000 0000	32
Power-down configuration register	DBPDNCNF	DB_ADDR+H'180	H'0000 0000	32
DDR-PHY unit control register 0	DBPDCNT0	DB_ADDR+H'200	H'0000 0000	32
DDR-PHY unit control register 1	DBPDCNT1	DB_ADDR+H'204	H'0000 0000	32
DDR-PHY unit control register 2	DBPDCNT2	DB_ADDR+H'208	H'0000 0000	32
DDR-PHY unit control register 3	DBPDCNT3	DB_ADDR+H'20C	H'0000 0000	32
DDR-PHY unit lock register	DBPDLCK	DB_ADDR+H'280	H'0000 0000	32
DDR-PHY unit register address	DBPDRGA	DB_ADDR+H'290	H'0000 0000	32
DDR-PHY unit register access	DBPDRGD	DB_ADDR+H'2A0	H'0000 0000	32
Bus control unit 0 control register 0	DBBS0CNT0	DB_ADDR+H'300	H'0000 0000	32
Bus control unit 0 control register 1	DBBS0CNT1	DB_ADDR+H'304	H'0000 0000	32

Table 4.3 (2) Register States in Each Operating Mode

Register Abbreviation	Power-On Reset	Manual Reset	Sleep	Software Standby	Deep Standby
DBSVCR0	Initialized	Retained	Retained	Retained	Initialized
DBSVCR1	Initialized	Retained	Retained	Retained	Initialized
DBSTATE	Initialized	Retained	Retained	Retained	Initialized
DBACEN	Initialized	Retained	Retained	Retained	Initialized
DBRFEN	Initialized	Retained	Retained	Retained	Initialized
DBCMD	Initialized	Retained	Retained	Retained	Initialized
DBWAIT	Initialized	Retained	Retained	Retained	Initialized
DBKIND	Initialized	Retained	Retained	Retained	Initialized
DBCONF	Initialized	Retained	Retained	Retained	Initialized
DBTR0	Initialized	Retained	Retained	Retained	Initialized
DBTR1	Initialized	Retained	Retained	Retained	Initialized
DBTR2	Initialized	Retained	Retained	Retained	Initialized
DBTR3	Initialized	Retained	Retained	Retained	Initialized
DBTR4	Initialized	Retained	Retained	Retained	Initialized
DBTR5	Initialized	Retained	Retained	Retained	Initialized
DBTR6	Initialized	Retained	Retained	Retained	Initialized
DBTR7	Initialized	Retained	Retained	Retained	Initialized
DBTR8	Initialized	Retained	Retained	Retained	Initialized
DBTR9	Initialized	Retained	Retained	Retained	Initialized
DBTR10	Initialized	Retained	Retained	Retained	Initialized
DBTR11	Initialized	Retained	Retained	Retained	Initialized
DBTR12	Initialized	Retained	Retained	Retained	Initialized
DBTR13	Initialized	Retained	Retained	Retained	Initialized
DBTR14	Initialized	Retained	Retained	Retained	Initialized
DBTR15	Initialized	Retained	Retained	Retained	Initialized
DBTR16	Initialized	Retained	Retained	Retained	Initialized
DBTR17	Initialized	Retained	Retained	Retained	Initialized
DBTR18	Initialized	Retained	Retained	Retained	Initialized
DBTR19	Initialized	Retained	Retained	Retained	Initialized
DBBL	Initialized	Retained	Retained	Retained	Initialized

Register Abbreviation	Power-On Reset	Manual Reset	Sleep	Software Standby	Deep Standby
DBADJ0	Initialized	Retained	Retained	Retained	Initialized
DBADJ1	Initialized	Retained	Retained	Retained	Initialized
DBADJ2	Initialized	Retained	Retained	Retained	Initialized
DBRFCNF0	Initialized	Retained	Retained	Retained	Initialized
DBRFCNF1	Initialized	Retained	Retained	Retained	Initialized
DBRFCNF2	Initialized	Retained	Retained	Retained	Initialized
DBCALCNF	Initialized	Retained	Retained	Retained	Initialized
DBCALTR	Initialized	Retained	Retained	Retained	Initialized
DBRNK0	Initialized	Retained	Retained	Retained	Initialized
DBPDNCNF	Initialized	Retained	Retained	Retained	Initialized
DBPDCNT0	Initialized	Retained	Retained	Retained	Initialized
DBPDCNT1	Initialized	Retained	Retained	Retained	Initialized
DBPDCNT2	Initialized	Retained	Retained	Retained	Initialized
DBPDCNT3	Initialized	Retained	Retained	Retained	Initialized
DBPDLCK	Initialized	Retained	Retained	Retained	Initialized
DBPDRGA	Initialized	Retained	Retained	Retained	Initialized
DBPDRGD	Undefined	Retained	Retained	Retained	Undefined
DBBS0CNT0	Initialized	Retained	Retained	Retained	Initialized
DBBS0CNT1	Initialized	Retained	Retained	Retained	Initialized

4.2 Register Descriptions

- Legend:

Initial value: Register value after a reset

R/W: Readable/writable. The written value can be read.

R: Read-only. The write value should always be the initial value.

W: Write-only. Reading this bit is prohibited. When the bit is reserved, the write value should always be the initial value.

All access to registers is made in longword units.

Write the initial value to the reserved bits.

- For real number x, ceil(x) represents the smallest whole number that is not smaller than x.
- For real number x, floor(x) represents the largest whole number that is not greater than x.

4.2.1 SHwy Version Control Register 0 (DBSVCR0)

The SHwy version control register 0 (DBSVCR0) is indispensable for all the target modules supporting SHwy. DBSVCR0 contains the DBSC3 address map and module identifier.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TOP_MB								BOT_MB							
Initial value:	0	1	1	1	1	1	1	1	0	0	0	0	1	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VC_ID															
Initial value:	0	1	0	0	1	0	0	0	0	0	1	0	0	0	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	TOP_MB	H'7F	R	Memory Block Upper Limit Address (upper 8 bits) These bits indicate the upper limit address of the address space assigned as the SDRAM area. The set value is specific to the product type.
23 to 16	BOT_MB	H'08	R	Memory Block Lower Limit Address (upper 8 bits) These bits indicate the lower limit address of the address area assigned as the SDRAM area. The set value is specific to the product type.
15 to 0	VC_ID	H'4823	R	DBSC3 Module Identifier These bits indicate the identifier specific to the DBSC3 module.

4.2.2 SHwy Version Control Register 1 (DBSVCR1)

The SHwy version control register 1 (DBSVCR1) is indispensable for all the target modules supporting SHwy.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	VC_VERS															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	IN ACTIVE	—	—	BAD _OPC	—	—	BAD_ ADDR	ERR _SNT	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R/W	R	R	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	VC_VERS	H'0000	R	DBSC3 Module Version Information These bits indicate the DBSC3 module version information. These bits cannot be modified.
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	INACTIVE	0	R/W	Error Source: Data Access to SDRAM in SDRAM Disable State This bit is set to 1 when a data access is made to the SDRAM when the ACEN bit in SDRAM access enable register (DBACEN) is 0 (SDRAM access is disabled). This bit is cleared to 0 when 0 is written to.
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	BAD_OPC	0	R/W	Error Source: Unsupported SHwy Command This bit is set to 1 when an unsupported SHwy command is used to access the DBSC3 control register area or memory area. This bit is cleared to 0 when 0 is written to.
4, 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2	BAD_ADDR	0	R/W	Error Source: Access to Undefined Register This bit is set to 1 when an address is accessed where no control register is assigned in the DBSC3 control register area. This bit is cleared to 0 when 0 is written to.
1	ERR_SNT	0	R/W	Error Bit: SHwy Error Response Issue This bit is set to 1 when an error response is returned from DBSC3 to SHwy. The error source can be analyzed using the INACTIVE, BAD_OPC, and BAD_ADDR register values. This bit is cleared to 0 when 0 is written to.
0	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Notes: 1. If an address other than in the DBSC3 control register area or memory area is accessed, operations are not guaranteed.
2. When this register is written to, the values of the INACTIVE, BAD_OPC, BAD_ADDR, and ERR_SNT bits are updated to the written values.

4.2.3 DBSC3 Status Register (DBSTATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ENDN	—	—	—	—	—	—	—	BKUP
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	ENDN	0	R	Endian Monitor Bit Enables monitoring of the current endian. 0: Big endian 1: Little endian
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	BKUP	Undefined	R	Power-Supply Backup Monitor Bit This bit permits reflecting the level of the input signal on the SDBUP pin. Enables monitoring of the power-supply backup status. 0: The low level is being input to the SDBUP pin. 1: The high level is being input to the SDBUP pin.

4.2.4 SDRAM Access Enable Register (DBACEN)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ACEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	ACEN	0	R/W	SDRAM Access Enable Bit This bit enables data access to SDRAM. Make data access to SDRAM after setting this bit to 1. When this bit is 0, do not access the SDRAM area. 0: Disables SDRAM access. 1: Enables SDRAM access.

Notes: 1. When a request to access the SDRAM area occurs while the ACEN bit is 0, the DBSC3 returns an error response.
2. When making the setting to disable access (writing 0 to this register), a Precharge All or Precharge command for the SDRAM may be issued automatically.

4.2.5 Auto-Refresh Enable Register (DBRFEN)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ARFEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	ARFEN	0	R/W	Auto-Refresh Enable Bit This bit starts or stops the auto-refresh function. Writing 1 to this bit resets the refresh counter (refresh history) in the DBSC3 and starts the auto-refresh function. While this bit is 1, the DBSC3 issues a refresh command at regular intervals. The refresh cycle time and other settings depend on the values which exist in refresh configuration registers 0 to 2 (DBRFCNF0 to DBRFCNF2) when 1 is written to this bit. 0: Stops the auto-refresh function. 1: Starts the auto-refresh function.

4.2.6 Manual Command-Issuing Register (DBCMD)

The manual command-issuing register (DBCMD) is used to issue the required commands for the sequence of initializing the SDRAM and of transitions to and from the self-refresh mode. The command corresponding to the OPC bits is issued once as a result of writing to this register. For instance, issuing the refresh command twice requires that ″001100″ be written to the OPC bits twice. Writing to this register must not proceed while access to SDRAM is enabled (ACEN = 1 in the DBACEN register). The timing with which an operation is complete (i.e. the timing with which the specified SDRAM command is output to SDRAM from DBSC3) may be later than the response of DBSC3 to writing to this register. Reading from the DBWAIT register will result in the actual output of the specified SDRAM command to SDRAM.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	OPC						—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ARG															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
29 to 24	OPC	000000	R/W	Operation Code Bits These bits specify the type of command to be issued. Refer to table 4.4. If Wait is specified for this bit, then valid SDRAM commands are not output, and no processing is performed except reserving the time until the next operation.
23 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	ARG	H'0000	R/W	Parameter Bits The meaning can differ according to the operation code indicated by the OPC bits. When OPC indicates ModeRegisterSet (MRS0 to MRS3), the ARG bits specify the value to be issued on the address pins (MA) of SDRAM. The value should be specified in bits 0 to 12 in ARG and bits 13 to 15 should be fixed at 0. When OPC is any other value, the ARG bits specify the minimum interval to issuing of the next command in SDRAM cycles. When ARG = 0, however, the values default to those given in ″Interval″ column of table 4.4.

When this register is used to issue command, the issuing of a subsequent SDRAM command is delayed by a certain period of time from the moment at which each operation is completed (i.e. from the time when the specified SDRAM command is output to SDRAM). This facilitates securing of the required amounts of time between commands in the issuing of multiple consecutive commands.
These periods are given in the "Interval (Number of SDRAM Cycles)" column of table 4.4. They can also be customized by using the ARG bits (except in cases where the OPC bits indicate MRS0 to MRS3).

Table 4.4 Manual Command Issue Functions

OPC	Code	Operation	Interval (Number of SDRAM Cycles)	ARG Function
00 0000	Wait	Issue "Device Deselected" (and insert waiting time)	4	Customizing the interval (for ARG = 0, the value is that given at left)
00 0010	ZQCS	Issue "ZQ Calibration Short" (DDR3-SDRAM only)	4	
00 0011	ZQCL	Issue "ZQ Calibration Long" (DDR3-SDRAM only)	4	
00 1011	PreA	Issue "Precharge All"	TRPA	
00 1100	Ref	Issue "Refresh"	TRFC	
01 0000	PDEn	Power Down Entry	4	
01 0001	PDXt	Power Down Exit	4	
01 1000	SREn	Self-Refresh Entry	4	
01 1001	SRXt	Self-Refresh Exit	TRFC	
10 0000	RstL	Set $\overline{\text{MRESET}}$ Pin to Low (DDR3-SDRAM only)	4	
10 0001	RstH	Set $\overline{\text{MRESET}}$ Pin to High (DDR3-SDRAM only)	4	
10 1000	MRS0	Issue "ModeRegisterSet" (for MRS/MR0)"	TMOD	Specifying the setting for a mode register in SDRAM
10 1001	MRS1	Issue "ModeRegisterSet" (for EMRS1/MR1)	TMOD	
10 1010	MRS2	Issue "ModeRegisterSet" (for EMRS2/MR2)	TMOD	
10 1011	MRS3	Issue "ModeRegisterSet" (for EMRS3/MR3)	TMOD	

Notes: 1. TRPA, TRFC and TMOD in the ″Interval″ column of table 4.4 indicate the bits in the timing registers. The intervals in these cases are determined by the corresponding register settings.

2. Only write to this register after having disabled SDRAM access (ACEN in DBACEN = 0).

3. Only write to this register after having stopped the auto-refresh function (ARFEN in DBRFEN register = 0). However, sequences described in section 4.3, DBSC3 Operation, are not limited to these. If OPC = Wait is written to this register during automatic refresh operations, the auto-refresh function may issue a refresh command during the time secured for the Wait command.

4.2.7 Operation Completion Waiting Register (DBWAIT)

When this register is read, the read value is returned to the CPU only after all commands that have been specified by using the DBCMD register up to that time have been issued. Thus, the CPU operation and SDRAM command issuing can be synchronized.

This register can be used to guarantee correctness for the relationship between the timing with which SDRAM commands are issued by DBSC3 and timing that is not managed by DBSC3 (e.g. clock control).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	WAIT
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	WAIT	0	R	Operation Completion Waiting Bits The value is meaningless. This bit is always read as 0.

4.2.8 SDRAM Kind Setting Register (DBKIND)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	DDCG		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	DDCG	000	R/W	SDRAM Kind Bits These bits can set the kind of SDRAM. Set the designated value according to the type. 000: Initial value (when using the DBSC3, set these bits to either of the following values (value for DDR2 or DDR3)). 101: DDR2-SDRAM 111: DDR3-SDRAM Other settings are prohibited.

Notes: 1. This register must only be written from within the initialization sequence.

2. Writing to this register should only be performed when the following conditions are met.
   - Access to SDRAM is disabled (i.e. the ACEN bit in the DBACEN register is 0)
   - Auto-refresh is disabled (i.e. the ARFEN bit in the DBRFEN register is 0)

4.2.9 SDRAM Configuration Setting Register (DBCONF)

This register sets the memory configuration to be used.
Refer to section 4.4, Setting the SDRAM Configuration Setting Register, for details on the memory configurations supported by DBSC3. For details on the relationship between the external pins of the SDRAM and the logical addresses of this LSI, refer to section 4.5, Relation between External Pins and Logical Addresses.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	AWRW0					—	—	—	AWRK0	—	—	AWBK0	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	AWCL0				—	—	—	—	—	—	DW0	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28 to 24	AWRW0	00000	R/W	Row Address Bit Width Setting Bits These bits specify the width, in bits, of row addresses. 00000: Initial value (set these bits to one of the following values (12 to 14 bits)). 01100: 12 bits 01101: 13 bits 01110: 14 bits Other settings are prohibited.
23 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
20	AWRK0	0	R/W	Number of Ranks Setting Bit Specifies the number of ranks. This LSI supports only rank count = 1 (rank count = 1 indicates that a single memory device is connected to the same data signal line). 0: 1 rank 1: Setting prohibited
19, 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17, 16	AWBK0	00	R/W	Number of Banks Setting Bits These bits specify the number of banks. 00: Initial value (set these bits to either of the following values (4 or 8 banks)). 10: 4 banks 11: 8 banks Other settings are prohibited.
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 8	AWCL0	0000	R/W	Column Address Bit Width Setting Bits These bits specify the width, in bits, of column addresses. 0000: Initial value (set these bits to either of the following values (9 or 10 bits)). 1001: 9 bits 1010: 10 bits Other settings are prohibited.
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1, 0	DW0	00	R/W	External Data Bus Width Setting Bits These bits specify the width of the external bus. Only the 16-bit data bus width is supported in this LSI. 00: Initial value (set these bits to the following value (16 bits)). 01: 16 bits Other settings are prohibited.

Notes: 1. Memory Configuration Supported
16-bit bus configuration connected with one 16-bit wide SDRAM module or two 8-bit wide SDRAM modules
2. This register must only be written from within the initialization sequence.
3. Writing to this register should only be performed when the following conditions are met:
- Access to SDRAM is disabled (i.e. the ACEN bit in the DBACEN register is 0)
- Auto-refresh is disabled (i.e. the ARFEN bit in the DBRFEN register is 0)
4. Set the value defined for each SDRAM type to be connected.

4.2.10 SDRAM Timing Register 0 (DBTR0)

SDRAM timing register 0 (DBTR0) is used to set a timing parameter for the SDRAM.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	CL			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	CL	0000	R/W	CAS Latency Setting Bits These bits are for setting the CAS latency of the SDRAM. The CL bits should be set to 3, 4, 5, or 6 cycles for DDR2-SDRAM and to 5 or 6 cycles for DDR3-SDRAM. 0000: Initial value (when using the DBSC3, set these bits to one of the following values (3 to 6 cycles)). 0011: 3 cycles 0100: 4 cycles 0101: 5 cycles 0110: 6 cycles Other settings are prohibited.

Notes: 1. The setting is in cycles of the SDRAM operating clock.

2. Writing to this register should only be performed when the following conditions are met.
   - Access to SDRAM is disabled (i.e. the ACEN bit in the DBACEN register is 0)
   - Auto-refresh is disabled (i.e. the ARFEN bit in the DBRFEN register is 0)
3. Set the value defined for each SDRAM type to be connected.

4.2.11 SDRAM Timing Register 1 (DBTR1)

The SDRAM timing register 1 (DBTR1) is used to set a timing parameter for the SDRAM.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	CWL			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	CWL	0000	R/W	CAS Write Latency Setting Bits These bits are for setting the CAS write latency of the SDRAM. The CWL bits should be set to a value of (cycle count set by CL bits − 1) for DDR2-SDRAM. The CWL bits should be set to 5 or 6 cycles for DDR3-SDRAM. However, a setting of CL = 5 and CWL = 6 is not allowed. 0000: Initial value (when using the DBSC3, set these bits to one of the following values (2 to 6 cycles)). 0010: 2 cycles 0011: 3 cycles 0100: 4 cycles 0101: 5 cycles 0110: 6 cycles Other settings are prohibited.

Notes: 1. The setting is in cycles of the SDRAM operating clock.
2. Writing to this register should only be performed when the following conditions are met.
   - Access to SDRAM is disabled (i.e. the ACEN bit in the DBACEN register is 0)
   - Auto-refresh is disabled (i.e. the ARFEN bit in the DBRFEN register is 0)
3. Set the value defined for each SDRAM type to be connected.

4.2.12 SDRAM Timing Register 2 (DBTR2)

The SDRAM timing register 2 (DBTR2) is used to set a timing parameter for the SDRAM.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	AL			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	AL	0000	R/W	Additive Latency Setting Bits These bits are for setting the additive latency of the SDRAM. This LSI supports only AL = 0. 0000: 0 cycles Other settings are prohibited.

Notes: 1. The setting is in cycles of the SDRAM operating clock.

2. Writing to this register should only be performed when the following conditions are met.
   - Access to SDRAM is disabled (i.e. the ACEN bit in the DBACEN register is 0)
   - Auto-refresh is disabled (i.e. the ARFEN bit in the DBRFEN register is 0)

4.2.13 SDRAM Timing Register 3 (DBTR3)

The SDRAM timing register 3 (DBTR3) is used to set a timing parameter for the SDRAM.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	TRCD			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	TRCD	0000	R/W	ACT-READ/WRITE Interval Setting Bits These bits indicate the minimum interval from an ACT command to a READ/WRITE command. 0000: Initial value (when using the DBSC3, set these bits to one of the following values (3 to 10 cycles)). 0011: 3 cycles 0100: 4 cycles : 1010: 10 cycles Other settings are prohibited.

Notes: 1. The setting is in cycles of the SDRAM operating clock.

2. Writing to this register should only be performed when the following conditions are met.
   - Access to SDRAM is disabled (i.e. the ACEN bit in the DBACEN register is 0)
   - Auto-refresh is disabled (i.e. the ARFEN bit in the DBRFEN register is 0)
3. Set the value defined for each SDRAM type to be connected.

4.2.14 SDRAM Timing Register 4 (DBTR4)

The SDRAM timing register 4 (DBTR4) is used to set a timing parameter for the SDRAM.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	TRPA			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	TRP			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
19 to 16	TRPA	0000	R/W	PREA Time Setting Bits These bits indicate the minimum interval from a PRE ALL (precharge all banks) command to an ACT/REF command. The value set in these bits must be greater than or equal to that in the TRP bits. 0000: Initial value (when using the DBSC3, set these bits to one of the following values (3 to 10 cycles)). 0011: 3 cycles 0100: 4 cycles : 1010: 10 cycles Other settings are prohibited.
15 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	TRP	0000	R/W	PRE Time Setting Bits These bits indicate the minimum interval from a PRE (precharge) command to an ACT/REF command. 0000: Initial value (when using the DBSC3, set these bits to one of the following values (3 to 10 cycles)). 0011: 3 cycles 0100: 4 cycles : 1010: 10 cycles Other settings are prohibited.

Notes: 1. The setting is in cycles of the SDRAM operating clock.
2. The following condition must be satisfied: TRPA ≥ TRP, TRC-TRP ≤ 32.
3. Writing to this register should only be performed when the following conditions are met.
- Access to SDRAM is disabled (i.e. the ACEN bit in the DBACEN register is 0)
- Auto-refresh is disabled (i.e. the ARFEN bit in the DBRFEN register is 0)
4. Set the value defined for each SDRAM type to be connected.

4.2.15 SDRAM Timing Register 5 (DBTR5)

The SDRAM timing register 5 (DBTR5) is used to set a timing parameter for the SDRAM.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	TRC					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5 to 0	TRC	000000	R/W	ACT-ACT/REF Interval Setting Bits These bits indicate the minimum interval from one ACT command to another ACT command (for the same bank) or to a REF command. 000000: Initial value (when using the DBSC3, set these bits to one of the following values (10 to 38 cycles)). 001010: 10 cycles 001011: 11 cycles : 100110: 38 cycles Other settings are prohibited.

Notes: 1. The setting is in cycles of the SDRAM operating clock.
2. The following condition must be satisfied: TRC-TRP $\leq$ 32.
3. Writing to this register should only be performed when the following conditions are met.
   - Access to SDRAM is disabled (i.e. the ACEN bit in the DBACEN register is 0)
   - Auto-refresh is disabled (i.e. the ARFEN bit in the DBRFEN register is 0)
4. Set the value defined for each SDRAM type to be connected.

4.2.16 SDRAM Timing Register 6 (DBTR6)

The SDRAM timing register 6 (DBTR6) is used to set a timing parameter for the SDRAM.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	TRAS					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5 to 0	TRAS	000000	R/W	ACT-PRE Interval Setting Bits These bits indicate the minimum interval from an ACT command to a PRE command. 000000: Initial value (when using the DBSC3, set these bits to one of the following values (7 to 28 cycles)). 000111: 7 cycles 001000: 8 cycles : 011100: 28 cycles Other settings are prohibited.

Notes: 1. The setting is in cycles of the SDRAM operating clock.

2. Writing to this register should only be performed when the following conditions are met.
   - Access to SDRAM is disabled (i.e. the ACEN bit in the DBACEN register is 0)
   - Auto-refresh is disabled (i.e. the ARFEN bit in the DBRFEN register is 0)
3. Set the value defined for each SDRAM type to be connected.

4.2.17 SDRAM Timing Register 7 (DBTR7)

The SDRAM timing register 7 (DBTR7) is used to set a timing parameter for the SDRAM.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	TRRD			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	TRRD	0000	R/W	ACT(A)-ACT(B) Interval Setting Bits These bits indicate the minimum interval between ACT commands issued for different banks. 0000: Initial value (when using the DBSC3, set these bits to one of the following values (2 to 5 cycles)). 0010: 2 cycles 0011: 3 cycles : 0101: 5 cycles Other settings are prohibited.

Notes: 1. The setting is in cycles of the SDRAM operating clock.

2. Writing to this register should only be performed when the following conditions are met.
   - Access to SDRAM is disabled (i.e. the ACEN bit in the DBACEN register is 0)
   - Auto-refresh is disabled (i.e. the ARFEN bit in the DBRFEN register is 0)
3. Set the value defined for each SDRAM type to be connected.

4.2.18 SDRAM Timing Register 8 (DBTR8)

The SDRAM timing register 8 (DBTR8) is used to set a timing parameter for the SDRAM.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TFAW							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	TFAW	H'00	R/W	4 Activate Window Length Setting Bits These bits indicate the length of the 4 activate window. 00000000: Initial value (when using the DBSC3, set these bits to one of the following values (8 to 24 cycles)). 00001000: 8 cycles 00001001: 9 cycles : 00011000: 24 cycles Other settings are prohibited.

Notes: 1. The setting is in cycles of the SDRAM operating clock.
2. The following condition must be satisfied: TFAW $\geq 4 \times$ TRRD.
3. Writing to this register should only be performed when the following conditions are met.
   - Access to SDRAM is disabled (i.e. the ACEN bit in the DBACEN register is 0)
   - Auto-refresh is disabled (i.e. the ARFEN bit in the DBRFEN register is 0)
4. Set the value defined for each SDRAM type to be connected.

4.2.19 SDRAM Timing Register 9 (DBTR9)

The SDRAM timing register 9 (DBTR9) is used to set a timing parameter for the SDRAM.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	TRDPR			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	TRDPR	0000	R/W	READ-PRE Interval Setting Bits These bits indicate the minimum interval from a READ command to a PRE command. 0000: Initial value (when using the DBSC3, set these bits to one of the following values (2 to 8 cycles)). 0010: 2 cycles 0011: 3 cycles : 1000: 8 cycles Other settings are prohibited.

Notes: 1. The setting is in cycles of the SDRAM operating clock.
2. The following condition must be satisfied: TRDPR ≥ BL/2.
3. If the SDRAM in use conforms to the JEDEC standard, the value of the bits can be calculated from the following formula.
TRDPR = BL/2 + max {2, ceil(tRTP / tCK)} – 2
4. Writing to this register should only be performed when the following conditions are met.
- Access to SDRAM is disabled (i.e. the ACEN bit in the DBACEN register is 0)
- Auto-refresh is disabled (i.e. the ARFEN bit in the DBRFEN register is 0)

4.2.20 SDRAM Timing Register 10 (DBTR10)

The SDRAM timing register 10 (DBTR10) is used to set a timing parameter for the SDRAM.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	TWR			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	TWR	0000	R/W	Write-Recovery Period Setting Bits These bits indicate the write-recovery period. 0000: Initial value (when using the DBSC3, set these bits to one of the following values (2 to 12 cycles)). 0010: 2 cycles 0011: 3 cycles : 1100: 12 cycles Other settings are prohibited.

Notes: 1. The setting is in cycles of the SDRAM operating clock.

2. Writing to this register should only be performed when the following conditions are met.
   - Access to SDRAM is disabled (i.e. the ACEN bit in the DBACEN register is 0)
   - Auto-refresh is disabled (i.e. the ARFEN bit in the DBRFEN register is 0)
3. Set the value defined for each SDRAM type to be connected.

4.2.21 SDRAM Timing Register 11 (DBTR11)

The SDRAM timing register 11 (DBTR11) is used to set a timing parameter for the SDRAM.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	TRDWR					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5 to 0	TRDWR	000000	R/W	READ-WRITE Interval Setting Bits These bits indicate the minimum interval from a READ command to a WRITE command.*[2] 000000: Initial value (when using the DBSC3, set these bits to one of the following values (4 to 15 cycles)). 000100: 4 cycles 000101: 5 cycles : 001111: 15 cycles Other settings are prohibited.

Notes: 1. The setting is in cycles of the SDRAM operating clock.

2. The following condition must be satisfied.

MODT Pin Usage	DDR2-SDRAM	DDR3-SDRAM
MODT pin is used	TRDWR ≥ BL/2 + 4	TRDWR ≥ CL – CWL + BL/2 + 4
MODT pin is not used	TRDWR ≥ BL/2 + 2	TRDWR ≥ CL – CWL + BL/2 + 2

3. Writing to this register should only be performed when the following conditions are met.
   - Access to SDRAM is disabled (i.e. the ACEN bit in the DBACEN register is 0)
   - Auto-refresh is disabled (i.e. the ARFEN bit in the DBRFEN register is 0)

4.2.22 SDRAM Timing Register 12 (DBTR12)

The SDRAM timing register 12 (DBTR12) is used to set a timing parameter for the SDRAM.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	TWRRD					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5 to 0	TWRRD	000000	R/W	WRITE-READ Interval Setting Bits These bits indicate the minimum interval from a WRITE command to a READ command.*[2] 000000: Initial value (when using the DBSC3, set these bits to one of the following values (6 to 18 cycles)). 000110: 6 cycles 000111: 7 cycles : 010010: 18 cycles Other settings are prohibited.

Notes: 1. The setting is in cycles of the SDRAM operating clock.

2. DDR2-SDRAM: TWRRD ≥ CL – 1 + BL/2.
   DDR3-SDRAM: TWRRD ≥ CWL + BL/2

3. If the SDRAM in use conforms to the JEDEC standard, the value of the bits can be calculated from the following formula.
   TWRRD = CWL + BL/2 + ceil(tWTR / tCK)

4. Writing to this register should only be performed when the following conditions are met.
   - Access to SDRAM is disabled (i.e. the ACEN bit in the DBACEN register is 0)
   - Auto-refresh is disabled (i.e. the ARFEN bit in the DBRFEN register is 0)

4.2.23 SDRAM Timing Register 13 (DBTR13)

The SDRAM timing register 13 (DBTR13) is used to set a timing parameter for the SDRAM.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TRFC							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	TRFC	H'00	R/W	REF-ACT/REF Interval Setting Bits These bits indicate the minimum interval from a REF (refresh) command to an ACT/REF command. H'00: Initial value (when using the DBSC3, set these bits to one of the following values (15 to 255 cycles)). H'0F: 15 cycles H'10: 16 cycles : H'FF: 255 cycles Other settings are prohibited.

Notes: 1. The setting is in cycles of the SDRAM operating clock.
2. Writing to this register should only be performed when the following conditions are met.
- Access to SDRAM is disabled (i.e. the ACEN bit in the DBACEN register is 0)
- Auto-refresh is disabled (i.e. the ARFEN bit in the DBRFEN register is 0)

3. Set the value defined for each SDRAM type to be connected.

4.2.24 SDRAM Timing Register 14 (DBTR14)

The SDRAM timing register 14 (DBTR14) is used to set a timing parameter for the SDRAM.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	TCKEHDLL							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TCKEH							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	TCKEHDLL	H'00	R/W	CKEH (DLL-LOCK) Period Setting Bits These bits indicate the minimum interval from the time the CKE signal goes high until the issuing of a further valid command (that requires the locked DLL). H'00: Initial value (when using the DBSC3, set these bits to one of the following values (2 to 15 cycles)). H'02: 2 cycles H'03: 3 cycles : H'0F: 15 cycles Other settings are prohibited.
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	TCKEH	H'00	R/W	CKEH Period Setting Bits These bits indicate the minimum interval from the time the CKE signal goes high until the issuing of a further valid command. The following condition must be satisfied: TCKEH ≤ TCKEHDLL H'00: Initial value (when using the DBSC3, set these bits to one of the following values (2 to 15 cycles)). H'02: 2 cycles H'03: 3 cycles : H'0F: 15 cycles Other settings are prohibited.

Notes: 1. The setting is in cycles of the SDRAM operating clock.

2. When the power-down mode is to be used, this register must be set accordingly. If the SDRAM in use conforms to the JEDEC standard, the value of the bits can be calculated from the following formula.
   In the case of DDR3-SDRAM:
   TCKEHDLL = ceil (tXPDLL / tCK)
   TCKEH = ceil (tXP/tCK)
   In the case of DDR2-SDRAM:
   TCKEHDLL = ceil (tXARD / tCK)
   TCKEH = ceil (tXP/tCK)

3. Writing to this register should only be performed when the following conditions are met.
   - Access to SDRAM is disabled (i.e. the ACEN bit in the DBACEN register is 0)
   - Auto-refresh is disabled (i.e. the ARFEN bit in the DBRFEN register is 0)

4.2.25 SDRAM Timing Register 15 (DBTR15)

The SDRAM timing register 15 (DBTR15) is used to set a timing parameter for the SDRAM.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	TCKEL			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	TCKEL	0000	R/W	CKEL Period Setting Bits These bits indicate the minimum time from the MCKE signal going low until it goes high. 0000: Initial value (when using the DBSC3, set these bits to one of the following values (2 to 15 cycles)). 0010: 2 cycles 0011: 3 cycles : 1111: 15 cycles Other settings are prohibited.

Notes: 1. The setting is in cycles of the SDRAM operating clock.

2. Writing to this register should only be performed when the following conditions are met.
   - Access to SDRAM is disabled (i.e. the ACEN bit in the DBACEN register is 0)
   - Auto-refresh is disabled (i.e. the ARFEN bit in the DBRFEN register is 0)

4.2.26 SDRAM Timing Register 16 (DBTR16)

The SDRAM timing register 16 (DBTR16) is used to set a timing parameter for the SDRAM.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	DQIENLTNCY		—	—	DQLOFFSET		—	—	DQL					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	DQENLTNCY		—	—	—	—	—	—	—	—	WDQL			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
29, 28	DQIENLTNCY	00	R/W	dqienltncy Setting Bits These bits set the latency from issuing a read command to the DDR-PHY unit up to outputting the dqi_en signal. One cycle should be set in this LSI. 00: Initial value (when using the DBSC3, set these bits to the following value (1 cycle)). 01: 1 cycle Other settings are prohibited.
27, 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25, 24	DQLOFFSET	00	R/W	dqltncy Offset Setting Bits These bits set the additional latency necessary for DBSC3 internal operation when a read command is issued to the DDR-PHY unit. Two cycles should be set in this LSI. 00: Initial value (when using the DBSC3, set these bits to the following value (2 cycles)). 10: 2 cycles Other settings are prohibited.

Bit	Bit Name	Initial Value	R/W	Description
23, 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21 to 16	DQL	000000	R/W	dqltncy Setting Bits These bits set the latency from issuing a read command to the DDR-PHY unit up to having the read data returned from the DDR-PHY unit. This setting must satisfy the following. DQL = CL + 6 cycles 000000: Initial value (when using the DBSC3, set these bits to one of the following values (9 to 12 cycles)). 001001: 9 cycles 001010: 10 cycles 001011: 11 cycles 001100: 12 cycles Other settings are prohibited.
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13, 12	DQENLTNCY	00	R/W	dqenltncy Setting Bits These bits set the latency from issue of a write command to DDR-PHY unit until the dqi_en signal is output. Set 1 cycle for this product. 00: Initial value (when using the DBSC3, set these bits to the following value (1 cycle)). 01: 1 cycle Other settings are prohibited.
11 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	WDQL	0000	R/W	wdqltncy Setting Bits These bits set the latency from issuing a write command up to outputting the write data. One cycle should be set in this LSI. 0000: Initial value (when using the DBSC3, set these bits to the following value (1 cycle)). 0001: 1 cycle Other settings are prohibited.

Notes: 1. The setting is in cycles of the SDRAM operating clock.

2. Writing to this register should only be performed when the following conditions are met.
   - Access to SDRAM is disabled (i.e. the ACEN bit in the DBACEN register is 0)
   - Auto-refresh is disabled (i.e. the ARFEN bit in the DBRFEN register is 0)

4.2.27 SDRAM Timing Register 17 (DBTR17)

The SDRAM timing register 17 (DBTR17) is used to set a timing parameter for the SDRAM.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	TMOD					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21 to 16	TMOD	000000	R/W	MRS Time Setting Bits These bits indicate the minimum interval from an MRS (mode register set) command to a subsequent command. 000000: Initial value (when using the DBSC3, set these bits to one of the following values (2 to 15 cycles)). 000010: 2 cycles 000011: 3 cycles : 001111: 15 cycles Other settings are prohibited.
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Notes: 1. The setting is in cycles of the SDRAM operating clock.

2. Writing to this register should only be performed when the following conditions are met.
   - Access to SDRAM is disabled (i.e. the ACEN bit in the DBACEN register is 0)
   - Auto-refresh is disabled (i.e. the ARFEN bit in the DBRFEN register is 0)
3. Set the value defined for each SDRAM type to be connected.

4.2.28 SDRAM Timing Register 18 (DBTR18)

The SDRAM timing register 18 (DBTR18) is used to set a timing parameter for the SDRAM.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	RODTL			—	—	—	—	—	RODTA		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	WODTL			—	—	—	—	—	WODTA		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 24	RODTL	000	R/W	MODT Pin Assert Period Setting Bits at Read These bits set the assert period of the MODT signal that is output when a read command is output. These bits can be set between the range of +0 and +7, on the basis of the (burst length/2) value of the corresponding read command. For DDR2, the cycle offset is fixed at 1. For DDR3, it is fixed at 0. 000: BL/2 cycles + cycle offset 001: BL/2 + cycle offset + 1 cycle : 111: BL/2 + 7 cycles + cycle offset
23 to 19	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
18 to 16	RODTA	000	R/W	MODT Pin Assert Start Timing Setting Bits at Read These bits set the assert start timing for the MODT signal that is output when a read command is output. These bits can be set between the range of −1 and +3, on the basis of the output timing of the corresponding read command. For DDR2, the delay offset is fixed at 0. For DDR3, 0 or 1 cycle should be specified through the db_odt_mode bit in DBPDCNT0. 000: Simultaneous with (read command + delay offset) 001: 1 cycle after (read command + delay offset) 010: 2 cycles after (read command + delay offset) 011: 3 cycles after (read command + delay offset) 111: 1 cycle before (read command + delay offset) Other settings are prohibited.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 8	WODTL	000	R/W	MODT Pin Assert Period Setting Bits at Write These bits set the assert period of the MODT signal that is output when a write command is output. These bits can be set between the range of +0 and +7, on the basis of the (burst length/2) value of the corresponding write command. For DDR2, the cycle offset is fixed at 1. For DDR3, it is fixed at 0. 000: BL/2 cycles + cycle offset 001: BL/2 + 1 cycles + cycle offset : 111: BL/2 + 7 cycles + cycle offset
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2 to 0	WODTA	000	R/W	MODT Pin Assert Start Timing Setting Bits at Write These bits set the assert start timing for the MODT signal that is output when a write command is output. These bits can be set between the range of −1 and +3, on the basis of the output timing of the corresponding write command. For DDR2, the delay offset is fixed at 0. For DDR3, 0 or 1 cycle should be specified through the db_odt_mode bit in DBPDCNT0. 000: Simultaneous with (write command + delay offset) 001: 1 cycle after (write command + delay offset) 010: 2 cycles after (write command + delay offset) 011: 3 cycles after (write command + delay offset) 111: 1 cycle before (write command + delay offset) Other settings are prohibited.

Notes: 1. The setting is in cycles of the SDRAM operating clock.

2. Writing to this register should only be performed when the following conditions are met.
   - Access to SDRAM is disabled (i.e. the ACEN bit in the DBACEN register is 0)
   - Auto-refresh is disabled (i.e. the ARFEN bit in the DBRFEN register is 0)

4.2.29 SDRAM Timing Register 19 (DBTR19)

The SDRAM timing register 19 (DBTR19) is used to set a timing parameter for the SDRAM.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TZQCS							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	TZQCS	H'00	R/W	Short Calibration Period Setting Bits These bits specify the minimum interval from a ZQCS (short calibration) command to the next command. This bit setting is valid only for DDR3-SDRAM. Specify 64 cycles in this LSI. 00000000: Initial value (when using the DBSC3, set these bits to the following value (64 cycles)). 01000000: 64 cycles Other settings are prohibited.

Notes: 1. The setting is in cycles of the SDRAM operating clock.

2. Writing to this register should only be performed when the following conditions are met.
   - Access to SDRAM is disabled (i.e. the ACEN bit in the DBACEN register is 0)
   - Auto-refresh is disabled (i.e. the ARFEN bit in the DBRFEN register is 0)

4.2.30 SDRAM Operation Setting Register (DBBL)

The SDRAM operation setting register is used to set a burst operation mode of the memory.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	BL	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	BL	00	R/W	Burst Length Setting Bits These bits specify the burst length of SDRAM. For DDR3-SDRAM, setting the BL bits to B'10 (fixed to 4) is prohibited. 00: Fixed to 8 10: Fixed to 4 Other settings are prohibited.

Note: Writing to this register should only be performed when the following conditions are met.

- Access to SDRAM is disabled (i.e. the ACEN bit in the DBACEN register is 0)
- Auto-refresh is disabled (i.e. the ARFEN bit in the DBRFEN register is 0)

4.2.31 DBSC3 Operation Adjustment Register 0 (DBADJ0)

DBADJ0 sets adjustments for the DBSC3 operation.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	CASHIFT	—	—	—	CAMODE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	CASHIFT	0	R/W	Command/Address Output Shift Setting Bit Instructs the DDR-PHY unit whether to shift the SDRAM command/address output. When the CASHIFT bit is 1, the DDR-PHY unit shifts the command/address output backward by 1/2 MCK cycles. 0: Command/address output timing is not shifted. 1: Command/address output timing is shifted backward by 1/2 MCK cycles.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
0	CAMODE	0	R/W	Command/Address Output Mode Setting Bit Sets the output mode for the SDRAM command/address. When the CAMODE bit is 0, the DBSC3 outputs a single command per clock cycle. When the CAMODE bit is 1, the DBSC3 outputs a single command per two clock cycles. In this case, command signals and address signals, except for the $\overline{CS}$ signal of SDRAM, are kept constant for a period of two clock cycles. During this period, the $\overline{CS}$ signal becomes low only in the latter one clock cycle. 0: One command output in 1 clock cycle 1: One command output in 2 clock cycles

Note: This register should be written to only when SDRAM access is disabled (ACEN = 0 in the DBACEN register).

4.2.32 DBSC3 Operation Adjustment Register 1 (DBADJ1)

DBADJ1 sets adjustments for the DBSC3 operation.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	AOOEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
0	AOOEN	0	R/W	Out-of-Order Process Enable Bit Enables or disables the process to change the order in which commands are issued to SDRAM. When the command issuing order change process is enabled, the DBSC3 performs the process to change the order of issuing commands for improving the efficiency of SDRAM data bus use. Accordingly, commands are issued to SDRAM in an order different from the order in which the requests were accepted. When the command issuing order change process is disabled, the DBSC3 issues commands to SDRAM in the same order the requests were accepted. The SDRAM performance deteriorates when read access and write access are performed alternately. To avoid this, the order of access can be changed so that as many read (or write) operations as possible are performed in succession. This is called the out-of-order process. Even when the out-of-order process is enabled, the order among read operations alone is not changed from that among read requests. Likewise, the order among write operations alone is not changed. Even after the out-of-order process, the order of read and write requests to a single address is guaranteed. If the out-of-order process is enabled, the efficiency of SDRAM data bus use improves compared to when the out-of-order process is disabled, but the worst latency increases. 0: Command issuing order change process is disabled. 1: Command issuing order change process is enabled.

Note: This register should be written to only when SDRAM access is disabled (ACEN = 0 in the DBACEN register).

4.2.33 DBSC3 Operation Adjustment Register 2 (DBADJ2)

DBADJ2 sets adjustments for the DBSC3 operation.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ACAPC								—	—	—	—	ACAPX			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	ACAPC	H'00	R/W	Bits for Setting Request Count Acceptable by Device Control Unit These bits set the number of requests acceptable by the device control unit in the DBSC3 in 64-bit data access units. The DBSC3 has a preceding PRE-ACT process that hides a penalty of the Precharge-Activate process which occurs at a page miss by overlapping the penalty with the read or write operation being currently executed. To perform this, a certain amount of requests have to be stocked. However, this setting also increases the number of requests that can be accepted, and the execution of subsequently generated requests that are high in the order of priority follows processing for requests that have already been accepted, so this can worsen the latency for the higher priority requests. Adjust the setting of the ACAPC bits to be above the value obtained from the equation below and for a balance between throughput and latency. Settings below the value from the equation will adversely affect throughput. Minimum value of ACAPC setting = ACAPC (TRP + TRCD)/2 H'00: Initial value (when setting is unnecessary) H'01: 1 request : H'20: 32 requests Other settings are prohibited.
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	ACAPX	H'0	R/W	Bits for Setting Transaction Count Acceptable by Device Control Unit These bits set the number of requests acceptable by the device control unit in the DBSC3 in transaction units. 'Transaction' refers to a single round of access by the CPU or DMAC, and since the number of requests included in a single transaction varies from 1 to 16 according to the unit of access, the number of requests cannot be controlled by merely setting the number of transactions. Therefore, set these bits for a generous number of transactions and control the upper limit on the number of requests through the setting of the ACAPC bits described above. Operation with the setting at its the initial value (H'0) is the same as operation with the maximum setting (H'8). If a value set in the ACAPC bits is small, setting the bits may worsen the throughput. H'0: Initial value (when setting is unnecessary) H'1: 1 transaction : H'8: 8 transactions Other settings are prohibited.

Note: This register should be written to only when SDRAM access is disabled (ACEN = 0 in the DBACEN register).

4.2.34 Refresh Configuration Register 0 (DBRFCNF0)

Refresh configuration register 0 (DBRFCNF0) is used to set the timing for refreshing of the SDRAM.

If you change the value of this register while auto-refresh is active (i.e. the ARFEN bit in the DBRFEN register is 1), the auto-refresh facility must be re-activated to reflect the new value in refreshing. To re-activate the auto-refresh facility, write 1 to the ARFEN bit in the DBRFEN register.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	REFTHF											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	REFTHF	H'000	R/W	Forcible Auto-Refresh Threshold Setting Bits These bits indicate the timing for forcible refreshing regardless of bus requests. The value represented by these bits affects the amount of jitter in the refresh interval and performance in access to memory. A smaller value means less jitter in the refresh interval but may reduce performance in access. For details on the amount of jitter in the refresh interval, see section 4.2.35, Refresh Configuration Register 1 (DBRFCNF1). The following condition must be satisfied: REFTHF ≥ (TCKEL + TCKEH) + REFTH0 (REFTH0 = max (TRDPR, CWL + BL/2 + TWR, TRAS, TRC − TRP) + (TRPA + 24) H'000: Initial value (when using the DBSC3, set these bits to one of the following values (128 to 511 cycles)). H'080: 128 cycles : H'1FF: 511 cycles Other settings are prohibited.

Notes: 1. The setting is in cycles of the SDRAM operating clock.

2. If auto-refresh is activated (i.e. the ARFEN bit in the DBRFEN register is set to 1) after a value smaller than the minimum value defined in the above table has been set in this register, correct operation cannot be guaranteed.

4.2.35 Refresh Configuration Register 1 (DBRFCNF1)

Refresh configuration register 1 (DBRFCNF1) is used to set the timing for refreshing of the SDRAM.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	REFPMAX				—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	REFINT															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 24	REFPMAX	0000	R/W	Maximum Post Number of Refresh Commands Setting Bits These bits indicate the maximum number of refresh commands (post number) accumulated by auto-refresh. As long as the number of refresh commands that has been accumulated is smaller than REFPMAX, refresh commands are issued while there are no bus requests. 0000: 0 (minimum amount of jitter in the refresh interval) 0001: 1 : 1000: 8 (maximum amount of jitter in the refresh interval) Other settings are prohibited.
23 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	REFINT	H'0000	R/W	Average Refresh Interval Setting Bits These bits indicate the average interval for issuing of refresh commands. When the REFINTS bit of the DBRFCNF2 register is 0, the average interval (in cycles) is REFINT. When the REFINTS bit of the DBRFCNF2 register is 1, on the other hand, the average interval (in cycles) is floor (REFINT/2). This average interval is hereafter referred to as REFINT_E. Thus, REFINT_E = REFINT >> REFINTS (>>: logical right-shift operator). The following condition must be satisfied: REFINT $\geq$ (REFTHF $\times$ 2) << REFINTS H'0000: Initial value (when using the DBSC3, set these bits to one of the following values (128 to 16383 cycles)). H'0080: 128 cycles H'0081: 129 cycles : H'3FFF: 16383 cycles Other settings are prohibited.

Refresh configuration register 1 (DBRFCNF1) is used to set the timing for refreshing of the SDRAM.

If you change the value of this register while auto-refresh is active (i.e. the ARFEN bit in the DBRFEN register is 1), the auto-refresh facility must be re-activated to reflect the new value in refreshing. To re-activate the auto-refresh facility, write 1 to the ARFEN bit in the DBRFEN register.

In the descriptions below, the "number of cycles" unless otherwise specified means the value measured with the SDRAM clock.

(1) Register Settings and Refresh Generation Timing

The following is an example of the settings in refresh configuration registers 1 and 2 and the timing of refresh generation.
In the explanations, "a ± b" indicates the range of values from a - b to a + b.

- To minimize variation of the Refresh Interval

  Set REFPMAX to 0.

  In this case, the time between one round of refresh generation and the next will be REFINT_E ± REFTHF cycles. More generally, taking n as a positive integer, the time from one round of refresh generation to refresh generation n rounds later is n × REFINT_E ± REFTHF cycles. However, this is on the assumption of no writing to the DBRFEN register during this period.
- To make the refresh interval flexible

  Set REFPMAX to a value greater than or equal to 1.

  REFINT holds the setting for the average refresh interval (tREFI in the normal range of operating temperatures), which is given in memory-vendor datasheets as an integer number of cycles. Set a value which has been rounded down from this integer. Set REFINTS to 0 or 1 according to the temperature at the time.

  In this case, taking n as a positive integer, the time from one round of refresh generation to refresh generation n rounds later is (n × REFINT_E + REFPMAX × REFINT_E cycles) or shorter. However, this is on the assumption of no writing to the DBRFEN register during this period.

Notes: 1. The setting is in cycles of the SDRAM operating clock.
2. If auto-refresh is activated (i.e. the ARFEN bit in the DBRFEN register is set to 1) after a value smaller than the minimum value defined in the above table has been set in this register, correct operation cannot be guaranteed.

4.2.36 Refresh Configuration Register 2 (DBRFCNF2)

Refresh configuration register 2 (DBRFCNF2) is used to set the timing for refreshing of the SDRAM.

If you change the value of this register while auto-refresh is active (i.e. the ARFEN bit in the DBRFEN register is 1), the auto-refresh facility must be re-activated to reflect the new value in refreshing. To re-activate the auto-refresh facility, write 1 to the ARFEN bit in the DBRFEN register.

In the descriptions below, the number of cycles means data measured with the SDRAM clock unless otherwise specified.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	REF INTS
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	REFINTS	0	R/W	Average Refresh Interval Adjustment Bit When this bit is 0, the average interval (in cycles) is REFINT. When this bit is 1, on the other hand, the average interval (in cycles) is floor (REFINT/2). 0: Average interval is REFINT 1: Average interval is 1/2 REFINT

4.2.37 DDR3-SDRAM Calibration Configuration Register (DBCALCNF)

DBCALCNF controls the function for calibrating DDR3-SDRAM at regular intervals.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	CALEN	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CALINT															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	CALEN	0	R/W	DDR3-SDRAM Calibration Enable Bit While this bit is set to 1, calibration of DDR3-SDRAM is executed (ZQCS command is executed) at regular intervals. 0: DDR3-SDRAM calibration is disabled. 1: DDR3-SDRAM calibration is enabled. (Setting the CALEN bit to 1 is prohibited for DDR2-SDRAM.)
23 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	CALINT	H'0000	R/W	DDR3-SDRAM Calibration Frequency Setting Bits These bits adjust the frequency of DDR3-SDRAM calibration. When the CALINT bits are set to n, the ZQCS command is issued once every time the refresh command has been issued for n times by the auto-refresh function. H'0000: Initial value (when using the DBSC3, set these bits to one of the following values (1 to 65535 times)). H'0001: Executed at every auto-refresh. H'0002: Executed at every 2 times of auto-refresh. : H'FFFF: Executed at every 65535 times of auto-refresh. Other settings are prohibited.

Notes: 1. Even when the CALEN bit is set to 1, the ZQCS command is not issued when the auto-refresh function is stopped (ARFEN = 0 in the DBRFEN register).
2. Writing to this register should only be performed when the following conditions are met.
- Access to SDRAM is disabled (i.e. the ACEN bit in the DBACEN register is 0)
- Auto-refresh is disabled (i.e. the ARFEN bit in the DBRFEN register is 0)

4.2.38 DDR3-SDRAM Calibration Timing Register (DBCALTR)

DBCALTR specifies the command interval limitations for DDR3-SDRAM calibration executed at regular intervals.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	TCALRZ											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	TCALZR											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 16	TCALRZ	H'000	R/W	DDR3-SDRAM Calibration Timing Setting (REF-ZQCS Interval Setting) These bits specify the minimum interval between REF and ZQCS commands for calibration execution. H'000: Initial value (when using the DBSC3, set these bits to one of the following values (128 to 4095 cycles)). H'080: 128 cycles : H'FFF: 4095 cycles Other settings are prohibited.
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
11 to 0	TCALZR	H'000	R/W	DDR3-SDRAM Calibration Timing Setting (ZQCS-REF Interval Setting) These bits specify the minimum interval between a ZQCS command for calibration and a REF command for the next auto-refresh execution. H'000: Initial value (when using the DBSC3, set these bits to one of the following values (128 to 4095 cycles)). H'080: 128 cycles : H'FFF: 4095 cycles Other settings are prohibited.

Notes: 1. The following conditions must be satisfied:
TCALRZ ≥ max(128, TRFC + 7 × TFAW ÷ 4 + tACTANY + 32)
TCALZR ≥ max(128, TZQCS + 7 × TFAW ÷ 4 + tACTANY + 32)
REFINT ≥ (TCALRZ + TCALZR + REFTHF × 2) << REFINTS
(tACTANY = max(TRCD, TRAS, TRC-TRPA, TFAW))

2. This register value has no effect when CALEN = 0 in DBCALCNF.
3. The setting is in cycles of the SDRAM operating clock.
4. Writing to this register should only be performed when the following conditions are met.
   - Access to SDRAM is disabled (i.e. the ACEN bit in the DBACEN register is 0)
   - Auto-refresh is disabled (i.e. the ARFEN bit in the DBRFEN register is 0)

4.2.39 ODT Operation Setting Register (DBRNK0)

DBRNK0 is used to enable or disable output from the MODT pin to SDRAM.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RODT OUT0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	WODT OUT0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	RODTOUT0	0	R/W	MODT Pin Output Enable at Read Enables or disables MODT pin output at read. When MODT output is enabled, the db_odt_dis bit in DBPDCNT0 should be set to 0 (enabled). 0: MODT pin output is disabled at read. 1: MODT pin output is enabled at read. (The db_odt_dis bit in DBPDCNT0 should also be set enabled.)
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	WODTOUT0	0	R/W	MODT Pin Output Enable at Write Enables or disables MODT pin output at write. When MODT output is enabled, the db_odt_dis bit in DBPDCNT0 should be set to 0 (enabled). 0: MODT pin output is disabled at write. 1: MODT pin output is enabled at write. (The db_odt_dis bit in DBPDCNT0 should also be set enabled.)

Note: 1. Writing to this register should only be performed when the following conditions are met.
- Access to SDRAM is disabled (i.e. the ACEN bit in the DBACEN register is 0)
- Auto-refresh is disabled (i.e. the ARFEN bit in the DBRFEN register is 0)

4.2.40 Power-Down Configuration Register (DBPDNCNF)

DBPDNCNF controls the auto power-down function.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PDWAIT								—	—	—	PDDLL	—	—	PDMODE	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 8	PDWAIT	H'00	R/W	Power-Down Wait Bits These bits set the number of cycles it takes to enter power-down mode after memory accesses no longer occur. H'00: Initial value (when using the DBSC3, set these bits to one of the following values (2 to 255 cycles)). H'02: 2 cycles : H'FF: 255 cycles Other settings are prohibited.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	PDDLL	0	R/W	Power-Down DLL Control Bit Turns on or off the DLL of SDRAM when entering a power-down mode. 0: DLL is turned off at a precharged power-down. DLL is turned on at an active power-down. 1: DDR2: DLL is turned off at a power-down. DDR3: DLL is turned on at a power-down.

Bit	Bit Name	Initial Value	R/W	Description
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	PDMODE	00	R/W	Power-Down Mode Bits When these bits are set to 01 and there has been no memory access for a certain period, the CKE pin is set to the low level causing SDRAM to enter a power-down mode. 00: Auto power-down mode is off 01: Auto power-down mode is on Other settings are prohibited.

Notes: 1. The setting is in cycles of the SDRAM operating clock.
2. When returning from a power-down mode, a penalty of MAX (3, TCKEH) – 3 cycles occurs.
3. Writing to this register should only be performed when the following conditions are met.
- Access to SDRAM is disabled (i.e. the ACEN bit in the DBACEN register is 0)
- Auto-refresh is disabled (i.e. the ARFEN bit in the DBRFEN register is 0)

4.2.41 DDR-PHY Unit Control Register 0 (DBPDCNT0)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	db_odt_mode	db_odt_dis	db_odten_sel		db_odt_tsel	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	db_offset		—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21	db_odt_mode	0	R/W	MODT Pin Timing Control DDR2-SDRAM: Set this bit to 0. DDR3-SDRAM: 0: The delay offset for MODT pin output is set to 0 cycles. 1: The delay offset for MODT pin output is set to 1 cycle.
20	db_odt_dis	0	R/W	Disable Control for ODT Function in DDR-PHY and MODT Pin Enables or disables the ODT function in DDR-PHY at read and the MODT pin function. 0: Usage of the ODT function is enabled. 1: The ODT function is not used.
19, 18	db_odten_sel	00	R/W	MODT Pin Assert Interval Setting at Read 00: Initial value Other settings are prohibited.
17, 16	db_odt_tsel	00	R/W	Resistance Control for ODT in DDR-PHY These bits select the terminal resistance for ODT in DDR-PHY. 00: 150 Ω (DDR2), 120 Ω (DDR3) 01: 75 Ω (DDR2), 60 Ω (DDR3) Other settings are prohibited.

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	db_offset	00	R/W	00: Initial value (when using the DBSC3, set these bits to the following value. 10: Common setting for DDR2 and DDR3 Other settings are prohibited.
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

4.2.42 DDR-PHY Unit Control Register 1 (DBPDCNT1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	db_ewc_close_offset0		db_ewc_strength0	db_ewc_open_offset0			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6, 5	db_ewc_close_offset0	00	R/W	EnableWindow Negate Control 0 Settings other than those below are prohibited. 00: Common setting for DDR2 and DDR3

Bit	Bit Name	Initial Value	R/W	Description
4	db_ewc_ strength0	0	R/W	EnableWindow Control Signal Strength Specification 0 0: MDQS1 or MDQS0 timing measurement pin strength: 40 Ω When using DDR3, set the db_ewc_strength0 bit to 0. 1: MDQS1 or MDQS0 timing measurement pin strength: 30 Ω When using DDR2, set the db_ewc_strength0 bit to 1.
3 to 0	db_ewc_open _offset0	0000	R/W	EnableWindow Assert Control 0 0000: Initial value (when using the DBSC3, set these bits to the following value. 0100: Common setting for DDR2 and DDR3 Other settings are prohibited.

4.2.43 DDR-PHY Unit Control Register 2 (DBPDCNT2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: When setting the bits in this register, first read this register with a 32-bit access to check the initial values of the bits and then set the initial values for the bits that cannot be written to.

4.2.44 DDR-PHY Unit Control Register 3 (DBPDCNT3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	db_ stby_n	db_ comhiz	—	—	db_io backup	db_dll enable2	db_dll enable1	db_dll reset_n	—	—	—	—	db_add_ strength	db_dqdm_ strength	db_dqs_ strength	db_ck_ strength
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	db_io enable2	db_io enable1	db_calib _start	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	db_stby_n	0	R/W	PLL Standby Specifies the operating state of the MCK pin and internal PLL. Power consumption can be kept low by using this function in combination with the self-refresh state. 0: MCK pin is in the standby state and internal PLL is stopped. 1: MCK pin and internal PLL are operating. The PLL oscillation settling time (at least 200 μs) must be secured when changing this bit from 0 to 1.
30	db_comhiz	0	R/W	Hi-Z Specification By setting this signal to the high level, external I/O pins, except for the MCK0/$\overline{\text{MCK0}}$, MCKE, and $\overline{\text{MRESET}}$ pins, enter the high impedance state. Power consumption can be kept low by using this function in combination with the self-refresh state. 0: External I/O pins are operating. 1: External I/O pins, except for MCK0/$\overline{\text{MCK0}}$, MCKE, and $\overline{\text{MRESET}}$, are in the high impedance state.
29, 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27	db_ iobackup	0	R/W	This bit is used at power-supply backup. Refer to section 4.3.7, SDRAM Power-Supply Backup Function.
26	db_ dllenable2	0	R/W	ENABLE2 Signal of DLL Refer to section 4.3, DBSC3 Operation.
25	db_ dllenable1	0	R/W	ENABLE1 Signal of DLL Refer to section 4.3, DBSC3 Operation.

Bit	Bit Name	Initial Value	R/W	Description
24	db_dllreset_n	0	R/W	RESET Signal of DLL Refer to section 4.3, DBSC3 Operation.
23 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
19	db_add_strength	0	R/W	Address System Pin Strength Specification 0: 40 Ω (When using DDR3, set to 40 Ω.) 1: 30 Ω (When using DDR2, set to 30 Ω.)
18	db_dqdm_strength	0	R/W	Data System (DQ, DM) Pin Strength Specification 0: 40 Ω (When using DDR3, set to 40 Ω.) 1: 30 Ω (When using DDR2, set to 30 Ω.)
17	db_dqs_strength	0	R/W	Data System (DQS) Pin Strength Specification 0: 40 Ω (When using DDR3, set to 40 Ω.) 1: 30 Ω (When using DDR2, set to 30 Ω.)
16	db_ck_strength	0	R/W	Data System (CK) Pin Strength Specification 0: 40 Ω (When using DDR3, set to 40 Ω.) 1: 30 Ω (When using DDR2, set to 30 Ω.)
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	db_ioenable2	0	R/W	ENABLE2 Signal of IO Refer to section 4.3, DBSC3 Operation.
13	db_ioenable1	0	R/W	ENABLE1 Signal of IO Refer to section 4.3, DBSC3 Operation.
12	db_calib_start	0	R/W	IO Calibration Setting Signal Specifies whether to execute DBSC3 calibration. 0: DBSC3 calibration is disabled. 1: DBSC3 calibration is always executed. For the setting procedure, refer to section 4.3, DBSC3 Operation.
11 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: DBSC3 calibration is always executed for the MCK0 and $\overline{\text{MCK0}}$ pins. For the other pins, calibration is executed when a REF command is issued.

4.2.45 DDR-PHY Unit Lock Register (DBPDLCK)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PLOCK															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	PLOCK	H'0000	R/W	Write H'A55A to these bits only when performing the open/short test on the MZQ pin. In normal operation, the bits should be fixed to H'0000; do not access these bits. For details, refer to section 4.3.10, Open/Short Test for MZQ Pin.

4.2.46 DDR-PHY Unit Register Address (DBPDRGA)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	PRA							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	PRA	H'00	R/W	Write H'00 or H'14 to these bits only when performing the open/short test on the MZQ pin. In normal operation, the bits should be fixed to H'00; do not access these bits. For details, refer to section 4.3.10, Open/Short Test for MZQ Pin.

4.2.47 DDR-PHY Unit Register Access (DBPDRGD)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PRD															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PRD															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PRD	Undefined	R/W	Write H'A5000000 to these bits or read this register value only when performing the open/short test on the MZQ pin. In normal operation, the bits should be fixed to H'00000000; do not access these bits. For details, refer to section 4.3.10, Open/Short Test for MZQ Pin.

4.2.48 Bus Control Unit 0 Control Register 0 (DBBS0CNT0)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

4.2.49 Bus Control Unit 0 Control Register 1 (DBBS0CNT1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	BKADB						BKADP						BKADM	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13 to 8	BKADB	000000	R/W	Address Location Bits for Upper Bank Address These bits specify the address location of the upper bank address. This bit setting is valid when the BKADM bits specify bank addresses to be handled as discontinuous addresses. The BA1 address location is set as the upper bank address. In an 8-bank product, BA2 is handled as the upper address of BA1. 000000: Make this setting when the BKADM bits are set to 00 001101: Address 13 is set to BA1 (address 14 is set to BA2) 001110: Address 14 is set to BA1 (address 15 is set to BA2) 001111: Address 15 is set to BA1 (address 16 is set to BA2) 010000: Address 16 is set to BA1 (address 17 is set to BA2) Other settings are prohibited.

Bit	Bit Name	Initial Value	R/W	Description
7 to 2	BKADP	000000	R/W	Address Location Bits for Bank Address These bits specify the address location of the bank address. According to the memory configuration used, either the upper bits of the column address are handled as the bank address or the address location of the bank address is set. 000000: Upper bits of column address (CAS) 001010: 1 Kbyte (address 10 is set to BA0) 001011: 2 Kbytes (address 11 is set to BA0) 001100: 4 Kbytes (address 12 is set to BA0) Other settings are prohibited.
1, 0	BKADM	00	R/W	Bank Address Usage Method Bits These bits set how to use the bank address. Whether to handle bank addresses as continuous addresses or discontinuous addresses can be set. When bank addresses are handled as continuous addresses, the address location of the bank address is set with the BKADP bits. When bank addresses are handled as discontinuous addresses, BA0 and BA1 can be handled as different addresses. Set BA0 with the BKADP bits and set BA1 with the BKADB bits. In an 8-bank product, BA2 and BA1 are handled as continuous addresses; BA2 is handled as the upper address of BA1. 00: Continuous addresses (BA0, BA1, and BA2 are specified with the BKADP bits) 01: Discontinuous addresses (lower one bit) (BA0 is specified with the BKADP bits, and BA1 and BA2 are specified with the BKADB bits) 10: Setting prohibited 11: Setting prohibited

Notes: 1. For the combinations that can be set by this register, refer to section 4.6, Address Location Specification for Bank Address.
2. This register can be written to only in the startup sequence.
3. Writing to this register should only be performed when the following condition is met.
- Access to SDRAM is disabled (i.e. the ACEN bit in the DBACEN register is 0)

4.3 DBSC3 Operation

4.3.1 Initialization Sequence

Before permitting accesses to the SDRAM after a power-on reset, the SDRAM should be initialized according to the appropriate sequence as shown below. Since the shown wait time between steps is merely an example, provide the necessary wait time given in the datasheet supplied by the memory vendor. The following sequence assumes that a high level is applied to the $\overline{\text{MBKPRST}}$ pin.

The cycle count is represented with the number of clock cycles of the SDRAM operating clock.

(1) DDR2-SDRAM

1. Make initial settings in DDR-PHY unit control registers 0, 1, and 3 (DBPDCNT0, DBPDCNT1, and DBPDCNT3). There is no particular order for the setting of the following steps (1) to (3).
   (1) Make settings in the db_offset, db_odt_tsel, db_odten_sel, db_odt_dis, and db_odt_mode bits in DDR-PHY unit control register 0 (DBPDCNT0).
   (2) Make settings in the db_ewc_open_offset0, db_ewc_strength0, and db_ewc_close_offset0 bits in DDR-PHY unit control register 1 (DBPDCNT1).
   (3) Set the db_calib_start, db_ioenable1, db_ioenable2, db_dllreset_n, db_dllenable1, db_dllenable2, db_iobackup, and db_comhiz bits to 0, and the db_ck_strength, db_dqs_strength, db_dqdm_strength, db_add_strength, and db_stby_n bits to 1 in DDR-PHY unit control register 3 (DBPDCNT3).
2. If the db_stby_n bit is changed from 0 to 1 in step 1 (3), it is necessary to wait for at least 200 µs before a stable MCK clock can be output. To secure this waiting time, issue the Wait code by the manual command-issuing register (DBCMD) and wait for a response after reading the operation completion waiting register (DBWAIT). The waiting times in the following steps should also be secured by the same procedure.
3. Set the db_calib_start bit in DDR-PHY unit control register 3 (DBPDCNT3) to 1.
4. After at least 32 clock cycles have elapsed following the setting in step 3, set the db_dllenable1 bit in DBPDCNT3 to 1.
5. After at least 100 µs have elapsed following the setting in step 4, set the db_dllenable2 bit in DBPDCNT3 to 1.
6. After at least 16 clock cycles have elapsed following the setting in step 5, set the db_dllreset_n bit in DBPDCNT3 to 1.
7. After at least 200 µs have elapsed following the setting in step 3, (step 6 not needed), set the db_ioenable1 bit in DBPDCNT3 to 1.

8. After at least one clock cycle has elapsed following the setting in step 7, set the db_ioenable2 bit in DBPDCNT3 to 1.
9. Wait until at least 10,000 clock cycles have elapsed following the setting in step 6.
10. Set the memory type in the SDRAM kind setting register (DBKIND).
11. Set up the SDRAM configuration setting register (DBCONF), SDRAM timing registers 0 to 19 (DBTR0 to DBTR19), SDRAM operation setting register (DBBL), and ODT operation setting register (DBRNK0).
12. Use the manual command-issuing register (DBCMD) to set the CKE pin of the SDRAM to high level. The value written to this register should be opc = PDXt, arg = cycles equivalent to 400 ns.
13. Use the manual command-issuing register (DBCMD) to issue a PREA (precharge all) command. The value written to this register should be opc = PreA, arg = 0.
14. Use the manual command-issuing register (DBCMD) to issue an EMRS (MR2) command.
15. Use the manual command-issuing register (DBCMD) to issue an EMRS (MR3) command.
16. Use the manual command-issuing register (DBCMD) to issue an EMRS (MR1) command. The settings should be made such that additive latency is 0, and DLL enable is enabled.
17. Use the manual command-issuing register (DBCMD) to issue a MRS (MR0) command. The settings should be made such that operating mode is normal, DLL reset is provided, burst length is 4, and burst type is sequential. CAS latency should be set according to the CL bit value in SDRAM timing register 0 (DBTR0). WR should be set according to the TWR bit value in SDRAM timing register 10 (DBTR10). PD should be set according to the PDDLL bit value in the power-down configuration register (DBPDNCNF).
18. Use the manual command-issuing register (DBCMD) to issue a PREA (precharge all) command. The value written to this register should be opc = PreA, arg = 0.
19. Use the manual command-issuing register (DBCMD) to issue a REF (refresh) command. The value written to this register should be opc = Ref, arg = 0.
20. Use the manual command-issuing register (DBCMD) to issue a REF (refresh) command. The value written to this register should be opc = Ref, arg = 0.
21. Use the manual command-issuing register (DBCMD) to issue a MRS (MR0) command. At this time, no DLL reset should be set.
22. Use the manual command-issuing register (DBCMD) to insert a waiting time. The value written to this register should be opc = Wait, arg = 200 clock cycles.
23. Use the manual command-issuing register (DBCMD) to issue an EMRS (MR1) command. At this time, set the OCD calibration program to the OCD calibration default and the others to the same values as those in step 16.
24. Use the manual command-issuing register (DBCMD) to issue an EMRS (MR1) command. At this time, set the OCD calibration program to the OCD calibration mode exit and the others to the same values as those in step 16.

25. If this is necessary, make settings in DBSC3 operation adjustment registers 0 to 2 (DBADJ0 to DBADJ2), bus control unit 0 control registers 0 and 1 (DBBS0CNT0 and DBBS0CNT1), and power-down configuration register (DBPDNCNF).
26. Make settings in refresh configuration registers 0 to 2 (DBRFCNF0 to DBRFCNF2).
27. Set the ARFEN bit to 1 in the auto-refresh enable register (DBRFEN).
28. Set the ACEN bit to 1 (access enabled) in the SDRAM access enable register (DBACEN).
29. Read the operation completion waiting register (DBWAIT) and wait for the response.

(2) DDR3-SDRAM

1. Before canceling a power-on reset, apply a low-level signal to the SDBUP pin.
2. Make initial settings in DDR-PHY unit control registers 0, 1, and 3 (DBPDCNT0, DBPDCNT1, and DBPDCNT3). There is no particular order for the setting of the following steps (1) to (3).
   (1) Make settings in the db_offset, db_odt_tsel, db_odten_sel, db_odt_dis, and db_odt_mode bits in DDR-PHY unit control register 0 (DBPDCNT0).
   (2) Make settings in the db_ewc_open_offset0, db_ewc_strength0, and db_ewc_close_offset0 bits in DDR-PHY unit control register 1 (DBPDCNT1).
   (3) Set the db_calib_start, db_ioenable1, db_ioenable2, db_ck_strength, db_dqs_strength, db_dqdm_strength, db_add_strength, db_dllreset_n, db_dllenable1, db_dllenable2, db_iobackup, and db_comhiz bits in DDR-PHY unit control register 3 (DBPDCNT3) to 0. Set the db_stby_n bit to 1.
3. If the db_stby_n bit is changed from 0 to 1 in step 2 (3), it is necessary to wait for at least 200 µs before a stable MCK clock can be output. To secure this waiting time, issue the Wait code by the manual command-issuing register (DBCMD) and wait for a response after reading the operation completion waiting register (DBWAIT). The waiting times in the following steps should also be secured by the same procedure.
4. Set db_calib_start in DDR-PHY unit control register 3 (DBPDCNT3) to 1.
5. After at least 32 clock cycles have elapsed following the setting in step 4, set db_dllenable1 of DBPDCNT3 to 1.
6. After at least 100 µs have elapsed following the setting in step 5, set db_dllenable2 of DBPDCNT3 to 1.
7. Apply a high-level signal to the SDBUP pin. Except in this initialization sequence, a high level should be applied to the SDBUP pin.
8. After at least 16 clock cycles have elapsed following the setting in step 6, set db_dllreset_n of DBPDCNT3 to 1.
9. After at least 200 µs have elapsed following the setting in step 4 (step 8 not needed), set db_ioenable1 of DBPDCNT3 to 1.

10. After at least 1 clock cycle has elapsed following the setting in step 9, set db_ioenable2 of DBPDCNT3 to 1.
11. Wait until at least 10,000 clock cycles have elapsed following the setting in step 8.
12. Set the memory type in the SDRAM kind setting register (DBKIND).
13. Set up the SDRAM configuration setting register (DBCONF), SDRAM timing registers 0 to 19 (DBTR0 to DBTR19), SDRAM operation setting register (DBBL), and ODT operation setting register (DBRNK0).
14. Use the manual command-issuing register (DBCMD) to insert a waiting time. The value written to this register should be opc = Wait, arg = cycles equivalent to 100 µs.
15. Use the manual command-issuing register (DBCMD) to set the $\overline{\text{MRESET}}$ pin of the SDRAM to high level. The value written to this register should be opc = RstH, arg = cycles equivalent to 100 µs.
16. Use the manual command-issuing register (DBCMD) to insert a waiting time. Write opc = Wait, arg = cycles equivalent to 100 µs to this register four times.
17. Use the manual command-issuing register (DBCMD) to set the MCKE pin of the SDRAM to high level. The value written to this register should be opc = PDXt, arg = tXPR (normally tRFC + cycles equivalent to 10 ns).
18. Use the manual command-issuing register (DBCMD) to issue a MRS (MR2) command. Adjust the setting for CWL to the setting of the CWL bits in SDRAM timing register 1 (DBTR1).
19. Use the manual command-issuing register (DBCMD) to issue a MRS (MR3) command. Set the MPR for normal operation.
20. Use the manual command-issuing register (DBCMD) to issue a MRS (MR1) command. Set the additive latency to 0 and DLL enable to enabled.
21. Use the manual command-issuing register (DBCMD) to issue a MRS (MR0) command. The settings should be made such that operating mode is normal, DLL reset is provided, and burst type is sequential. CAS latency should be set according to the CL bit value in SDRAM timing register 0 (DBTR0). WR should be set according to the TWR bit value in SDRAM timing register 10 (DBTR10). PD should be set according to the PDDLL bit value in the power-down configuration register (DBPDNCNF).
22. Use the manual command-issuing register (DBCMD) to issue a ZQ Calibration Long command. The value written to this register should be opc = ZQCL, arg = max {tZQinit, tDLLK – tMOD} (normally 512 clock cycles).
23. Use the manual command-issuing register (DBCMD) to issue a REF (refresh) command. The value written to this register should be opc = Ref, arg = 0.
24. Use the manual command-issuing register (DBCMD) to issue a REF (refresh) command. The value written to this register should be opc = Ref, arg = 0.

25. If this is necessary, make settings in DBSC3 operation adjustment registers 0 to 2 (DBADJ0 to DBADJ2), bus control unit 0 control registers 0 and 1 (DBBS0CNT0 and DBBS0CNT1), and power-down configuration register (DBPDNCNF).
26. Make settings in refresh configuration registers 0 to 2 (DBRFCNF0 to DBRFCNF2).
27. Make settings in the DDR3-SDRAM calibration configuration register (DBCALCNF).
28. Set the ARFEN bit to 1 in the auto-refresh enable register (DBRFEN).
29. Set the ACEN bit to 1 (access enabled) in the SDRAM access enable register (DBACEN).
30. Read the operation completion waiting register (DBWAIT) and wait for the response.

4.3.2 Self-Refresh Operation

If it is not necessary to access the SDRAM, the SDRAM can be put in self-refresh mode to reduce power consumption while still retaining data contents.

The following shows an example of the sequence; settings should be actually made according to the datasheet supplied by the memory vendor.

(1) DDR2-SDRAM

The following procedure is used to make a transition to self-refresh mode in DDR2-SDRAM. The settings in steps 6 to 8 can be made simultaneously.

1. Halt all data accesses to the SDRAM. Disable the CPU interrupts to prevent the CPU from processing the interrupts and accessing data in SDRAM.
2. Set the ACEN bit in the SDRAM access enable register (DBACEN) to 0 (access disabled).
3. Use the manual command-issuing register (DBCMD) to issue a PREA (precharge all) command. The value written to this register should be opc = PreA, arg = 0.
4. Use the manual command-issuing register (DBCMD) to issue a Self-Refresh Entry command. The value written to this register should be opc = SREn, arg = 0.
5. Set the ARFEN bit in the auto-refresh enable register (DBRFEN) to 0.
6. Power consumption of the DLL can be reduced by setting db_dllreset_n, db_dllenable1, and db_dllenable2 in DDR-PHY unit control register 3 (DBPDCNT3) to 0.
7. Power consumption of the PLL can be reduced and MCK0/$\overline{\text{MCK0}}$ can be placed at the low level by setting db_stby_n in DDR-PHY unit control register 3 (DBPDCNT3) to 0.
8. Pins other than MCK0/$\overline{\text{MCK0}}$ and MCKE can be placed in the high-impedance (Hi-Z) state and power consumption can be reduced by setting db_comhiz in DDR-PHY unit control register 3 (DBPDCNT3) to 1.
9. Enable the CPU interrupts.

Execute the following procedure to end self-refresh operation in DDR2-SDRAM.

1. Disable the CPU interrupts.
2. If db_stby_n in DDR-PHY unit control register 3 (DBPDCNT3) is set to 0 to reduce the power consumption of the PLL, set the bit to 1 and wait for at least 200 µs. To secure this waiting time, issue the Wait code by the manual command-issuing register (DBCMD) and wait for a response after reading the operation completion waiting register (DBWAIT). The waiting times in the following steps should also be secured by the same procedure.

3. If db_dllreset_n, db_dllenable1, and db_dllenable2 in DDR-PHY unit control register 3 (DBPDCNT3) are set to 0 to reduce the power consumption of the DLL, execute the following steps (1) to (4).
   (1) Set db_dllenable1 to 1.
   (2) After at least 100 µs have elapsed, set db_dllenable2 to 1.
   (3) After at least 16 clock cycles have elapsed, set db_dllreset_n to 1.
   (4) Wait for at least 10,000 clock cycles.
4. If db_comhiz in DDR-PHY unit control register 3 (DBPDCNT3) is set to 1, set it to 0 and wait for at least 5 clock cycles.
5. Use the manual command-issuing register (DBCMD) to issue a Self-Refresh Exit command. The value written to this register should be opc = SRXt, arg = 0.
6. Set the ARFEN bit in the auto-refresh enable register (DBRFEN) to 1.
7. Use the manual command-issuing register (DBCMD) to insert the period of waiting until access to the SDRAM is enabled. The value written to this register should be opc = Wait, arg = 200 clock cycles.
8. Read the operation completion waiting register (DBWAIT) and wait for a response to be returned.
9. Set the ACEN bit in the SDRAM access enable register (DBACEN) to 1 (access enabled).
10. Enable the CPU interrupts.

(2) DDR3-SDRAM

The following procedure is used to make a transition to self-refresh mode in DDR3-SDRAM. The settings in steps 9 to 11 can be made simultaneously.

1. Halt all data accesses to the SDRAM. Disable the CPU interrupts to prevent the CPU from processing the interrupts and accessing data in SDRAM.
2. Continue applying a high level to the $\overline{\text{MBKPRST}}$ and SDBUP pins. Keep the high level also during self-refreshing and release from self-refreshing.
3. Set the ACEN bit in the SDRAM access enable register (DBACEN) to 0 (access disabled).
4. Use the manual command-issuing register (DBCMD) to issue a PREA (precharge all) command. The value written to this register should be opc = PreA, arg = 0.
5. Use the manual command-issuing register (DBCMD) to issue a Self-Refresh Entry command. The value written to this register should be opc = SREn, arg = 0.
6. Set the ARFEN bit in the auto-refresh enable register (DBRFEN) to 0.
7. Use the manual command-issuing register (DBCMD) to insert a time to wait for the clock to stop. The value written to this register should be opc = Wait, arg = tCKSRE (normally, max{5 clock cycles, cycles equivalent to 10 ns}).

8. Read the operation completion waiting register (DBWAIT) and wait for a response to be returned.
9. Power consumption of the DLL can be reduced by setting db_dllreset_n, db_dllenable1, and db_dllenable2 in DDR-PHY unit control register 3 (DBPDCNT3) to 0.
10. Power consumption of the PLL can be reduced and MCK0/$\overline{\text{MCK0}}$ can be placed at the low level by setting db_stby_n in DDR-PHY unit control register 3 (DBPDCNT3) to 0.
11. Pins other than MCK0/$\overline{\text{MCK0}}$, MCKE, and $\overline{\text{MRESET}}$ can be placed in the high-impedance (Hi-Z) state and power consumption can be reduced by setting db_comhiz in DDR-PHY unit control register 3 (DBPDCNT3) to 1.
12. Enable the CPU interrupts.

Execute the following procedure to end self-refresh operation in DDR3-SDRAM.

1. Disable the CPU interrupts.
2. If db_stby_n in DDR-PHY unit control register 3 (DBPDCNT3) is set to 0 to reduce the power consumption of the PLL, set the bit to 1 and wait for at least 200 µs. To secure this waiting time, issue the Wait code by the manual command-issuing register (DBCMD) and wait for a response after reading the operation completion waiting register (DBWAIT). The waiting times in the following steps should also be secured by the same procedure.
3. If db_dllreset_n, db_dllenable1, and db_dllenable2 in DDR-PHY unit control register 3 (DBPDCNT3) are set to 0 to reduce the power consumption of the DLL, execute the following steps (1) to (4).
   (1) Set db_dllenable1 to 1.
   (2) After at least 100 µs have elapsed, set db_dllenable2 to 1.
   (3) After at least 16 clock cycles have elapsed, set db_dllreset_n to 1.
   (4) Wait for at least 10,000 clock cycles.
4. If db_comhiz in DDR-PHY unit control register 3 (DBPDCNT3) is set to 1, set it to 0 and wait for at least 5 clock cycles.
5. Use the manual command-issuing register (DBCMD) to insert the time to wait until release from self-refreshing. The value written to this register should be opc= Wait, arg = tCKSRX (normally, max{5 clock cycles, cycles equivalent to 10 ns}).
6. Read the operation completion waiting register (DBWAIT) and wait for a response to be returned.
7. Use the manual command-issuing register (DBCMD) to issue a Self-Refresh Exit command. The value written to this register should be opc = SRXt, arg = 0.
8. Set the ARFEN bit in the auto-refresh enable register (DBRFEN) to 1.
9. Use the manual command-issuing register (DBCMD) to insert the period of waiting until access to the SDRAM is enabled. The value written to this register should be opc = Wait, arg = tXSDLL (normally 512 clock cycles).

10. Read the operation completion waiting register (DBWAIT) and wait for a response to be returned.
11. Set the ACEN bit in the SDRAM access enable register (DBACEN) to 1 (access enabled).
12. Enable the CPU interrupts.

4.3.3 Changing Refresh Settings during Operation

In the DBSC3, the refresh settings (refresh interval, etc.) can be changed during operation.

1. Write to refresh configuration registers 0 to 2 (DBRFCNF0 to DBRFCNF2) as required to change the refresh settings.
2. Write 1 to the ARFEN bit in the auto-refresh enable register (DBRFEN).

   After the write operation in step 2, the new settings written in step 1 are reflected in the refresh operation.

4.3.4 Auto Power-Down

In the DBSC3, the SDRAM power consumption can be reduced by setting the CKE pin to low level while there are no SDRAM accesses.

Auto power-down is performed by using the power-down configuration register (DBPDNCNF) to set the number of cycles to elapse before transiting to a power-down mode after no memory accesses occur, and then set the PDMODE bits to 01.

However, if a memory access occurs, the CKE pin is driven to high level and an access command is issued. This will generate a penalty cycle which does not occur in a mode other than auto power-down mode.

4.3.5 Power-Down

If there is no access to SDRAM, entering to the power-down mode can place the SDRAM internal clock in the inactive state, which can effectively lower the power consumption in non-access cycles. Even in a power-down mode, the clock and power need to be supplied.

An example at entering to the power-down mode is listed as follows.

1. Set the ACEN bit in the SDRAM access enable register (DBACEN) to 0 (access disabled).
2. Use the manual command-issuing register (DBCMD) to issue a Power Down Entry command. The value written to this register should be opc = PDEn, arg = 0.

The power-down mode cancellation is performed with the following procedure.

1. Use the manual command-issuing register (DBCMD) to issue a Power Down Exit command. The value written to this register should be opc = PDXt, arg = 0.
2. Set the ACEN bit in the SDRAM access enable register (DBACEN) to 1 (access enabled).

To keep holding the SDRAM data in a power-down mode, a refresh command needs to be issued at regular intervals, similar to as in normal operation. In the DBSC3, by entering a power-down mode with the auto-refresh function operating (ARFEN = 1 in the DBRFEN register), refresh is performed regularly even while in a power-down mode and the SDRAM data is held.

Since SDRAM access is disabled during a power-down mode, requesting a SDRAM data access to the DBSC3 causes an error.

4.3.6 Retaining SDRAM Memory Contents in Software Standby Mode

In software standby mode, the clock in the LSI stops and the entire LSI is in standby state. The SDRAM memory contents however can be retained by performing the following processing. For details on the software standby function, entering software standby mode, and returning from software standby mode, refer to section 9.2, Overview of Power-Down Modes, in section 9, Operating Modes and Power-Down Modes.

To retain the SDRAM data, the procedure to perform before entering software standby mode and the procedure to perform when returning from software standby mode are described below.

(1) DDR2-SDRAM

Use the following procedure to make a transition to DDR2-SDRAM software standby mode.

1. Halt all data accesses to the SDRAM. Disable the CPU interrupts to prevent the CPU from accessing the SDRAM due to the interrupt handling.
2. Set the ACEN bit in the SDRAM access enable register (DBACEN) to 0 (access disabled).
3. Use the manual command-issuing register (DBCMD) to issue a PREA (precharge all) command. The value written to this register should be opc = PreA, arg = 0.
4. Use the manual command-issuing register (DBCMD) to issue a Self-Refresh Entry command. The value written to this register should be opc = SREn, arg = 0.
5. Set the ARFEN bit in the auto-refresh enable register (DBRFEN) to 0.
6. Set db_dllreset_n, db_dllenable1, db_dllenable2, and db_stby_n to 0 and db_comhiz to 1 in DDR-PHY unit control register 3 (DBPDCNT3).
7. Make a transition to software standby mode. Refer to section 9.2, Overview of Power-Down Modes, in section 9, Operating Modes and Power-Down Modes.

Use the following procedure to recover from DDR2-SDRAM software standby mode.

1. Perform the recovery processing from software standby mode. Refer to section 9.2, Overview of Power-Down Modes, in section 9, Operating Modes and Power-Down Modes.
2. Disable the CPU interrupts.
3. Set db_stby_n in DDR-PHY unit control register 3 (DBPDCNT3) to 1 and wait for at least 200 µs. To secure this waiting time, issue the Wait code by the manual command-issuing register (DBCMD) and wait for a response after reading the operation completion waiting register (DBWAIT). The waiting times in the following steps should also be secured by the same procedure.

4. Execute the following steps (1) to (4).
   (1) Set db_dllenable1 in DDR-PHY unit control register 3 (DBPDCNT3) to 1.
   (2) After at least 100 µs have elapsed, set db_dllenable2 in DDR-PHY unit control register 3 (DBPDCNT3) to 1.
   (3) After at least 16 clock cycles have elapsed, set db_dllreset_n in DDR-PHY unit control register 3 (DBPDCNT3) to 1.
   (4) Wait for at least 10,000 clock cycles.
5. Set db_comhiz in DDR-PHY unit control register 3 (DBPDCNT3) to 0 and wait for at least 5 clock cycles.
6. Use the manual command-issuing register (DBCMD) to issue a Self-Refresh Exit command. The value written to this register should be opc = SRXt, arg = 0.
7. Set the ARFEN bit in the auto-refresh enable register (DBRFEN) to 1.
8. Use the manual command-issuing register (DBCMD) to insert the period of waiting until access to the SDRAM is enabled. The value written to this register should be opc = Wait, arg = 200 clock cycles.
9. Read the operation completion waiting register (DBWAIT) and wait for a response to be returned.
10. Set the ACEN bit in the SDRAM access enable register (DBACEN) to 1 (access enabled).
11. Enable the CPU interrupts.

(2) DDR3-SDRAM

Use the following procedure to make a transition to DDR3-SDRAM software standby mode.

1. Continue applying a high level to the $\overline{\text{MBKPRST}}$ and SDBUP pins. Keep the high level also during software standby mode and release from software standby mode.
2. Halt all data accesses to the SDRAM. Disable the CPU interrupts to prevent the CPU from accessing the SDRAM due to the interrupt handling.
3. Set the ACEN bit in the SDRAM access enable register (DBACEN) to 0 (access disabled).
4. Use the manual command-issuing register (DBCMD) to issue a PREA (precharge all) command. The value written to this register should be opc = PreA, arg = 0.
5. Use the manual command-issuing register (DBCMD) to issue a Self-Refresh Entry command. The value written to this register should be opc = SREn, arg = 0.
6. Set the ARFEN bit in the auto-refresh enable register (DBRFEN) to 0.
7. Use the manual command-issuing register (DBCMD) to insert a time to wait for the clock to stop. The value written to this register should be opc = Wait, arg = tCKSRE (normally, max{5 clock cycles, cycles equivalent to 10 ns}).

8. Read the operation completion waiting register (DBWAIT) and wait for a response to be returned.
9. Set db_dllreset_n, db_dllenable1, db_dllenable2, and db_stby_n to 0 and db_comhiz to 1 in DDR-PHY unit control register 3 (DBPDCNT3).
10. Make a transition to software standby mode. Refer to section 9.2, Overview of Power-Down Modes, in section 9, Operating Modes and Power-Down Modes.

Use the following procedure to recover from DDR3-SDRAM software standby mode.

1. Perform the recovery processing from software standby mode. Refer to section 9.2, Overview of Power-Down Modes, in section 9, Operating Modes and Power-Down Modes.
2. Disable the CPU interrupts.
3. Set db_stby_n in DDR-PHY unit control register 3 (DBPDCNT3) to 1 and wait for at least 200 µs. To secure this waiting time, issue the Wait code by the manual command-issuing register (DBCMD) and wait for a response after reading the operation completion waiting register (DBWAIT). The waiting times in the following steps should also be secured by the same procedure.
4. Execute the following steps (1) to (4).
   (1) Set db_dllenable1 in DDR-PHY unit control register 3 (DBPDCNT3) to 1.
   (2) After at least 100 µs have elapsed, set db_dllenable2 in DDR-PHY unit control register 3 (DBPDCNT3) to 1.
   (3) After at least 16 clock cycles have elapsed, set db_dllreset_n in DDR-PHY unit control register 3 (DBPDCNT3) to 1.
   (4) Wait for at least 10,000 clock cycles.
5. Set db_comhiz in DDR-PHY unit control register 3 (DBPDCNT3) to 0 and wait for at least 5 clock cycles.
6. Use the manual command-issuing register (DBCMD) to insert the time to wait until release from self-refreshing. The value written to this register should be opc= Wait, arg = tCKSRX (normally, max{5 clock cycles, cycles equivalent to 10 ns}).
7. Read the operation completion waiting register (DBWAIT) and wait for a response to be returned.
8. Use the manual command-issuing register (DBCMD) to issue a Self-Refresh Exit command. The value written to this register should be opc = SRXt, arg = 0.
9. Set the ARFEN bit in the auto-refresh enable register (DBRFEN) to 1.
10. Use the manual command-issuing register (DBCMD) to insert the period of waiting until access to the SDRAM is enabled. The value written to this register should be opc = Wait, arg = tXSDLL (normally, 512 clock cycles).

11. Read the operation completion waiting register (DBWAIT) and wait for a response to be returned.
12. Set the ACEN bit in the SDRAM access enable register (DBACEN) to 1 (access enabled).
13. Enable the CPU interrupts.

4.3.7 SDRAM Power-Supply Backup Function

The SDRAM power-supply backup function utilizes the SDRAM self-refresh state to turn off the power supply to the most modules including the DBSC3 (except the VDD_DDR power), while maintaining the data in the SDRAM. By using this function, not only is it possible to cut power consumption, but the time needed to transfer data once again to the SDRAM can be eliminated, since the valid data is maintained within the SDRAM (see figure 4.1). In order to realize this function, a separate external control circuit (microcomputer or similar) is needed to monitor the states of this LSI and the memory.

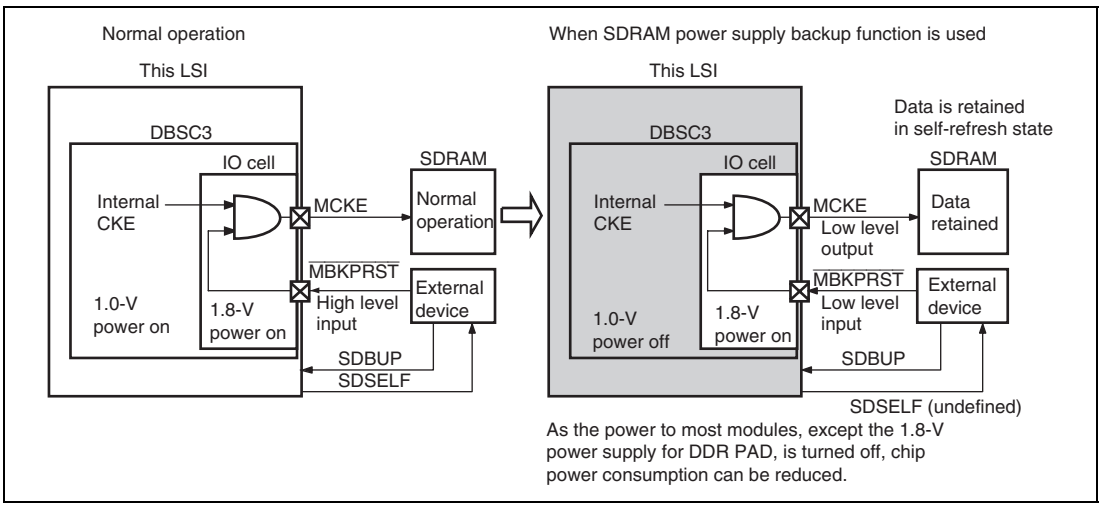


Figure 4.1 SDRAM Power-Supply Backup Function

To implement the power-supply backup function, a control signal, $\overline{\text{MBKPRST}}$, is necessary to hold MCKE at low level even when power other than for VDD_DDR is turned off. In addition, the SDBUP signal controls the inhibition of $\overline{\text{MRESET}}$ signal assertion for the DDR3-SDRAM. While the $\overline{\text{MBKPRST}}$ signal is at low level, the MCKE pin can be held at low level even when the power supply within the chip is in the turned-off state. By holding the SDBUP signal at high level, the $\overline{\text{MRESET}}$ pin for the DDR3-SDRAM can be held at high level. After putting the SDRAM into the self-refresh state, the $\overline{\text{MBKPRST}}$ and SDBUP signals can be used to hold the MCKE signal at low level and to hold the $\overline{\text{MRESET}}$ pin for the DDR3-SDRAM at high level, so the SDRAM self-refresh state can be maintained even when the power supply in the chip is turned off.

To cancel the power-supply backup state, perform a power-on reset. As a result, the DBSC3 registers are initialized, and so the self-refresh control circuit is also initialized. Even in this state, the MCKE pin can be held at low level by holding the $\overline{\text{MBKPRST}}$ signal at low level, and the $\overline{\text{MRESET}}$ pin for the DDR3-SDRAM can be held at high level by holding the SDBUP signal at high level. Power-on reset causes the DBSC3 to fix the internal CKE signal at low level, so that after power-on reset is released the $\overline{\text{MBKPRST}}$ signal is raised to high level. (If not in the power-supply backup state, $\overline{\text{MBKPRST}}$ is always at the high level and there is no problem).

Thus the power-supply backup state is cancelled through a power-on reset, and so the software must decide whether the normal SDRAM initialization sequence is necessary, or whether the LSI was in the power-supply backup state. For this decision, the DBSC3 drives the SDSELF signal high to notify external devices that self-refresh state has been entered. This SDSELF signal should be monitored by an external control circuit such as a microcomputer, and at recovery from power-supply backup state, a high-level signal should be input to this LSI through the SDBUP pin when DDR3-SDRAM is used or a status signal should be input through a general input pin when DDR2-SDRAM is used. After a power-on reset, the software should monitor the state signal input from an external control circuit (SDBUP signal for DDR3-SDRAM or general input signal for DDR2-SDRAM) and judge whether the state should be the power-supply backup state or whether SDRAM initialization is necessary. Note that the SDSELF signal is undefined in power-supply backup state.

These procedures are explained below.

(1) DDR2-SDRAM

A transition to the DDR2-SDRAM power-supply backup mode is performed with the following procedure.

1. The pin function controller (PFC) should be set up so that the SDSELF signal is selected.
2. Halt all data accesses to the SDRAM. Disable the CPU interrupts to prevent the CPU from accessing the SDRAM due to the interrupt handling.
3. Set the ACEN bit in the SDRAM access enable register (DBACEN) to 0 (access disabled).
4. Use the manual command-issuing register (DBCMD) to issue a PREA (precharge all) command. The value written to this register should be opc = PreA, arg = 0.
5. Use the manual command-issuing register (DBCMD) to issue a Self-Refresh Entry command. The value written to this register should be opc = SREn, arg = 0.
6. Set the ARFEN bit in the auto-refresh enable register (DBRFEN) to 0.

7. After at least 10 clock cycles have elapsed, set the db_iobackup bit in DDR-PHY unit control register 3 (DBPDCNT3) to 1 and wait for at least 30 clock cycles. To secure this waiting time, issue the Wait code by the manual command-issuing register (DBCMD) and wait for a response after reading the operation completion waiting register (DBWAIT).
8. Set db_dllreset_n, db_dllenable1, db_dllenable2, and db_stby_n to 0 and db_comhiz to 1 in DDR-PHY unit control register 3 (DBPDCNT3).
9. The DBSC3 drives the SDSELF signal high to convey that the SDRAM has entered the self-refresh state. Upon receiving this notification, the external device should change the $\overline{\text{MBKPRST}}$ signal from high level to low level.
10. At least 1 μs after the $\overline{\text{MBKPRST}}$ signal has been set to low level, turn off the unnecessary power, other than the VDD_DDR power.

Recovery from the DDR2-SDRAM power-supply backup mode is performed with the following procedure.

1. Turn on the power supply to the LSI while the $\overline{\text{MBKPRST}}$ signal is at the low level.
2. Input a power-on reset to the LSI.
3. After release of power-on reset, the external device should change the $\overline{\text{MBKPRST}}$ signal from low level to high level. Before driving the signal high, be sure to confirm that the $\overline{\text{PRESETOUT}}$ signal output from this LSI has changed to high level.
4. Through the general input pin, check whether it is in an initialization sequence of the SDRAM or in recovery from power-supply backup mode. For normal initialization sequence of the SDRAM, execute the procedure described in section 4.3.1, Initialization Sequence.
5. Make initial settings in DDR-PHY unit control registers 0, 1, and 3 (DBPDCNT0, DBPDCNT1, and DBPDCNT3). There is no particular order for the setting of the following steps (1) to (3).
   (1) Make settings in the db_offset, db_odt_tsel, db_odten_sel, db_odt_dis, and db_odt_mode bits in DDR-PHY unit control register 0 (DBPDCNT0).
   (2) Make settings in the db_ewc_open_offset0, db_ewc_strength0, and db_ewc_close_offset0 bits in DDR-PHY unit control register 1 (DBPDCNT1).
   (3) Set the db_calib_start, db_ioenable1, db_ioenable2, db_dllreset_n, db_dllenable1, db_dllenable2, db_iobackup, and db_comhiz bits to 0, and the db_ck_strength, db_dqs_strength, db_dqdm_strength, db_add_strength, and db_stby_n bits to 1 in DDR-PHY unit control register 3 (DBPDCNT3).
6. It is necessary to wait for at least 200 μs before a stable MCK clock can be output. To secure this waiting time, issue the Wait code by the manual command-issuing register (DBCMD) and wait for a response after reading the operation completion waiting register (DBWAIT). The waiting times in the following steps should also be secured by the same procedure.
7. Set the db_calib_start bit in DDR-PHY unit control register 3 (DBPDCNT3) to 1.

8. After at least 32 clock cycles have elapsed following the setting in step 7, set db_dllenable1 in DBPDCNT3 to 1.
9. After at least 100 µs have elapsed following the setting in step 8, set db_dllenable2 in DBPDCNT3 to 1.
10. After at least 16 clock cycles have elapsed following the setting in step 9, set db_dllreset_n in DBPDCNT3 to 1.
11. After at least 200 µs have elapsed following the setting in step 7, set db_ioenable1 in DBPDCNT3 to 1.
12. After at least 1 clock cycle has elapsed following the setting in step 11, set db_ioenable2 in DBPDCNT3 to 1.
13. Wait until 10,000 clock cycles have elapsed following the setting in step 9.
14. Use the following steps (1) and (2) to issue the refresh command twice. This is absolutely necessary for initializing the DDR-PHY.
    (1) Write opc = Ref, arg = 0 in the manual command-issuing register (DBCMD).
    (2) Write opc = Ref, arg = 0 in the manual command-issuing register (DBCMD) again.
15. Use the SDRAM kind setting register (DBKIND) to set the type of memory.
16. Set up the SDRAM configuration setting register (DBCONF), SDRAM timing registers 0 to 19 (DBTR0 to DBTR19), SDRAM operation setting register (DBBL), and ODT operation setting register (DBRNK0).
17. Use the manual command-issuing register (DBCMD) to make the DBSC3 recognize that self-refreshing is proceeding. The value written to this register should be opc = SREn, arg = 0.
18. If this is necessary, make settings in DBSC3 operation adjustment registers 0 to 2 (DBADJ0 to DBADJ2), bus control unit 0 control registers 0 and 1 (DBBS0CNT0 and DBBS0CNT1), and power-down configuration register (DBPDNCNF).
19. Set up refresh configuration registers 0 to 2 (DBRFCNF0 to DBRFCNF2).
20. Use the manual command-issuing register (DBCMD) to issue a Self-Refresh Exit command. The value written to this register should be opc = SRXt, arg = 0.
21. Set the ARFEN bit in the auto-refresh enable register (DBRFEN) to 1.
22. Use the manual command-issuing register (DBCMD) to insert the period of waiting until access to the SDRAM is enabled. The value written to this register should be opc = Wait, arg = 200.
23. Set the ACEN bit in the SDRAM access enable register (DBACEN) to 1 (access enabled).
24. Read the operation completion waiting register (DBWAIT) and wait for a response to be returned.

(2) DDR3-SDRAM

A transition to the DDR3-SDRAM power-supply backup mode is performed with the following procedure.

1. The pin function controller (PFC) should be set up so that the SDSELF signal is selected. The SDBUP pin should be driven to high level.
2. Halt all data accesses to the SDRAM. Disable the CPU interrupts to prevent the CPU from accessing the SDRAM due to the interrupt handling.
3. Set the ACEN bit in the SDRAM access enable register (DBACEN) to 0 (access disabled).
4. Use the manual command-issuing register (DBCMD) to issue a PREA (precharge all) command. The value written to this register should be opc = PreA, arg = 0.
5. Use the manual command-issuing register (DBCMD) to issue a Self-Refresh Entry command. The value written to this register should be opc = SREn, arg = 0.
6. Set the ARFEN bit in the auto-refresh enable register (DBRFEN) to 0.
7. After at least 10 clock cycles have elapsed, set the db_iobackup bit in DDR-PHY unit control register 3 (DBPDCNT3) to 1 and wait for at least 30 clock cycles. To secure this waiting time, issue the Wait code by the manual command-issuing register (DBCMD) and wait for a response after reading the operation completion waiting register (DBWAIT).
8. Set db_dllreset_n, db_dllenable1, db_dllenable2, and db_stby_n to 0 and db_comhiz to 1 in DDR-PHY unit control register 3 (DBPDCNT3).
9. The DBSC3 drives the SDSELF signal high to convey that the SDRAM has entered the self-refresh state. Upon receiving this notification, the external device should change the $\overline{\text{MBKPRST}}$ signal from high level to low level.
10. At least 1 μs after the $\overline{\text{MBKPRST}}$ signal has been set to low level, turn off the unnecessary power, other than the VDD_DDR power.

Recovery from the DDR3-SDRAM power-supply backup mode is performed with the following procedure.

1. Turn on the power supply to the LSI while the $\overline{\text{MBKPRST}}$ signal is at the low level and the SDBUP signal is at the high level.
2. Input a power-on reset to the LSI.
3. After release of power-on reset, the external device should change the $\overline{\text{MBKPRST}}$ signal from low level to high level. Before driving the signal high, be sure to confirm that the $\overline{\text{PRESETOUT}}$ signal output from this LSI has changed to high level. Be sure to keep the SDBUP signal at the high level even after this change.

4. Refer to the value of the BKUP bit in the DBSC3 status register (DBSTATE) to determine whether it is in an initialization sequence of the SDRAM or in recovery from power-supply backup mode. For normal initialization sequence of the SDRAM (the BKUP bit in DBSTATE is 0), execute the procedure described in section 4.3.1, Initialization Sequence.
5. Make initial settings in DDR-PHY unit control registers 0, 1, and 3 (DBPDCNT0, DBPDCNT1, and DBPDCNT3). There is no particular order for the setting of the following steps (1) to (3).
   (1) Make settings in the db_offset, db_odt_tsel, db_odten_sel, db_odt_dis, and db_odt_mode bits in DDR-PHY unit control register 0 (DBPDCNT0).
   (2) Make settings in the db_ewc_open_offset0, db_ewc_strength0, and db_ewc_close_offset0 bits in DDR-PHY unit control register 1 (DBPDCNT1).
   (3) Set the db_calib_start, db_ioenable1, db_ioenable2, db_ck_strength, db_dqs_strength, db_dqdm_strength, db_add_strength, db_dllreset_n, db_dllenable1, db_dllenable2, db_iobackup, and db_comhiz bits in DDR-PHY unit control register 3 (DBPDCNT3) to 0. Set the db_stby_n bit to 1.
6. It is necessary to wait for at least 200 µs before a stable MCK clock can be output. To secure this waiting time, issue the Wait code by the manual command-issuing register (DBCMD) and wait for a response after reading the operation completion waiting register (DBWAIT). The waiting times in the following steps should also be secured by the same procedure.
7. Set the db_calib_start bit in DDR-PHY unit control register 3 (DBPDCNT3) to 1.
8. After at least 32 clock cycles have elapsed following the setting in step 7, set db_dllenable1 in DBPDCNT3 to 1.
9. After at least 100 µs have elapsed following the setting in step 8, set db_dllenable2 in DBPDCNT3 to 1.
10. After at least 16 clock cycles have elapsed following the setting in step 9, set db_dllreset_n in DBPDCNT3 to 1.
11. After at least 200 µs have elapsed following the setting in step 7, set db_ioenable1 in DBPDCNT3 to 1.
12. After at least 1 clock cycle has elapsed following the setting in step 11, set db_ioenable2 in DBPDCNT3 to 1.
13. Wait until 10,000 clock cycles have elapsed following the setting in step 9.
14. Use the following steps (1) and (2) to issue the refresh command twice. This is absolutely necessary for initializing the DDR-PHY.
   (1) Write opc = Ref, arg = 0 in the manual command-issuing register (DBCMD).
   (2) Write opc = Ref, arg = 0 in the manual command-issuing register (DBCMD) again.
15. Use the SDRAM kind setting register (DBKIND) to set the type of memory.

16. Set up the SDRAM configuration setting register (DBCONF), SDRAM timing registers 0 to 19 (DBTR0 to DBTR19), SDRAM operation setting register (DBBL), and ODT operation setting register (DBRNK0).
17. Use the manual command-issuing register (DBCMD) to make the DBSC3 recognize that self-refreshing is proceeding as well as to insert the time to wait until release from self-refreshing. The value written to this register should be opc = SREn, arg = tCKSRX (normally, max {5 clock cycles, cycles equivalent to 10 ns}).
18. If this is necessary, make settings in DBSC3 operation adjustment registers 0 to 2 (DBADJ0 to DBADJ2), bus control unit 0 control registers 0 and 1 (DBBS0CNT0 and DBBS0CNT1), and power-down configuration register (DBPDNCNF).
19. Set up refresh configuration registers 0 to 2 (DBRFCNF0 to DBRFCNF2).
20. Set up the DDR3-SDRAM calibration configuration register (DBCALCNF).
21. Use the manual command-issuing register (DBCMD) to issue a Self-Refresh Exit command. The value written to this register should be opc = SRXt, arg = 0.
22. Set the ARFEN bit in the auto-refresh enable register (DBRFEN) to 1.
23. Use the manual command-issuing register (DBCMD) to insert the period of waiting until access to the SDRAM is enabled. The value written to this register should be opc = Wait, arg = tXSDLL (normally, 512 clock cycles).
24. Set the ACEN bit in the SDRAM access enable register (DBACEN) to 1 (access enabled).
25. Read the operation completion waiting register (DBWAIT) and wait for a response to be returned.

4.3.8 Retaining SDRAM Memory Contents in Deep Standby Mode when Power-Supply Backup Function is Also Used

When returning from deep standby mode, the LSI internal state is equivalent to the state at a power-on reset. The SDRAM memory contents however can be retained by performing the following processing.

This section describes the procedures for retaining SDRAM memory contents in deep standby mode when the power-supply backup function is also used. When the power-supply backup function is not used, the SDRAM memory contents can be retained in deep standby mode without using an external control circuit. For this procedure, refer to section 4.3.9, Retaining SDRAM Memory Contents in Deep Standby Mode when Power-Supply Backup Function is Not Used.

An external control circuit (microcomputer or similar) such as that used in section 4.3.7, SDRAM Power-Supply Backup Function, is separately required for implementing this function. For details, refer to section 4.3.7, SDRAM Power-Supply Backup Function.

For details on the deep standby function, entering deep standby mode, and returning from deep standby mode, refer to section 9.2, Overview of Power-Down Modes, in section 9, Operating Modes and Power-Down Modes.

To retain the SDRAM data, the procedure to perform before entering deep standby mode and the procedure to perform when returning from deep standby mode are described below.

(1) DDR2-SDRAM

Use the following procedure to make a transition to DDR2-SDRAM deep standby mode.

1. The pin function controller (PFC) should be set up so that the SDSELF signal is selected.
2. Halt all data accesses to the SDRAM. Disable the CPU interrupts to prevent the CPU from accessing the SDRAM due to the interrupt handling.
3. Set the ACEN bit in the SDRAM access enable register (DBACEN) to 0 (access disabled).
4. Use the manual command-issuing register (DBCMD) to issue a PREA (precharge all) command. The value written to this register should be opc = PreA, arg = 0.
5. Use the manual command-issuing register (DBCMD) to issue a Self-Refresh Entry command. The value written to this register should be opc = SREn, arg = 0.
6. Set the ARFEN bit in the auto-refresh enable register (DBRFEN) to 0.

7. After at least 10 clock cycles have elapsed, set the db_iobackup bit in DDR-PHY unit control register 3 (DBPDCNT3) to 1 and wait for at least 30 clock cycles. To secure this waiting time, issue the Wait code by the manual command-issuing register (DBCMD) and wait for a response after reading the operation completion waiting register (DBWAIT).
8. Set db_dllreset_n, db_dllenable1, db_dllenable2, and db_stby_n to 0 and db_comhiz to 1 in DDR-PHY unit control register 3 (DBPDCNT3).
9. The DBSC3 drives the SDSELF signal high to convey that the SDRAM has entered the self-refresh state. Upon receiving this notification, the external device should change the $\overline{\text{MBKPRST}}$ signal from high level to low level.
10. At least 1 µs after the $\overline{\text{MBKPRST}}$ signal has been set to low level, make a transition to deep standby mode.

Use the following procedure to recover from DDR2-SDRAM deep standby mode.

1. Generate a deep standby cancellation source with the $\overline{\text{MBKPRST}}$ pin kept at the low level.
2. Perform the recovery processing described in section 9.2, Overview of Power-Down Modes, in section 9, Operating Modes and Power-Down Modes. Note that DDR2-SDRAM can be accessed regardless of the IOKEEP flag state in DSFR.
3. After deep standby mode is canceled, the external control circuit changes the $\overline{\text{MBKPRST}}$ signal from low level to high level. Before driving the signal high, be sure to confirm that the $\overline{\text{PRESETOUT}}$ signal output from this LSI has changed to high level.
4. Through the general input pin, check whether it is in an initialization sequence of the SDRAM or in recovery from deep standby mode. For normal initialization sequence of the SDRAM, execute the procedure described in section 4.3.1, Initialization Sequence.
5. Make initial settings in DDR-PHY unit control registers 0, 1, and 3 (DBPDCNT0, DBPDCNT1, and DBPDCNT3). There is no particular order for the setting of the following steps (1) to (3).
   (1) Make settings in the db_offset, db_odt_tsel, db_odten_sel, db_odt_dis, and db_odt_mode bits in DDR-PHY unit control register 0 (DBPDCNT0).
   (2) Make settings in the db_ewc_open_offset0, db_ewc_strength0, and db_ewc_close_offset0 bits in DDR-PHY unit control register 1 (DBPDCNT1).
   (3) Set the db_calib_start, db_ioenable1, db_ioenable2, db_dllreset_n, db_dllenable1, db_dllenable2, db_iobackup, and db_comhiz bits to 0 and the db_ck_strength, db_dqs_strength, db_dqdm_strength, db_add_strength, and db_stby_n bits to 1 in DDR-PHY unit control register 3 (DBPDCNT3).
6. It is necessary to wait for at least 200 µs before a stable MCK clock can be output. To secure this waiting time, issue the Wait code by the manual command-issuing register (DBCMD) and wait for a response after reading the operation completion waiting register (DBWAIT). The waiting times in the following steps should also be secured by the same procedure.

7. Set the db_calib_start bit in DDR-PHY unit control register 3 (DBPDCNT3) to 1.
8. After at least 32 clock cycles have elapsed following the setting in step 7, set db_dllenable1 in DBPDCNT3 to 1.
9. After at least 100 µs have elapsed following the setting in step 8, set db_dllenable2 in DBPDCNT3 to 1.
10. After at least 16 clock cycles have elapsed following the setting in step 9, set db_dllreset_n in DBPDCNT3 to 1.
11. After at least 200 µs have elapsed following the setting in step 7, set db_ioenable1 in DBPDCNT3 to 1.
12. After at least 1 clock cycle has elapsed following the setting in step 11, set db_ioenable2 in DBPDCNT3 to 1.
13. Wait until 10,000 clock cycles have elapsed following the setting in step 9.
14. Use the following steps (1) and (2) to issue the refresh command twice. This is absolutely necessary for initializing the DDR-PHY.
    (1) Write opc = Ref, arg = 0 in the manual command-issuing register (DBCMD).
    (2) Write opc = Ref, arg = 0 in the manual command-issuing register (DBCMD) again.
15. Use the SDRAM kind setting register (DBKIND) to set the type of memory.
16. Set up the SDRAM configuration setting register (DBCONF), SDRAM timing registers 0 to 19 (DBTR0 to DBTR19), SDRAM operation setting register (DBBL), and ODT operation setting register (DBRNK0).
17. Use the manual command-issuing register (DBCMD) to make the DBSC3 recognize that self-refreshing is proceeding. The value written to this register should be opc = SREn, arg = 0.
18. If this is necessary, make settings in DBSC3 operation adjustment registers 0 to 2 (DBADJ0 to DBADJ2), bus control unit 0 control registers 0 and 1 (DBBS0CNT0 and DBBS0CNT1), and power-down configuration register (DBPDNCNF).
19. Set up refresh configuration registers 0 to 2 (DBRFCNF0 to DBRFCNF2).
20. Use the manual command-issuing register (DBCMD) to issue a Self-Refresh Exit command. The value written to this register should be opc = SRXt, arg = 0.
21. Set the ARFEN bit in the auto-refresh enable register (DBRFEN) to 1.
22. Use the manual command-issuing register (DBCMD) to insert the period of waiting until access to the SDRAM is enabled. The value written to this register should be opc = Wait, arg = 200.
23. Set the ACEN bit in the SDRAM access enable register (DBACEN) to 1 (access enabled).
24. Read the operation completion waiting register (DBWAIT) and wait for a response to be returned.

(2) DDR3-SDRAM

Use the following procedure to make a transition to DDR3-SDRAM deep standby mode.

1. The pin function controller (PFC) should be set up so that the SDSELF signal is selected. The SDBUP pin should be driven to high level.
2. Halt all data accesses to the SDRAM. Disable the CPU interrupts to prevent the CPU from accessing the SDRAM due to the interrupt handling.
3. Set the ACEN bit in the SDRAM access enable register (DBACEN) to 0 (access disabled).
4. Use the manual command-issuing register (DBCMD) to issue a PREA (precharge all) command. The value written to this register should be opc = PreA, arg = 0.
5. Use the manual command-issuing register (DBCMD) to issue a Self-Refresh Entry command. The value written to this register should be opc = SREn, arg = 0.
6. Set the ARFEN bit in the auto-refresh enable register (DBRFEN) to 0.
7. After at least 10 clock cycles have elapsed, set the db_iobackup bit in DDR-PHY unit control register 3 (DBPDCNT3) to 1 and wait for at least 30 clock cycles. To secure this waiting time, issue the Wait code by the manual command-issuing register (DBCMD) and wait for a response after reading the operation completion waiting register (DBWAIT).
8. Set db_dllreset_n, db_dllenable1, db_dllenable2, and db_stby_n to 0 and db_comhiz to 1 in DDR-PHY unit control register 3 (DBPDCNT3).
9. The DBSC3 drives the SDSELF signal high to convey that the SDRAM has entered the self-refresh state. Upon receiving this notification, the external device should change the $\overline{\text{MBKPRST}}$ signal from high level to low level.
10. At least 1 μs after the $\overline{\text{MBKPRST}}$ signal has been set to low level, make a transition to deep standby mode.

Use the following procedure to recover from DDR3-SDRAM deep standby mode.

1. Generate a deep standby cancellation source with the $\overline{\text{MBKPRST}}$ pin kept at the low level and the SDBUP pin kept at the high level.
2. Perform the recovery processing described in section 9.2, Overview of Power-Down Modes in section 9, Operating Modes and Power-Down Modes. Note that DDR3-SDRAM can be accessed regardless of the IOKEEP flag state in DSFR.
3. After deep standby mode is canceled, the external control circuit changes the $\overline{\text{MBKPRST}}$ signal from low level to high level. Before driving the signal high, be sure to confirm that the $\overline{\text{PRESETOUT}}$ signal output from this LSI has changed to high level. Be sure to keep the SDBUP signal at the high level even after this change.

4. Refer to the value of the BKUP bit in the DBSC3 status register (DBSTATE) to determine whether it is in an initialization sequence of the SDRAM or in recovery from deep standby mode. For normal initialization sequence of the SDRAM (the BKUP bit in DBSTATE is 0), execute the procedure described in section 4.3.1, Initialization Sequence.
5. Make initial settings in DDR-PHY unit control registers 0, 1, and 3 (DBPDCNT0, DBPDCNT1, and DBPDCNT3). There is no particular order for the setting of the following steps (1) to (3).
   (1) Make settings in the db_offset, db_odt_tsel, db_odten_sel, db_odt_dis, and db_odt_mode bits in DDR-PHY unit control register 0 (DBPDCNT0).
   (2) Make settings in the db_ewc_open_offset0, db_ewc_strength0, and db_ewc_close_offset0 bits in DDR-PHY unit control register 1 (DBPDCNT1).
   (3) Set the db_calib_start, db_ioenable1, db_ioenable2, db_ck_strength, db_dqs_strength, db_dqdm_strength, db_add_strength, db_dllreset_n, db_dllenable1, db_dllenable2, db_iobackup, and db_comhiz bits in DDR-PHY unit control register 3 (DBPDCNT3) to 0. Set the db_stby_n bit to 1.
6. It is necessary to wait for at least 200 µs before a stable MCK clock can be output. To secure this waiting time, issue the Wait code by the manual command-issuing register (DBCMD) and wait for a response after reading the operation completion waiting register (DBWAIT). The waiting times in the following steps should also be secured by the same procedure.
7. Set the db_calib_start bit in DDR-PHY unit control register 3 (DBPDCNT3) to 1.
8. After at least 32 clock cycles have elapsed following the setting in step 7, set db_dllenable1 in DBPDCNT3 to 1.
9. After at least 100 µs have elapsed following the setting in step 8, set db_dllenable2 in DBPDCNT3 to 1.
10. After at least 16 clock cycles have elapsed following the setting in step 9, set db_dllreset_n in DBPDCNT3 to 1.
11. After at least 200 µs have elapsed following the setting in step 7, set db_ioenable1 in DBPDCNT3 to 1.
12. After at least 1 clock cycle has elapsed following the setting in step 11, set db_ioenable2 in DBPDCNT3 to 1.
13. Wait until 10,000 clock cycles have elapsed following the setting in step 9.
14. Use the following steps (1) and (2) to issue the refresh command twice. This is absolutely necessary for initializing the DDR-PHY.
   (1) Write opc = Ref, arg = 0 in the manual command-issuing register (DBCMD).
   (2) Write opc = Ref, arg = 0 in the manual command-issuing register (DBCMD) again.
15. Use the SDRAM kind setting register (DBKIND) to set the type of memory.

16. Set up the SDRAM configuration setting register (DBCONF), SDRAM timing registers 0 to 19 (DBTR0 to DBTR19), SDRAM operation setting register (DBBL), and ODT operation setting register (DBRNK0).
17. Use the manual command-issuing register (DBCMD) to make the DBSC3 recognize that self-refreshing is proceeding as well as to insert the time to wait until release from self-refreshing. The value written to this register should be opc = SREn, arg = tCKSRX (normally, max{5 clock cycles, cycles equivalent to 10 ns}).
18. If this is necessary, make settings in DBSC3 operation adjustment registers 0 to 2 (DBADJ0 to DBADJ2), bus control unit 0 control registers 0 and 1 (DBBS0CNT0 and DBBS0CNT1), and power-down configuration register (DBPDNCNF).
19. Set up refresh configuration registers 0 to 2 (DBRFCNF0 to DBRFCNF2).
20. Set up the DDR3-SDRAM calibration configuration register (DBCALCNF).
21. Use the manual command-issuing register (DBCMD) to issue a Self-Refresh Exit command. The value written to this register should be opc = SRXt, arg = 0.
22. Set the ARFEN bit in the auto-refresh enable register (DBRFEN) to 1.
23. Use the manual command-issuing register (DBCMD) to insert the period of waiting until access to the SDRAM is enabled. The value written to this register should be opc = Wait, arg = tXSDLL (normally, 512 clock cycles).
24. Set the ACEN bit in the SDRAM access enable register (DBACEN) to 1 (access enabled).
25. Read the operation completion waiting register (DBWAIT) and wait for a response to be returned.

4.3.9 Retaining SDRAM Memory Contents in Deep Standby Mode when Power-Supply Backup Function is Not Used

When returning from deep standby mode, the LSI internal state is equivalent to the state at a power-on reset. The SDRAM memory contents however can be retained by performing the following processing.

This section describes the procedures for retaining SDRAM memory contents in deep standby mode when the power-supply backup function is not used. When the power-supply backup function is used, refer to section 4.3.8, Retaining SDRAM Memory Contents in Deep Standby Mode when Power-Supply Backup Function is Also Used.

When the power-supply backup function is not used, no external control circuit (such as a microcomputer) is required.

For details on the deep standby function, entering deep standby mode, and returning from deep standby mode, refer to section 9.2, Overview of Power-Down Modes, in section 9, Operating Modes and Power-Down Modes.

(1) Handling of SDSELF, SDBUP, and $\overline{\text{MBKPRST}}$ Pins

The SDSELF pin is not used for either DDR2-SDRAM or DDR3-SDRAM. The $\overline{\text{MBKPRST}}$ pin should be fixed to high level (for example, pulled up to the VDD_DDR power) for both SDRAM types.

When DDR2-SDRAM is used, the SDBUP pin should be fixed to low level. When DDR3-SDRAM is used, a general I/O pin of this LSI should be selected for SDBUP control, connected to the SDBUP pin on the board, and pulled down. A general I/O pin satisfying all of the following conditions should be selected.

- The general input function is selected after a power-on reset.
- The pull-down function is disabled after a power-on reset.
- The mode setting function is not selected after a power-on reset.

(2) Initial Settings for Controlling SDBUP Pin

When DDR3-SDRAM is used, initial settings for controlling the SDBUP pin should be included in the initialization sequence to be executed during the power-on reset processing. Replace step 7 described in section 4.3.1 (2), DDR3-SDRAM, with the four steps described below. The following shows an example where the DACK0 pin is selected as the general I/O pin for controlling SDBUP. When selecting another pin, refer to section 36, GPIO, and section 37, Pin Function Controller (PFC), and use the control registers corresponding to the selected pin. When DDR2-SDRAM is used, the following steps are not necessary.

1. Set the POSNEG1[3] bit to 0 to select positive logic for the DACK0 pin output.
2. Set the IOINTSEL1[3] bit to 0 to select the general I/O function for the DACK0 pin.
3. Set the OUTDT1[3] bit to 1 to output a high level through the DACK0 pin.
4. Set the INOUTSEL1[3] bit to 1 to select the general output function for the DACK0 pin.

(3) DDR2-SDRAM

Use the following procedure to make a transition to DDR2-SDRAM deep standby mode.

1. Halt all data accesses to the SDRAM. Disable the CPU interrupts to prevent the CPU from accessing the SDRAM due to the interrupt handling.
2. Set the ACEN bit in the SDRAM access enable register (DBACEN) to 0 (access disabled).
3. Use the manual command-issuing register (DBCMD) to issue a PREA (precharge all) command. The value written to this register should be opc = PreA, arg = 0.
4. Use the manual command-issuing register (DBCMD) to issue a Self-Refresh Entry command. The value written to this register should be opc = SREn, arg = 0.
5. Set the ARFEN bit in the auto-refresh enable register (DBRFEN) to 0.
6. After at least 10 clock cycles have elapsed, set the db_iobackup bit in DDR-PHY unit control register 3 (DBPDCNT3) to 1 and wait for at least 30 clock cycles. To secure this waiting time, issue the Wait code by the manual command-issuing register (DBCMD) and wait for a response after reading the operation completion waiting register (DBWAIT).
7. Set db_dllreset_n, db_dllenable1, db_dllenable2, and db_stby_n to 0 and db_comhiz to 1 in DDR-PHY unit control register 3 (DBPDCNT3).
8. Make a transition to deep standby mode.

Use the following procedure to recover from DDR2-SDRAM deep standby mode.

1. Generate a deep standby cancellation source.

2. Perform the recovery processing described in section 9.2, Overview of Power-Down Modes in section 9, Operating Modes and Power-Down Modes. If DSFR is set to H'0000 at this time, execute the procedure described in section 4.3.1, Initialization Sequence. If DSFR is not set to H'0000, the current state is recovery from deep standby mode; execute the following procedure for recovery from deep standby mode. Note that DDR2-SDRAM can be accessed regardless of the IOKEEP flag state in DSFR.
3. Clear the IOKEEP flag in DSFR.
4. Make initial settings in DDR-PHY unit control registers 0, 1, and 3 (DBPDCNT0, DBPDCNT1, and DBPDCNT3). There is no particular order for the setting of the following steps (1) to (3).
   (1) Make settings in db_offset, db_odt_tsel, db_odten_sel, db_odt_dis, and db_odt_mode in DDR-PHY unit control register 0 (DBPDCNT0).
   (2) Make settings in db_ewc_open_offset0, db_ewc_strength0, and db_ewc_close_offset0 in DDR-PHY unit control register 1 (DBPDCNT1).
   (3) Set db_calib_start, db_ioenable1, db_ioenable2, db_dllreset_n, db_dllenable1, db_dllenable2, db_iobackup, and db_comhiz to 0, and db_ck_strength, db_dqs_strength, db_dqdm_strength, db_add_strength, and db_stby_n to 1 in DDR-PHY unit control register 3 (DBPDCNT3).
5. It is necessary to wait for at least 200 µs before a stable MCK clock can be output. To secure this waiting time, issue the Wait code by the manual command-issuing register (DBCMD) and wait for a response after reading the operation completion waiting register (DBWAIT). The waiting times in the following steps should also be secured by the same procedure.
6. Set db_calib_start in DDR-PHY unit control register 3 (DBPDCNT3) to 1.
7. After at least 32 clock cycles have elapsed following the setting in step 6, set db_dllenable1 in DBPDCNT3 to 1.
8. After at least 100 µs have elapsed following the setting in step 7, set db_dllenable2 in DBPDCNT3 to 1.
9. After at least 16 clock cycles have elapsed following the setting in step 8, set db_dllreset_n in DBPDCNT3 to 1.
10. After at least 200 µs have elapsed following the setting in step 6, set db_ioenable1 in DBPDCNT3 to 1.
11. After at least one clock cycle has elapsed following the setting in step 10, set db_ioenable2 in DBPDCNT3 to 1.
12. Wait until 10,000 clock cycles have elapsed following the setting in step 8.
13. Use the following steps (1) and (2) to issue the refresh command twice. This is absolutely necessary for initializing the DDR-PHY.
   (1) Write opc = Ref, arg = 0 in the manual command-issuing register (DBCMD).
   (2) Write opc = Ref, arg = 0 in the manual command-issuing register (DBCMD) again.

14. Use the SDRAM kind setting register (DBKIND) to set the type of memory.
15. Set up the SDRAM configuration setting register (DBCONF), SDRAM timing registers 0 to 19 (DBTR0 to DBTR19), SDRAM operation setting register (DBBL), and ODT operation setting register (DBRNK0).
16. Use the manual command-issuing register (DBCMD) to make the DBSC3 recognize that self-refreshing is proceeding. The value written to this register should be opc = SREn, arg = 0.
17. If this is necessary, make settings in DBSC3 operation adjustment registers 0 to 2 (DBADJ0 to DBADJ2), bus control unit 0 control registers 0 and 1 (DBBS0CNT0 and DBBS0CNT1), and power-down configuration register (DBPDNCNF).
18. Set up refresh configuration registers 0 to 2 (DBRFCNF0 to DBRFCNF2).
19. Use the manual command-issuing register (DBCMD) to issue a Self-Refresh Exit command. The value written to this register should be opc = SRXt, arg = 0.
20. Set the ARFEN bit in the auto-refresh enable register (DBRFEN) to 1.
21. Use the manual command-issuing register (DBCMD) to insert the period of waiting until access to the SDRAM is enabled. The value written to this register should be opc = Wait, arg = 200.
22. Set the ACEN bit in the SDRAM access enable register (DBACEN) to 1 (access enabled).
23. Read the operation completion waiting register (DBWAIT) and wait for a response to be returned.

(4) DDR3-SDRAM

Use the following procedure to make a transition to DDR3-SDRAM deep standby mode. The initialization sequence after a power-on reset should have been executed with the inclusion of the steps described in section 4.3.9 (2), Initial Settings for Controlling SDBUP Pin.

1. Halt all data accesses to the SDRAM. Disable the CPU interrupts to prevent the CPU from accessing the SDRAM due to the interrupt handling.
2. Set the ACEN bit in the SDRAM access enable register (DBACEN) to 0 (access disabled).
3. Use the manual command-issuing register (DBCMD) to issue a PREA (precharge all) command. The value written to this register should be opc = PreA, arg = 0.
4. Use the manual command-issuing register (DBCMD) to issue a Self-Refresh Entry command. The value written to this register should be opc = SREn, arg = 0.
5. Set the ARFEN bit in the auto-refresh enable register (DBRFEN) to 0.
6. After at least 10 clock cycles have elapsed, set the db_iobackup bit in DDR-PHY unit control register 3 (DBPDCNT3) to 1 and wait for at least 30 clock cycles. To secure this waiting time, issue the Wait code by the manual command-issuing register (DBCMD) and wait for a response after reading the operation completion waiting register (DBWAIT).

7. Set db_dllreset_n, db_dllenable1, db_dllenable2, and db_stby_n to 0 and db_comhiz to 1 in DDR-PHY unit control register 3 (DBPDCNT3).
8. Make a transition to deep standby mode.

Use the following procedure to recover from DDR3-SDRAM deep standby mode.

1. Generate a deep standby cancellation source.
2. Perform the recovery processing described in section 9.2, Overview of Power-Down Modes in section 9, Operating Modes and Power-Down Modes. If DSFR is set to H'0000 at this time, execute the procedure described in section 4.3.1, Initialization Sequence. If DSFR is not set to H'0000, the current state is recovery from deep standby mode; execute the following procedure for recovery from deep standby mode. Here, do not clear the IOKEEP flag in DSFR (settings can be made for DDR3-SDRAM regardless of the IOKEEP flag state in DSFR).
3. Make initial settings in DDR-PHY unit control registers 0, 1, and 3 (DBPDCNT0, DBPDCNT1, and DBPDCNT3). There is no particular order for the setting of the following steps (1) to (3).
   (1) Make settings in the db_offset, db_odt_tsel, db_odten_sel, db_odt_dis, and db_odt_mode bits in DDR-PHY unit control register 0 (DBPDCNT0).
   (2) Make settings in the db_ewc_open_offset0, db_ewc_strength0, and db_ewc_close_offset0 bits in DDR-PHY unit control register 1 (DBPDCNT1).
   (3) Set the db_calib_start, db_ioenable1, db_ioenable2, db_ck_strength, db_dqs_strength, db_dqdm_strength, db_add_strength, db_dllreset_n, db_dllenable1, db_dllenable2, db_iobackup, and db_comhiz bits in DDR-PHY unit control register 3 (DBPDCNT3) to 0. Set the db_stby_n bit to 1.
4. It is necessary to wait for at least 200 µs before a stable MCK clock can be output. To secure this waiting time, issue the Wait code by the manual command-issuing register (DBCMD) and wait for a response after reading the operation completion waiting register (DBWAIT). The waiting times in the following steps should also be secured by the same procedure.
5. Set db_calib_start in DDR-PHY unit control register 3 (DBPDCNT3) to 1.
6. After at least 32 clock cycles have elapsed following the setting in step 5, set db_dllenable1 in DBPDCNT3 to 1.
7. After at least 100 µs have elapsed following the setting in step 6, set db_dllenable2 in DBPDCNT3 to 1.
8. Execute the four steps described in section 4.3.9 (2), Initial Settings for Controlling SDBUP Pin.
9. After at least 16 clock cycles have elapsed following the setting in step 7, set db_dllreset_n in DBPDCNT3 to 1.

10. After at least 200 µs have elapsed following the setting in step 5, set db_ioenable1 in DBPDCNT3 to 1.
11. After at least one clock cycle has elapsed following the setting in step 10, set db_ioenable2 in DBPDCNT3 to 1.
12. Wait until 10,000 clock cycles have elapsed following the setting in step 9.
13. Use the following steps (1) and (2) to issue the refresh command twice. This is absolutely necessary for initializing the DDR-PHY.
    (1) Write opc = Ref, arg = 0 in the manual command-issuing register (DBCMD).
    (2) Write opc = Ref, arg = 0 in the manual command-issuing register (DBCMD) again.
14. Use the SDRAM kind setting register (DBKIND) to set the type of memory.
15. Set up the SDRAM configuration setting register (DBCONF), SDRAM timing registers 0 to 19 (DBTR0 to DBTR19), SDRAM operation setting register (DBBL), and ODT operation setting register (DBRNK0).
16. Use the manual command-issuing register (DBCMD) to make the DBSC3 recognize that self-refreshing is proceeding as well as to insert the time to wait until release from self-refreshing. The value written to this register should be opc = SREn, arg = tCKSRX (normally, max{5 clock cycles, cycles equivalent to 10 ns}).
17. If this is necessary, make settings in DBSC3 operation adjustment registers 0 to 2 (DBADJ0 to DBADJ2), bus control unit 0 control registers 0 and 1 (DBBS0CNT0 and DBBS0CNT1), and power-down configuration register (DBPDNCNF).
18. Set up refresh configuration registers 0 to 2 (DBRFCNF0 to DBRFCNF2).
19. Set up the DDR3-SDRAM calibration configuration register (DBCALCNF).
20. Use the manual command-issuing register (DBCMD) to issue a Self-Refresh Exit command. The value written to this register should be opc = SRXt, arg = 0.
21. Set the ARFEN bit in the auto-refresh enable register (DBRFEN) to 1.
22. Use the manual command-issuing register (DBCMD) to insert the period of waiting until access to the SDRAM is enabled. The value written to this register should be opc = Wait, arg = tXSDLL (normally, 512 clock cycles).
23. Set the ACEN bit in the SDRAM access enable register (DBACEN) to 1 (access enabled).
24. Read the operation completion waiting register (DBWAIT) and wait for a response to be returned.
25. Clear the IOKEEP flag in DSFR.

4.3.10 Open/Short Test for MZQ Pin

The MZQ pin should be connected to GND via a 120-Ω resistor (tolerance of 1% or better). The state of this pin connection can be tested through register settings using the following procedure.

1. Initialize the DDR-PHY unit and controller, and enable DDR memory access (refer to section 4.3.1, Initialization Sequence).
2. Write H'0000A55A to the DDR-PHY unit lock register (DBPDLCK: H'FE800280) (PLOCK = H'A55A)
3. Write H'00000000 to the DDR-PHY unit register address (DBPDRGA: H'FE800290) (PRA = H'00)
4. Write H'A5000000 to the DDR-PHY unit register access (DBPDRGD: H'FE8002A0) (PRD = H'A5000000)
5. Write H'00000014 to the DDR-PHY unit register address (DBPDRGA: H'FE800290) (PRA = H'14)
6. Read the DDR-PHY unit register access (DBPDRGD: H'FE8002A0) and check the connection state indicated in the PRD[21:16] and PRD[5:0] bits.

When DDR2 is Used:

PRD[21:16] and PRD[5:0] (Read Values)	MZQ Pin State
PRD[21:16] < H'0C or PRD[5:0] < H'0E	Open (resistance is large) or short-circuit with power supply
PRD[21:16] > H'2E or PRD[5:0] > H'2E	Short-circuit with GND (resistance is small)
H'0C ≤ PRD[21:16] ≤ H'2E and H'0E ≤ PRD[5:0] ≤ H'2E	Normal

When DDR3 is Used:

PRD[21:16] and PRD[5:0] (Read Values)	MZQ Pin State
PRD[21:16] < H'09 or PRD[5:0] < H'0C	Open (resistance is large) or short-circuit with power supply
PRD[21:16] > H'34 or PRD[5:0] > H'34	Short-circuit with GND (resistance is small)
H'09 ≤ PRD[21:16] ≤ H'34 and H'0C ≤ PRD[5:0] ≤ H'34	Normal

7. Write H'00000000 to the DDR-PHY unit register address (DBPDRGA: H'FE800290) (PRA = H'00)
8. Write H'00000000 to the DDR-PHY unit register access (DBPDRGD: H'FE8002A0) (PRD = H'00000000)

9. Write H'00000000 to the DDR-PHY unit lock register (DBPDLCK: H'FE800280) (PLOCK = H'0000)
10. End of the MZQ pin test (normal access can be done).

4.4 Setting the SDRAM Configuration Setting Register

The setting values of the SDRAM configuration setting register (DBCONF) are shown below.

4.4.1 DDR2-SDRAM (16-Bit External Bus)

Table 4.5 SDRAM Configuration Setting Register (DDR2-SDRAM)

Memory Configuration	Bank [No. of Banks]	Row [No. of Bits]	Column [No. of Bits]	DBCONF Setting				
				AWRK0	AWBK0	AWRW0	AWCL0	DW0
16 M × 16 bits, 256 Mbits (1 module)	4	13	9	0	10	01101	1001	01
32 M × 8 bits, 256 Mbits (2 modules)	4	13	10	0	10	01101	1010	01
32 M × 16 bits, 512 Mbits (1 module)	4	13	10	0	10	01101	1010	01
64 M × 8 bits, 512 Mbits (2 modules)	4	14	10	0	10	01110	1010	01
64 M × 16 bits, 1 Gbits (1 module)	8	13	10	0	11	01101	1010	01
128 M × 8 bits, 1 Gbits (2 modules)	8	14	10	0	11	01110	1010	01
128 M × 16 bits, 2 Gbits (1 module)	8	14	10	0	11	01110	1010	01

4.4.2 DDR3-SDRAM (16-Bit External Bus)

Table 4.6 SDRAM Configuration Setting Register (DDR3-SDRAM)

Memory Configuration	Bank [No. of Banks]	Row [No. of Bits]	Column [No. of Bits]	DBCONF Setting				
				AWRK0	AWBK0	AWRW0	AWCL0	DW0
32 M × 16 bits, 512 Mbits (1 module)	8	12	10	0	11	01100	1010	01
64 M × 8 bits, 512 Mbits (2 modules)	8	13	10	0	11	01101	1010	01
64 M × 16 bits, 1 Gbits (1 module)	8	13	10	0	11	01101	1010	01
128 M × 8 bits, 1 Gbits (2 modules)	8	14	10	0	11	01110	1010	01
128 M × 16 bits, 2 Gbits (1 module)	8	14	10	0	11	01110	1010	01

4.5 Relation between External Pins and Logical Addresses

Following shows the relation between SDRAM external pins and logical addresses on the SHwy transaction.

4.5.1 DDR2-SDRAM

Table 4.7 Relation between SDRAM Address Pins and Logical Addresses when the External Data Bus Width is Set to 16 Bits

(When One 16-bit-width SDRAM Module or Two 8-bit-width SDRAM Modules are Connected)

Memory Type		MBA2	MBA1	MBA0	MA13	MA12	MA11	MA10	MA9	MA8	MA7	MA6	MA5	MA4	MA3	MA2	MA1	MA0
16 M × 16 bits	ROW	—	A11	A10	—	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14	A13	A12
	COL	—	A11	A10	—	—	—	AP	—	A9	A8	A7	A6	A5	A4	A3	A2	A1
32 M × 8 bits	ROW	—	A12	A11	—	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14	A13
	COL	—	A12	A11	—	—	—	AP	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1
32 M × 16 bits	ROW	—	A12	A11	—	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14	A13
	COL	—	A12	A11	—	—	—	AP	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1
64 M × 8 bits	ROW	—	A12	A11	A26	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14	A13
	COL	—	A12	A11	—	—	—	AP	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1
64 M × 16 bits	ROW	A13	A12	A11	—	A26	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14
	COL	A13	A12	A11	—	—	—	AP	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1
128 M × 8 bits	ROW	A13	A12	A11	A27	A26	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14
	COL	A13	A12	A11	—	—	—	AP	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1
128 M × 16 bits	ROW	A13	A12	A11	A27	A26	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14
	COL	A13	A12	A11	—	—	—	AP	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1

Notes: 1. A31 to A0 are the logical address bits in byte-unit. A31 represents the MSB, and A0 the LSB.
2. AP is an abbreviation of auto precharge option.
3. 64 M × 16 bits, 128 M × 8 bits, and 128 M × 16 bits are 8-bank products.

4.5.2 DDR3-SDRAM

Table 4.8 Relation between SDRAM Address Pins and Logical Addresses when the External Data Bus Width is Set to 16 Bits

(When One 16-bit-width SDRAM Module or Two 8-bit-width SDRAM Modules are Connected)

Memory Type		MBA2	MBA1	MBA0	MA13	MA12	MA11	MA10	MA9	MA8	MA7	MA6	MA5	MA4	MA3	MA2	MA1	MA0
32 M × 16 bits	ROW	A13	A12	A11	—	—	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14
	COL	A13	A12	A11	—	$\overline{BC}$	—	AP	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1
64 M × 8 bits	ROW	A13	A12	A11	—	A26	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14
	COL	A13	A12	A11	—	$\overline{BC}$	—	AP	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1
64 M × 16 bits	ROW	A13	A12	A11	—	A26	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14
	COL	A13	A12	A11	—	$\overline{BC}$	—	AP	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1
128 M × 8 bits	ROW	A13	A12	A11	A27	A26	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14
	COL	A13	A12	A11	—	$\overline{BC}$	—	AP	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1
128 M × 16 bits	ROW	A13	A12	A11	A27	A26	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14
	COL	A13	A12	A11	—	$\overline{BC}$	—	AP	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1

Notes: 1. A31 to A0 are the logical address bits in byte-unit. A31 represents the MSB, and A0 the LSB.
2. AP is an abbreviation of auto precharge option.
3. $\overline{BC}$ is an abbreviation of burst chop.

4.6 Address Location Specification for Bank Address

Specifying the address location of the bank address is described here. The address location of the bank address can be specified independent of the SDRAM configuration. The bank addresses can be specified as either continuous addresses or discontinuous addresses.

4.6.1 Bank Address Setting Combinations

The values that can be set in the bus control unit 0 control register 1 (DBBS0CNT1) are shown below.

Table 4.9 DBBS0CNT1 Settings

BKADM	BKADP	BKADB
00	000000	000000
00	001010	000000
00	001011	000000
00	001100	000000
01	000000	001101
01	000000	001110
01	000000	001111
01	000000	010000
01	001010	001101
01	001010	001110
01	001010	001111
01	001010	010000
01	001011	001101
01	001011	001110
01	001011	001111
01	001011	010000
01	001100	001101
01	001100	001110
01	001100	001111
01	001100	010000

4.6.2 Settings for the Contiguous Address

Set BKADM to 00 for the contiguous address. In this case, the bank address location is specified by BKADP. The bank address location settings can be selected either setting in the upper column address or specifying the address location. In a case that the address location is specified, specifies the lower bit location of the bank address (BA0).

The following shows a case that one DDR2-SDRAM (32 M × 16 bits) is connected.

Table 4.10 BKADM = 00, BKADP = 000000

Type		MBA2	MBA1	MBA0	MA13	MA12	MA11	MA10	MA9	MA8	MA7	MA6	MA5	MA4	MA3	MA2	MA1	MA0
32 M × 16 bits	ROW	—	A12	A11	—	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14	A13
	COL	—	A12	A11	—	—	—	AP	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1

Table 4.11 BKADM = 00, BKADP = 001010

Type		MBA2	MBA1	MBA0	MA13	MA12	MA11	MA10	MA9	MA8	MA7	MA6	MA5	MA4	MA3	MA2	MA1	MA0
32 M × 16 bits	ROW	—	A11	A10	—	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14	A13
	COL	—	A11	A10	—	—	—	AP	A12	A9	A8	A7	A6	A5	A4	A3	A2	A1

Table 4.12 BKADM = 00, BKADP = 001100

Type		MBA2	MBA1	MBA0	MA13	MA12	MA11	MA10	MA9	MA8	MA7	MA6	MA5	MA4	MA3	MA2	MA1	MA0
32 M × 16 bits	ROW	—	A13	A12	—	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14	A11
	COL	—	A13	A12	—	—	—	AP	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1

4.6.3 Settings for the Non-Contiguous Address

Set BKADM to 01 for the non-contiguous address. In this case, the lower bank address location is specified by BKADP and the upper bank address location is specified by BKADB. The lower bank address location settings can be selected either setting in the upper column address or specifying the address location. In a case that the address location is specified, specifies the lower bit location of the bank address (BA0). The address location for the upper bank address location can be specified. In a case that the address location is specified, specifies the upper bit location of the bank address (BA1). BA2 is located in the upper of BA1 for the 8-bank product.

The following shows a case that one DDR2-SDRAM (64 M × 16 bits) is connected.

Table 4.13 BKADM = 01, BKADP = 000000, BKADB = 001101

Type		MBA2	MBA1	MBA0	MA13	MA12	MA11	MA10	MA9	MA8	MA7	MA6	MA5	MA4	MA3	MA2	MA1	MA0
64 M × 16 bits	ROW	A14	A13	A11	—	A26	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A12
	COL	A14	A13	A11	—	—	—	AP	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1

Table 4.14 BKADM = 01, BKADP = 001100, BKADB = 010000

Type		MBA2	MBA1	MBA0	MA13	MA12	MA11	MA10	MA9	MA8	MA7	MA6	MA5	MA4	MA3	MA2	MA1	MA0
64 M × 16 bits	ROW	A17	A16	A12	—	A26	A25	A24	A23	A22	A21	A20	A19	A18	A15	A14	A13	A11
	COL	A17	A16	A12	—	—	—	AP	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1

Section 5 Direct Memory Access Controller (SHwy-DMAC)

5.1 Overview

The direct memory access controller (DMAC) connected to the SHwy bus can be used instead of the CPU to perform high-speed data transfers among external memory (DDR-SDRAM) and on-chip memory.

5.1.1 Features

- Two channels
- Address space: Supports 32-bit address spaces.
- Transfer data capacity
  - A value from 4 bytes (H'0000_0004) to 536,870,912 bytes (H'2000_0000) can be set in 4-byte units.
- Transfer data length
  - Automatically selected from 4, 8, 16, or 32 bytes according to the transfer source and destination addresses and remaining transfer data size.
- Dual address mode
- Priority: Fixed channel priority mode
- Interrupt request: DMA transfer end interrupt (DMATEn), transfer source transfer error interrupt (DMASEn), and transfer destination transfer error interrupt (DMADEn) can be generated in each channel (n = 0 and 1; corresponds to each channel).
- Data transfer
  - Contiguous region transfer, stride transfer, and gather/scatter transfer are possible among resources on the SHwy bus.
- Command chain
  - Multiple data transfers can be executed continuously according to the data transfer instruction set in the specified address.
- SHwy transactions which can be issued
    - LOAD (4-, 8-, 16-, or 32-byte transfer) and STORE (4-, 8-, 16-, or 32-byte transfer)

5.1.2 Block Diagram

Figure 5.1 shows a block diagram of the DMAC.

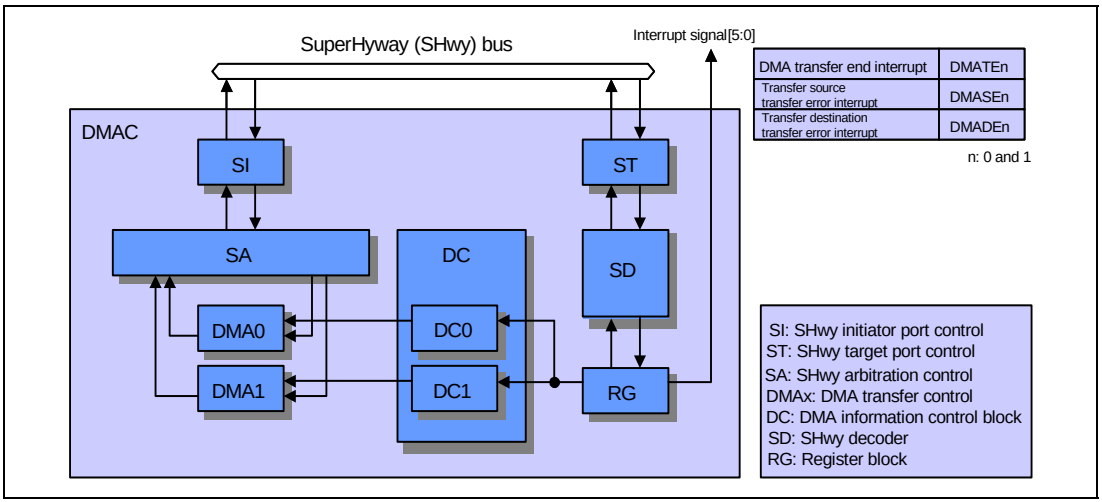


Figure 5.1 Block Diagram of DMAC

5.1.3 External Pins

There are no external pins associated with the DMAC.

5.1.4 Register Configuration

Table 5.1 shows the register configuration. For the relationship between a channel and its registers, DMASAR in ch0 is shown as DMASAR0. The registers should be accessed only by the access size shown in the table.

Do not write to addresses other than those listed in the table below, otherwise normal operation cannot be guaranteed.

Table 5.1 Register Configuration

Register Name	Abbreviation	R/W	P4 Address	Area 7 Address	Access Size
DMA operation register	DMAOR	R/W	H'FE00_0010	H'1E00_0010	32
DMA source address registers 0 and 1	DMASAR0 and DMASAR1	R/W	ch0: H'FE00_0020 ch1: H'FE00_0120	ch0: H'1E00_0020 ch1: H'1E00_0120	32
DMA destination address registers 0 and 1	DMADAR0 and DMADAR1	R/W	ch0: H'FE00_0028 ch1: H'FE00_0128	ch0: H'1E00_0028 ch1: H'1E00_0128	32
DMA byte count registers 0 and 1	DMABCNTR0 and DMABCNTR1	R/W	ch0: H'FE00_0030 ch1: H'FE00_0130	ch0: H'1E00_0030 ch1: H'1E00_0130	32
DMA stride count registers 0 and 1	DMASBCNTR0 and DMASBCNTR1	R/W	ch0: H'FE00_0034 ch1: H'FE00_0134	ch0: H'1E00_0034 ch1: H'1E00_0134	32
DMA stride registers 0 and 1	DMASTRR0 and DMASTRR1	R/W	ch0: H'FE00_0038 ch1: H'FE00_0138	ch0: H'1E00_0038 ch1: H'1E00_0138	32
DMA command chain address registers 0 and 1	DMACCAR0 and DMACCAR1	R/W	ch0: H'FE00_0040 ch1: H'FE00_0140	ch0: H'1E00_0040 ch1: H'1E00_0140	32
DMA channel control registers 0 and 1	DMACHCR0 and DMACHCR1	R/W	ch0: H'FE00_0048 ch1: H'FE00_0148	ch0: H'1E00_0048 ch1: H'1E00_0148	32
DMA channel status registers 0 and 1	DMACHSR0 and DMACHSR1	R/(W)*	ch0: H'FE00_004C ch1: H'FE00_014C	ch0: H'1E00_004C ch1: H'1E00_014C	32

Note: * To clear the SE, DE, and TE bits, only 1 can be written to.

Table 5.2 Register States in Each Operating Mode

Abbreviation	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
DMAOR	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
DMASAR0 and DMASAR1	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
DMADAR0 and DMADAR1	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
DMABCNTR0 and DMABCNTR1	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
DMASBCNTR0 and DMASBCNTR1	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
DMASTRR0 and DMASTRR1	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
DMACCAR0 and DMACCAR1	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
DMACHCR0 and DMACHCR1	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
DMACHSR0 and DMACHSR1	H'00000000	H'00000000	Retained	Retained	Retained	Initialized

Note: Initialized: The value is the one written in "Power-on Reset".

5.2 Register Descriptions

5.2.1 DMA Operation Register (DMAOR)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMAE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Descriptions
31	DMAE	0	R/W	DMA Master Enable Enables or disables DMA transfers for all channels. Set this bit to 1 to use the DMA transfer function. When this bit is cleared to 0, transfers on all channels are stopped. If transfer has been stopped, even if this bit is set to 1 again, the stopped DMA transfer will not be resumed. When transfer has been stopped, the transfers up to two addresses before the values indicated in DMADAR0 and DMADAR1 will be completed, but the data values of the previous and current addresses cannot be guaranteed.
30 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

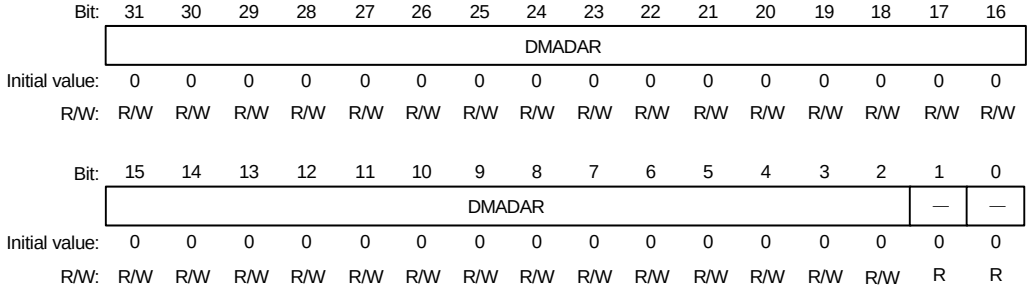
5.2.2 DMA Source Address Registers 0 and 1 (DMASAR0 and DMASAR1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMASAR															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMASAR														—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Descriptions
31 to 2	DMASAR	H'0000 0000	R/W	These bits specify the transfer source address in DMA transfer. During DMA transfer, these bits indicate the transfer source address that is being currently issued to the SHwy bus. These bits correspond to a 32-bit physical address space. Note that only a multiple of four can be specified. A quarter of the target value (a value excluding the lower two bits) should be specified in this field.
1, 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

5.2.3 DMA Destination Address Registers 0 and 1 (DMADAR0 and DMADAR1)



Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMADAR															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMADAR														—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Descriptions
31 to 2	DMADAR	H'0000 0000	R/W	These bits specify the transfer destination address in DMA transfer. During DMA transfer, these bits indicate the transfer destination address that is being currently issued to the SHwy bus. These bits correspond to a 32-bit physical address space. Note that only a multiple of four can be specified. A quarter of the target value (a value excluding the lower two bits) should be specified in this field.
1, 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

5.2.4 DMA Byte Count Registers 0 and 1 (DMABCNTR0 and DMABCNTR1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	BCNT												
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BCNT														—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Descriptions
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28 to 2	BCNT	H'000 0000	R/W	These bits specify the transfer byte count. If these bits are specified as 0, 2^29 (= 536,870,912) bytes are transferred. Note that only a multiple of four can be specified. A quarter of the target value (a value excluding the lower two bits) should be specified in this field. The number of bytes transferred from the transfer source is counted.
1, 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

5.2.5 DMA Stride Count Registers 0 and 1 (DMASBCNTR0 and DMASBCNTR1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SBCINI														—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SBCNT														—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Descriptions
31 to 18	SBCINI	H'0000	R/W	Initial Stride Counter These bits specify the initial value of the stride counter. The initial value of the number of data bytes to be transferred as a block in stride transfer or gather/scatter transfer is set. Note that only a multiple of four can be specified. A quarter of the target value (a value excluding the lower 2 bits) should be specified in this field.
17, 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 2	SBCNT	H'0000	R/W	Stride Counter These bits specify the stride counter which indicates the number of data bytes to be transferred as a block in stride transfer or gather/scatter transfer. During data transfer, these bits indicate the number of remaining bytes to be transferred. When BCNT ≠ 0 and SBCNT = 0, data transfer is continued by loading the SBINI value. Note that only a multiple of four can be specified. A quarter of the target value (a value excluding the lower 2 bits) should be specified in this field.
1, 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

5.2.6 DMA Stride Registers 0 and 1 (DMASTRR0 and DMASTRR1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SS														—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DS														—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Descriptions
31 to 18	SS	H'0000	R/W	Stride Width of Transfer Source Address These bits specify the stride width of the transfer source address. Note that only a multiple of four can be specified. A quarter of the target value (a value excluding the lower 2 bits) should be specified in this field.
17, 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 2	DS	H'0000	R/W	Stride Width of Transfer Destination Address These bits specify the stride width of the transfer destination address. Note that only a multiple of four can be specified. A quarter of the target value (a value excluding the lower 2 bits) should be specified in this field.
1, 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

5.2.7 DMA Command Chain Address Registers 0 and 1 (DMACCAR0 and DMACCAR1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CCA															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CCA											—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Descriptions
31 to 5	CCA	H'000 0000	R/W	Command Chain Address These bits specify the address of the command stream of the first command chain when command chains are executed. During command chain execution, these bits specify the address of the command stream of the command chain to be executed next. Note that only a 32-byte boundary can be specified as a command chain address. This field specifies a value excluding the lower 5 bits.
4 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

5.2.8 DMA Channel Control Registers 0 and 1 (DMACHCR0 and DMACHCR1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CHE	—	CCRE	—	—	—	SA SRE	DA SRE	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R/W	R	R	R	R/W	R/W	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Descriptions
31	CHE	0	R/W	DMA Channel Enable Enables or disables a channel. If this bit is set to 1, the data transfer of the corresponding channel is started. Note however that the data transfer is not performed while a bit indicating a transfer end (TE) or a transfer error (DE or SE) is set to 1. When this bit is cleared to 0, the data transfer will be stopped. If transfer has been stopped, even if this bit is set to 1 again, the stopped DMA transfer will not be resumed. When transfer has been stopped, the transfers up to two addresses before the values indicated in DMADAR0 and DMADAR1 will be completed, but the data values of the previous and current addresses cannot be guaranteed. This bit is not cleared to 0 when data transfer ends or is stopped, except for when an error response occurs in response to LOAD transfer performed for acquiring a command during command chain execution. 0: Disables data transfer 1: Enables data transfer
30	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Descriptions
29	CCRE	0	R/W	Command Chain Enable Enables or disables a command chain. If a data transfer is requested while this bit is set to 1, the data transfer is performed by reading a command from an address specified by DMACCAR. 0: Disables a command chain 1: Enables a command chain
28 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25	SASRE	0	R/W	Transfer Source Address Stride Enable Enables or disables the stride registers for the transfer source address. 0: Disables SS bits in DMASTRR0 and DMASTRR1 1: Enables SS bits in DMASTRR0 and DMASTRR1
24	DASRE	0	R/W	Transfer Destination Address Stride Enable Enables or disables the stride registers for the transfer destination address. 0: Disables DS bits in DMASTRR0 and DMASTRR1 1: Enables DS bits in DMASTRR0 and DMASTRR1
23 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

5.2.9 DMA Channel Status Registers 0 and 1 (DMACHSR0 and DMACHSR1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	SEE	—	DEE	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R	R/W	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	SE	—	DE	—	—	—	—	—	IE	—	—	TE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/WC1	R	R/WC1	R	R	R	R	R	R/W	R	R	R/WC1

Bit	Bit Name	Initial Value	R/W	Descriptions
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27	SEE	0	R/W	Transfer Source Transfer Error Interrupt Enable Enables or disables an interrupt caused by a transfer error during data transfer from the transfer source. If this bit is set to 1, an interrupt is requested when the SE bit is set to 1. 0: Disables an interrupt request 1: Enables an interrupt request
26	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
25	DEE	0	R/W	Transfer Destination Transfer Error Interrupt Enable Enables or disables an interrupt caused by a transfer error during data transfer to the transfer destination. If this bit is set to 1, an interrupt is requested when the DE bit is set to 1. 0: Disables an interrupt request 1: Enables an interrupt request

Bit	Bit Name	Initial Value	R/W	Descriptions
24 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11	SE	0	R/WC1	Transfer Source Transfer Error Flag Indicates that a transfer error has occurred during data transfer from the transfer source. When an error occurs, the DMA transfer being executed is stopped. When this bit is set to 1, DMA transfers are disabled even if the CHE bit in DMCHCR0 and DMCHCR1 is set to 1. [Clearing condition] Writing 1 to this bit clears the flag. Writing 0 to this bit is ignored. 0 should be written to this bit except for when clearing this bit.
10	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
9	DE	0	R/WC1	Transfer Destination Transfer Error Flag Indicates that a transfer error has occurred during data transfer to the transfer destination. When an error occurs, the DMA transfer being executed is stopped. When this bit is set to 1, DMA transfers are disabled even if the CHE bit in DMCHCR0 and DMCHCR1 is set to 1. [Clearing condition] Writing 1 to this bit clears the flag. Writing 0 to this bit is ignored. 0 should be written to this bit except for when clearing this bit.

Bit	Bit Name	Initial Value	R/W	Descriptions
8 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	IE	0	R/W	DMA Transfer End Interrupt Enable Enables or disables an interrupt caused by the end of a DMA transfer. If this bit is set to 1, an interrupt is requested when the TE bit is set to 1. 0: Disables an interrupt request 1: Enables an interrupt request
2, 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	TE	0	R/WC1	DMA Transfer End Flag Indicates the transfer end flag. If a data transfer is completed after DMBCNTR0 and DMBCNTR1 are cleared to 0, this bit is set to 1. If a data transfer is terminated by a transfer error, or if a data transfer is forcibly terminated by clearing the CHE bit in DMCHCR0 and DMCHCR1 to 0, this bit will not be set to 1. In addition, if the next transfer is specified by using a command chain (CCRE = 1), this bit will not be set to 1 even if the current data transfer is completed normally. When this bit is set to 1, DMA transfers are disabled even if the CHE bit in DMCHCR0 and DMCHCR1 is set to 1. 0: Indicates that a data transfer is in progress or has been stopped 1: Indicates that a data transfer is completed (by DMBCNTR0 and DMBCNTR1 = 0) [Clearing condition] Writing 1 to this bit clears the flag. Writing 0 to this bit is ignored. 0 should be written to this bit except for when clearing this bit.

5.3 Operation

Table 5.3 shows the relationship between the operation type and control register setting. The operation is controlled by four bits: bits 31, 29, 25, and 24 in DMA channel control registers 0 and 1 (DMACHCR0 and DMACHCR1).

Table 5.3 Relationship between Operation Type and Control Register Setting

	Bit 31: CHE	Bit 29: CCRE	Bit 25: SASRE	Bit 24: DASRE
Operation Type	**DMA Channel Enable**	**Command Chain Enable**	**Transfer Source Address Stride Enable**	**Transfer Destination Address Stride Enable**
No operation	0	—	—	—
Contiguous region transfer	1	0	0	0
Stride transfer	1	0	1	1
Gather transfer	1	0	1	0/1
Scatter transfer	1	0	0/1	1
Command chain	1	1	—	—
Setting prohibited	Combinations other than above			

[Legend]
—: Don't care

In gather transfer, the transfer destination start address is the same in any block when the DASRE bit in DMACHCR is 0, and when the DASRE bit in DMACHCR is 1 and the DS bits in DMASTRR are 0.

Similar in scatter transfer, the transfer source start address is the same in any block when the SASRE bit in DMACHCR is 0, and when the SASRE bit in DMACHCR is 1 and the SS bits in DMASTRR are 0.

5.3.1 Channel Priority

When the DMAC receives transfer requests on two or more channels, every time data transfer for a read cycle or write cycle of one transfer unit (byte, word, longword, 8-byte, 16-byte, or 32-byte units) has finished, data transfer is started on the channel which has the highest priority among the channels in which transfer is enabled.

When the DMAC receives transfer requests on two or more channels simultaneously, data transfer is started according to the determined priority.

Simultaneous transfer requests are handled in the order of CH1 > CH0.

5.3.2 Contiguous Region Transfer

After transfer conditions have been set in registers and contiguous region transfer has been specified in a DMA channel control register, data transfer is performed with the following procedure. The registers for ch0 are set in the following procedure. Similar settings should be made when using ch1.

1. Check if transfer is enabled.

   Transfer is enabled under the following conditions: the DMAE bit in DMAOR is 1, the CHE bit in DMACHCR0 is 1, the SE bit in DMACHSR0 is 0, the DE bit in DMACHSR0 is 0, and the TE bit in DMACHSR0 is 0. According to table 5.3, set DMA channel control register 0 to specify contiguous region transfer.
2. If transfer is enabled, data transfer starts. The DMABCNTR0 value is decremented for each transfer.
3. When the specified number of bytes has been transferred (when DMABCNTR0 = 0), the transfer ends normally and the TE bit in DMACHSR0 is set to 1. If the IE bit in DMACHSR0 is set to 1 at this time, a DMA transfer end interrupt is sent to the CPU.

   If a transfer destination transfer error or a transfer source transfer error occurs, the data transfer is stopped. At this time, the SE bit or DE bit in DMACHSR0 is set to 1. Data transfers are also stopped when the CHE bit in DMACHCR0 is cleared to 0.

The transfer size is automatically determined by the addresses set in DMASAR0, DMASAR1, DMADAR0, and DMADAR1, and the remaining data size that is still to be transferred.

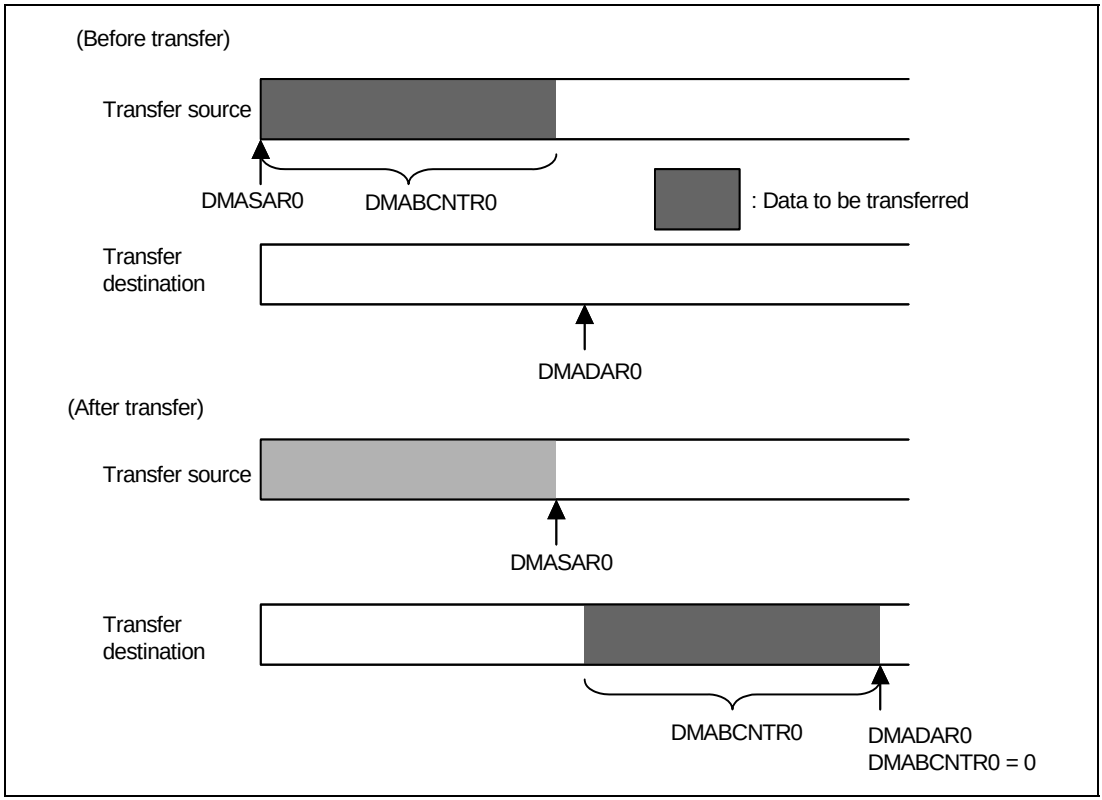


Figure 5.2 Contiguous Region Transfer

5.3.3 Stride, Gather, or Scatter Transfer

After transfer conditions have been set in registers and stride, gather, or scatter transfer has been specified in a DMA channel control register, data transfer is performed with the following procedure. The registers for ch0 are set in the following procedure. Similar settings should be made when using ch1.

1. Check if transfer is enabled.

   Transfer is enabled under the following conditions: the DMAE bit in DMAOR is 1, the CHE bit in DMACHCR0 is 1, the SE bit in DMACHSR0 is 0, the DE bit in DMACHSR0 is 0, and the TE bit in DMACHSR0 is 0. According to table 5.3, set DMA channel control register 0 to specify stride, gather, or scatter transfer.
2. If transfer is enabled, data transfer starts. The DMABCNTR0 value and the value of the SBCNT bits in DMASBCNTR0 are decremented for each transfer.
3. When DMABCNTR0 is not 0 and the SBCNT bits in DMASBCNTR0 are 0, the SS bit value in DMASTRR is added to DMASAR0 and the DS bit value in DMASTRR0 is added to DMADAR0 to determine the next transfer source and transfer destination addresses.

   Then, the value set in the SBCINI bits in DMASBCNTR0 is set to the SBCNT bits in DMASBCNTR0 and the processing returns to step 2.
4. When the specified number of bytes has been transferred (when DMABCNTR0 = 0), the transfer ends normally and the TE bit in DMACHSR0 is set to 1. If the IE bit in DMACHSR0 is set to 1 at this time, a DMA transfer end interrupt is sent to the CPU.

   If a transfer source transfer error or a transfer destination transfer error occurs, the data transfer is stopped. At this time, the SE bit or DE bit in DMACHSR0 is set to 1. Data transfers are also stopped when the CHE bit in DMACHCR0 is cleared to 0.

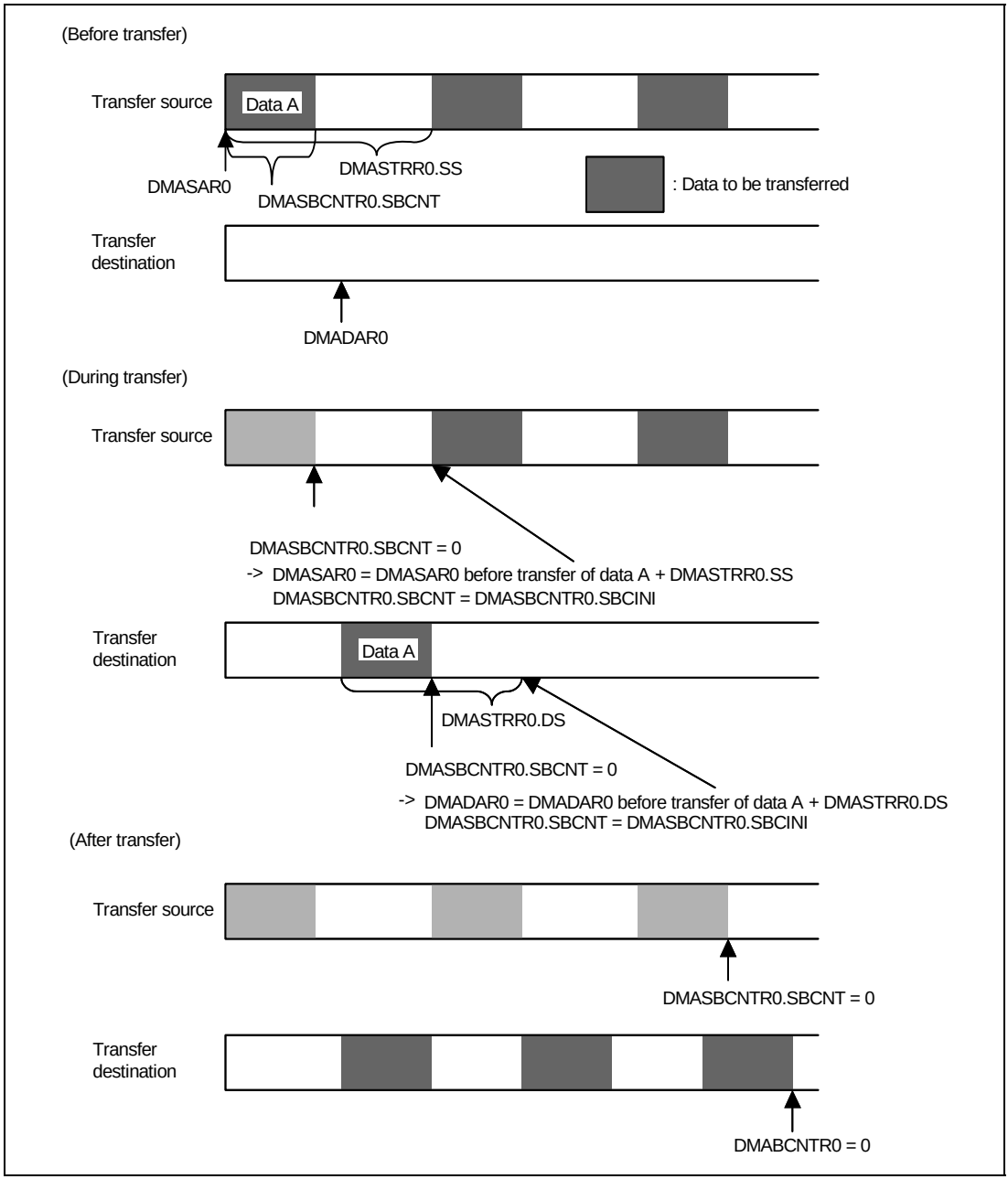


Figure 5.3 Stride Transfer

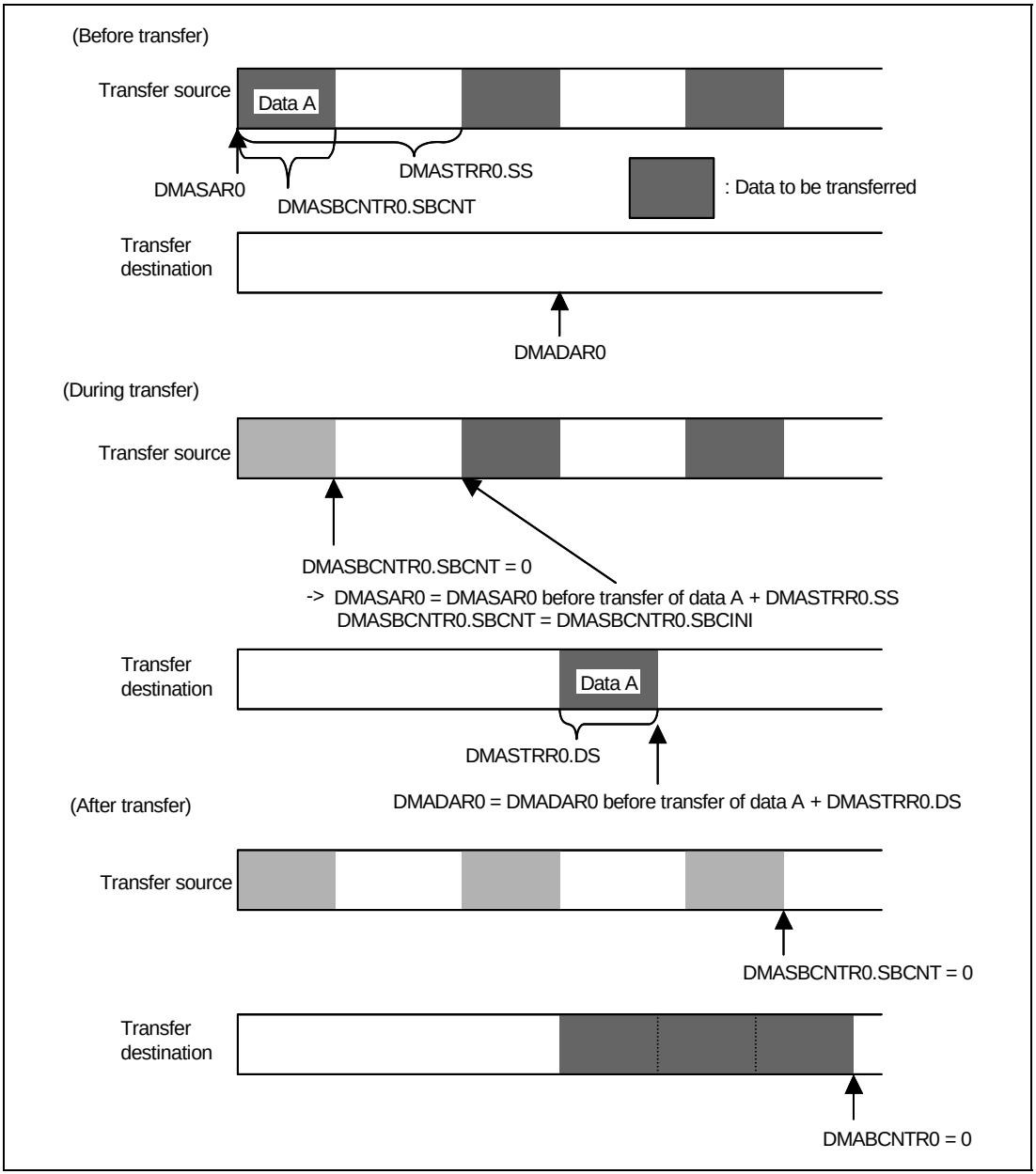


Figure 5.4 Gather Transfer

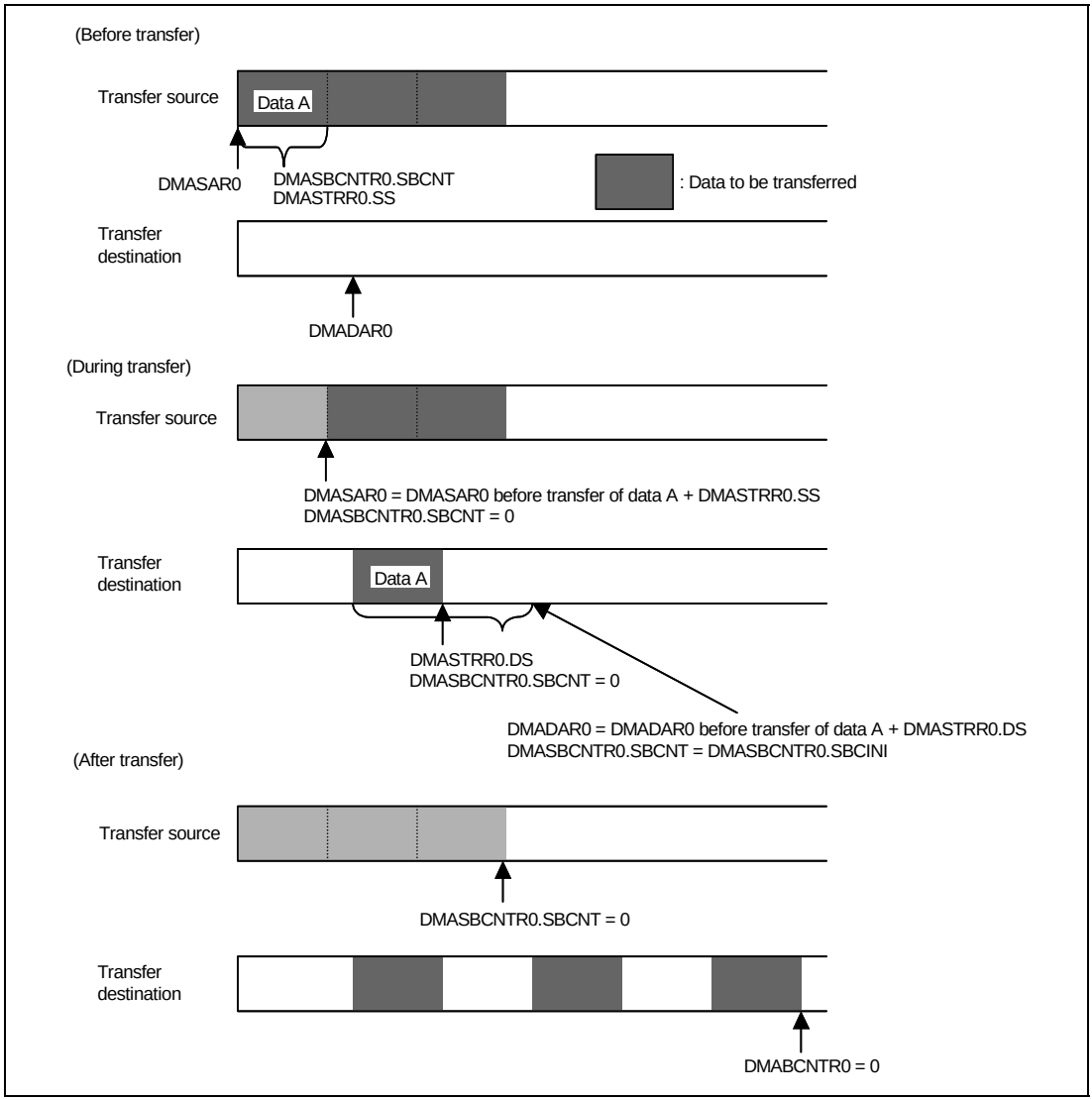


Figure 5.5 Scatter Transfer

5.3.4 Command Chain

After transfer conditions have been set in registers and a command chain has been specified in a DMA channel control register, data transfer is performed with the following procedure. The registers for ch0 are set in the following procedure. Similar settings should be made when using ch1.

1. Check if transfer is enabled.

   Transfer is enabled under the following conditions: the DMAE bit in DMAOR is 1, the CHE bit in DMACHCR0 is 1, the SE bit in DMACHSR0 is 0, the DE bit in DMACHSR0 is 0, and the TE bit in DMACHSR0 is 0. According to table 5.3, set DMA channel control register 0 to specify a command chain.
2. If transfer is enabled, read a data transfer command from the address set in DMACCAR and set the registers.
3. Execute the specified data transfer command.
4. Even if execution of the specified data transfer command ends normally, if the CCRE bit in DMACHCR0 is 1, there is a command that needs to be executed next. In such a case, the TE bit in DMACHSR0 is not set to 1 and no interrupt is generated even if the IE bit in DMACHSR0 is 1.
5. When the CHE bit in DMACHCR0 is 1, the DE bit in DMACHSR0 is 0, the SE bit in DMACHSR0 is 0, the TE bit in DMACHSR0 is 0, and the CCRE bit in DMACHCR0 is 1, the processing returns to step 2.
6. If the CCRE bit in DMACHCR0 is 0 when execution of the data transfer command has finished, data transfer by command chains is completed. At this time, the TE bit in DMACHSR0 is set to 1, and a DMA transfer end interrupt is generated if the IE bit in DMACHSR0 is 1.

   If a transfer source transfer error or a transfer destination transfer error occurs, the data transfer is stopped. At this time, the SE bit or DE bit in DMACHSR0 is set to 1. Data transfers are also stopped when the CHE bit in DMACHCR0 is cleared to 0.

   If an error occurs while reading the data transfer command in step 2, the SE bit in DMACHSR0 is set to 1 and the CHE bit in DMACHCR0 is cleared to 0.

Figure 5.6 shows the command stream format.

The CHE bit that is set at H'00 must always be set to 1. The bits that are set at H'04 are reserved. Set these bits to H'0000_0008. In the last command stream of the command chain, 27'H0 must always be set to the CCA bits that are set at H'10.

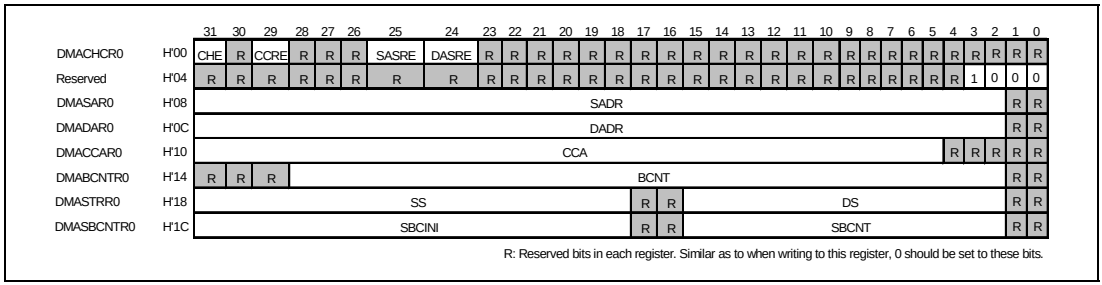


Figure 5.6 Command Stream Format in Command Chain

In a command chain, indicating the start address of the next command in the CCA bits enables DMA transfers to be performed continuously, as shown in figure 5.7.

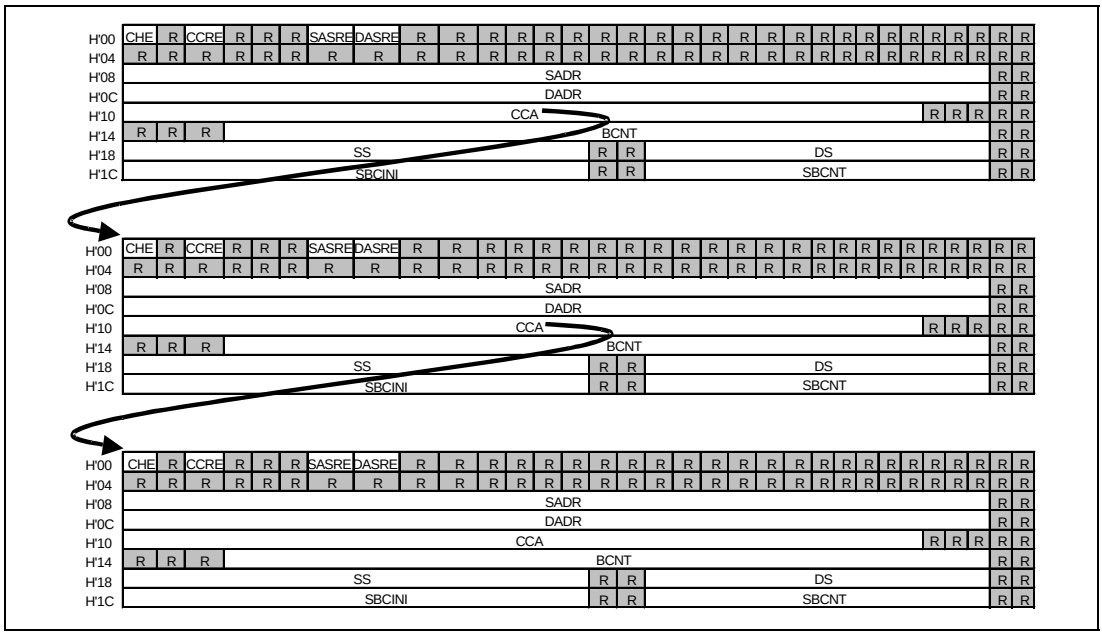


Figure 5.7 Command Chain

5.4 Usage Notes

Note the following when using this DMAC.

5.4.1 DMA Transfer Halfway Finished

In the following cases, subsequent DMA transfers cannot be resumed when a DMA transfer is finished in the middle of processing.

- When the DMAE bit in DMAOR is cleared to 0
- When the CHE bit in DMACHCRn (n: 0 and 1) is cleared to 0
- When a transfer source transfer error or a transfer destination transfer error is generated

To re-execute DMA transfer, set the registers again to start DMA transfer.

The procedure for setting the registers again is shown below. The values to be set again in the registers are determined from the DMADAR0 and DMADAR1 values at the time DMA transfer was stopped.

1. Confirm the DMADAR0 and DMADAR1 values, determine the values to be set again in DMASAR0 and DMASAR1 and DMADAR0 and DMADAR1, and set these registers.

   Subtract 7'h20 (corresponding to 32-byte transfer data) from DMADAR0 and DMADAR1, and set the obtained values to DMADAR0 and DMADAR1. Set addresses that correspond to DMADAR0 and DMADAR1 to DMASAR0 and DMASAR1.
2. Set values to DMABCNTR0 and DMABCNTR1.
3. Check if transfer is enabled.

   Transfer is enabled and data transfer starts under the following conditions: the DME bit in DMAOR is 1, the CHE bit in DMACHCR0 is 1, the SE bit in DMACHSR0 is 0, the DE bit in DMACHSR0 is 0, and the TE bit in DMACHSR0 is 0.

5.4.2 Interrupt Generated during DMA Transfer

If the DMAE bit in DMAOR or the CHE bit in DMACHCRn (n: 0 and 1) is cleared to 0 during DMA transfer when the IE bit in DMACHSRn, the SEE bit in DMACHSRn, and the DEE bit in DMACHSRn are 1, an interrupt may be generated before the DMA transfer is stopped. Handle the generated interrupt properly.

5.4.3 Modules with Limitation on Access Size

When you execute DMA transfer to or from any of the modules listed in table 5.4, please do so in accord with condition 1 or 2, depending on the module's classification in table 5.4. In the descriptions of the conditions, n is a positive integer, and m = 0 or 1. Under the other conditions, non-permitted transactions may be issued to the following modules.

Do not use the DMAC to handle transfer to or from the DBG module and TMU module.

Table 5.4 Conditions for DMA Transfer

Condition	1	2
Module	DMA	LRAM (SH-4A)
	HPB	
	R-GPVG	
	DU	
	VIN	
	INTC	
	USB	
	DBSC	

Condition 1: 4-byte data transfer

- Contiguous region transfer
  Setting prohibited.
- Stride, gather, or scatter transfer
  Set the SBCINI and SBCNT bits in DMASBCNTRm to 14'h1.

Condition 2: 32-byte data transfer

- Contiguous region transfer
  If a module within the scope of this condition is the source for the transfer, set DMASARm to a 32-byte boundary address.
  If a module within the scope of this condition is the destination for the transfer, set DMADARm to a 32-byte boundary address.
  Set DMABCNTRm to 32n bytes.
- Stride, gather, or scatter transfer
  Set a 32-byte transfer.
  If a module within the scope of this condition is the source for the transfer, set DMASARm to a 32-byte boundary address and the SS bit in DMASTRRm to 32n/4 bytes.

If a module within the scope of this condition is the destination for the transfer, set DMADARm to a 32-byte boundary address and the DS bit in DMASTRRm to 32n/4 bytes. Set the SBCINI and SBCNT bits in DMASBCNTRm to 32n/4 bytes.

Set DMABCNTRm to 32n bytes.

5.4.4 Module Stop

During the DMAC operation, the module should not be stopped by the CPG register setting. If stopped, accuracy of data being transferred cannot be guaranteed.

Section 6 HPB

6.1 Overview

The HPB is a functional module that efficiently transfers PIO access from the CPU (the SuperHyway bus) to the HPB bus, which is a peripheral module interface, by performing an interface conversion from the SuperHyway bus protocol to the HPB bus protocol. With built-in 28 HPB-DMAC channels, the HPB performs DMA transfer operations between functional modules on the HPB bus and DDR2_SDRAM/DDR3_SDRAM. To facilitate these transfer operations, the HPB has a double-bus structure in which two separate buses are provided. One is the HPB P1BUS, which is connected to the functional modules on the HPB bus that exclusively perform PIO accesses; and the other is the HPB P2BUS to HPB P6BUS and HPB P8BUS, which are connected to the functional modules that perform DMA transfers by using the HPB-DMACs. As a result, if a PIO access to the module connected to the HPB P1BUS occurs simultaneously with a DMA transfer to the module connected to HPB P2BUS to HPB P6BUS and HPB P8BUS, no contention occurs on the HPB bus.

6.2 Features

- Receiving CPU (SuperHyway bus) Accesses
  - PIO accesses (LOAD and STORE transactions) from the CPU (SuperHyway bus) can be received.
  - Four planes of buffers are incorporated to receive PIO accesses from the CPU (SuperHyway bus).
  - PIO accesses to HPB internal registers from the CPU (SuperHyway bus) can be received.
- Transferring HPB Bus Accesses
  - The alignment transformation function is provided for PIO accesses from the CPU (SuperHyway bus) using endian setting information and access size information.
  - Access contentions on the HPB are arbitrated between PIO accesses from the CPU (SuperHyway bus) and DMA transfers from the HPB-DMAC.
- Performing DMA Transfers by HPB-DMAC
  - DMA transfers are performed between a functional module on the HPB bus and the corresponding SuperHyway (DDR2_SDRAM/DDR3_SDRAM).

    (For DMA transfer control, see section 6A, Direct Memory Access Controller for Local Bus and Peripheral Modules (LBSC-DMAC/HPB-DMAC).)

6.3 Block Diagram

The HPB connects with a number of functional modules under the control of the HPB. To ensure efficient data transfers to the individual functional modules, an internal structure is adopted that enhances parallel bus operations.

Figure 6.1 shows an access route diagram with respect to the functional modules running under the control of the HPB; the list below highlights the main features of the enhanced parallel operations:

- Modules that exclusively perform PIO accesses and modules that perform DMA transfers through the use of the HPB-DMAC are provided on mutually independent HPB buses
- SuperHyway target port control (SHwy-IF (Target Port))
  - Four planes of buffers are incorporated for accesses from the SuperHyway
  - LOAD and STORE transactions from the SuperHyway are supported.
- SuperHyway initiator port control (SHwy-IF (Initiator Port))
  - Conversion from HPB-DMAC protocol (HPB bus protocol) to SuperHyway protocol is performed.
- SuperHyway initiator port access arbitration
  - Bus arbitration function is supported between accesses from the HPB-DMACs to memory (SuperHyway)
- Router
  - Accesses from the HPB-DMACs are assigned to appropriate functional modules.
- SuperHyway-HPB bus conversion
  - Alignment transformation is performed for write data using endian setting information and access size information provided from the SuperHyway.
  - Bus arbitration is performed between the HPB bus accesses from the SuperHyway and those from the HPB-DMAC.
- DMA transfer control
  - For DMA transfer control, see section 6A, Direct Memory Access Controller for Local Bus and Peripheral Modules (LBSC-DMAC/HPB-DMAC).

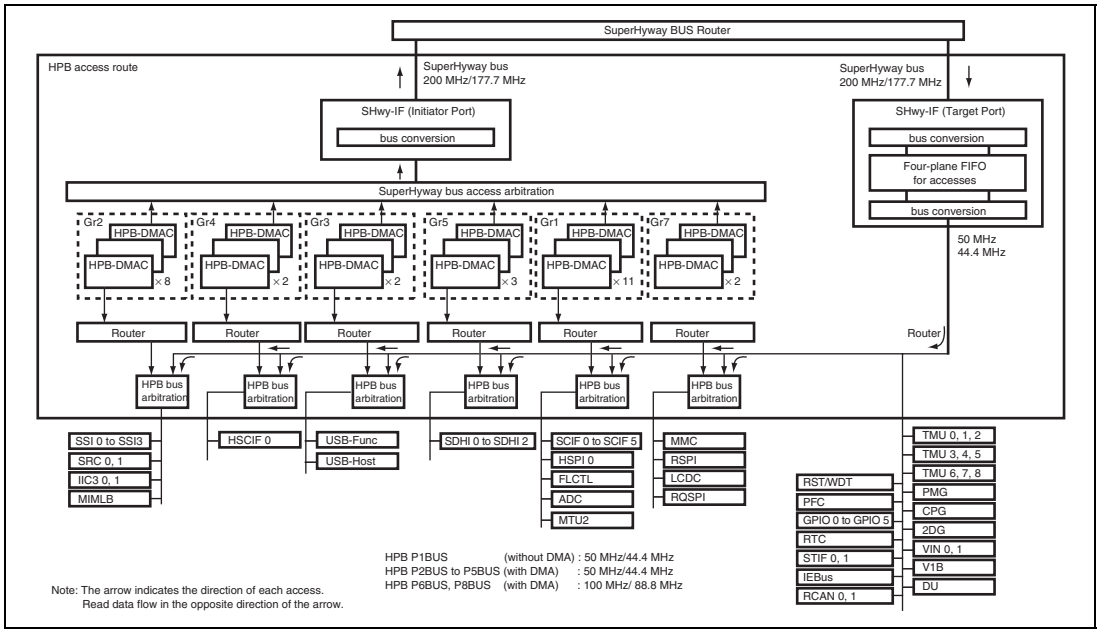


Figure 6.1 Access Route Diagram with Respect to Functional Modules Running under Control of HPB

6.4 Supported Areas

The memory map of the HPB area is shown in figure 6.2. The HPB area is allocated from H'FFC00000 to H'FFFFFFFF.

H'FFC00000	HPB internal registers
H'FFC40000	GPIO 0
H'FFC41000	GPIO 1
H'FFC42000	GPIO 2
H'FFC43000	GPIO 3
H'FFC44000	GPIO 4
H'FFC45000	GPIO 5
H'FFC46000	Nothing allocated
H'FFC50000	VIN0
H'FFC51000	VIN1
H'FFC52000	V1B
H'FFC53000	Nothing allocated
H'FFC60000	LCDC
H'FFC70000	IIC3(ch0)
H'FFC71000	IIC3(ch1)
H'FFC72000	Nothing allocated
H'FFC7F000	PMG
H'FFC80000	CPG
H'FFCC0000	RST/WDT
H'FFD00000	Nothing allocated
H'FFD80000	TMU 0,1,2
H'FFD81000	TMU 3,4,5
H'FFD82000	TMU 6,7,8
H'FFD83000	Nothing allocated
H'FFD85000	MIMLB
H'FFD86000	Nothing allocated
H'FFE00000	SSI 0
H'FFE10000	SSI 1
H'FFE20000	SSI 2
H'FFE30000	SSI 3
H'FFE40000	SCIF 0
H'FFE41000	SCIF 1
H'FFE42000	SCIF 2
H'FFE43000	SCIF 3
H'FFE44000	SCIF 4
H'FFE45000	SCIF 5
H'FFE46000	Nothing allocated
H'FFE48000	HSCIF0
H'FFE49000	Nothing allocated

H'FFE4C000	SDHI 0
H'FFE4D000	SDHI 1
H'FFE4E000	SDHI 2
H'FFE4F000	MMC
H'FFE50000	Nothing allocated
H'FFE60000	USB Function
H'FFE70000	USB Host
H'FFE80000	2DG
H'FFEA0000	Nothing allocated
H'FFEE0000	STIF0
H'FFEE8000	STIF1
H'FFEF0000	Nothing allocated
H'FFF20000	SRC0
H'FFF30000	SRC1
H'FFF40000	Nothing allocated
H'FFF80000	DU
H'FFFC0000	PFC
H'FFFC1000	Nothing allocated
H'FFFC2000	RSPI
H'FFFC3000	RQSPI
H'FFFC4000	Nothing allocated
H'FFFC5000	RTC
H'FFFC6000	MTU2
H'FFFC7000	HSPI
H'FFFC8000	Nothing allocated
H'FFFC9000	IEBus
H'FFFCA000	FLCTL
H'FFFCB000	ADC
H'FFFCC000	Nothing allocated
H'FFFD0000	RCAN0
H'FFFD1000	RCAN1
H'FFFD2000	Nothing allocated
H'FFFE0000	SSS
H'FFFE1000 H'FFFFFFFF	Nothing allocated

Figure 6.2 HPB Area Memory Map

6.5 Register Descriptions

All of the HPB internal registers are mapped in the HPB bus area. Table 6.1 (1) shows configuration of the HPB internal registers. Details of the registers are described in the following sections. For details of registers related to the HPB-DMAC, see section 6A, Direct Memory Access Controller for Local Bus and Peripheral Modules (LBSC-DMAC/HPB-DMAC).

Table 6.1 (1) Configuration of HPB Internal Registers

Address	Register Name	Abbreviation	Access Type	Access Size	Register Location (Module)
H'FFC004C0	RCAN0 control register	RCAN0CTL	R/W	32	HPB
H'FFC004C4	RCAN1 control register	RCAN1CTL	R/W	32	HPB

Note: Do not write to addresses other than listed above, otherwise normal operation cannot be guaranteed. Values read from other addresses are undefined.

Table 6.1 (2) Register State in Each Operating Mode

Abbreviation	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
RCAN0CTL	H'0000 0000	H'0000 0000	Retained	Retained	—	Initialized
RCAN1CTL	H'0000 0000	H'0000 0000	Retained	Retained	—	Initialized

Note: Initialized: The value is the one written in "Power-on Reset".

6.5.1 RCAN0 Control Register (RCAN0CTL)

Function: RCAN0CTL is used to set the clock mode of RCAN0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ASYNC
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	ASYNC	0	R/W	Selects the clock mode of RCAN0 (synchronous or asynchronous). 0: Synchronous mode (synchronized to the internal clock) 1: Asynchronous mode (synchronized to the external clock)

6.5.2 RCAN1 Control Register (RCAN1CTL)

Function: RCAN1CTL is used to set the clock mode of RCAN1.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ASYNC
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	ASYNC	0	R/W	Selects the clock mode of RCAN1. (synchronous or asynchronous) 0: Synchronous mode (synchronized to the internal clock) 1: Asynchronous mode (synchronized to the external clock)

6.6 Operation

6.6.1 Organization and Basic Operation

(1) SuperHyway Request Packet Reception Operation

The HPB being connected to the SuperHyway bus, receives PIO access from the CPU via the SuperHyway bus. The HPB's SuperHyway target port determines the functional module to be accessed based on an address contained in the SuperHyway request packet, and issues bus access after converting the interface to a bus protocol in the HPB area. In this request packet reception operation, a given speed is converted to an HPB area bus frequency before and after a built-in four-plane FIFO. The target port outputs response packets in response to request packets. If a response packet is received when the access FIFO is full, the port, instead of returning a response, enters the wait state.

(2) SuperHyway Reception Transaction

The HPB checks the transaction associated with a given request packet, and only accepts allowable access requests. Table 6.2 lists transactions that can be accepted by the HPB.

Table 6.2 Transactions That Can be Accepted by HPB

Transaction Name	HPB	Transaction Name	HPB
LOAD (1 byte)	Supported	STORE (1 byte)	Supported
LOAD (2 bytes)	Supported	STORE (2 bytes)	Supported
LOAD (4 bytes)	Supported	STORE (4 bytes)	Supported
LOAD (8 bytes)	Not supported	STORE (8 bytes)	Not supported
LOAD (16 bytes)	Not supported	STORE (16 bytes)	Not supported
LOAD (32 bytes)	Not supported	STORE (32 bytes)	Not supported

Even among the acceptable transactions listed in this table, actually receivable transaction can vary depending on the specifications for the functional modules in the HPB area. The HPB does not keep track of acceptable transactions by functional module.

(3) HPB P1BUS

Functional modules running on the HPB bus include those which perform PIO reception operations without performing DMA operation, or those which autonomously perform DMA operation toward the SuperHyway bus without performing DMA operations on the HPB bus. All operations by these modules on the HPB bus are limited to PIO access reception operation, and the modules are multi-connected to the HPB P1BUS.

(4) HPB P2BUS to HPB P6BUS, HPB P8BUS

Functional modules running on the HPB bus include those which perform DMA operations by issuing a DMA request and by means of the HPB-DMAC. Operations by these modules on the HPB bus generate PIO access or DMA access by the HPB-DMAC. If PIO access and DMA access occur simultaneously, contention arbitration control is performed by a fixed priority scheme in which DMA access always takes priority. These modules are multi-connected to the HPB P2BUS to HPB P6BUS and HPB P8BUS under the control of the HPB.

(5) DMA Operation

The HPB incorporates 28 HPB-DMAC channels. These HPB-DMACs support DMA operations between SuperHyway bus and the functional modules that issue DMA requests on the HPB bus. For performing a DMA operation, it is possible to select the functional module to be supported on the HPB bus by means of register settings in the HPB-DMAC. While DDR2_SDRAM/DDR3_SDRAM is assumed as a destination for data transfer on the SuperHyway bus side, this destination is determined by the DMA transfer address that is assigned to the HPB-DMAC. For details on how to start the HPB-DMAC, see section 6A, Direct Memory Access Controller for Local Bus and Peripheral Modules (LBSC-DMAC/HPB-DMAC). Table 6.3 shows the available HPB-DMAC-to-SuperHyway access transactions.

Table 6.3 HPB-DMAC-to-SuperHyway Access Transactions

Transaction Name	HPB	Transaction Name	HPB
LOAD (1 byte)	Not supported	STORE (1 byte)	Supported
LOAD (2 bytes)	Not supported	STORE (2 bytes)	Supported
LOAD (4 bytes)	Supported	STORE (4 bytes)	Supported
LOAD (8 bytes)	Supported	STORE (8 bytes)	Supported
LOAD (16 bytes)	Supported	STORE (16 bytes)	Supported
LOAD (32 bytes)	Supported	STORE (32 bytes)	Supported

6.6.2 Endian and Data Alignment

(1) Endian Transformation during PIO Access

While the SuperHyway supports both the big endian and the little endian, the HPB bus is fixed to the big endian. For this reason, the HPB performs data alignment transformations compatible with the endian according to endian signals and access size, while simultaneously converting bus widths. Figure 6.3 shows a data alignment transformation diagram for HPB bus access.

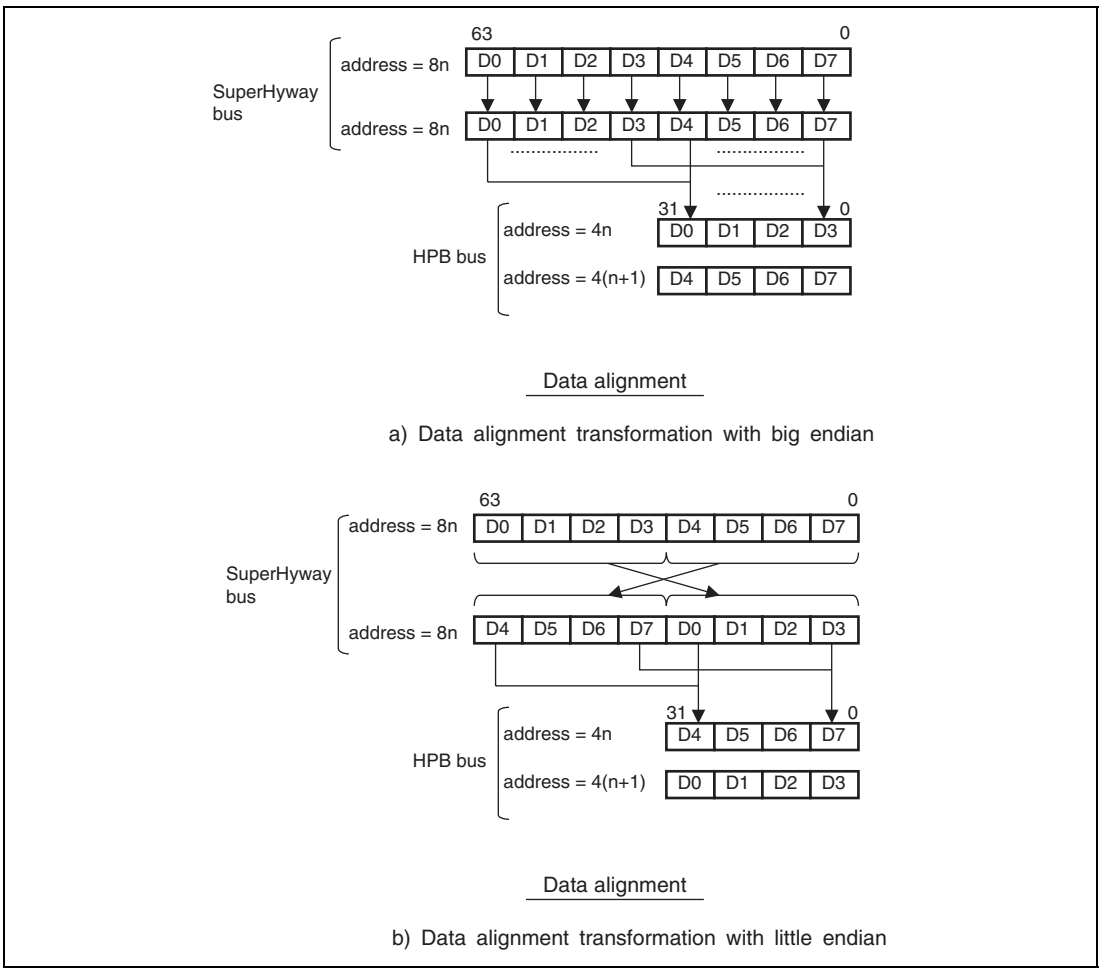


Figure 6.3 Data Alignment Transformation Diagram for HPB Bus Access

The following describes data alignment performed on an actual HPB bus. Figure 6.4 shows a concept diagram on data alignment transformation associated with the assert status of each byte enable signal. For write operations, the HPB copies valid data to all byte lanes on the HPB bus so that every functional module can fetch and read data from any byte lane. In contrast, the HPB does not copy data to byte lanes for read operations. To return data on alignment that is compatible with either endian, the functional modules under the control of each HPB should return data to all byte lanes, similar to the situation where the HPB performs write operations.

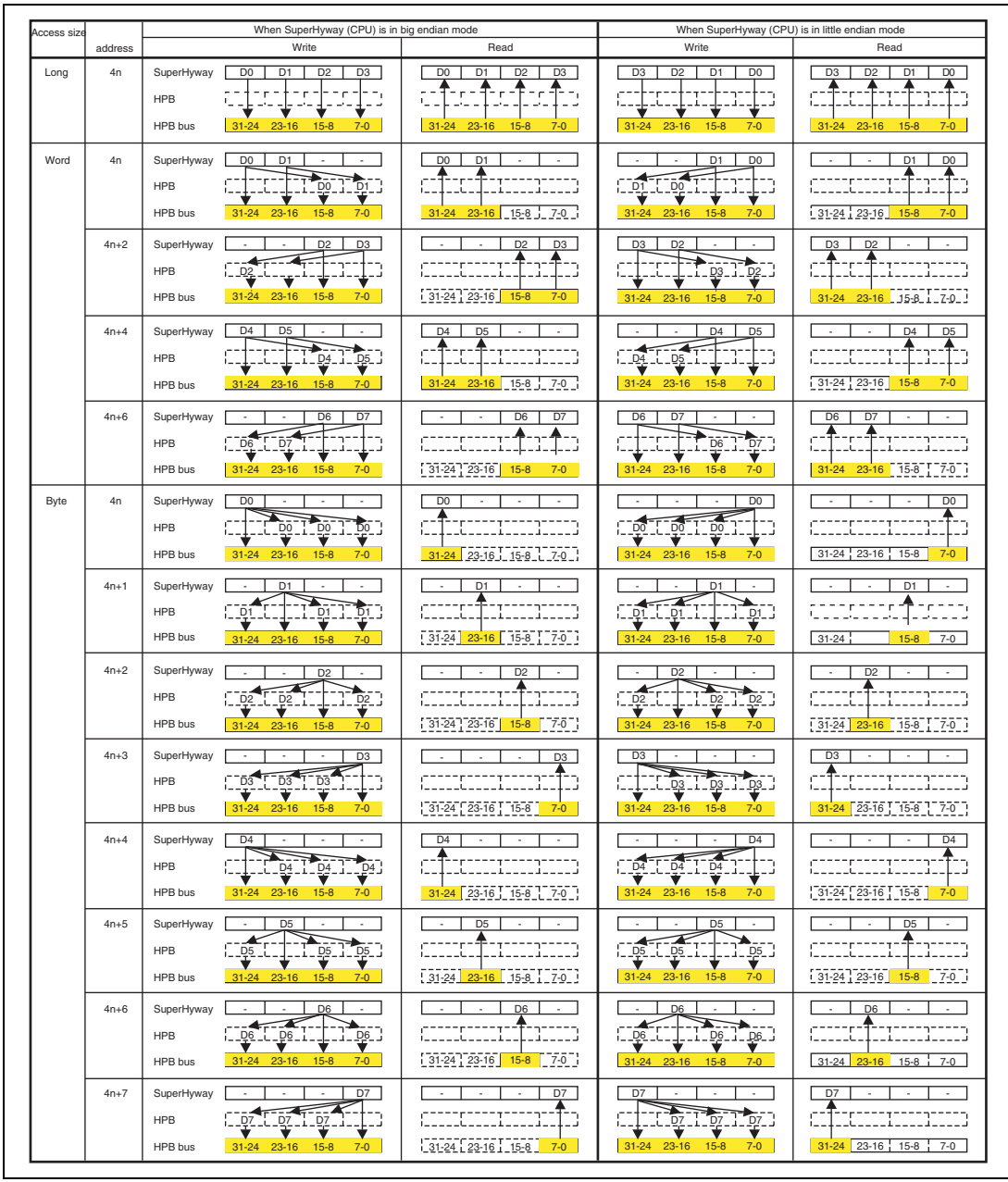


Figure 6.4 Concept Diagram on Data Alignment Transformation Associated with Assert Status of Each Byte Enable Signal

(2) Endian Transformation during DMA Access

During DMA transfer between a functional module connected on the HPB bus and the SuperHyway bus, during DMA transfer to the HPB bus, data alignment transformations are performed by the HPB-DMAC according to the endian and the size of the transfer bus involved. For details, see section 6A, Direct Memory Access Controller for Local Bus and Peripheral Modules (LBSC-DMAC/HPB-DMAC).

6.6.3 HPB Bus Timing Chart

(1) Timing Chart of HPB Bus Access from SuperHyway Bus

Figure 6.5 shows a timing chart of HPB bus access from the SuperHyway bus. The HPB bus, which is a 50-/44.4-MHz bus (the frequencies are 100/88.8 MHz in which case the bus is an HPB P6BUS or HPB P8BUS), has 1/4 the speed of the SuperHyway bus speed (the speed is 1/2 of the SuperHyway bus speed in which case the bus is an HPB P6BUS or HPB P8BUS) During HPB_BUS access the HPB performs clock speed conversion in addition to interface conversion. On the HPB bus, there is a minimum of one clock cycle of interval between two access events. A single bus access event involves a minimum of two clock cycles. By contrast, if a pwait signal is received from the access destination functional module, the bus access is extended; note, however, that the pread signal cannot be extended. When the pwait signal is negated, the bus access is terminated at the next clock cycle.

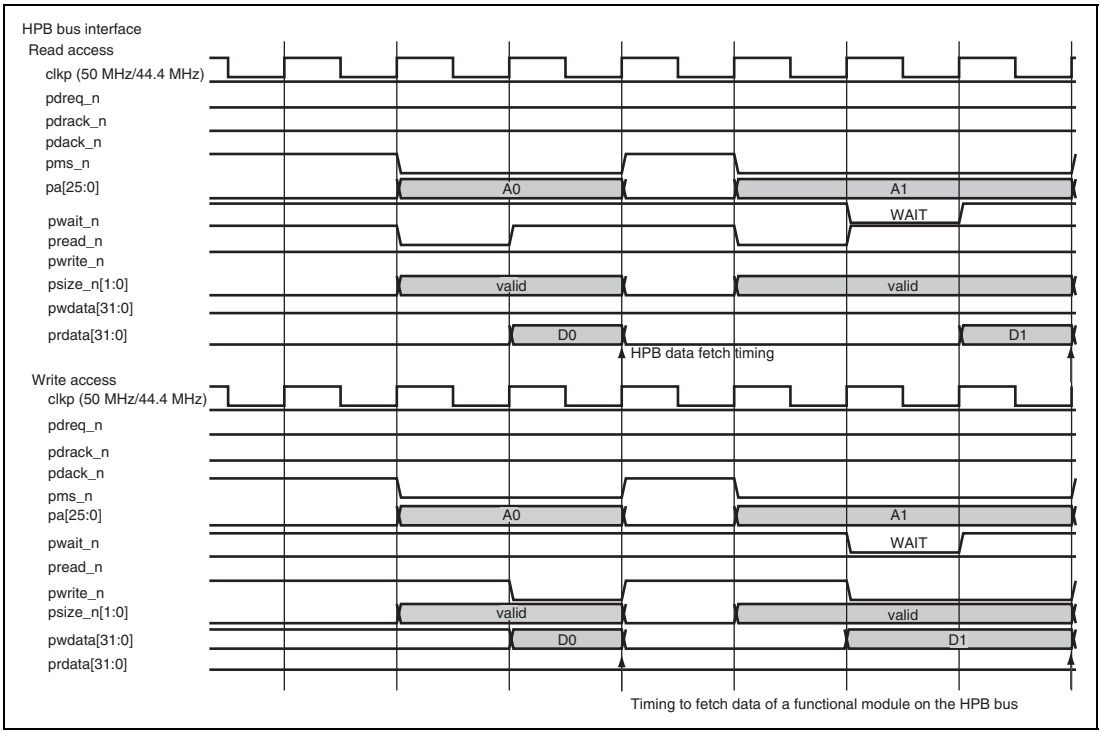


Figure 6.5 Timing Chart of HPB Bus Access from SuperHyway Bus

(2) Timing Chart of HPB Bus Access (DMA Operation) from HPB-DMAC

Figure 6.6 shows a timing chart of HPB bus access from the HPB-DMAC. The DMA access interface on the HPB bus has the configuration in which three DMA signals, pdreq, pdrack, and pdack, are provided in addition to the interface signals that are used in normal PIO operations. Upon receipt of pdreq from a functional module on the HPB bus, the HPB-DMAC returns a pdrack to acknowledge the receipt of a DMA request from the functional module on the HPB bus. This causes the functional module on the HPB bus to negate pdreq. The HPB-DMAC starts bus access by asserting a pdack at the pdrack negate timing at the earliest. Although figure 6.6 only includes DMA operation, even when a PIO operation is inserted, there is a minimum of one clock cycle of interval between two access events. Each DMA bus access involves a minimum of two clock cycles. By contrast, if a pwait signal is received from the access destination functional module, the bus access is extended; note, however, that the pdack and pread signals cannot be extended. When the pwait signal is negated, the bus access is terminated at the next clock cycle.

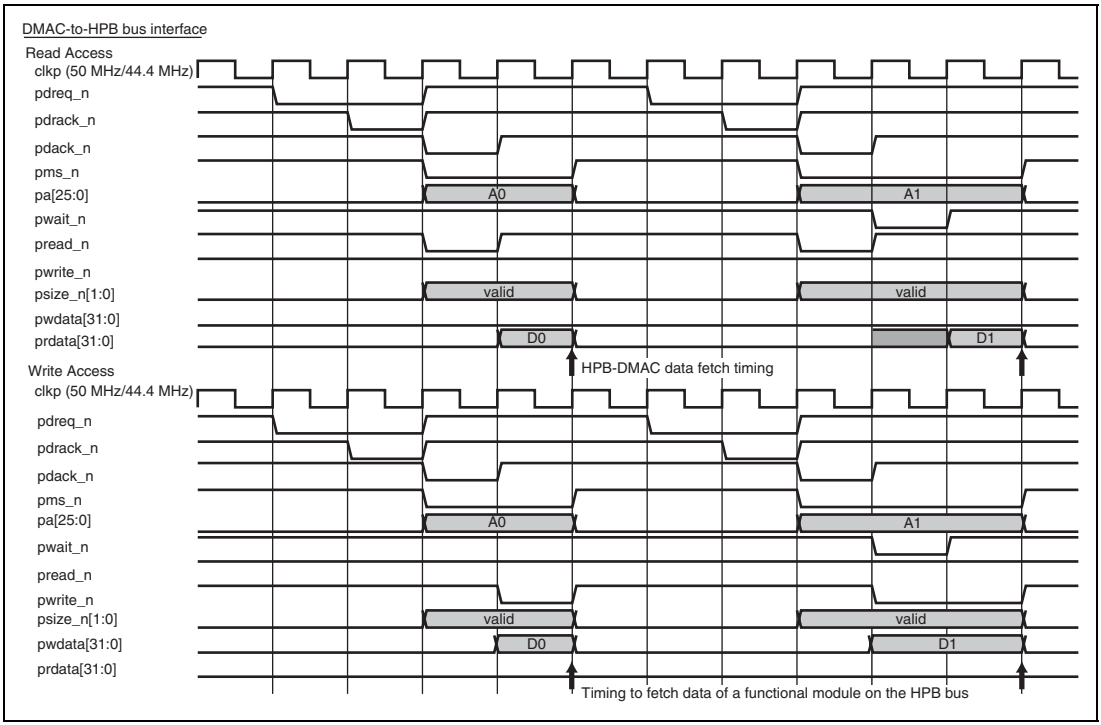


Figure 6.6 Timing Chart of HPB Bus Access (DMA Operation) from HPB-DMAC

6.6.4 Bus Arbitration

(1) Bus Arbitration on HPB Bus Output

The bus arbitration circuit arbitrates any bus access contention between DMA access operation and PIO operation. The bus arbitration method employed is a fixed priority scheme, wherein DMA access operations are performed in priority over PIO access operations. However, because functional modules on the HPB bus that do not have DMA requests are multi-connected to the HPB P1BUS to which functional modules that only accept PIO access are connected, no bus contention occurs on the HPB P1BUS. Consequently, no bus arbitration exists involving such modules when the HPB bus is accessed.

(2) Bus Arbitration on SuperHyway Bus Output

The HPB incorporates 28 HPB-DMAC channels, which can operate concurrently. In order to perform transfers to the DDR2_SDRAM/DDR3_SDRAM, these DMACs access the SuperHyway bus. Consequently, these access events cause access contention within the HPB when an attempt is made to output data to the SuperHyway bus. To resolve this issue, bus arbitration is performed on a round-robin, equal priority basis so that bus access to the SuperHyway bus is performed sequentially by selected channels.

Section 6A Direct Memory Access Controller for Local Bus and Peripheral Modules (LBSC-DMAC/HPB-DMAC)

6A.1 Overview

The LBSC-DMAC and HPB-DMAC are DMA controllers with the common architecture, sharing essentially the same methodology of operation. The LBSC-DMAC performs DMA transfer between the external bus (EX-BUS) and DDR2-SDRAM/DDR3-SDRAM, while the HPB-DMAC performs DMA transfer between peripherals on the HPB bus and DDR2-SDRAM/DDR3-SDRAM. The LBSC-DMAC is assigned channel numbers 0 to 2, and the HPB-DMAC is assigned channel numbers 0 to 28 (channel 26 is not in use), so that a total of thirty-one channels are provided within the chip. For each of these DMA channels, a different transfer destination can be selected, for independent parallel operation.

Different modes of data transfer mode are selectable for each channel. In this section, the character "n" refers to one of the thirty-one DMA channels for the DMACs.

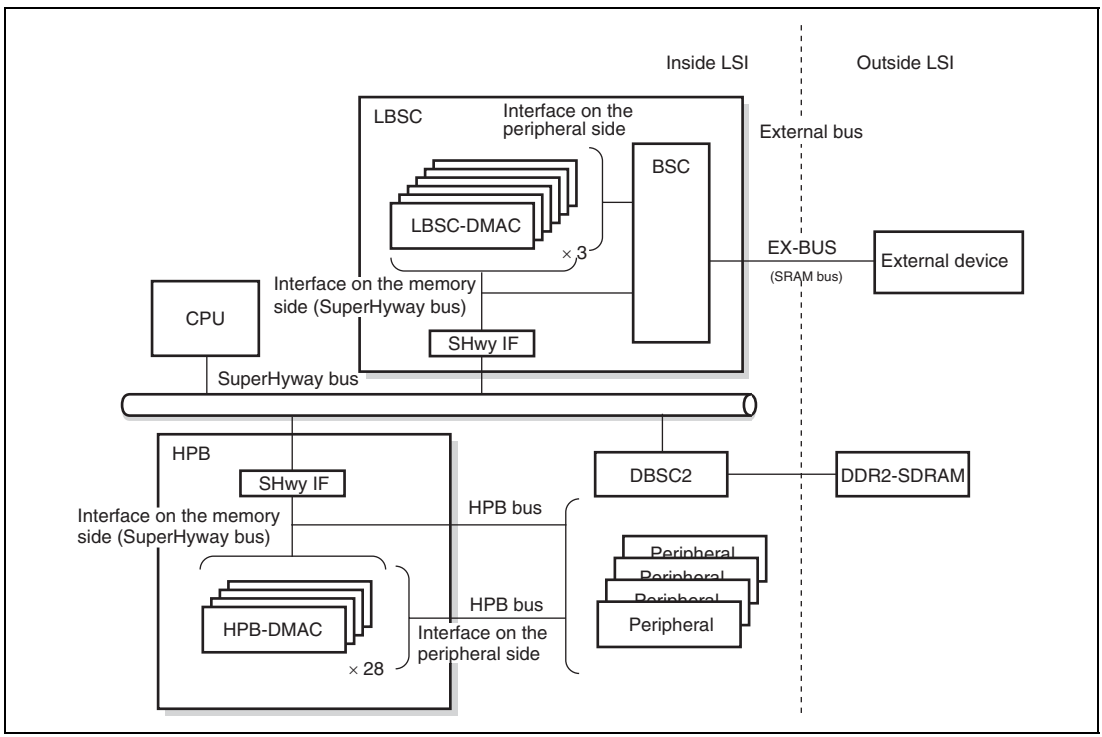


Figure 6A.1 Context of LBSC-DMAC and HPB-DMAC

Figure 6A.1 shows the context of the LBSC-DMAC and HPB-DMAC. The DMACs are connected to DDR2-SDRAM/DDR3-SDRAM outside the chip via the DBSC2, connected via the memory (SuperHyway bus)-side interface, and to peripherals connected to the HPB bus and external bus via the peripheral-side interface. Selection of these peripherals is performed by setting registers within each of the DMACs. Of these, the LBSC-DMAC supports DMA transfers with general-purpose external devices on the EX-BUS (SRAM bus), and transfer can be performed under handshaking control by the DMA request and DMA acknowledge signals (these channels also support an auto-request mode with no handshaking).

6A.2 Features

The DMAC has the following features.

- Number of channels: Three channels (channels 0 to 2) for the LBSC-DMAC, and twenty-eight channels (channels 0 to 28) for the HPB-DMAC (channel 26 is not in use)
- Address space: Physical address space
- Transfer direction: Capable of transfer from a peripheral module to a memory (SuperHyway bus), or from a memory (SuperHyway bus) to a peripheral module
- Transfer data length:
  - For peripherals: Selectable from 1, 2, or 4 bytes.
  - For memory (SuperHyway bus):

    When the SWMD bit in the DCR register is cleared to 0: Access size specified for memory (SupwerHyway bus) (For details on each channel access size, see section 6A.4.8, DMA Control Register (DCR))
    When the SWMD bit in the DCR register is set to 1: Selectable from 1, 2, or 4 bytes
    When final packing processing is performed while the PKMD bit in the DCR register is set to 1: 1 byte or the access size specified for memory (SuperHyway bus)
- Transfer burst length: 1 or 8 (transfer with a burst length of 8 is supported only for LBSC-DMAC channels 0, 1 and HPB-DMAC channels 19, 20, and 23 to 28.)
- Transfer count:
  - Maximum: 16 M (16,777,216) times or 64 M (67,108,864) (64 M is supported only for channel 0 of LBSC-DMAC)
  - Minimum: One time

- Address mode: Dual address mode
  - — Dual address mode

    Both the transfer source and transfer destination are accessed by using addresses.

    Values set in the DMAC's internal registers indicate the addresses for both the transfer source and the transfer destination. (Registers: DMA source address register (DSAR0 or DSAR1), DMA destination address register (DDAR0 or DDAR1), and DMA control register (DCR; bits SPDAM and DPDAM).
- Transfer requests: The external requests, peripheral requests, auto-requests, and timer requests are supported.
  - — External requests

    Requests from three external DREQ pins. Low or high level detection or edge detection can be specified for the request signals (DREQ) in the LBSC register. Either the active high or active low logic can be specified by LBSC for the request acknowledge level signal (DRACK) and the acknowledge level signal (DACK). (DRACK supports only channel 0 of the LBSC-DMACs.)
  - — Peripheral request

    Transfer requests from on-chip peripheral modules.
  - — Auto-request

    Initiates DMA transfer according to the DMAC internal timing.
  - — Timer request

    A transfer request is generated at an interval specified by a timer in the DMAC.
- Transfer modes: Single transfer and continuous transfer modes are supported.
  - — Single transfer mode

    DMA transfer ends when transfer is completed for the transfer count specified by the DMA transfer count register.
  - — Continuous transfer mode

    Available in all channels. If there is a next DMA transfer request (DNXT) when transfer is completed for the transfer count specified by the DMA transfer count register, the next DMA transfer information is fetched and the next DMA transfer is continued. If no next DMA transfer request (DNXT) is found, the DMAC waits until a next DMA transfer request is specified. The continuous transfer mode is terminated by the DQEND bit in the DMA command register (DCMDR).

    DMA information can be specified in two modes: one mode uses one of two sets of DMA information registers repeatedly, and the other uses the two sets alternately.

The DMAC also provides the automatic continuous transfer mode. The automatic continuous transfer mode is enabled by setting the ACMD bit in the DMA control register (DCR) to 1 while the continuous transfer mode is enabled (the CT bit in DCR is set to 1). In this mode, when transfer is completed for the transfer count specified by the DMA transfer count register, the DMAC fetches the next DMA transfer information and continues DMA transfer regardless of whether there is a next DMA transfer request (DNXT). This mode is terminated by the DQEND bit in the DMA command register (DCMDR).

- Transfer end interrupt: An interrupt request can be sent to the CPU on completion of the number of transfers specified for each DMA information unit.
- Supports DMA operation for UltraATA (LBSC-DMAC channels 0 and 1)
  - Timer monitoring function for temporarily queued requests, write-data setup/hold setting function, and CRC indicating function in DMA operations for UltraATA.

6A.3 DMA Transfer Method in LBSC-DMAC/HPB-DMAC

The relation of each DMA channel to peripherals is shown below. Transfers can be performed with between various functional blocks over different channels, but if destination is on the memory (SuperHyway) side, DDR2-SDRAM/DDR3-SDRAM via the DBSC2 and so on becomes the destination for transfer according to the address setting.

Table 6A.1 DMA Transfer Specifications

Channel	Application	Communication Type	Selection of Communication Remote Device
LBSC-DMA00	Communications with general-use device via EX-BUS	Dual address transfer, single/8-burst transfer DREQ/DACK handshake Channel 0 can select DRACK. Channels 0 and 1 can select UltraATA DMA function.	DREQ/DACK number corresponds to DMAC channel number. (External bus spaces allocated to each DMAC are specified by the LBSC internal register.)
LBSC-DMA01			
LBSC-DMA02		Dual address transfer, single transfer DREQ/DACK handshake	
HPB-DMA00	Communications with a peripheral via the internal HPB P2BUS	Dual address transfer, single transfer DREQ/DACK handshake	Internal peripheral for communication is selected by the internal registers in each DMAC. (SCIF0 to 5, HSPI, FLCTL, ADC, or MTU2 is selectable.)
HPB-DMA01			
HPB-DMA02			
HPB-DMA03			
HPB-DMA04			
HPB-DMA05			
HPB-DMA06			
HPB-DMA07			
HPB-DMA08			
HPB-DMA09			
HPB-DMA10			

Channel	Application	Communication Type	Selection of Communication Remote Device
HPB-DMA11	Communications with peripherals via the internal HPB P4BUS	Dual address transfer, single transfer DREQ/DACK handshake	Internal peripheral for communication is selected by the internal registers in each DMAC. (SSI 0 to 3, SRS0,1, IIC30, 1, or MIMLB is selectable.)
HPB-DMA12			
HPB-DMA13			
HPB-DMA14			
HPB-DMA15			
HPB-DMA16			
HPB-DMA17			
HPB-DMA18			
HPB-DMA19	Communications with peripherals via the internal HPB P3BUS	Dual address transfer, single/8-burst transfer (recommended) DREQ/DACK handshake	Internal peripheral for communication is selected by the internal registers in each DMAC. (USB Function 0 or 1 is selectable.)
HPB-DMA20			
HPB-DMA21	Communications with peripherals via the internal HPB P5BUS	Dual address transfer, single transfer DREQ/DACK handshake	Internal peripheral for communication is selected by the internal registers in each DMAC. (HSCIF is selectable.)
HPB-DMA22			
HPB-DMA23	Communications with peripherals via the internal HPB P6BUS	Dual address transfer, single/8-burst transfer DREQ/DACK handshake	Internal peripheral for communication is selected by the internal registers in each DMAC. (SDHI0 to 2 are selectable.)
HPB-DMA24			
HPB-DMA25			
HPB-DMA26	Reserved		
HPB-DMA27	Communications with peripherals via the internal HPB P8BUS	Dual address transfer, single/8-burst transfer DREQ/DACK handshake	Internal peripheral for communication is selected by the internal registers in each DMAC. (MMC, RSPI, or RQSPI is selectable.)
HPB-DMA28			

6A.4 Register Configuration

Registers in the LBSC-DMAC and HPB-DMAC are mapped to the SH register map space. The DMAC has thirty-one channels; some registers are prepared for each individual channel and some are used by all channels in common.

- LBSC-DMAC Register Map

Table 6A.2 (1) List of LBSC-DMAC Registers

Address (Bytes)	Name		Abbreviation	Access Type	Access Size
H'FF801000 + H'40 × [n]	[Individual] DMA source address register 0	DMA information register set 0	DSAR0	R/W	32
H'FF801004 + H'40 × [n]	[Individual] DMA destination address register 0		DDAR0	R/W	32
H'FF801008 + H'40 × [n]	[Individual] DMA transfer count register 0		DTCR0	R/W	32
H'FF80100C + H'40 × [n]	[Individual] DMA source address register 1	DMA information register set 1	DSAR1	R/W	32
H'FF801010 + H'40 × [n]	[Individual] DMA destination address register 1		DDAR1	R/W	32
H'FF801014 + H'40 × [n]	[Individual] DMA transfer count register 1		DTCR1	R/W	32
H'FF801018 + H'40 × [n]	[Individual] DMA source address status register		DSASR	R	32
H'FF80101C + H'40 × [n]	[Individual] DMA destination address status register		DDASR	R	32
H'FF801020 + H'40 × [n]	[Individual] DMA transfer count status register		DTCSR	R	32
H'FF801028 + H'40 × [n]	[Individual] DMA control register		DCR	R/W	32

Address (Bytes)	Name	Abbreviation	Access Type	Access Size
H'FF80102C + H'40 × [n]	[Individual] DMA command register	DCMDR	—/W	32
H'FF801030 + H'40 × [n]	[Individual] DMA forced stop register	DSTPR	—/W	32
H'FF801034 + H'40 × [n]	[Individual] DMA status register	DSTSR	R	32
H'FF801038 + H'40 x [n]	[Individual] DMA channel debugging register	DDBGR	R/W	32
H'FF80103C + H'40 x [n]	[Individual] DMA channel debugging register 2	DDBGR2	R/W	32
H'FF801400	[Common to LBSC-DMAC] DMA timer control register	DTIMR	R/W	32
H'FF801404	[Common to LBSC-DMAC] DMA request mask control register	DRMSKR	R/W	32
H'FF80140C	[Common to LBSC-DMAC] DMA memory access priority level control register	DMLVLR	R/W	32
H'FF801410	[Common to LBSC-DMAC] DMA transfer end interrupt status register	DINTSR	R	32
H'FF801414	[Common to LBSC-DMAC] DMA transfer end interrupt status clear register	DINTCR	—/W	32
H'FF801418	[Common to LBSC-DMAC] DMA transfer end interrupt enable register	DINTMR	R/W	32
H'FF801420	[Common to LBSC-DMAC] DMA activation status register	DACTSR	R	32
H'FF801424 to H'FF80142C	[Common to LBSC-DMAC] DMA00 to DMA02 channel software-reset register	LSRSTR0 to LSRSTR2	R/WC1	32
H'FF801480	[Common to LBSC-DMAC] External-DMA data alignment control register	DMALGR	R/W	32
H'FF801490	[Common to LBSC-DMAC] LBSC-DMA SHwy priority control register	LBSC-DMASPR	R/W	32
H'FF8014C0	[Common to LBSC-DMAC] UltraATA DMA mode register	UATMR	R/W	32
H'FF8014C4	[Common to LBSC-DMAC] UltraATA write cycle setting register	UATWCR	R/W	32
H'FF8014C8	[Common to LBSC-DMAC] UltraATA timeout period setting register	UATTSR	R/W	32

Address (Bytes)	Name	Abbreviation	Access Type	Access Size
H'FF8014CC	[Common to LBSC-DMAC] UltraATA error indication register	UATTER	R/W	32
H'FF8014D0	[Common to LBSC-DMAC] UltraATA error interrupt enable register	UATIER	R/W	32
H'FF8014D4	[Common to LBSC-DMAC] UltraATA CRC code indication register	UATCRCR	R	32
H'FF8014E0	[Common to LBSC-DMAC] UltraATA DMA mode register 2	UATMR2	R/W	32
H'FF8014E4	[Common to LBSC-DMAC] UltraATA DMA mode register 3	UATMR3	R/W	32
H'FF800030	[Common to LBSC-DMAC] UltraATA transfer mode register	UATTMR	R/W	32

[Legend]

[n]: LBSC-DMAC channel number

Notes: 1. The CPU should access the above register in longword units (32 bits). The CPU should not access the above register in byte or word units.

2. Do not write to any addresses other than listed above. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than listed above are undefined.

Table 6A.2 (2) Register State in Each Operating Mode

Abbreviation	Power-on Reset (by the $\overline{\text{PRESET}}$ pin, WDT, or H-UDI)	Manual Reset (by the $\overline{\text{PRESET}}$ pin, WDT, or multiple exception)	Sleep	Software Standby	Module Standby	Deep Standby
DSAR0	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DDAR0	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DTCR0	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DSAR1	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DDAR1	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DTCR1	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DSASR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DDASR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DTCSR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DCR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DCMDR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DSTPR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DSTSR	H'0000_0020	H'0000_0020	Retained	Retained	—	Initialized
DDBGR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DDBGR2	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DTIMR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DRMSKR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DMLVLR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DINTSR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DINTCR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DINTMR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DACTSR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
LSRSTR0 to LSRSTR2	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DMALGR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
LBSC-DMASPR	H'0000_0888	H'0000_0888	Retained	Retained	—	Initialized
UATMR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
UATWCR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized

Abbreviation	Power-on Reset (by the $\overline{\text{PRESET}}$ pin, WDT, or H-UDI)	Manual Reset (by the $\overline{\text{PRESET}}$ pin, WDT, or multiple exception)	Sleep	Software Standby	Module Standby	Deep Standby
UATTSR	H'FFFF_FFFF	H'FFFF_FFFF	Retained	Retained	—	Initialized
UATTER	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
UATIER	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
UATCRCR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
UATMR2	H'FFFF_FFFF	H'FFFF_FFFF	Retained	Retained	—	Initialized
UATMR3	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
UATTMR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized

Note: Initialized: The value is the one written in "Power-on Reset".

- HPB-DMAC Register Map

Table 6A.3 (1) List of HPB-DMAC Registers

Address (Bytes)	Name		Abbreviation	Access Type	Access Size
H'FFC08000 + H'40 × [n]	[Individual] DMA source address register 0	DMA information register set 0	DSAR0	R/W	32
H'FFC08004 + H'40 × [n]	[Individual] DMA destination address register 0		DDAR0	R/W	32
H'FFC08008 + H'40 × [n]	[Individual] DMA transfer count register 0		DTCR0	R/W	32
H'FFC0800C + H'40 × [n]	[Individual] DMA source address register 1	DMA information register set 1	DSAR1	R/W	32
H'FFC08010 + H'40 × [n]	[Individual] DMA destination address register 1		DDAR1	R/W	32
H'FFC08014 + H'40 × [n]	[Individual] DMA transfer count register 1		DTCR1	R/W	32
H'FFC08018 + H'40 × [n]	[Individual] DMA source address status register		DSASR	R	32
H'FFC0801C + H'40 × [n]	[Individual] DMA destination address status register		DDASR	R	32

Address (Bytes)	Name	Abbreviation	Access Type	Access Size
H'FFC08020 + H'40 × [n]	[Individual] DMA transfer count status register	DTCSR	R	32
H'FFC08024 + H'40 × [n]	[Individual] DMA port selection register	DPTR	R/W	32
H'FFC08028 + H'40 × [n]	[Individual] DMA control register	DCR	R/W	32
H'FFC0802C + H'40 × [n]	[Individual] DMA command register	DCMDR	—/W	32
H'FFC08030 + H'40 × [n]	[Individual] DMA forced stop register	DSTPR	—/W	32
H'FFC08034 + H'40 × [n]	[Individual] DMA status register	DSTSR	R	32
H'FFC08038 +H'40 × [n]	[Individual] DMA channel debugging register	DDBGR	R/W	32
H'FFC0803C +H'40 × [n]	[Individual] DMA channel debugging register 2	DDBGR2	R/W	32
H'FFC08800	[Common to HPB-DMAC] DMA timer control register	DTIMR	R/W	32
H'FFC0880C	[Common to HPB-DMAC] DMA transfer end interrupt status register	DINTSR	R	32
H'FFC08810	[Common to HPB-DMAC] DMA transfer end interrupt status clear register	DINTCR	—/W	32
H'FFC08814	[Common to HPB-DMAC] DMA transfer end interrupt enable register	DINTMR	R/W	32
H'FFC08818	[Common to HPB-DMAC] DMA activation status register	DACTSR	R	32
H'FFC0881C to H'FFC0888C	[Common to HPB-DMAC] HPB DMA00 to DMA28 channel software-reset registers	HSRSTR0 to HSRSTR28	R/WC1	32
H'FFC08890 to H'FFC0889C	[Common to HPB-DMAC] HPB-DMA SHwy priority control registers 0 to 3	HPB-DMASPR0 to HPB-DMASPR3	R/W	32
H'FFC088A0	[Common to HPB-DMAC] HPB-DMA access priority level control register	HPB-DMLVLR	R/W	32

[Legend]

[n]: HPB-DMAC channel number

Notes: 1. The CPU should access the above register in longword units (32 bits). The CPU should not access the above register in byte or word units.

2. Do not write to any addresses other than listed above. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than listed above are undefined.

Table 6A.3 (2) Register State in Each Operating Mode

Abbreviation	Power-on Reset (by the $\overline{\text{PRESET}}$ pin, WDT, or H-UDI)	Manual Reset (by the $\overline{\text{PRESET}}$ pin, WDT, or multiple exception)	Sleep	Software Standby	Module Standby	Deep Standby
DSAR0	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DDAR0	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DTCR0	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DSAR1	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DDAR1	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DTCR1	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DSASR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DDASR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DTCSR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DPTR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DCR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DCMDR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DSTPR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DSTSR	H'0000_0020	H'0000_0020	Retained	Retained	—	Initialized
DDBGR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DDBGR2	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DTIMR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DINTSR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DINTCR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DINTMR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
DACTSR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
HSRSTR0 to HSRSTR28	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
HPB-DMASPR0	H'8888_8888	H'8888_8888	Retained	Retained	—	Initialized
HPB-DMASPR1	H'8888_8888	H'8888_8888	Retained	Retained	—	Initialized

Abbreviation	Power-on Reset (by the $\overline{\text{PRESET}}$ pin, WDT, or H-UDI)	Manual Reset (by the $\overline{\text{PRESET}}$ pin, WDT, or multiple exception)	Sleep	Software Standby	Module Standby	Deep Standby
HPB-DMASPR2	H'8888_8888	H'8888_8888	Retained	Retained	—	Initialized
HPB-DMASPR3	H'0008_8088	H'0008_8088	Retained	Retained	—	Initialized
HPB-DMLVLR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized

Note: Initialized: The value is the one written in "Power-on Reset".

6A.4.1 DMA Source Address Registers 0, 1 (DSAR0, DSAR1)

Each register specifies the DMA start address of the transfer source.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DSA[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DSA[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DSA[31:0]	All 0	R/W	DMA Transfer Source Start Address The transfer source start address indicates a memory address, a peripheral module address, or an external module address according to the SMDL bit value in the DMA control register (DCR). When SMDL = 0: Transfer source address = Memory (SuperHyway bus) address When SMDL = 1: Transfer source address = Peripheral address

Notes: 1. When the address setting is a memory address, the following address boundary should be used.

	LBSC-DMAC Channel Number		HPB-DMAC Channel Number		
SWMD bit in the DCR register	0, 1	2	4 to 10	0 to 3	11 to 25, 27, 28
0	32-byte boundary	16-byte boundary	8-byte boundary	16-byte boundary	32-byte boundary
1	4-byte boundary				

2. When the address setting is an external module address, and in addition the SPDS or DPDS bit in DCR selects a 16-bit access size, up to a 16-bit boundary can be set. In this case, if an 8-bit boundary is set, upon writing the lower 1 bit is ignored.

3. When the address set is an external module address, and in addition the SPDS or DPDS bit in DCR selects an 8-bit access size, up to an 8-bit boundary can be set.
4. In the case of note 2, when the DMAC external module data access size is smaller than the external bus width setting in LBSC, the DMALGR0/1 register in the DMAC can be used to specify whether there is a change in data alignment (data access byte lane according to the address value). See sections 6A.4.22, External DMA Data Alignment Control Register (DMALGR) and 6B, LBSC within Bus Bridge.
5. When an address setting is an external module address, the upper address bits 31 to 26 are not connected to the external bus. These are provided for ease in understanding the contents of software settings. Also, CS1 and EX_CS0 to EX_CS5 space identification using these upper bits is not performed. DMAC access destination space is identified through the LBSC external DMAC channel area allocation register.
6. When an address setting is for a peripheral device, the upper address is provided for ease of understanding of the contents of software settings, and is not used for identification of access destination space specific to the peripheral.

6A.4.2 DMA Destination Address Registers 0, 1 (DDAR0, DDAR1)

Each register specifies the DMA start address of the transfer destination.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DDA[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DDA[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DDA[31:0]	All 0	R/W	DMA Transfer Destination Start Address The transfer destination start address indicates a memory address, a peripheral module address, or an external module address according to the DMDL bit value in the DMA control register (DCR). When DMDL = 0: Transfer destination address = Memory address When DMDL = 1: Transfer destination address = Peripheral module or external module address

Note: See the notes 1 to 6 in section 6A.4.1, DMA Source Address Registers 0, 1 (DSAR0, DSAR1). Those notes also apply to these registers.

6A.4.3 DMA Transfer Count Registers 0, 1 (DTCR0, DTCR1)

Each register specifies the DMA transfer count.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	DTC[25:16]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit::	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25 to 0	DTC[25:0]	All 0	R/W	DMA Transfer Count These bits specify the DMA transfer count (number of bytes, words, or longwords). LBSC-DMAC channel 0: The maximum count is DTC = 0h, which indicates 64 M (67,108,864) times. Other than LBSC-DMAC channel 0: The maximum count is DTC = 0h, which indicates 16 M (16,777,216) times.

Note: This register specifies the transfer count on the peripheral side for transfer from a peripheral to a memory (SuperHyway bus) or from a memory (SuperHyway bus) to a peripheral. For 8-burst DMA operation, one count for each 8-burst operation.

6A.4.4 DMA Source Address Status Register (DSASR)

DSASR indicates the transfer source address.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DSAS[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DSAS[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DSAS[31:0]	All 0	R	These bits indicate the latest source address for which DMA transfer has been completed.

6A.4.5 DMA Destination Address Status Register (DDASR)

DDASR indicates the transfer destination address.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DDAS[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DDAS[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DDAS[31:0]	All 0	R	These bits indicate the latest destination address for which DMA transfer has been completed.

6A.4.6 DMA Transfer Count Status Register (DTCSR)

DTCSR indicates the remaining count of the current transfer.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	DTCS[25:16]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTCS[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25 to 0	DTCS[25:0]	All 0	R	Remaining DMA Transfer Count These bits indicate the remaining count of the current DMA transfer (number of bytes, words, or longwords). Note that bits 25 to 0 are used only for the LBSC-DMAC channel 0 and bits 23 to 0 are used for the other channels. The number of transferred bytes depends on the peripheral's data bus width.

Note: This register indicates the remaining transfer count on the peripheral side for transfer from a peripheral to a memory (SuperHyway bus) or from a memory (SuperHyway bus) to a peripheral. In 8-burst DMA operation, 8-burst transfer is counted as one.

6A.4.7 DMA Port Select Register (DPTR)

DPTR selects the peripheral for DMA transfer. (HPB-DMAC dedicated register)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	SDPT[4:0]					—	—	—	DDPT[4:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 8	SDPT[4:0]	H'00	R/W	Transfer Source Peripheral Access Port Select (See table 6A.4.)
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4 to 0	DDPT[4:0]	H'00	R/W	Transfer Destination Peripheral Access Port Select (See table 6A.5.)

Notes: 1. It is prohibited to specify the same module in multiple channels and to transfer data in the same direction in those channels.
2. When the SMDL bit in DCR is cleared to 0 (when a memory is selected), the SDPT bit setting is ignored. When the DMDL bit in DCR is cleared to 0 (when a memory is selected), the DDPT bit setting is ignored.
3. Setting the values other than those to which a module is assigned for each channel is prohibited. If those values are set, operation cannot be guaranteed.

Table 6A.4 Transfer Source Peripheral Access Port Select

SDPT[12:8]	HPB-DMAC channels 0 to 3	HPB-DMAC channels 4 to 10	HPB-DMAC channels 11 to 18	HPB-DMAC channels 19 and 20	HPB-DMAC channels 21 and 22	HPB-DMAC channels 23 to 25	HPB-DMAC channels 27 and 28
H'0	SCIF0	SCIF0	SSI0	USBF0	HSCIF	—	MMC0
H'1	SCIF1	SCIF1	SSI1	USBF1	—	—	MMC1
H'2	SCIF2	SCIF2	SSI2	—	—	—	RSPI
H'3	SCIF3	SCIF3	SSI3	—	—	—	—
H'4	SCIF4	SCIF4	—	—	—	—	RQSPI
H'5	SCIF5	SCIF5	—	—	—	—	—
H'6	—	—	—	—	—	—	—
H'7	—	—	—	—	—	—	—
H'8	—	—	SRC0*[1]	—	—	—	—
H'9	—	—	SRC1*[2]	—	—	—	—
H'A	HSPI	HSPI	IIC30	—	—	—	—
H'B	—	—	IIC31	—	—	—	—
H'C	—	—	MIMLB*[3]	—	—	—	—
H'D	—	—	—	—	—	—	—
H'E	FLCTL0	FLCTL0	MIMLB*[4]	—	—	—	—
H'F	FLCTL1	FLCTL1	—	—	—	—	—
H'10	ADC	ADC	—	—	—	—	—
H'11	MTU20	MTU20	—	—	—	SDHI0	—
H'12	MTU21	MTU21	—	—	—	—	—
H'13	MTU22	MTU22	—	—	—	SDHI1	—
H'14	MTU23	MTU23	—	—	—	—	—
H'15	MTU24	MTU24	—	—	—	SDHI2	—

Notes: Setting is prohibited for " —" and H'16 to H'1F.

1. Output FIFO full (converted data are to be written back)
2. Output FIFO full (converted data are to be written back)
3. Stream reception
4. Packet reception

Table 6A.5 Transfer Destination Peripheral Access Port Select

DDPT[4:0]	HPB-DMAC channels 0 to 3	HPB-DMAC channels 4 to 10	HPB-DMAC channels 11 to 18	HPB-DMAC channels 19 and 20	HPB-DMAC channels 21 and 22	HPB-DMAC channels 23 to 25	HPB-DMAC channels 27 and 28
H'0	SCIF0	SCIF0	SSI0	USBF0	HSCIF	—	MMC0
H'1	SCIF1	SCIF1	SSI1	USBF1	—	—	MMC1
H'2	SCIF2	SCIF2	SSI2	—	—	—	RSPI
H'3	SCIF3	SCIF3	SSI3	—	—	—	—
H'4	SCIF4	SCIF4	—	—	—	—	RQSPI
H'5	SCIF5	SCIF5	—	—	—	—	—
H'6	—	—	—	—	—	—	—
H'7	—	—	—	—	—	—	—
H'8	—	—	SRC0*[1]	—	—	—	—
H'9	—	—	SRC1*[2]	—	—	—	—
H'A	HSPI	HSPI	IIC30	—	—	—	—
H'B	—	—	IIC31	—	—	—	—
H'C	—	—	MIMLB*[3]	—	—	—	—
H'D	—	—	MIMLB*[4]	—	—	—	—
H'E	FLCTL0	FLCTL0	—	—	—	—	—
H'F	FLCTL1	FLCTL1	—	—	—	—	—
H'10	—	—	—	—	—	SDHI0	—
H'11	MTU20	MTU20	—	—	—	—	—
H'12	MTU21	MTU21	—	—	—	SDHI1	—
H'13	MTU22	MTU22	—	—	—	—	—
H'14	MTU23	MTU23	—	—	—	SDHI2	—
H'15	MTU24	MTU24	—	—	—	—	—

Notes: Setting is prohibited for "—" and H'16 to H'1F.

1. Input FIFO empty (data are to be input to the SRC)
2. Input FIFO empty (data are to be input to the SRC)
3. Stream transmission
4. Packet transmission

6A.4.8 DMA Control Register (DCR)

DCR specifies the transfer mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	DTAMD	DTAC	DTAU	DTAU1	SWMD	BTMD	PKMD	—	CT	ACMD	DIP
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	SMDL	SPDAM	SDRMD[1:0]		SPDS[1:0]		—	—	DMDL	DPDAM	DDRMD[1:0]		DPDS[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26	DTAMD	0	R/W	Specifies the data alignment conversion mode when a memory is accessed in DMA transfer (see section 6A.5.7, Data Alignment in SuperHyway Bus Interface). 0: Conversion according to the combination of the input pin (little: endian mode) and the peripheral bus width. 1: Conversion according to the combination of DTAC (DMA data alignment conversion), DTAU (DMA data alignment unit), and DTAU1 (8-byte conversion in 4-byte units).
25	DTAC	0	R/W	Enables or disables data alignment conversion when a memory is accessed in DMA transfer (see section 6A.5.7, Data Alignment in SuperHyway Bus Interface). This setting is valid when DTAMD = 1. 0: Disables data alignment conversion. 1: Enables data alignment conversion.

Bit	Bit Name	Initial Value	R/W	Description
24	DTAU	0	R/W	Specifies the unit for data alignment conversion (see section 6A.5.7, Data Alignment in SuperHyway Bus Interface). This setting is valid when DTAMD = 1. 0: Byte units 1: Word units
23	DTAU1	0	R/W	Specifies whether 8-byte data alignment is performed in 4 bytes unit (see section 6A.5.7, Data Alignment in SuperHyway Bus Interface). The setting value is valid when DTAMD = 1. 0: Not performed. 1: Performed.
22	SWMD	0	R/W	Specifies memory (SuperHyway bus) access size. 0: * (clearing this bit to 0 is recommended when DDR2-SDRAM is specified.) 1: 4 bytes (set this bit to 1 when address on the HPB side is specified.)
21	BTMD	0	R/W	Specifies the burst DMA transfer (only for LBSC-DMAC channels 0, 1 and HPB-DMAC channels 19, 20, and 23 to 28). Burst DMA transfer is performed for peripherals. 0: Does not transfer in burst mode. 1: Transfers in burst mode (burst length is fixed to eight).
20	PKMD	0	R/W	Enables or disables packing of data read from a peripheral for DMA transfer from the peripheral to the SuperHyway bus. 0: Disables packing. 1: Enables packing.
19	—	0	R	Reserved These bits are always read as 0. The write value should always be 0.
18	CT	0	R/W	Specifies continuous DMA transfer. 0: Does not transfer in continuous mode. 1: Transfers in continuous mode.

Bit	Bit Name	Initial Value	R/W	Description
17	ACMD	0	R/W	Specifies automatic continuous DMA transfer (valid only when CT = 1). 0: Does not transfer in automatic continuous mode (checks DNXT = 1 in DCMDR). 1: Transfers in automatic continuous mode (regardless of the DNXT bit in DCMDR).
16	DIP	0	R/W	Specifies the valid DMA information set(s). 0: Uses one DMA information set repeatedly. 1: Uses two DMA information sets alternately.
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13	SMDL	0	R/W	Selects the transfer source module. 0: Memory (SuperHyway bus) 1: Peripheral
12	SPDAM	0	R/W	Specifies whether to fix or increment the transfer source peripheral address. 0: Fixes the peripheral address at the value specified in DSAR0 or DSAR1. 1: Increments the peripheral address (increments by one for 8-bit transfer, by two for 16-bit transfer, or by 4 for 32-bit transfer). When SMDL is set to 0, this bit should be set to 0 or the same value as set in DPDAM.
11, 10	SDRMD[1:0]	00	R/W	These bits specify the DMA request mode for the transfer source. 00: Module request (external request or peripheral module request) 01: Auto-request 10: Timer request 11: Setting prohibited When SMDL is set to 0, these bits should be set to 00 or the same value as set in DDRMD.

Bit	Bit Name	Initial Value	R/W	Description
9, 8	SPDS[1:0]	00	R/W	These bits specify the data bus width for the transfer source peripheral. 00: 8 bits 01: 16 bits 10: 32 bits 11: Setting prohibited When SMDL is set to 0, these bits should be set to 00 or the same value as set in DPDS.
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	DMDL	0	R/W	Selects the transfer destination module. 0: Memory (SuperHyway bus) 1: Peripheral
4	DPDAM	0	R/W	Specifies whether to fix or increment the transfer destination peripheral address. 0: Fixes the peripheral address at the value specified in DDAR0 or DDAR1. 1: Increments the peripheral address (increments by one for 8-bit transfer, by two for 16-bit transfer, or by 4 for 32-bit transfer). When DMDL is set to 0, this bit should be set to 0 or the same value as set in SPDAM.
3, 2	DDRMD[1:0]	00	R/W	These bits specify the DMA request mode for the transfer destination. 00: Module request (external request or peripheral module request) 01: Auto-request 10: Timer request 11: Setting prohibited When DMDL is set to 0, these bits should be set to 00 or the same value as set in SDRMD.

Bit	Bit Name	Initial Value	R/W	Description
1, 0	DPDS[1:0]	00	R/W	These bits specify the data bus width for the transfer destination peripheral. 00: 8 bits 01: 16 bits 10: 32 bits 11: Setting prohibited When DMDL is set to 0, these bits should be set to 00 or the same value as set in SPDS.

Note: * The following table summarizes the memory (SuperHyway) access size in each DMAC channel when SWMD bit is cleared to 0.

DCR Register	LBSC-DMAC Channel Number		HPB-DMAC Channel Number		
SWMD Bit	0, 1	2	4 to 10	0 to 3	11 to 25, 27, 28
0	32 bytes	16 bytes	8 bytes	16 bytes	32 bytes

Notes: 1. When SMDL and DMDL = 1 and 0, data is transferred from a peripheral to a memory. When SMDL and DMDL = 0 and 1, data is transferred from a memory to a peripheral. Setting SMDL and DMDL = "1 and 1" or "0 and 0" is prohibited.

2. Not all DMAC functions can be applied to every peripheral. DMAC functions must be specified appropriately according to the functions and restrictions of each peripheral.

3. As the EX-BUS width is 16 bits, setting "10" (32 bits) is prohibited for the SPDS bits (data bus width for the transfer source peripheral) and DPDS bits (data bus width for the transfer destination peripheral) in the LBSC-DMAC.

6A.4.9 DMA Command Register (DCMDR)

DCMDR activates or stops DMA transfer.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	BD OUT	DQ SPD	DQ SPC	DM SPD	DM SPC	DQ END	DNXT	DMEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	—/W*	—/W*	—/W*	—/W*	—/W*	—/W*	—/W*	—/W*

Note: * The bit is writable. The read value is undefined.

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	BDOUT*[2]	0	—/W*[1]	1: Forcibly writes the data read from a peripheral to the SuperHyway bus side (Transfer direction: from a peripheral to the SuperHyway bus). Writing 1 to this bit forcibly writes to the SuperHyway bus side, and then terminates DMA transfer.
6	DQSPD	0	—/W*[1]	1: Temporarily stops transfer in DMA information units.
5	DQSPC	0	—/W*[1]	1: Cancels temporary transfer stop in DMA information units.
4	DMSPD	0	—/W*[1]	1: Temporarily stops transfer in bus cycle units.
3	DMSPC	0	—/W*[1]	1: Cancels temporary transfer stop in bus cycle units.
2	DQEND	0	—/W*[1]	1: Terminates continuous DMA transfer mode. Only the DMA information specified before is transferred, and then continuous transfer mode is terminated.
1	DNXT	0	—/W*[1]	1: Requests the next DMA transfer. In continuous transfer mode, after the current DMA information is transferred, the next DMA information is transferred.
0	DMEN	0	—/W*[1]	1: Activates DMA transfer.

Notes: 1. The bit is writable. The read value is undefined.

2. For use of BDOUT, see sections 6A.5.3, Packing Data Read from Peripheral or External Module, and 6A.5.4, Limitations on Packing of Data Read from Peripheral or External Module.

6A.4.10 DMA Forced Stop Register (DSTPR)

DSTPR forcibly terminates DMA transfer.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DM STP
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	—/W*

Note: * The bit is writable. The read value is undefined.

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	DMSTP	0	—/W*	1: Forcibly terminates DMA transfer. After the current bus cycle is completed, DMA transfer is terminated. (Values set for the DMA transfer status registers (DSASR, DDASR, and DTCSR) are retained.)

Note: * The bit is writable. The read value is undefined.

6A.4.11 DMA Status Register (DSTSR)

DSTSR indicates the DMA transfer status.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	NDP1	NDP0	DQ SPS	DM SPS	DQ STS	DR STS	DM STS
Initial value:	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

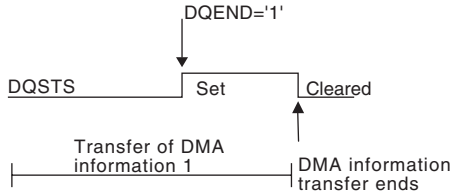
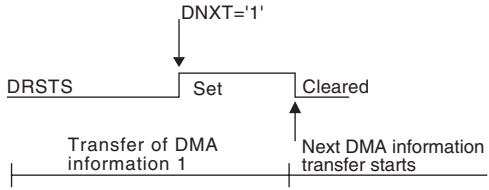
Bit	Bit Name	Initial Value	R/W	Description
31 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6	NDP1	0	R	Next DMA Transfer Information Register Status 1 0: Does not transfer DMA information in DMA information register set 1 in the next DMA information transfer. 1: Transfers DMA information in DMA information register set 1 in the next DMA information transfer.
5	NDP0	1	R	Next DMA Transfer Information Register Status 0 0: Does not transfer DMA information in DMA information register set 0 in the next DMA information transfer. 1: Transfers DMA information in DMA information register set 0 in the next DMA information transfer.
4	DQSPS	0	R	Temporary Stop Status of DMA Information Updating 0: Normal operation 1: DMA information updating is temporarily stopped.
3	DMSPS	0	R	Temporary Stop Status of DMA Transfer 0: Normal operation 1: DMA transfer is temporarily stopped.
2	DQSTS	0	R	DMA Acceptance End Status 0: DMA information can be accepted. 1: DMA information acceptance is stopped.

Bit	Bit Name	Initial Value	R/W	Description
1	DRSTS	0	R	DMA Transfer Request Status 0: Next DMA transfer has not been requested. 1: Next DMA transfer has been requested.
0	DMSTS	0	R	DMA Status 0: DMA transfer has been completed. 1: DMA transfer is active.

Note: Either NDP0 or NDP1 (next DMA transfer information register set 0 or 1) is always set to 1.

The following shows the transition of each bit status in DSTSR.

	Conditions of Status Transition		
	0 →	**1** →	**0**
NDP1	Initial state	Continuous transfer mode is selected and information register sets 0 and 1 are used alternately (DIP in DCR = 1): DMA transfer for information register set 0 is in progress.	Continuous transfer mode is selected and information register sets 0 and 1 are used alternately (DIP in DCR = 1): DMA transfer for information register set 1 is in progress.
NDP0	—	1. Initial state 2. Single transfer mode: Always 3. Continuous transfer mode is selected and information register set 0 is used repeatedly (DIP in DCR = 0): Always 4. Continuous transfer mode is selected and information register sets 0 and 1 are used alternately (DIP in DCR = 1): DMA transfer for information register set 1 is in progress.	Continuous transfer mode is selected and information register sets 0 and 1 are used alternately (DIP in DCR = 1): DMA transfer for information register set 0 is in progress.
DQSPS	Initial state	DMA transfer is temporarily stopped in DMA information units (DQSPD in DCMDR = 1) and the current information transfer is completed.	Temporary stop of DMA transfer in DMA information units is canceled (DQSPC in DCMDR = 1).
DMSPS	Initial state	DMA transfer is temporarily stopped in bus cycle units (DMSPD in DCMDR = 1) and the current bus cycle is completed.	Temporary stop of DMA transfer in bus cycle units is canceled (DMSPC in DCMDR = 1).

	Conditions of Status Transition		
	0 →	**1** →	**0**
DQSTS	Initial state	Continuous transfer mode is selected (CT in DCR = 1) and DMA continuous transfer is terminated (DQEND = 1) during transfer of DMA information 1. 	Transfer of DMA transfer information ends.
DRSTS	Initial state	Continuous transfer mode is selected (CT in DCR = 1) and next DMA transfer information is requested (DNXT = 1) during transfer of DMA information 1. 	Next DMA information transfer is started.

Conditions of Status Transition

	0 →	1 →	0
DMSTS	Initial state	DMA is activated (DMEN in DCMDR = 1).	End state (remains in the idle state). 1. Transfer end a. Single transfer mode: Transfer of one DMA information set is completed. b. Continuous transfer mode: DRSTS = 0 and DQSTS = 1 in DSTSR and transfer of current DMA information is completed. 2. Forced stop DMSTP in DSTPR is set to 1 and DMA transfer is terminated after the current bus cycle is completed.

6A.4.12 DMA Channel Debugging Register (DDBGR)

DDBGR is used for debugging.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DBG02	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	DBG01			—	DBG00		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	DBG02	0	R/W	Test Bit This bit is a test bit and cannot be written to. If this bit is written to, correct operation cannot be guaranteed.
30 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6 to 4	DBG01	000	R	Test Bits
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2 to 0	DBG00	000	R	Test Bits

6A.4.13 DMA Channel Debugging Register 2 (DDBGR2)

DDBGR2 is used to for debugging.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	DBG12	DBG11		—	—	DBG10									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DBG10															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30	DBG12	0	R/W	Test Bit This bit is a test bit and cannot be written to. If this bit is written to, correct operation cannot be guaranteed.
29, 28	DBG11	00	R/W	Test Bits These bits are test bits and cannot be written to. If these bits are written to, correct operation cannot be guaranteed.
27, 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25 to 0	DBG10	All 0	R	Test Bits

6A.4.14 DMA Timer Control Register (DTIMR)

DTIMR specifies the timer cycle in the DMAC.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTIM[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	DTIM[15:0]	All 0	R/W	DMAC Internal Timer Cycle Set Request mode: Specify the DMA request interval in timer request mode. Request interval: DTIM × peripheral-side bus clock cycle (ns) • Peripheral-side bus clock cycle (ns) for LBSC-DMAC: Approximately 22.5 ns (EX-BUS frequency 44.4 MHz for CPU 533.3 MHz) Approximately 20 ns (EX-BUS frequency 50 MHz for CPU 400 MHz) • Peripheral-side bus clock cycle (ns) for HPB-DMAC: Approximately 22.5 ns (HPB frequency 44.4 MHz for CPU 533.3 MHz) Approximately 20 ns (HPB frequency 50 MHz for CPU 400 MHz) • Peripheral-side bus clock cycle (ns) for HPB-DMAC: Approximately 11.25 ns (HPB frequency 88.8 MHz for CPU 533.3 MHz) Approximately 10 ns (HPB frequency 100 MHz for CPU 400 MHz) Note: Even in timer request mode, operation is the same as that in auto-request mode when the value of DTIM is 0.

6A.4.15 DMA Request Mask Control Register (DRMSKR)

DRMSKR specifies the timing for de-masking the DMA request signal in the external-bus DMAC (LBSC-DMAC dedicated register).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	DRMSK2[3:0]				DRMSK1[3:0]				DRMSK0[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 8	DRMSK2 [3:0]	0000	R/W	These bits specify the number of clock cycles from the completion of a DMA transfer (the time of $\overline{CS}$ signal negation) until de-masking for a subsequent DMA request, and each field corresponds to the LBSC-DMAC channel with the same number. Interval until de-masking the DMA request signal: DRMSK × external bus clock period (ns)
7 to 4	DRMSK1 [3:0]	0000	R/W	
3 to 0	DRMSK0 [3:0]	0000	R/W	

Note: Settings for DRMSK2 to DRMSK0 are only valid when level-detection is specified for the external DREQ signal (specified by EXDMCRy in the LBSC) for the channel. In edge-detection mode, set the bits to 0.

6A.4.16 DMA Memory Access Priority Level Control Register (DMLVLR)

DMLVLR specifies the access priority level (1 or 2) (LBSC-DMAC dedicated register).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	DM LV2	DM LV1	DM LV0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	DMLV2 to DMLV0	000	R/W	These bits specify the external bus arbitration priority group for each DMAC channel which corresponds to each bit. 0: Level 2 (Low: Group 2) 1: Level 1 (High: Group 1)

6A.4.17 DMA Transfer End Interrupt Status Register (DINTSR)

DINTSR indicates the DMA transfer end interrupt status (LBSC-DMAC dedicated register).

- LBSC-DMAC Dedicated Register

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	DTE2	DTE1	DTE0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	DTEn	000	R	DMA Transfer End Interrupt Status Each bit indicates the DMA transfer status (n: DMA channel number). 0: Initial state or data is being transferred before the count specified by DTCR is reached. 1: Transfer has been completed for the count specified by DTCR.

- HPB-DMAC Dedicated Register

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	DTE28	DTE27	—	DTE25	DTE24	DTE23	DTE22	DTE21	DTE20	DTE19	DTE18	DTE17	DTE16
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTE15	DTE14	DTE13	DTE12	DTE11	DTE10	DTE9	DTE8	DTE7	DTE6	DTE5	DTE4	DTE3	DTE2	DTE1	DTE0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29, 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28, 27, 25 to 0	DTEn	All 0	R	DMA Transfer End Interrupt Status Each bit indicates the DMA transfer status (n: DMA channel number). 0: Initial state or data is being transferred until the count specified by DTC is reached. 1: Transfer has been completed for the count specified by DTC.

6A.4.18 DMA Transfer End Interrupt Status Clear Register (DINTCR)

DINTCR clears the DMA transfer end interrupt status.

- LBSC-DMAC Dedicated Register

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	DTEC2	DTEC1	DTEC0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	—/W*	—/W*	—/W*

Note: * The bit is writable. The read value is undefined.

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	DTECn	000	—/W*	DMA Transfer End Interrupt Status Clear Writing 1 to each bit clears the corresponding DMA transfer end interrupt status (n: DMAC channel number). Writing 0 to these bits is ignored. Each bit is always read as 0.

Note: * The bit is writable. The read value is undefined.

- HPB-DMAC Dedicated Register

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	DTEC28	DTEC27	—	DTEC25	DTEC24	DTEC23	DTEC22	DTEC21	DTEC20	DTEC19	DTEC18	DTEC17	DTEC16
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	—/W*	—/W*	R	—/W*	—/W*	—/W*	—/W*	—/W*	—/W*	—/W*	—/W*	—/W*	—/W*

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTEC15	DTEC14	DTEC13	DTEC12	DTEC11	DTEC10	DTEC9	DTEC8	DTEC7	DTEC6	DTEC5	DTEC4	DTEC3	DTEC2	DTEC1	DTEC0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	—/W*	—/W*	—/W*	—/W*	—/W*	—/W*	—/W*	—/W*	—/W*	—/W*	—/W*	—/W*	—/W*	—/W*	—/W*	—/W*

Note: * The bit is writable. The read value is undefined.

Bit	Bit Name	Initial Value	R/W	Description
31 to 29, 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28, 27, 25 to 0	DTECn	All 0	—/W*	DMA Transfer End Interrupt Status Clear Writing 1 to each bit clears the corresponding DMA transfer end interrupt status (n: DMA channel number) Writing 0 to these bits is ignored. Each bit is always read as 0.

Note: * The bit is writable. The read value is undefined.

6A.4.19 DMA Transfer End Interrupt Enable Register (DINTMR)

DINTMR controls output of DMA transfer end interrupts.

- LBSC-DMAC Dedicated Register

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	DTEM2	DTEM1	DTEM0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	DTEMn	000	R/W	DMA Transfer End Interrupt Output Control An interrupt signal is output as a level signal (n: DMA channel number). 0: Does not output an interrupt on completion of a DMA transfer. 1: Outputs an interrupt on completion of a DMA transfer.

- HPB-DMAC Dedicated Register

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	DTEM 28	DTEM 27	—	DTEM 25	DTEM 24	DTEM 23	DTEM 22	DTEM 21	DTEM 20	DTEM 19	DTEM 18	DTEM 17	DTEM 16
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTEM 15	DTEM 14	DTEM 13	DTEM 12	DTEM 11	DTEM 10	DTEM 9	DTEM 8	DTEM 7	DTEM 6	DTEM 5	DTEM 4	DTEM 3	DTEM 2	DTEM 1	DTEM 0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29, 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28, 27, 25 to 0	DTEMn	All 0	R/W	DMA Transfer End Interrupt Output Control An interrupt signal is output as a level signal (n: DMA channel number). 0: Does not output an interrupt on completion of a DMA transfer. 1: Outputs an interrupt on completion of a DMA transfer.

6A.4.20 DMA Activation Status Register (DACTSR)

DACTSR indicates the activation status of each channel.

- LBSC-DMAC Dedicated Register

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	DS2	DS1	DS0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	DSn	000	R	DMA Channel n Status (n: DMA channel number) 0: Idle state 1: Active state

- HPB-DMAC Dedicated Register

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	DS28	DS27	—	DS25	DS24	DS23	DS22	DS21	DS20	DS19	DS18	DS17	DS16
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DS15	DS14	DS13	DS12	DS11	DS10	DS9	DS8	DS7	DS6	DS5	DS4	DS3	DS2	DS1	DS0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29, 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28, 27, 25 to 0	DSn	All 0	R	DMA Channel n Status (n: DMA channel number) 0: Idle state 1: Active state

6A.4.21 Software-Reset Registers (LSRSTR0 to LSRSTR2, HSRSTR0 to HSRSTR28)

Each register resets DMA channel n.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SRST
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/WC1*

Note: * The bit is readable and writable. Writing 1 to the bit initializes the bit. Writing 0 to the bit is ignored.

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SRST	0	R/WC1*	Software Reset Resets DMA channel n. 0: Writing 0 is ignored. 1: Resets DMA channel n. This register is always read as 0.

Note: Writing 1 resets the DMAC regardless of the DMA transfer status. The same registers are reset as when a power-on reset or manual reset is performed. Accordingly, a software reset should be used only while DMA transfer is not in progress (e.g., during system debugging). To stop operation, forced termination or temporary stop should be specified instead of software reset. In the registers used by all channels in common (DMA transfer end interrupt status register (DINTSR) and DMA transfer end interrupt enable register (DINTMR)), only the bits corresponding to the software-reset channel are initialized.

* The bit is readable and writable. Writing 1 to the bit initializes the bit. Writing 0 to the bit is ignored.

6A.4.22 External DMA Data Alignment Control Register (DMALGR)

DMALGR specifies whether there is a data alignment conversion in external bus access by the external bus DMAC. (LBSC-DMAC dedicated register)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	DMLG2				DMLG1				DMLG0			
					exbwe	exac	exbw		exbwe	exac	exbw		exbwe	exac	exbw	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	DMLGn exbwe	0	R/W	Specifies whether there is an EX-BUS data alignment conversion for each DMAC channel. (n: DMAC channel number) When the DMAC/DCR setting DMA access bus width is smaller than the external bus width for the relevant DMA area set in LBSC, 0: Access byte lanes fixed 1: Access byte lanes variable
	DMLGn exac	0	R/W	Specifies endian setting when there is alignment conversion with the exbwe bit set to 1. (n: DMAC channel number) 0: Big endian 1: Little endian
	DMLGn exbw	00	R/W	Specifies the unit for data alignment conversion with the exbwe bit set to 1. (n: DMAC channel number) 00: 8 bits 01: 16 bits 10: Setting prohibited 11: Invalid

Note: Does not need to be set when the bus width set in LBSC and the DMA access size set in DMAC/DCR are the same.
For details, see section 6B.6.3 (1), Data Alignment during LBSC-DMAC Access in section 6B, LBSC within Bus Bridge.

6A.4.23 LBSC-DMA SHwy Priority Control Register (LBSC-DMASPR)

LBSC-DMASPR specifies the SHwy bus access priority level for HPB-DMAC. (LBSC-DMAC dedicated register)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	SPRR2				SPRR1				SPRR0			
Initial value:	0	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	SPRRn	H'8	R/W	Specifies the SHwy bus access priority level for each DMAC channel (n: DMAC channel number). Priority: H'0 (lowest) to H'F (highest)

Note: Since SHwy bus access priority level settings relate to priority control for the SHwy bus overall, priority levels with other access modules must be confirmed when making settings.

6A.4.24 HPB-DMA SHwy Priority Control Register 0 (HPB-DMASPR0)

HPB-DMASPR0 specifies the SHwy bus access priority level for HPB-DMAC. (HPB-DMAC dedicated register)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SPRR7				SPRR6				SPRR5				SPRR4			
Initial value:	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPRR3				SPRR2				SPRR1				SPRR0			
Initial value:	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SPRRn	H'8	R/W	Specifies the SHwy bus access priority level for each DMAC channel (n: DMA channel number). Priority: 0 (lowest) to F (highest)

Notes: 1. Since SHwy bus access priority level settings relate to priority control for the SHwy bus overall, priority levels with other access modules must be confirmed when making settings.

2. The SPRRn bits should be set to H'8 or H'9 in this product.

6A.4.25 HPB-DMA SHwy Priority Control Register 1 (HPB-DMASPR1)

HPB-DMASPR1 specifies the SHwy bus access priority level for HPB-DMAC. (HPB-DMAC dedicated register)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SPRR15				SPRR14				SPRR13				SPRR12			
Initial value:	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPRR11				SPRR10				SPRR9				SPRR8			
Initial value:	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SPRRn	H'8	R/W	Specifies the SHwy bus access priority level for each DMAC channel (n: DMA channel number). Priority: 0 (lowest) to F (highest)

Note: See the table note in section 6A.4.24, HPB-DMA SHwy Priority Control Register 0 (HPB-DMASPR0).

6A.4.26 HPB-DMA SHwy Priority Control Register 2 (HPB-DMASPR2)

HPB-DMASPR2 specifies the SHwy bus access priority level for HPB-DMAC. (HPB-DMAC dedicated register)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SPRR23				SPRR22				SPRR21				SPRR20			
Initial value:	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPRR19				SPRR18				SPRR17				SPRR16			
Initial value:	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SPRRn	H'8	R/W	Specifies the SHwy bus access priority level for each DMAC channel (n: DMA channel number). Priority: 0 (lowest) to F (highest)

Note: See the table note in section 6A.4.24, HPB-DMA SHwy Priority Control Register 0 (HPB-DMASPR0).

6A.4.27 HPB-DMA SHwy Priority Control Register 3 (HPB-DMASPR3)

HPB-DMASPR3 specifies the SHwy bus access priority level for HPB-DMAC. (HPB-DMAC dedicated register)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	SPRR28			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPRR27				—	—	—	—	SPRR25				SPRR24			
Initial value:	1	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 20, 11 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
19 to 12, 7 to 0	SPRRn	H'8	R/W	Specifies the DMAC bus access priority level for each DMAC channel (n: DMAC channel number). Priority: H'0 (lowest) to H'F (highest)

Note: See the table note in section 6A.4.24, HPB-DMA SHwy Priority Control Register 0 (HPB-DMASPR0).

6A.4.28 HPB-DMA Access Priority Level Control Register (HPB-DMLVLR)

HPB-DMLVLR specifies the DMAC bus access priority level (1 or 2) for HPB-DMAC. (HPB-DMAC dedicated register)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	DMLV28	DMLV27	—	DMLV25	DMLV24	DMLV23	DMLV22	DMLV21	DMLV20	DMLV19	DMLV18	DMLV17	DMLV16
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMLV15	DMLV14	DMLV13	DMLV12	DMLV11	DMLV10	DMLV9	DMLV8	DMLV7	DMLV6	DMLV5	DMLV4	DMLV3	DMLV2	DMLV1	DMLV0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29, 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28, 27, 25 to 0	DMLVn	All 0	R/W	Specifies the SHwy bus access priority level for each DMAC channel (n: DMAC channel number). 0: Level 2 (low: group 2) 1: Level 1 (high: group 1)

Note: For the HPB-DMC access contention control, see section 6A.5.11, SuperHyway Bus and HPB Bus Access Priority Control by the HPB-DMAC.

6A.4.29 UltraATA DMA Mode Register (UATMR)

UATMR enables the UltraATA mode and specifies data alignment in UltraATA mode. (LBSC-DMAC dedicated register)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	UTDR		—	—	UTWE	UTRE	—	UTSL		UATM
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10, 7, 6, 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	UTDR	00	R/W	These bits select the external pin for the DREQ signal in the UltraATA interface. 00: No external pin is used. 01: The DREQ0 pin in the EX-BUS is used as the DREQ (DMARQ) pin. 10: The DREQ1 pin in the EX-BUS is used as the DREQ (DMARQ) pin. 11: No external pin is used.
5	UTWE	0	R/W	Enables or disables data alignment conversion for write operation in UltraATA mode. 0: Disables data alignment conversion for DMA write operation. 1: Enables data alignment conversion for DMA write operation. (Data alignment is in 2-byte units for every 4 bytes of data.)

Bit	Bit Name	Initial Value	R/W	Description
4	UTRE	0	R/W	Enables or disables data alignment conversion for read operation in UltraATA mode. 0: Disables data alignment conversion for DMA read operation. 1: Enables data alignment conversion for DMA read operation. (Data alignment is in 2-byte units for every 4 bytes of data.)
2, 1	UTSL	00	R/W	These bits select the external pin for the IORDY (DDMARDY/DSTROBE) signal in the UltraATA interface. 00: No external pin is used. 01: The EX_WAIT0 pin in the EX-BUS is used as the IORDY (DDMARDY/DSTROBE) pin. 10: The EX_WAIT1 pin in the EX-BUS is used as the IORDY (DDMARDY/DSTROBE) pin. 11: The EX_WAIT2 pin in the EX-BUS is used as the IORDY (DDMARDY/DSTROBE) pin.
0	UATM	0	R/W	Specifies the UltraATA operating mode. 0: Normal DMA mode 1: UltraATA DMA mode

Notes: 1. For the UltraATA DMA operation, refer to section 6B, LBSC within Bus Bridge.

2. When the UTSL bits (external pin selection for the IORDY (DDMARDY/DSTROBE) signal in the UltraATA interface) are set to 00 (initial value: no external pin is used), the IORDY (DDMARDY/DSTROBE) signal in the UltraATA interface is set to an internally-fixed value and does not work correctly; be sure to select one of EX_WAIT0 to EX_WAIT2 in the EX-BUS.
3. When the UTDR bits (external pin selection for the DREQ signal in the UltraATA interface) are set to 00 or 11 (no external pin is used), the DREQ signal in the UltraATA interface is set to an internally-fixed value and does not work correctly; be sure to select DREQ0 or DREQ1 in the EX-BUS.

6A.4.30 UltraATA Write Cycle Setting Register (UATWCR)

UARWCR specifies the setup and hold clock cycles of the write data relative to the HSTROBE signal in UltraATA DMA operation. (LBSC-DMAC dedicated register)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	UATWCYC		
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	UATWCYC	H'0	R/W	These bits specify the setup and hold clock cycles of the write data relative to the HSTROBE (DIOR) signal in UltraATA DMA operation. H'0: Setup = 1 bus clock cycle, Hold = 1 bus clock cycle H'1: Setup = 2 bus clock cycle, Hold = 1 bus clock cycle H'2: Setup = 2 bus clock cycle, Hold = 2 bus clock cycle H'3: Setup = 3 bus clock cycle, Hold = 2 bus clock cycle H'4: Setup = 3 bus clock cycle, Hold = 3 bus clock cycle H'5: Setup = 4 bus clock cycle, Hold = 3 bus clock cycle H'6: Setup = 4 bus clock cycle, Hold = 4 bus clock cycle H'7: Setup = 5 bus clock cycle, Hold = 4 bus clock cycle

Note: For the UltraATA DMA operation, refer to section 6B, LBSC within Bus Bridge.

6A.4.31 UltraATA Timeout Period Setting Register (UATTSR)

UATTSR specifies the period until timeout when communication temporarily stops in UltraATA DMA read operation. (LBSC-DMAC dedicated register)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	UATOVFT															
Initial Value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	UATOVFT															
Initial Value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	UATOVFT	H'FFFF_FFFF	R/W	These bits specify the period until HOST timeout when communication temporarily stops (no change in DSTROBE) in UltraATA DMA read operation. Timeout period (ns) = Set value converted to decimal × EX-BUS clock period (ns).

Note: For the UltraATA DMA operation, refer to section 6B, LBSC within Bus Bridge.

6A.4.32 UltraATA Error Indication Register (UATTER)

UATTER indicates the error status in UltraATA DMA operation. (LBSC-DMAC dedicated register)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PER	DER
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	PER	0	R/W	Indicates whether PIO access is executed for the area allocated to the ATA space during UltraATA DMA operation. (The PIO access attempted for the area allocated to the ATA space during UltraATA DMA operation is ignored.) 0: No PIO access has been executed for the area allocated to the ATA space during UltraATA DMA operation. 1: PIO access has been executed for the area allocated to the ATA space during UltraATA DMA operation.
0	DER	0	R/W	Indicates whether timeout occurs due to a temporary communication stop (no change in DSTROBE) during UltraATA DMA read operation. The timeout period is specified through UATTSR and UATMR2. (When reading) 0: No timeout error has occurred. 1: A timeout error has occurred (interrupt is generated when enabled through UATIER). (When writing) No timeout detection.

Note: For the UltraATA DMA operation, refer to section 6B, LBSC within Bus Bridge.

6A.4.33 UltraATA Error Interrupt Enable Register (UATIER)

UATIER enables interrupts when errors occur in UltraATA DMA operation. (LBSC-DMAC dedicated register)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PERE	DERE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	PERE	0	R/W	Specifies whether to output the interrupt notification signal while the PER bit in UATTER is set to 1. 0: The interrupt notification signal is not output. 1: The interrupt notification signal is output.
0	DERE	0	R/W	Specifies whether to output an interrupt notification signal while the DER bit in UATTER is set to 1. 0: The interrupt notification signal is not output. 1: The interrupt notification signal is output.

Note: For the UltraATA DMA operation, refer to section 6B, LBSC within Bus Bridge.

6A.4.34 UltraATA CRC Code Indication Register (UATCRCR)

UATCRCR indicates the CRC code of the communication data in UltraATA DMA operation. (LBSC-DMAC dedicated register)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRC															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	CRC	H'0000	R	These bits indicate the CRC code created from the transfer data in UltraATA DMA operation.

Note: For the UltraATA DMA operation, refer to section 6B, LBSC within Bus Bridge.

6A.4.35 UltraATA DMA Mode Register 2 (UATMR2)

UATMR2 enables the UltraATA mode (LBSC-DMAC dedicated register).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	UATM2															
Initial Value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	UATM2															
Initial Value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	UATM2	H'FFFF_FFFF	R/W	Specifies operation in UltraATA mode. Set these bits to H'0000 0006 when the UltraATA interface is in use. Operation is not guaranteed if a value other than H'0000 0006 is written here.

Note: For details, refer to figure 6B.20, Setting Procedure for the UltraATA DMA Transfer in section 6B, LBSC within Bus Bridge.

6A.4.36 UltraATA DMA Mode Register 3 (UATMR3)

UATMR3 enables the UltraATA mode (LBSC-DMAC dedicated register).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	UATM3			
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 4	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	UATM3	0000	R/W	Specifies operation in UltraATA mode. Set these bits to H'C when the UltraATA interface is in use. Operation is not guaranteed if a value other than H'C is written here.

Note: For details, refer to figure 6B.20, Setting Procedure for the UltraATA DMA Transfer in section 6B, LBSC within Bus Bridge.

6A.4.37 UltraATA Transfer Mode Register (UATTMR)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	DTCD	—	—	—	—	—	—	—	DBG0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 9	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
8	DTCD	0	R/W	Controls the operating mode for continuation in case of transfer termination operations during DMA operations for UltraATA. Some existing ATA devices handle the termination of transfer in the same way as a pause. If the DMA transfer count has not been reached, such devices should not abnormally end the transfer even if they acknowledge the termination request, but should restart the transfer upon reception of the next DREQ (DMARQ) signal from the device. This operation is called “transfer termination continuation mode”. 1: Transfer termination continuation mode is disabled. 0: Transfer termination continuation mode is enabled.
7 to 1	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
0	DBG0	0	R/W	Test Bit Writing 1 to this bit is prohibited since this is a test bit. This bit is always read as 0. The write value should always be 0.

Note: For the UltraATA DMA operation, refer to section 6B, LBSC within Bus Bridge.

6A.5 Operation

6A.5.1 DMA Transfer Procedure

The following describes the DMA transfer procedure.

1. Making the initial setting for the DMAC

   Select the peripheral for DMA transfer in the DMA port select register (DPTR) (only for the HPB-DMAC).

   Specify transfer conditions in the DMA control register (DCR).
2. Specifying DMA transfer information

   Make appropriate settings in the following registers according to the DIP bit setting (input mode: using one information set repeatedly or two information sets alternately) in DCR:

   DMA source address register (DSAR)

   DMA destination address register (DDAR)

   DMA transfer count register (DTCR)

   When the DIP bit specifies that one information set is used repeatedly, make the appropriate settings in DSAR0, DDAR0, and DTCR0.
3. Activating the DMAC

   Activate the DMAC by setting the DMEN bit in the DMA command register (DCMDR).
4. Reading the specified DMA transfer information

   The specified DMA transfer information is read from DMA information registers 0 and 1 in that order.
5. Clearing the DMA transfer request

   The DMA transfer request status signal is cleared.
6. Starting DMA transfer

   When the auto-request mode is selected as the transfer request mode, transfer automatically starts at the DMAC transfer timing after the transfer information is obtained.

   When the external request or peripheral request mode is selected, DMA transfer is performed for one bus access cycle when a transfer request is accepted.

   When the timer request is selected, transfer automatically starts at the intervals specified in the DMAC internal timer after the transfer information is obtained.
7. Issuing an interrupt for the end of a specified number of transfers

   In single transfer mode, DMA transfer stops when transfer is completed for the specified number of times, and the CPU is notified of the end of transfer through an interrupt.

In continuous transfer mode, the CPU is notified of the end of transfer in DMA transfer information units through an interrupt.

The interrupt signal is controlled according to the setting in the DMA transfer end interrupt enable register (DINTMR).

8. Reading the next DMA transfer information (continuous transfer mode)

   If the next DMA transfer request is specified, the information of the transfer is read and data is transferred in the same way as described in step 6.

   If no additional DMA transfer request is specified (DRSTS = 0), the continuous DMA transfer mode is terminated when DQSTS = 1, or the next DMA transfer request is waited for when DQSTS = 0.

9. Adding DMA transfer information (continuous transfer mode)

   If new DMA transfer information should be added, specify the information in the DMA transfer information set that will be used for the next transfer (the next DMA transfer information set can be checked with the NDP1 and NDP0 bits in the DMA status register (DSTSR)).

   If no DMA transfer information should be added, write 1 to the DQEND bit in the DMA command register (DCMDR) to terminate the continuous transfer mode.

Note: Actually, the source bus and destination bus operate independently.

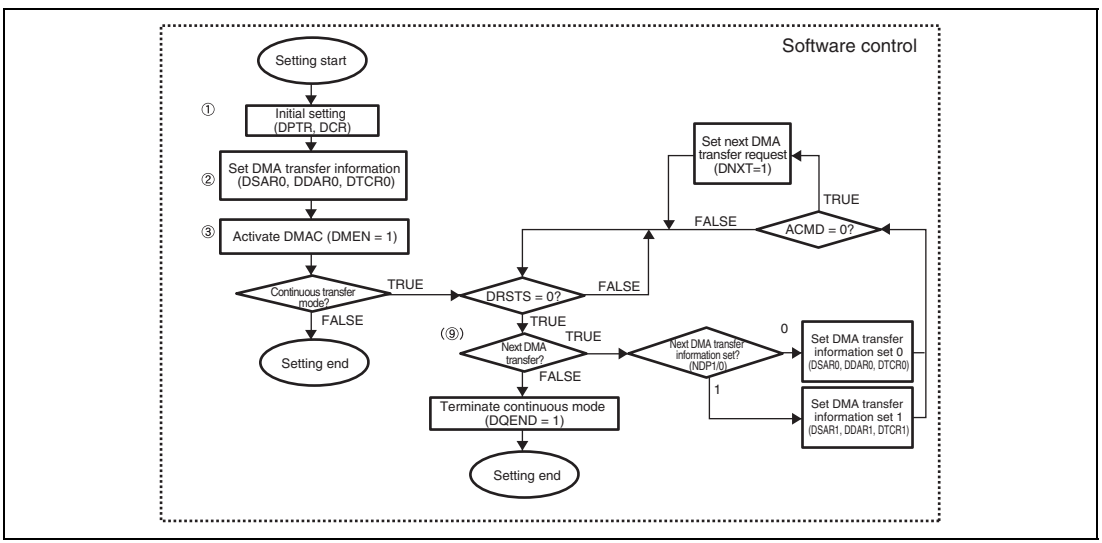


Figure 6A.2 DMA Transfer Flowchart (1)

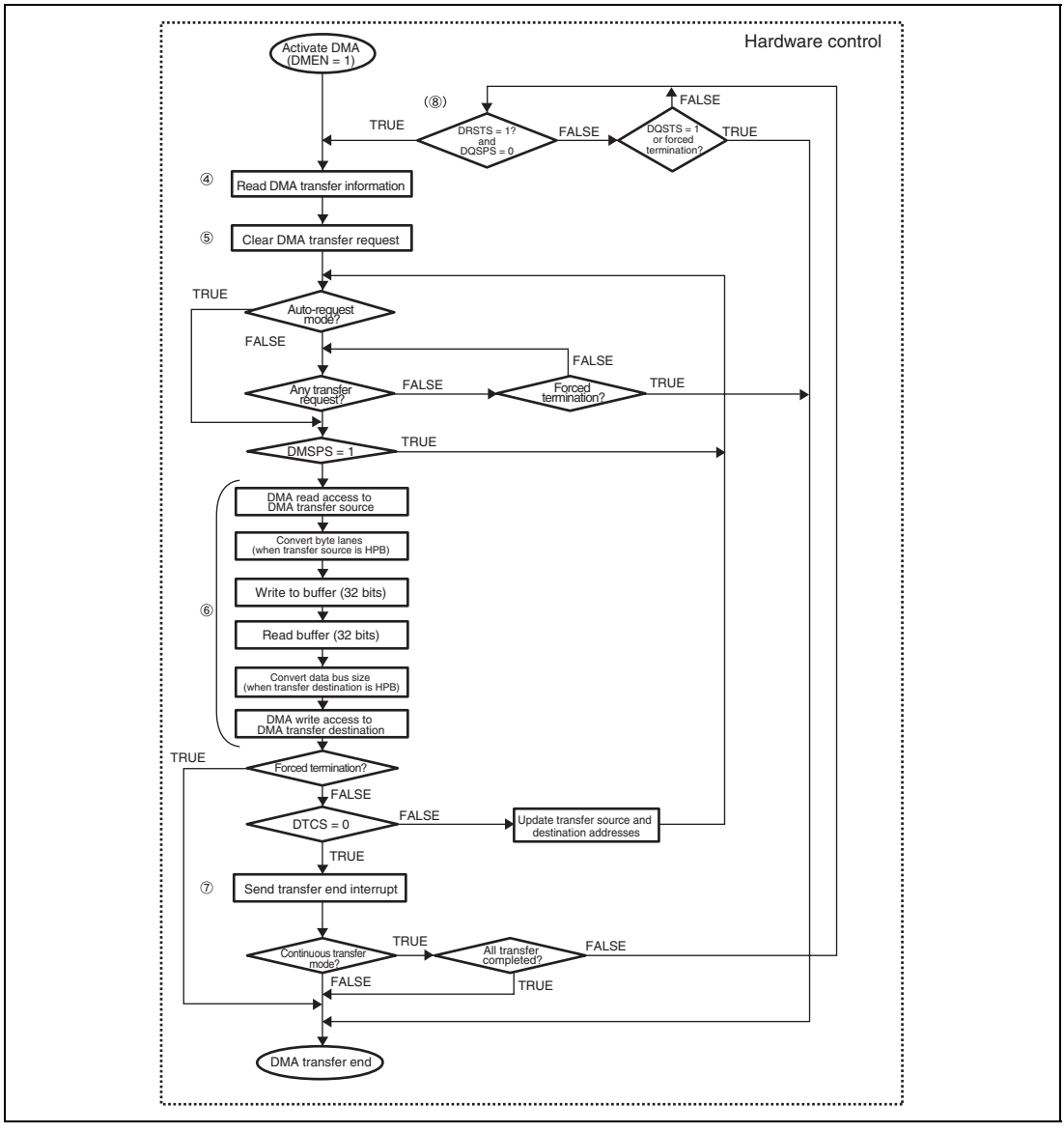


Figure 6A.2 DMA Transfer Flowchart (2)

6A.5.2 Continuous DMA Transfer Operation

The following shows the relationship between addition of DMA transfer information by software and DMA transfer information read and data transfer operation by hardware, which are described in step 9 in section 6A.5.1, DMA Transfer Procedure. Figures 6A.3 and 6A.4 show transfer using DMA information set 0 repeatedly, and figures 6A.5 and 6A.6 show transfer using DMA information sets 0 and 1 alternately.

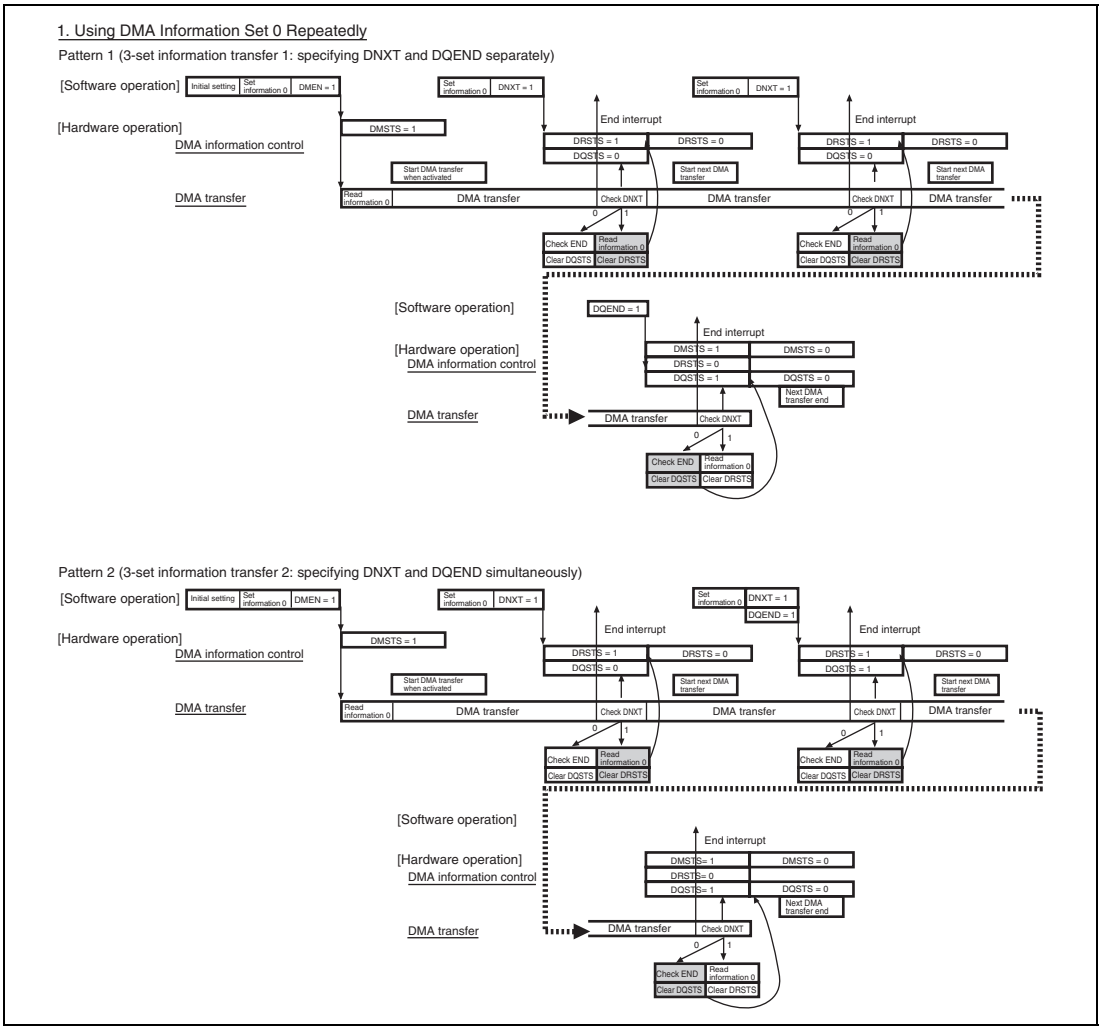


Figure 6A.3 Transfer Using DMA Information Set 0 Repeatedly (1)

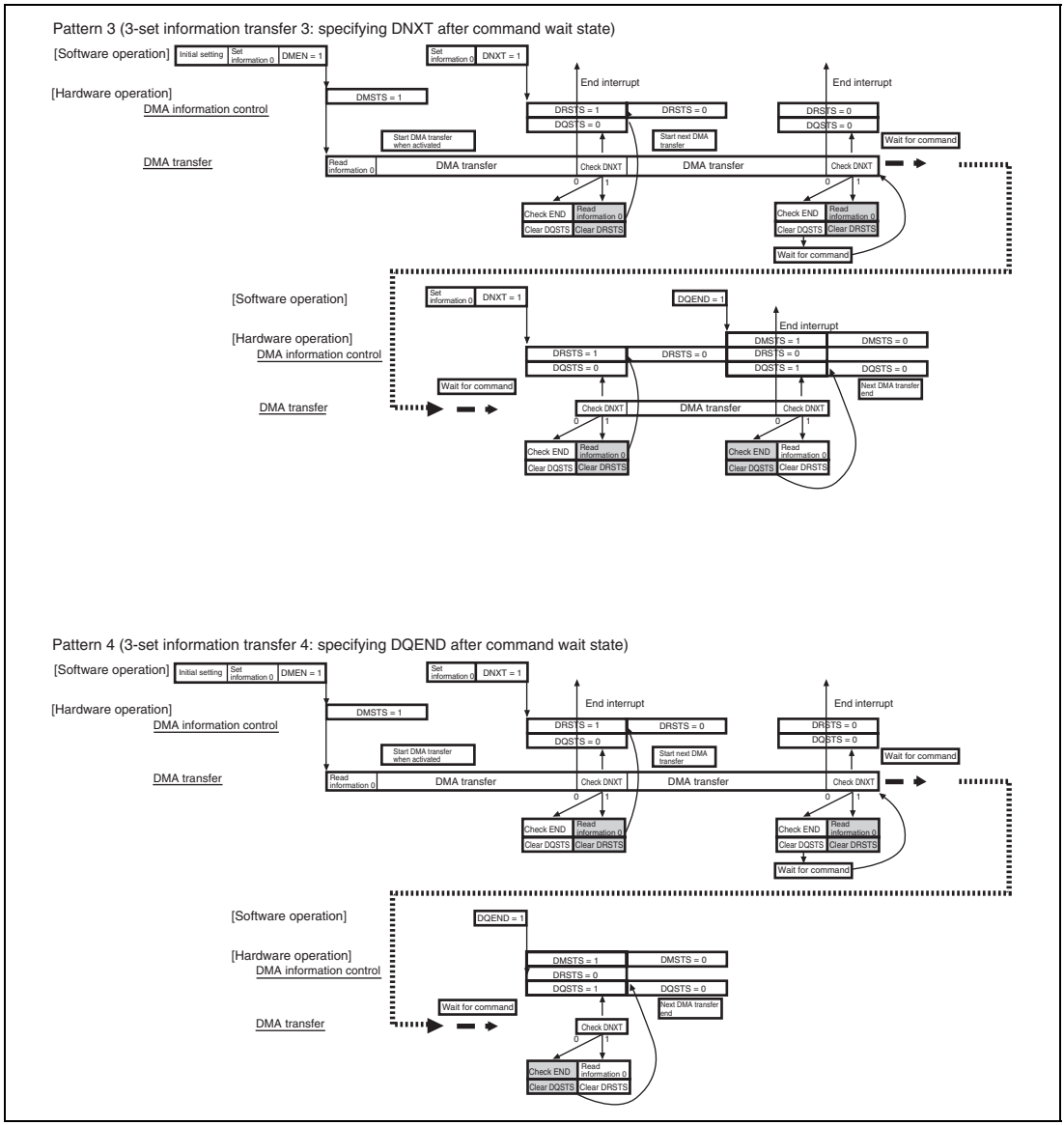


Figure 6A.4 Using DMA Information Set 0 Repeatedly (2)

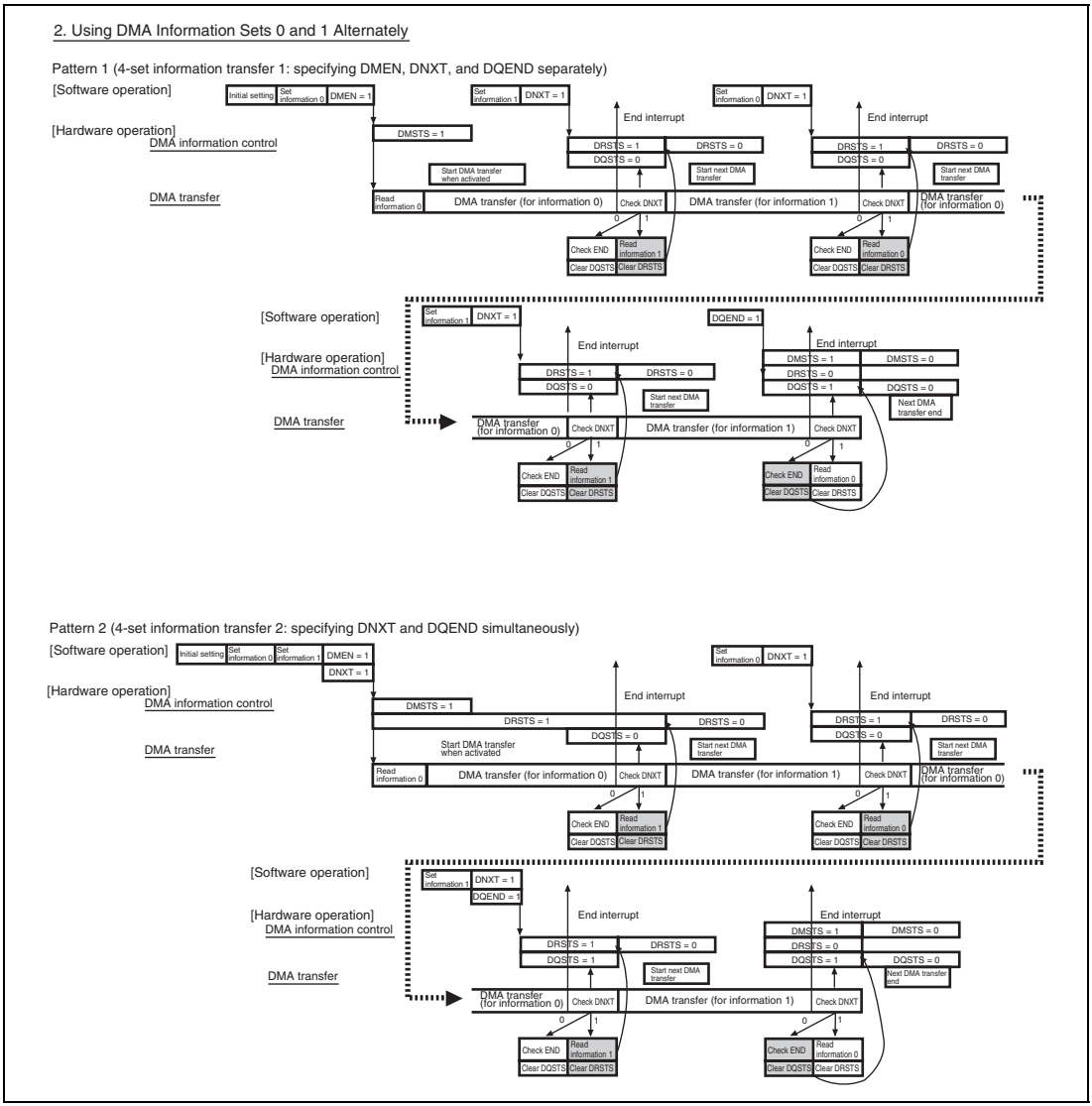


Figure 6A.5 Using DMA Information Sets 0 and 1 Alternately (1)

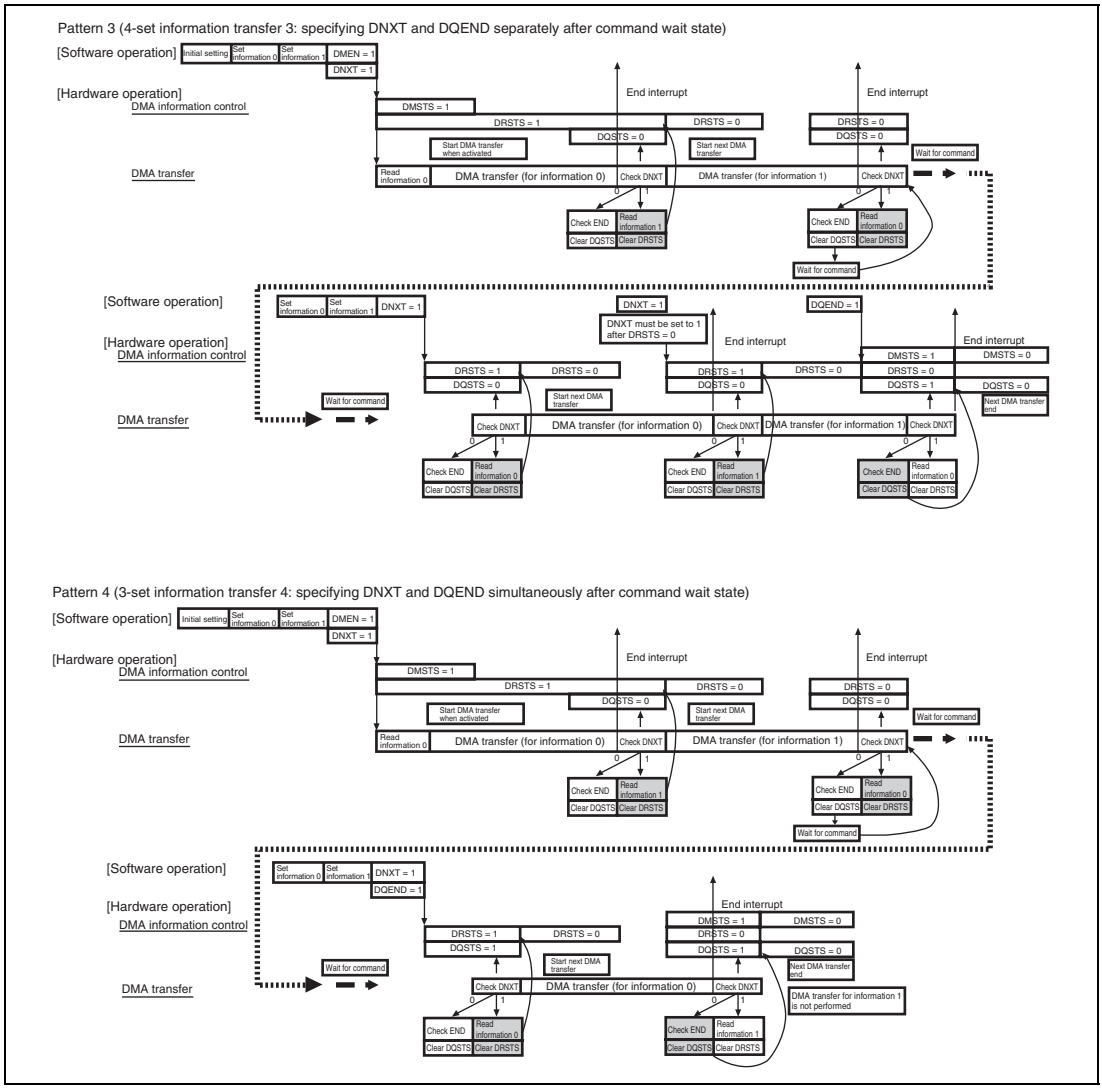


Figure 6A.6 Using DMA Information Sets 0 and 1 Alternately (2)

6A.5.3 Packing Data Read from Peripheral or External Module

Writing 1 to the PKMD bit when making necessary settings in the DMA control register (DCR) in the usual DMA activation procedure enables packing of data read from a peripheral or external module and then writing the data to memory (the SuperHyway side). Packing size can be specified as either 4 bytes or the number of bytes indicated under the note (*) for the DCR description according to the value of the SWMD bit in the DCR. However, when the destination for transfer on the SuperHyway side is memory (DDR2-SDRAM/DDR3-SDRAM), we recommend the number of bytes indicated in the note (*) for the DCR register description as the packing size, in order to use memory and the SuperHyway efficiently.
When the destination for transfer on the SuperHyway side by the LBSC-DMAC is a peripheral module attached to the HPB, since no more than four bytes will be written to a register of such a module, packing specification of the access size when SWMD in DCR = 0 is not available (for details on access sizes for the individual channels, see the note (*) for the DCR register description). On the other hand, the HPB-DMAC can transfer data in the units specified with SWMD in DCR = 0 when destination for transfer on the SuperHyway side is on the external bus and the destination device is SRAM etc.

During the packing operation, even if the DMAC holds data less than the specified packing size when the DMAC completes the specified count of transfer, the DMAC writes fetched data to memory and indicates the end of DMA transfer by setting the DTEn bit in the DMA transfer end interrupt status register (DINTSR). If the peripheral or external module completes a DMA request before the specified transfer count is reached, DMA transfer can be terminated by writing 1 to the BDOUT bit in the DMA command register (DCMDR). In this case, if data which is being packed remains in the DMAC, the DMAC writes the data to memory (SuperHyway side). The DTE[n] bit in the DMA transfer end interrupt status register (DINTSR) is set and a transfer end interrupt is generated. Note that zero padding does not occur during write to memory. If no data remains in the DMAC, the DMAC terminates DMA transfer without accessing memory and generates a transfer end interrupt in the same way as when data remains in the DMAC.

When transfer is terminated by the BDOUT bit setting in continuous transfer mode, the DMAC transfers the next DMA information if the next DMA information transfer is requested through the DNXT bit in DCMDR, and then terminates DMA transfer in the continuous transfer mode termination procedure. In this mode, DMA requests (dreq) from peripheral or external modules are masked (not accepted) until a transfer end interrupt occurs after the BDOUT bit is set to 1.

6A.5.4 Limitations on Packing of Data Read from Peripheral or External Module

- The transfer count is specified in DMA transfer count registers 0 and 1 (DTCR0 and DTCR1) in the DMAC.
- If DMA transfer from a peripheral or external module is completed before the specified transfer count is reached, the DMAC cannot distinguish whether data is being transferred or the transfer has been completed, and data of less than packing size may remain in the DMAC internal buffer. When the size of remaining data is the same as the specified packing size, the data is transferred to memory.
- Data remaining in the DMAC internal buffer is written to memory through a forced write executed by setting the BDOUT bit in the DMA command register (DCMDR). (Zero padding does not occur because the transfer destination is memory. For example, if 3-byte data remains in the DMAC un-transferred to memory, the 3-byte data is written to the memory as is.)
- A forced write can be triggered by a transfer-completed interrupt from an on-chip or external peripheral module. However, whether the transfer-completed interrupt actually indicates the completion of DMA transfer from the module depends on the specifications of the module. Accordingly, check the specifications of the module before executing a forced write.

6A.5.5 Notification of the End of DMA Transfer

The DMAC notifies the CPU via the INTC2 of the end of transfer through transfer end interrupt signal (a level signal) when transfer is completed for the transfer count specified in DMA transfer information in single transfer mode. In continuous transfer mode, the DMAC outputs transfer end interrupt signal every time transfer is completed for the transfer count specified in one DMA transfer information set.

The transfer end interrupt signal is controlled according to the setting in the DMA transfer end interrupt enable register (DINTMR). Writing 1 to the DMA transfer end interrupt status clear register clears the transfer end interrupt signal.

6A.5.6 DMA Transfer Stop, and Resume Procedures

This section describes the procedures for stopping and resuming DMA transfer.

- To Stop (Cancel) DMA Transfer during Operation

Step	Overview	Register Operation	Operation after Register Write
1 ↓	Specify forced termination for DMAC.	Write 1 to DMSTP of DSTPR in DMAC.	The DMAC stops DMA transfer as soon as the current DMA bus cycle is completed, and then enters the idle state. Unfinished transfer data remaining in the buffer is discarded. The registers retain the values. No end interrupt is issued.
2 ↓	Check that the DMAC has entered the idle state.	If DMSTS of DSTSR in DMAC is 0, the DMAC is in the idle state.	—
3	Specify forced termination for external devices.	(Depends on the external devices.)	The external devices stop sending DMA requests.

Note: Step 2 can be done after step 3.

- To Temporarily Stop (Pause) DMA Transfer during Operation and Then Resume It

Step	Overview	Register Operation	Operation after Register Write
1 ↓	Specify temporary stop for DMAC.	Write 1 to DMSPD of DCMDR in DMAC.	The DMAC temporarily stops DMA transfer as soon as the current DMA bus cycle is completed. The DMAC retains its internal status, including unfinished transfer data remaining in the buffer, without change.
2 ↓	Check that the DMAC has entered the temporary stop state.	If DMSPS of DSTSR in DMAC is 1, the DMAC is in the suspended state.	—
3 ↓	A certain period of time has elapsed (the LBSC may detect a DREQ signal from an external device during this period, but the DMAC keeps the temporary stop and the external devices continue to wait for DMA transfer).		
4	Specify cancel of temporary stop (resume transfer).	Write 1 to DMSPC of DCMDR in DMAC.	The paused state is canceled, and the DMAC resumes its operation with the DMA transfer for the DREQ detected by the LBSC.

Note: Step 2 can be done between steps 3 and 4.

- To Temporarily Stop (Pause) DMA Transfer during Operation and Then Terminate (Cancel) Operation

Step	Overview	Register Operation	Operation after Register Write
1 ↓	Specify temporary stop for DMAC.	Write 1 to DMSPD of DCMDR in DMAC.	The DMAC temporarily stops DMA transfer as soon as the current DMA bus cycle is completed. The DMAC retains its internal status, including unfinished transfer data remaining in the buffer, without change.
2 ↓	Check that the DMAC has entered the temporary stop state.	If DMSPS of DSTSR in DMAC is 1, the DMAC is in the suspended state.	—
3 ↓	A certain period of time has elapsed (the LBSC may detect a DREQ signal from an external device during this period, but the DMAC keeps the temporary stop and the external devices continue to wait for DMA transfer).		
4 ↓	Specify forced termination for DMAC.	Write 1 to DMSTP of DSTPR in DMAC.	The DMAC exits the temporary stop state and enters the idle state. Unfinished transfer data remaining in the buffer is discarded. The registers retain the values. No end interrupt is issued.
5 ↓	Check that the DMAC has entered the idle state.	If DMSTS of DSTSR in DMAC is 0, the DMAC is in the idle state.	—
6	Specify forced termination for external devices.	(Depends on the external devices.)	The external devices stop sending DMA requests.

Notes: 1. Step 2 can be done between steps 3 and 4.
2. Step 5 can be done after step 6.

6A.5.7 Data Alignment in SuperHyway Bus Interface

The SuperHyway bus is always accessed through a handshake using an access request and a request acknowledge. Alignment of data read or written during the access to memory through the SuperHyway bus is always converted.

When the DTAMD bit in the DMA control register (DCR) is 0, the SuperHyway bus data alignment is converted according to the endian mode signal (DMAC input signal: little) and the peripheral data bus width (the SPDS1 and SPDS0 bits or DPDS1 and DPDS0 bits in DCR). When the DTAMD bit in DCR is 1, data alignment is converted according to the DTAC, DTAU, and DTAU1 bit settings in DCR.

The following table shows the SuperHyway bus data alignment control according to the DTAMD bit in DCR, endian mode signal (DMAC input signal: little), peripheral data bus width (the SPDS1 and SPDS0 bits or DPDS1 and DPDS0 bits in DCR), and DTAC, DTAU, and DTAU1 bits in DCR.

Data Alignment Control According to Bit and Signal Settings

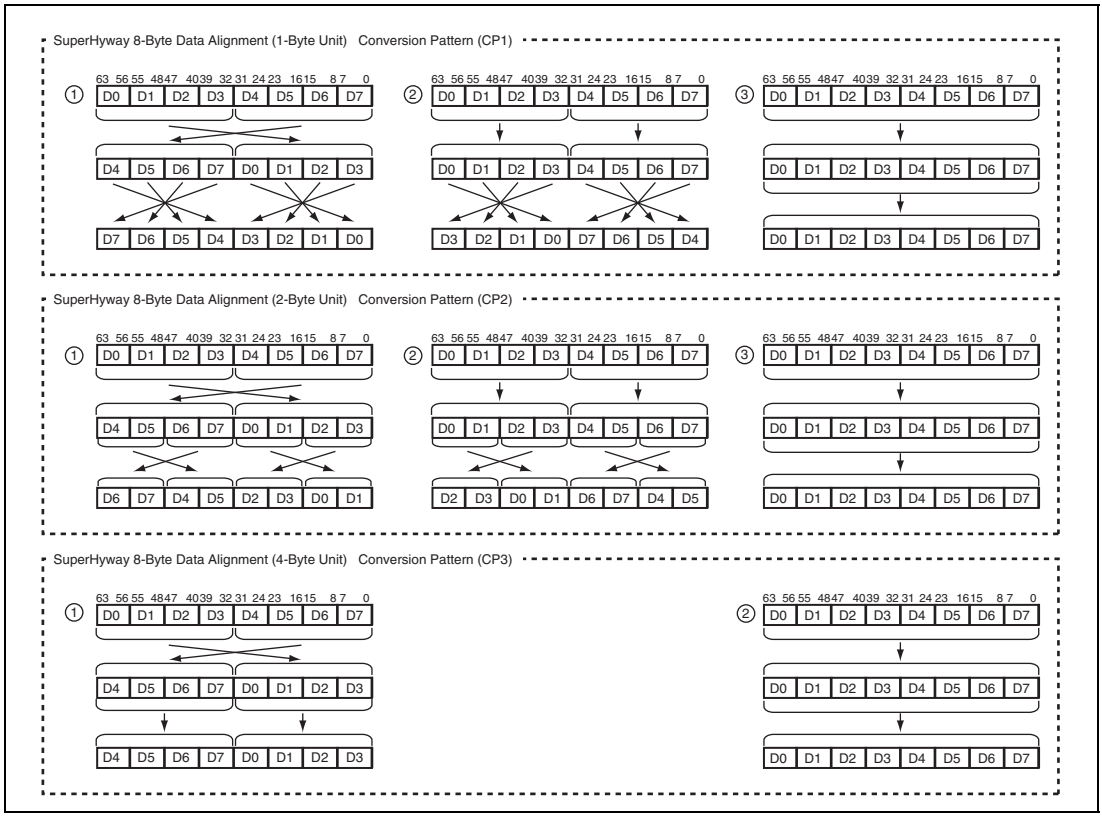
No.	DTAMD	little (MD[8])	PDS [1:0]	DTAC	DTAU	DTAU1	Data Alignment of 4 Bytes	Unit for 4-Byte Data Alignment	8-Byte Data Alignment in 4-Byte Unit	Conversion Pattern	Remarks
1	0	0	00 (8 bits)	*	*	*	Not controlled	8 bits	Not controlled	CP1 (3)	Standard conversion (Software must specify only PDS[1:0])
2	0	0	01 (16 bits)	*	*	*	Not controlled	16 bits	Not controlled	CP2 (3)	
3	0	0	10 (32 bits)	*	*	*	Not controlled	8 bits	Not controlled	CP3 (2)	
4	0	1	00 (8 bits)	*	*	*	Controlled	8 bits	Controlled	CP1 (1)	
5	0	1	01 (16 bits)	*	*	*	Controlled	16 bits	Controlled	CP2 (1)	
6	0	1	10 (32 bits)	*	*	*	Not controlled	8 bits	Controlled	CP3 (1)	

No.	DTAMD	little (MD[8])	PDS [1:0]	DTAC	DTAU	DTAU1	Data Alignment of 4 Bytes	Unit for 4-Byte Data Alignment	8-Byte Data Alignment in 4-Byte Unit	Conversion Pattern	Remarks
7	1	*	*	0	0	0	Not controlled	8 bits	Not controlled	CP1 (3)	Special conversion (Software must specify the alignment mode)
8	1	*	*	0	0	1	Not controlled	8 bits	Controlled	CP3 (1)	
9	1	*	*	0	1	0	Not controlled	16 bits	Not controlled	CP2 (3)	
10	1	*	*	0	1	1	Not controlled	16 bits	Controlled	CP3 (1)	
11	1	*	*	1	0	0	Controlled	8 bits	Not controlled	CP1 (2)	
12	1	*	*	1	0	1	Controlled	8 bits	Controlled	CP1 (1)	
13	1	*	*	1	1	0	Controlled	16 bits	Not controlled	CP2 (2)	
14	1	*	*	1	1	1	Controlled	16 bits	Controlled	CP2 (1)	

[Legend]

*: Don't care

The following shows data alignment conversion in the DMAC. Conversion pattern numbers in the table above correspond to the conversion numbers below.



6A.5.8 Data Alignment in HPB Bus Interface

The HPB bus is accessed in the data bus width specified in the SPDS or DPDS bit in the DMA control register (DCR), and big endian is always assumed.

6A.5.9 Data Alignment in EX-BUS Interface

Although data alignment mode for the external bus access follows by default the same approach as that for the HPB bus, setting the DMALGR register in the DMAC allows variable alignment mode for the external bus access as the mode of alignment conversion. For the operation in a variable alignment mode, see section 6B.6.3 (1), Data Alignment during LBSC-DMAC Access, in section 6B, LBSC within Bus Bridge.

6A.5.10 Timing Charts

Timing charts are given for DMA operation during the external SRAM bus operation. In the timing charts, the DREQ and DACK signal polarity is indicated as negative; however, these polarity settings, assertion of either the $\overline{CS}$ or DACK signal, and switching DREQ to a level signal or to edge detection mode, can be set. In all cases these settings should be made using LBSC internal registers.

(1) EX-BUS DMA Single-Read/Write Operation

In SRAM bus operation, upon receiving a DREQ signal a DRACK is asserted to send notification of detection of a DMA request. There are no DRACK signals for other than LBSC-DMAC channel 0. In these cases, the device which is the DMA request source must negate the DREQ signal until the end of the bus operation through the DACK signal indicating that the DMA transfer has actually started. The DMAC starts the next DREQ sampling from the next clock cycle after the end of the DMA bus transfer (LBSC-DMAC DMA request mask control register (DRMSKR) settings can be used to delay the start of the next sampling), and upon detection, starts the next DMA transfer bus operation.

In DMA single read and write operations, the number of clock cycles from $\overline{CS}$ assertion to $\overline{RD}$ (or $\overline{WE}$) signal assertion, the number of $\overline{RD}$ (or $\overline{WE}$) pulse clock cycles, and the number of clock cycles from $\overline{RD}$ (or $\overline{WE}$) negation to $\overline{CS}$ negation, can be set through the LBSC internal register. During the period of $\overline{RD}$ (or $\overline{WE}$) assertion, if a WAIT signal from the DMA request source is detected, then during the WAIT assertion period, the $\overline{RD}$ (or $\overline{WE}$) pulse width is extended. With respect to specification of WAIT signal sampling also, the internal LBSC register should be used.

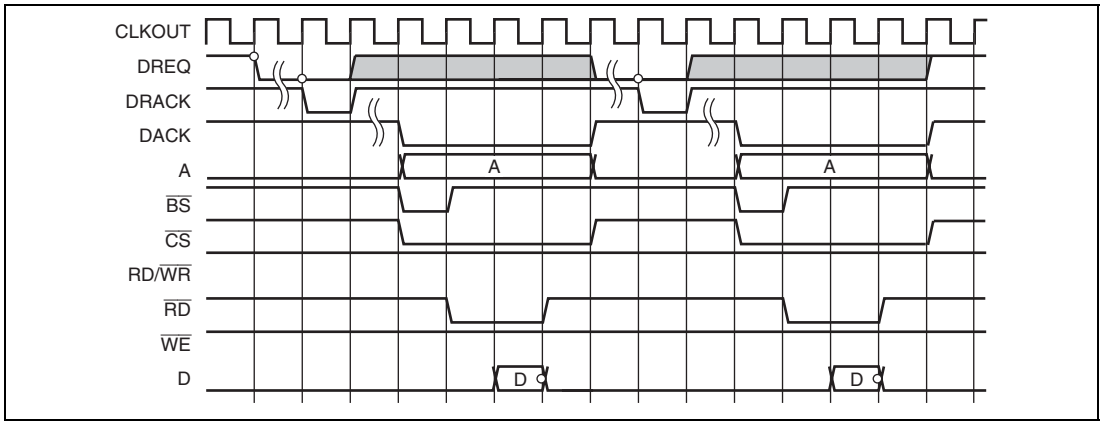


Figure 6A.7 External Bus DMA Read Operation (SRAM Bus Single Read)

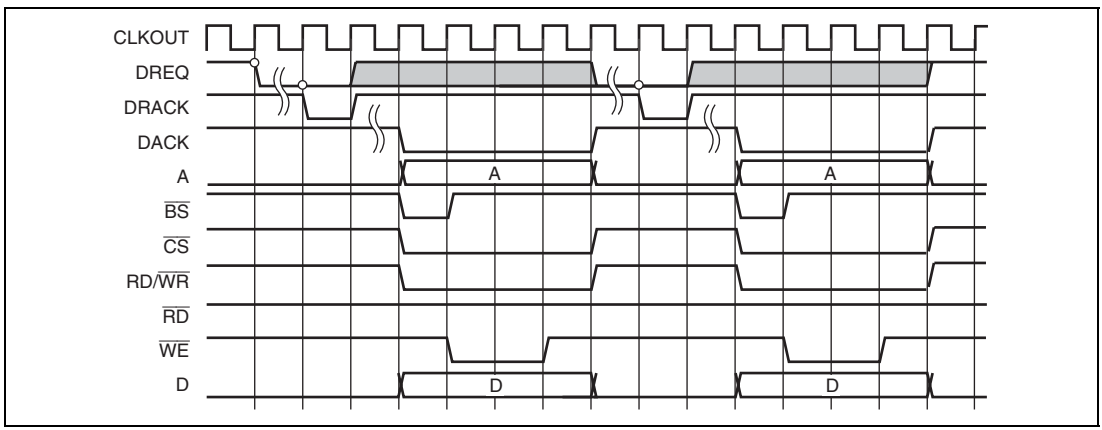


Figure 6A.8 External Bus DMA Write Operation (SRAM Bus Single Write)

(2) EX-BUS DMA 8-Burst Read/Write Operation

In DMA 8-burst operation, 8-burst bus transfer operation is performed upon detection of a single DREQ signal. In SRAM bus operation, upon receiving a DREQ signal a DRACK is asserted to send notification of detection of a DMA request. There are no DRACK signals for other than LBSC-DMAC channel 0, so that in these cases the device which is the DMA request source must negate the DREQ signal until the end of the 8-burst bus operation through the DACK signal indicating that the DMA transfer has actually started. The DMAC starts the next DREQ sampling from the next clock cycle after the end of the DMA bus operation (LBSC-DMAC DMA request mask control register (DRMSKR) settings can be used to delay the start of the next sampling), and upon detection, starts 8-burst bus operation for the next DMA transfer.

Also in DMA 8-burst operations, the number of clock cycles from $\overline{CS}$ assertion to $\overline{RD}$ (or $\overline{WE}$) signal assertion, the number of $\overline{RD}$ (or $\overline{WE}$) pulse clock cycles, and the number of clock cycles from $\overline{RD}$ (or $\overline{WE}$) negation to $\overline{CS}$ negation, can be set through the LBSC internal register. During the period of $\overline{RD}$ (or $\overline{WE}$) assertion, if a WAIT signal from the DMA request source is detected, then during the WAIT assertion period, the $\overline{RD}$ (or $\overline{WE}$) pulse width can be extended. With respect to WAIT signal control also, the internal LBSC register should be used.

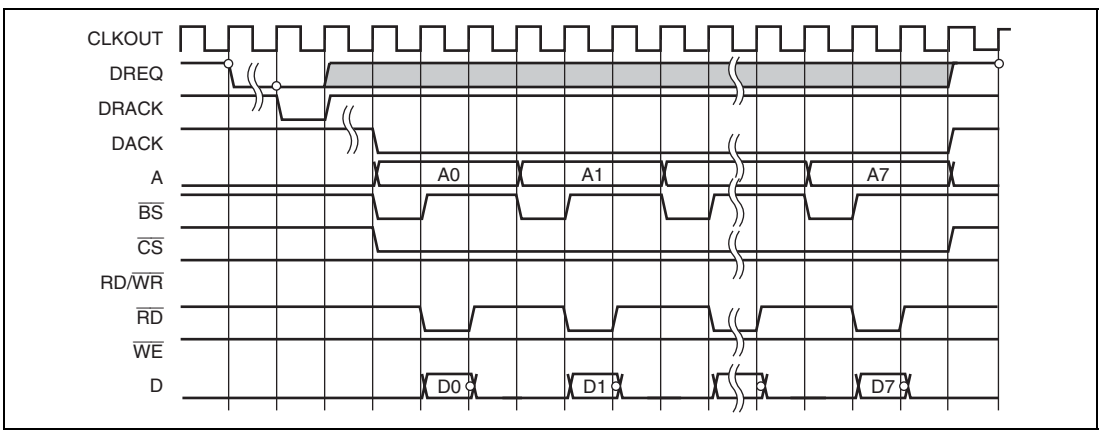


Figure 6A.9 External Bus DMA Read Operation (SRAM Bus Burst Read)

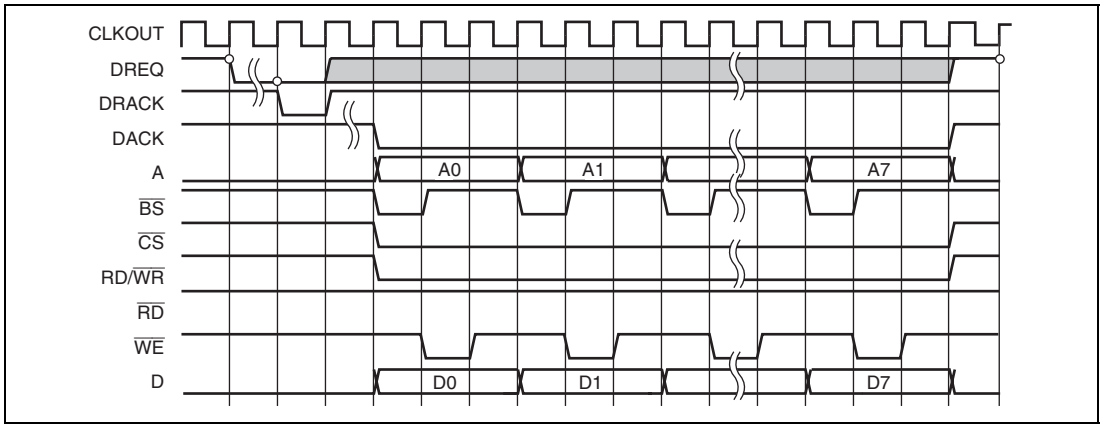


Figure 6A.10 External Bus DMA Write Operation (SRAM Bus Burst Write)

6A.5.11 SuperHyway Bus and HPB Bus Access Priority Control by the HPB-DMAC

To cope with contention among HPB-DMAC channels 0 to 28 over access to a SuperHyway bus or HPB bus, the HPB-DMAC divides the channels into two groups according to the settings of bits DMLV0 to DMLV28 in the HPB-DMAC access priority level control register (HPB-DMLVLR). In addition, the HPB-DMAC identifies the priority HPB-DMAC channel within each group by using a round-robin mechanism. Figure 6A.11 shows the concept of SuperHyway and HPB bus arbitration by the HPB-DMAC.

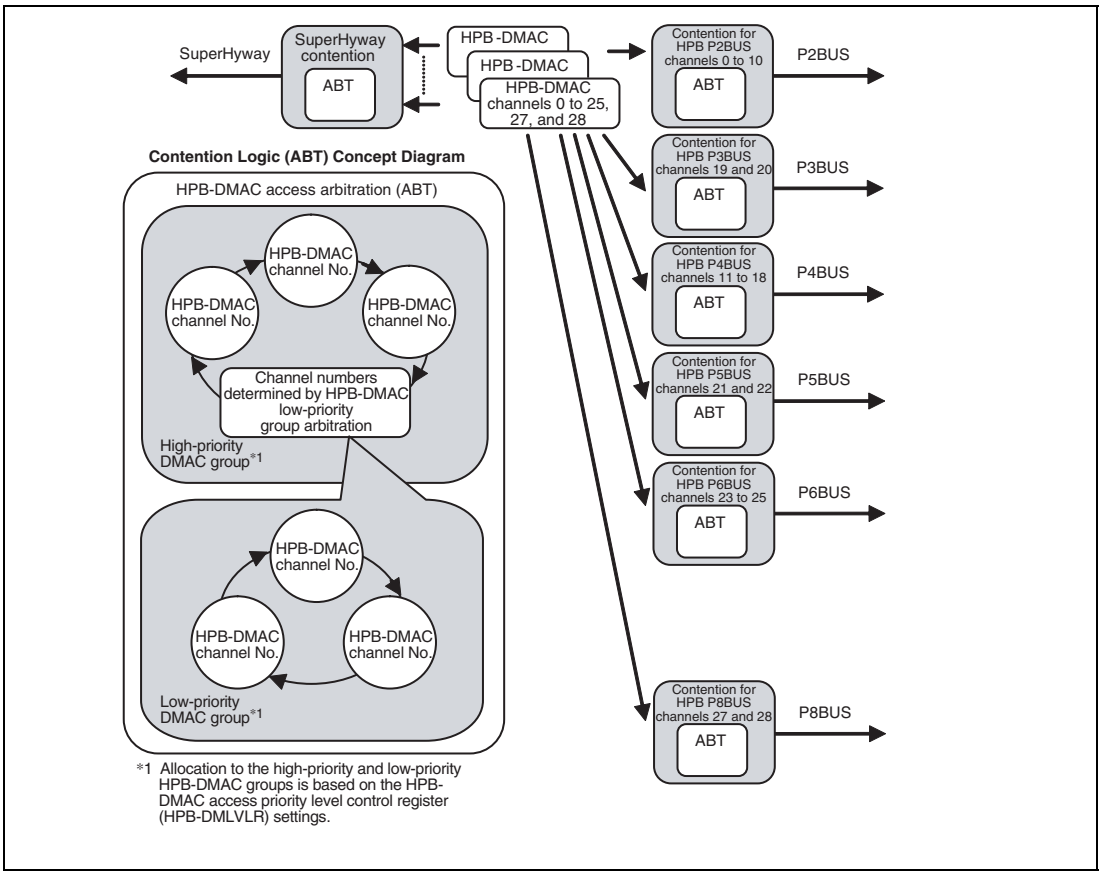


Figure 6A.11 Concept of SuperHyway and HPB Bus Arbitration by the HPB-DMAC

6A.5.12 Restrictions on Usage of Received Data Packing Function in DMA Transfer for HSPI

Although the registers of the HSPI have a 32-bit configuration, since the unit of data reception is eight bits, data packing can be used with DMA transfer. However, the following restrictions on usage apply.

When the HPB-DMAC (channels 0 to 10) is to be used, data packing is enabled by setting the PKMD bit of the DMA control register (DCR) to 1.

When data packing is to be used, set the DMA source address register of the HPB-DMAC (channels 0 to 10) to the relevant address below.

- HSPI

Set the address to H'FFFC 7013 (address in the P4 area) or H'1FFC 7013 (address in area 7).

6A.5.13 Points for Caution When DMA Transfer is Used with Modules Having Real-time Characteristics

When DMA transfer is used with modules that have real-time characteristics (SSI, HSPI, etc.), make settings for adjustment as described below. However, do not use more than eight DMA channels at a time with these settings.

1. Set the bits corresponding to the channel (DMLVx; x = 0 to 18) in the HPB-DMA access priority level control register (HPB-DMLVLR: see section 6A.4.28, HPB-DMA Access Priority Level Control Register (HPB-DMLVLR)) to 1, selecting level 1 (high; group 1).
2. Set the bits corresponding to the channel (SPRRx; x = 0 to 18) in the HPB-DMA SHwy priority control register 0 (HPB-DMASPR0: see section 6A.4.24, HPB-DMA SHwy Priority Control Register 0 (HPB-DMASPR0)), HPB-DMA SHwy priority control register 1 (HPB-DMASPR1: see section 6A.4.25, HPB-DMA SHwy Priority Control Register 1 (HPB-DMASPR1)), and HPB-DMA SHwy priority control register 2 (HPB-DMASPR2: see section 6A.4.26, HPB-DMA SHwy Priority Control Register 2 (HPB-DMASPR2)) to H'9 (selecting a priority level of the SHwy bus to H'9).

Section 6B LBSC within Bus Bridge

6B.1 Overview

The LBSC performs bus arbitration and necessary interface conversion for the accesses from the CPU (SuperHyway bus) and DMA accesses from LBSC-DMAC channels 0 to 2 and outputs them to the external buses. Further, for external bus access, various settings can be specified in the LBSC control registers for the selection of a connection interface type for each area on the external bus or for the adjustment of number of setup/hold cycles on addresses and chip select signals with respect to read/write enable signals. Thus, the LBSC configuration allows diversity in methodology for accessing various external devices that are assigned to their corresponding areas.

The frequency of the external bus clock CLKOUT signal is 44.4 MHz (1/12th of the 533.3 MHz CPU operating frequency) or 50.0 MHz (1/8th of the 400 MHz CPU operating frequency). The LBSC outputs bus signals in synchronization with the external bus clock.

6B.2 Features

The key features of the LBSC include:

- Support for areas 0, 1, and 6
  - Each area is allocated to EX_BUS, and SRAM, ATA, or byte control SRAM bus protocol can be selected.
  - Controls 64 Mbytes of area 6 divided into up to six areas (each area capacity is variable).
  - Controls 64 Mbytes of area 0 divided into up to seven areas (each area capacity is variable)(when MD7 pin = 0 and MD9 pin = 1).
  - Interface, bus size, and wait-cycle insertion can be controlled in each area.
  - Provides output synchronized with a bus clock frequency equal to 1/12th of the 533.3 MHz CPU operating frequency.
  - Provides output synchronized with a bus clock frequency equal to 1/8th of the 400 MHz CPU operating frequency.
- External DMA transfer (for details on DMAC, refer to section 6A, Direct Memory Access Controller for Local Bus and Peripheral Modules (LBSC-DMAC/HPB-DMAC).
  - Three channels
  - Support of devices with DACK signal
  - Support of edge-detection and level-detection external request signals
  - Synchronous/asynchronous DREQ and the polarity of DREQ, DACK, and DRACK can be inverted through register settings.

- SRAM interface
  - Wait-cycle insertion can be controlled through register settings.
  - Wait cycles can be inserted with the EX_WAIT pin.
  - Connectable bus size: 16 or 8 bits.
- Burst ROM interface (area 0 and CPU access only)
  - Wait-cycle insertion can be controlled through register settings.
  - Burst count can be specified through register settings (cases where this reaches an address branching point are automatically detected, after which the access is broken off).
  - Connectable bus size: 16 or 8 bits
- Byte-control SRAM interface (areas 1 and 6 only)
  - SRAM interface with byte control
  - Wait-cycle insertion can be controlled through register settings.
  - Wait-cycle insertion with the EX_WAIT pin
  - Connectable bus size: 16 or 8 bits
- ATA interface (areas 1 and 6 only)
  - Wait-cycle insertion can be controlled through register settings.
  - Support of PIO modes 0 to 4
  - Support of multi-word transfer
  - Support of Ultra DMA modes 0 to 4 transfer
  - Ready timeout detection (detection time (ns) = EX_BUS operating frequency (ns) × 100 clock cycles)

6B.3 Block Diagram

Figure 6B.1 is a block diagram of the LBSC. Placed on the SuperHyway bus, the LBSC outputs accesses from the CPU in sequence to an external bus according to the settings that are provided in internal registers of the LBSC. For the external bus EX_BUS, either of the SRAM or ATA bus protocol can be selected. In addition, the LBSC incorporates three LBSC-DMAC channels that control DMA transfers between an external bus and the DDR2-SDRAM; and INTC and INTC2 that control interruptions within and outside a chip. For details on the LBSC-DMAC, INTC, and INTC2, see the relevant sections. Since the CPU and the LBSC-DMAC generate access contention for an external bus, the BSC uses an arbiter unit to arbitrate such access requests. The access request that was selected by means of arbitration is converted into an external bus waveform by the bus interface unit before being output. The LBSC has an external wait control input that controls the pulse width. When a request for access is received by an external device, the external device uses this to control the wait for a response to suit the situation at the time of access-request reception.

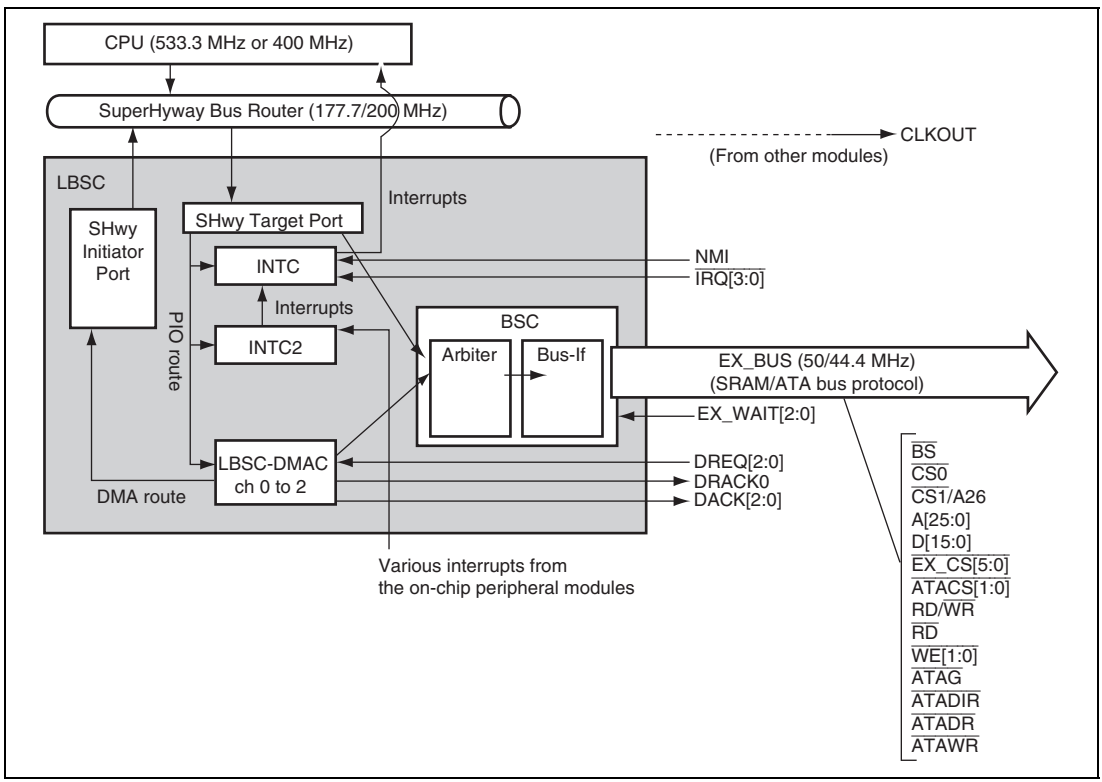


Figure 6B.1 Block Diagram of LBSC

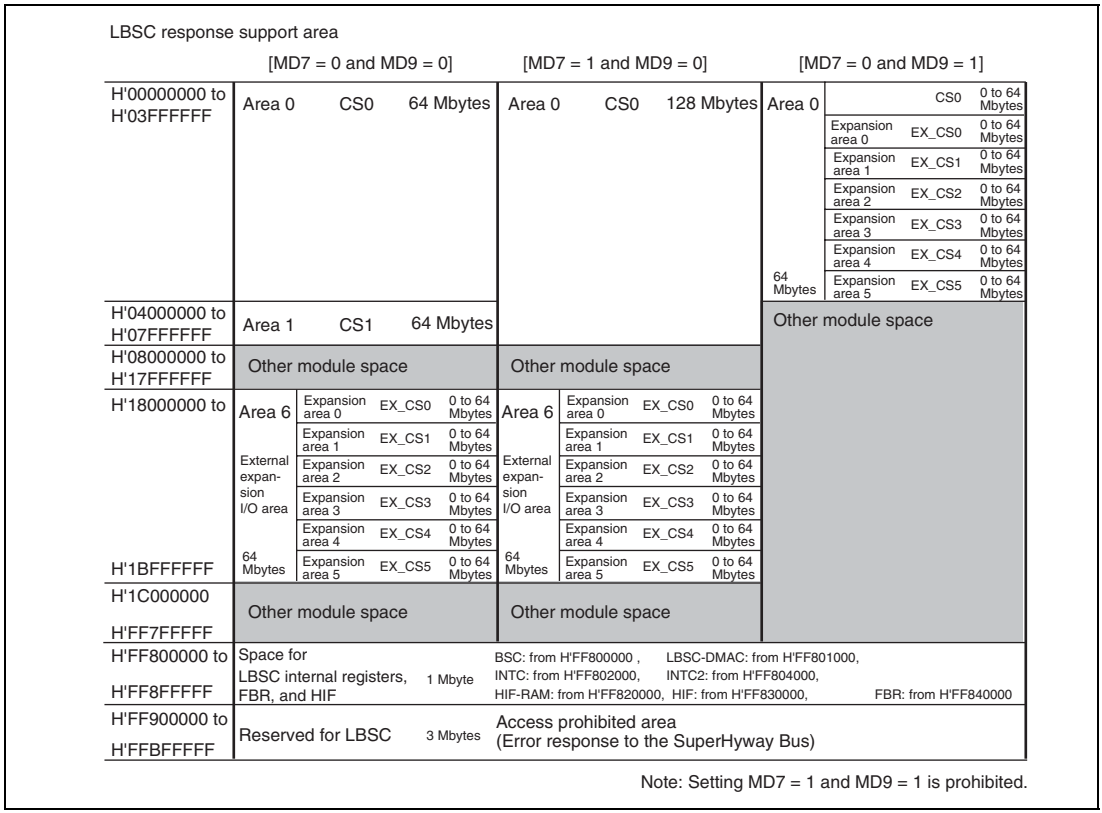
6B.4 LBSC Areas

6B.4.1 LBSC Support Areas

Figure 6B.2 shows LBSC response support areas from the CPU.

The basic configuration of the LBSC supports area 0 and area 1 as external spaces; and area 6 as an expansion I/O space. Area 6 can be subdivided into a maximum of six units, each having a maximum of 64 Mbytes as determined by register settings. The total capacity for area 6 is also up to 64 Mbytes. For area 0 can be made into a 128-Mbyte space depending on the specific LSI startup mode (MD7 mode pin) in use. In this case, however, area 1 is completely allocated as a space in area 0, and the $\overline{CS1}$ signal is changed to the signal equivalent to A[26] in a bus address (addition of one bit to the address signal changes the capacity from 64 Mbytes to 128 Mbytes). When the MD7 mode pin = 0 and the MD9 mode pin = 1 for LSI startup mode, the LBSC supports only area 0. This area can be subdivided into a maximum of seven units, each with a size of 64 Mbytes or less as determined by register settings. The total capacity for area 0 is also a maximum of 64 Mbytes.

Registers of the BSC, LBSC-DMAC, INTC, and INTC2 are provided in the internal register space of the LBSC.



Note: Setting MD7 = 1 and MD9 = 1 is prohibited.

Figure 6B.2 LBSC Response Support Areas from the CPU

6B.4.2 Functionality Supported in Each Area

Table 6B.1 lists the functions supported by the LBSC in each area on the EX_BUS.

Table 6B.1 Functions Supported in Each Area on EX_BUS

Area		Bus	Capacity	Operating Mode	Guard interval	WAIT Function
0		8/16 bits	64/128 Mbytes selectable Area 0 divided MD7 and MD9 pins specified	SRAM DMA Burst ROM	Disabled	Enabled
1		8/16 bits	Fixed to 64 Mbytes When area 0 = 128 Mbytes, no space exists.	SRAM DMA ATA (PIO) Byte-control SRAM	Enabled	Enabled
Expansion area	0	8/16 bits	0 to 64 Mbytes Total capacity of expansion areas: 64 Mbytes	SRAM DMA ATA (PIO) Byte-control SRAM	Enabled	Enabled
	1	8/16 bits	0 to 64 Mbytes Total capacity of expansion areas: 64 Mbytes	SRAM DMA ATA (PIO) Byte-control SRAM	Enabled	Enabled
	2	8/16 bits	0 to 64 Mbytes Total capacity of expansion areas: 64 Mbytes	SRAM DMA ATA (PIO) Byte-control SRAM	Enabled	Enabled

Notes: 1. The bus size for area 0 is specified with the LSI mode pins (MD[6:5]).
2. When using area 0 in 128-Mbyte mode (MD7), no space exists for area 1.
3. When using area 0 in division mode (MD7 and MD9), the capacity of each division is a maximum of 64 Mbytes. Total capacity of area 0 is also a maximum of 64 Mbytes.
4. When accessing through EX_BUS, A[0] is output as byte address even if 16-bit bus is selected.
5. One DMAC channel cannot be allocated to two areas simultaneously.
6. The capacity of one expansion area is a maximum of 64 Mbytes. Total capacity of expansion areas is also a maximum of 64 Mbytes.

6B.5 Register Descriptions

The LBSC set registers to control the interface, bus size, $\overline{RD}/\overline{WE}$ signal pulse cycles, and setup and hold cycles for the $\overline{CS}$ signal with respect to the $\overline{RD}/\overline{WE}$ signal, for each of externally connected devices. Table 6B.2 (1) lists registers of the LBSC. Note that correct operation is not guaranteed in principle if each register is modified during external bus access (for register modification in other cases, refer to the Note under specific register description).

Table 6B.2 (1) LBSC Register Configuration

Address (H'FF80_0***)	Register Name	Abbreviation	Access Type	Access Size (bit)	Remarks
200	Area 0 control register	CS0CTRL	R/W	32	
204	Area 1 control register	CS1CTRL	R/W	32	
208 to 21C	Expansion area x control register	ECSxCTRL	R/W	32	x = 0 to 5
220	Area 0 control 2 register	CS0CTRL2	R/W	32	
230	Area 0 RD/WE pulse control register	CSWCR0	R/W	32	
234	Area 1 RD/WE pulse control register	CSWCR1	R/W	32	
238 to 24C	Expansion area x RD/WE pulse control register	ECSWCRx	R/W	32	x = 0 to 5
250 to 258	LBSC-DMAC channel y RD/WE pulse control register	EXDMAWCRy	R/W	32	y = 0 to 2
280	Area 0 external wait control register	CSPWCR0	R/W	32	
284	Area 1 external wait control register	CSPWCR1	R/W	32	
288 to 29C	Expansion area x external wait control register	ECSPWCRx	R/W	32	x = 0 to 5
2A0	External wait input control register	EXWTSYNC	R/W	32	
2B0	Area 0 burst control register	CS0BSTCTL	R/W	32	
2B4	Area 0 burst pitch set register	CS0BTPH	R/W	32	
2C0	Area 1 guard setting register	CS1GDST	R/W	32	
2C4 to 2D8	Expansion area x guard setting register	ECSxGDST	R/W	32	x = 0 to 5
2F0 to 2F8	LBSC-DMAC channel y area allocation register	EXDMASETy	R/W	32	y = 0 to 2
310 to 318	LBSC-DMAC channel y control register	EXDMCRy	R/W	32	y = 0 to 2
330	BSC interrupt source status register	BCINTSR	R	32	

Address (H'FF80_0***)	Register Name	Abbreviation	Access Type	Access Size (bit)	Remarks
334	BSC interrupt source clear register	BCINTCR	—/W	32	
338	BSC interrupt enable register	BCINTMR	R/W	32	
340	EX_BUS priority level set register	EXBATLV	R/W	32	
344	External wait status register	EXWTSTS	R	32	
380	ATACS control register	ATACSCTRL	R/W	32	

Note: Do not write to any addresses other than listed above. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than those listed above are undefined.

Table 6B.2 (2) Register State in Each Operating Mode

Abbreviation	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
CS0CTRL	Undefined	Retained	Retained	Retained	—	Initialized
CS1CTRL	H'0000_0020	Retained	Retained	Retained	—	Initialized
ECSxCTRL	H'0000_0020	Retained	Retained	Retained	—	Initialized
CS0CTRL2	H'0000_4000	Retained	Retained	Retained	—	Initialized
CSWCR0	H'077F_077F	Retained	Retained	Retained	—	Initialized
CSWCR1	H'077F_077F	Retained	Retained	Retained	—	Initialized
ECSWCRx	H'077F_077F	Retained	Retained	Retained	—	Initialized
EXDMAWCRy	H'077F_077F	Retained	Retained	Retained	—	Initialized
CSPWCR0	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
CSPWCR1	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
ECSPWCRx	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
EXWTSYNC	H'0000_0000	Retained	Retained	Retained	—	Initialized
CS0BSTCTL	H'0000_0000	Retained	Retained	Retained	—	Initialized
CS0BTPH	H'0000_00F7	Retained	Retained	Retained	—	Initialized
CS1GDST	H'0000_0000	Retained	Retained	Retained	—	Initialized
ECSxGDST	H'0000_0000	Retained	Retained	Retained	—	Initialized
EXDMASETy	H'0000_0000	Retained	Retained	Retained	—	Initialized
EXDMCRy	H'0000_0000	Retained	Retained	Retained	—	Initialized

Abbreviation	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
BCINTSR	H'0000_0000	Retained	Retained	Retained	—	Initialized
BCINTCR	H'0000_0000	Retained	Retained	Retained	—	Initialized
BCINTMR	H'0000_0000	Retained	Retained	Retained	—	Initialized
EXBATLV	H'0000_0000	Retained	Retained	Retained	—	Initialized
EXWTSTS	Undefined	Undefined	Retained	Retained	—	Initialized
ATACSCTRL	H'0000_0000	Retained	Retained	Retained	—	Initialized

Note: Initialized: The value is the one written in "Power-on Reset".

6B.5.1 Area 0 Control Register (CS0CTRL)

Function: CS0CTRL specifies the interface in area 0 (EX_BUS).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ENDIAN	—	—	—	—	—	—	128B	—	—	CS0SZ		—	—	CS0IF	
Initial value:	—	0	0	0	0	0	0	—	0	0	—	—	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15	ENDIAN	—	R	Endian Indication These bits indicate the value specified by the LSI mode pin MD8. 0: Big endian 1: Little endian
14 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	128B	—	R	Area 0 Capacity Indication This bit indicates the value specified by the LSI mode pin MD7. 0: 64 Mbytes 1: 128 Mbytes
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
5, 4	CS0SZ	—	R	Area 0 Bus Size Indication These bits indicate the bus size specified by the LSI mode pins MD6 and MD5. 00: Setting prohibited 01: 8 bits 10: 16 bits 11: Setting prohibited
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	CS0IF	00	R/W	Area 0 Interface Selection 00: Standard (SRAM) 01: Burst ROM 10: Setting prohibited 11: Setting prohibited

Note: 1. Even when the burst ROM interface is selected by setting CS0IF = B'01, burst ROM operation is not available unless appropriate setting is made in CS0BSTCTL. Be sure to specify both CS0BSTCTL and CS0BTPH before using the burst ROM interface.

2. Setting of burst ROM is valid for CPU access only. For DMA transfer access to area 0, SRAM interface (enabled when EXDMAWCR/CSPWCR0 is valid) is usually selected.

6B.5.2 Area 1 Control Register (CS1CTRL)

Function: CS1CTRL specifies the interface in area 1 (EX_BUS).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CS1SZ		—	CS1 BRM	CS1IF	
Initial value:	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
15 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5, 4	CS1SZ	10	R/W	Area 1 Bus Size Selection 00: Setting prohibited 01: 8 bits 10: 16 bits 11: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	CS1BRM	0	R/W	Area 1 Byte-Control SRAM Mode Selection (valid only when CS1IF = 01) 0: Same cycle as $\overline{CS}$ 1: Same cycle as $\overline{RD}$

Bit	Bit Name	Initial Value	R/W	Description
1, 0	CS1IF	00	R/W	Area 1 Interface Selection 00: Standard (SRAM) 01: Byte-control SRAM 10: ATA 11: Setting prohibited

6B.5.3 Expansion Area x Control Register (ECSxCTRL (x = 0 to 5))

Function: ECSxCTRL specifies the interface and area capacity for expansion area x (EX_BUS).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	ECSxCP							—	—	ECSxSZ		—	ECSx BRM	ECSxIF	
Initial value:	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14 to 8	ECSxCP	0000000	R/W	Expansion Area x Capacity Setting (this bit field has no effect in ECS5CTRL. See the notes below). The value set in these bits × 1 Mbyte is the capacity of the respective expansion area.
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
5, 4	ECSxSZ	10	R/W	Expansion Area x Bus Size Selection 00: Setting prohibited 01: 8 bits 10: 16 bits 11: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	ECSxBRM	0	R/W	Expansion Area x Byte-Control SRAM Mode Selection (valid only when ECSxIF = 01) 0: Same cycle as $\overline{CS}$ 1: Same cycle as $\overline{RD}$
1, 0	ECSxIF	00	R/W	Expansion Area x Interface Selection 00: Standard (SRAM) 01: Byte-control SRAM 10: ATA 11: Setting prohibited

Notes: 1. If the sum of all ECSxCP settings exceeds 64, the excess capacity is ignored.
Example: When the registers are set to ECS0CP = 40 Mbytes, ECS1CP = 30 Mbytes, and ECS2CP = 20 Mbytes, the excess capacity is ignored. That is, ECS0CP = 40 Mbytes, ECS1CP = 24 Mbytes, and ECS2CP = 0 Mbytes.

2. The area-capacity setting in the ECS5CTRL register (ECS5CP bits) has no effect. Regardless of the setting, the capacity of expansion area 5 will be 64 Mbytes − (sum of capacities for ECS0CP to ECS4CP). Accordingly, If the sum of the ECS0CP to ECS4CP settings exceeds 64 Mbytes, the capacity for expansion area 5 will be 0 Mbytes.

3. Intermediate areas can be set to 0 Mbytes. Specify 0 Mbytes for the area where the $\overline{CS}$ signal cannot be used by selecting the pin multiplex exclusive signal. If the capacity other than 0 Mbytes is specified for the area, the allocated area cannot be used.

4. For the expansion area capacity setting and area division, refer to section 6B.6.1 (1), Address Generation/Alignment.

6B.5.4 Area 0 Control 2 Register (CS0CTRL2)

Function: CS0CTRL2 specifies the area capacity for area 0 (EX_BUS). (Enabled when MD7 pin = 0 and MD9 pin = 1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	CS0CP							—	—	—	—	—	—	—	—
Initial value:	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 15	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
14 to 8	CS0CP	H'40	R/W	Area 0 Capacity Setting (see the notes below). The value set in this bit × 1 Mbyte is the capacity of the area 0.
7 to 0	—	All 0	R	Reserved This bit is always read as 0. The write value should always be 0.

Notes: 1. If the sum of CS0CP and ECSxCP settings exceeds 64, the excess capacity is ignored.
Example: When the registers are set to CS0CP = 20 Mbytes, ECS0CP = 20 Mbytes, ECS1CP = 30 Mbytes, and ECS2CP = 20 Mbytes, the excess capacity is ignored. That is, CP0CP = 20 Mbytes, ECS0CP = 20 Mbytes, ECS1CP = 24 Mbytes, and ECS2CP = 0 Mbytes.

2. The area-capacity setting in the ECS5CTRL register (ECS5CP bits) has no effect. Regardless of the setting, the capacity of expansion area 5 will be 64 Mbytes − (sum of capacities for CS0CP and ECS0CP to ECS4CP). Accordingly, If the sum of the CS0CP and ECS0CP to ECS4CP settings exceeds 64 Mbytes, the capacity for expansion area 5 will be 0 Mbytes.
3. Intermediate areas can be set to 0 Mbytes. Specify 0 Mbytes for the area where the $\overline{CS}$ signal cannot be used by selecting the pin multiplex exclusive signal. If the capacity other than 0 Mbytes is specified for the area, the allocated area cannot be used.
4. For the area 0 and expansion area capacity settings and area division, refer to section 6B.6.1 (1), Address Generation/Alignment.

6B.5.5 Area 0 RD/WE Pulse Control Register (CSWCR0)

Function: CSWCR0 specifies the $\overline{RD}/\overline{WE}$ pulse cycles and setup and hold cycles for the $\overline{CS}$ signal and address during access to area 0 (EX_BUS). (The settings for read access are ignored when the burst ROM interface is selected.)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	WRITE CS SETUP CYCLE			—	WRITE CS HOLD CYCLE			WRITE PULSE CYCLE			
Initial value:	0	0	0	0	0	1	1	1	0	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	READ CS SETUP CYCLE			—	READ CS HOLD CYCLE			READ PULSE CYCLE			
Initial value:	0	0	0	0	0	1	1	1	0	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 24	WRITE CS SETUP CYCLE	111	R/W	These bits specify the $\overline{CS}$ and address setup cycles with respect to the $\overline{WE}$ signal during writing to area 0. 000: 0 cycle for setup period 001: 1 cycle for setup period 010: 2 cycles for setup period 011: 3 cycles for setup period 100: 4 cycles for setup period 101: 5 cycles for setup period 110: 6 cycles for setup period 111: 7 cycles for setup period
23	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
22 to 20	WRITE CS HOLD CYCLE	111	R/W	These bits specify the $\overline{CS}$ and address hold cycles with respect to the $\overline{WE}$ signal during writing to area 0. 000: 0 cycle for hold period 001: 1 cycle for hold period 010: 2 cycles for hold period 011: 3 cycles for hold period 100: 4 cycles for hold period 101: 5 cycles for hold period 110: 6 cycles for hold period 111: 7 cycles for hold period
19 to 16	WRITE PULSE CYCLE	1111	R/W	These bits specify the $\overline{WE}$ pulse cycles during writing to area 0. 0000: Setting prohibited 0001: 1-cycle pulse 0010: 2-cycle pulse 0011: 3-cycle pulse 0100: 4-cycle pulse 0101: 5-cycle pulse 0110: 6-cycle pulse 0111: 7-cycle pulse 1000: 8-cycle pulse 1001: 9-cycle pulse 1010: 10-cycle pulse 1011: 11-cycle pulse 1100: 12-cycle pulse 1101: 13-cycle pulse 1110: 14-cycle pulse 1111: 15-cycle pulse
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
10 to 8	READ CS SETUP CYCLE	111	R/W	These bits specify the $\overline{CS}$ and address setup cycles with respect to the $\overline{RD}$ signal during reading from area 0. 000: 0 cycle for setup period 001: 1 cycle for setup period 010: 2 cycles for setup period 011: 3 cycles for setup period 100: 4 cycles for setup period 101: 5 cycles for setup period 110: 6 cycles for setup period 111: 7 cycles for setup period
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6 to 4	READ CS HOLD CYCLE	111	R/W	These bits specify the $\overline{CS}$ and address hold cycles with respect to the $\overline{RD}$ signal during reading from area 0. 000: 0 cycle for hold period 001: 1 cycle for hold period 010: 2 cycles for hold period 011: 3 cycles for hold period 100: 4 cycles for hold period 101: 5 cycles for hold period 110: 6 cycles for hold period 111: 7 cycles for hold period

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	READ PULSE CYCLE	1111	R/W	These bits specify the $\overline{RD}$ pulse cycles during reading from area 0. 0000: Setting prohibited 0001: 1-cycle pulse 0010: 2-cycle pulse 0011: 3-cycle pulse 0100: 4-cycle pulse 0101: 5-cycle pulse 0110: 6-cycle pulse 0111: 7-cycle pulse 1000: 8-cycle pulse 1001: 9-cycle pulse 1010: 10-cycle pulse 1011: 11-cycle pulse 1100: 12-cycle pulse 1101: 13-cycle pulse 1110: 14-cycle pulse 1111: 15-cycle pulse

Notes: 1. A minimum of two clock cycles are required for one EX_BUS access cycle and therefore, the setting must satisfy this lower limit.

Setting less than two clock cycles for one access cycle is not expected in the design of this LSI, and if attempted correct operation is not guaranteed.

Example: When CSSetupCycle = B'000, CSHoldCycle = B'000, and PulseCycle = B'0001, correct operation is not guaranteed.

2. When controlling wait insertion through LSI external pins (EX_WAIT[2:0]), set PulseCycle to B'0010 or a larger value. If B'0001 or a smaller value is specified, wait insertion through an external pin is disabled.
3. When the burst ROM interface is specified for area 0, the read access-related settings for area 0 in this register is ignored and settings in CS0BTPH are enabled.
4. DMA transfer access is performed according to the settings in the RD/WE pulse control register (EXDMAWCRy (y = 0 to 2)) for the respective LBSC-DMAC channel.
5. For details, refer to section 6B.6.1, SRAM Interface (Basic Functionality).

6B.5.6 Area 1 RD/WE Pulse Control Register (CSWCR1)

Function: CSWCR1 specifies the $\overline{RD}/\overline{WE}$ pulse cycles and setup and hold cycles for the $\overline{CS}$ signal and address during access to area 1 (EX_BUS).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	WRITE CS SETUP CYCLE			—	WRITE CS HOLD CYCLE			WRITE PULSE CYCLE			
Initial value:	0	0	0	0	0	1	1	1	0	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	READ CS SETUP CYCLE			—	READ CS HOLD CYCLE			READ PULSE CYCLE			
Initial value:	0	0	0	0	0	1	1	1	0	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 24	WRITE CS SETUP CYCLE	111	R/W	These bits specify the $\overline{CS}$ and address setup cycles with respect to the $\overline{WE}$ signal during writing to area 1. 000: 0 cycle for setup period 001: 1 cycle for setup period 010: 2 cycles for setup period 011: 3 cycles for setup period 100: 4 cycles for setup period 101: 5 cycles for setup period 110: 6 cycles for setup period 111: 7 cycles for setup period
23	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
22 to 20	WRITE CS HOLD CYCLE	111	R/W	These bits specify the $\overline{CS}$ and address hold cycles with respect to the $\overline{WE}$ signal during writing to area 1. 000: 0 cycle for hold period 001: 1 cycle for hold period 010: 2 cycles for hold period 011: 3 cycles for hold period 100: 4 cycles for hold period 101: 5 cycles for hold period 110: 6 cycles for hold period 111: 7 cycles for hold period
19 to 16	WRITE PULSE CYCLE	1111	R/W	These bits specify the $\overline{WE}$ pulse cycles during writing to area 1. 0000: Setting prohibited 0001: 1-cycle pulse 0010: 2-cycle pulse 0011: 3-cycle pulse 0100: 4-cycle pulse 0101: 5-cycle pulse 0110: 6-cycle pulse 0111: 7-cycle pulse 1000: 8-cycle pulse 1001: 9-cycle pulse 1010: 10-cycle pulse 1011: 11-cycle pulse 1100: 12-cycle pulse 1101: 13-cycle pulse 1110: 14-cycle pulse 1111: 15-cycle pulse
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
10 to 8	REACS SETUP CYCLE	111	R/W	These bits specify the $\overline{CS}$ and address setup cycles with respect to the $\overline{RD}$ signal during reading from area 1. 000: 0 cycle for setup period 001: 1 cycle for setup period 010: 2 cycles for setup period 011: 3 cycles for setup period 100: 4 cycles for setup period 101: 5 cycles for setup period 110: 6 cycles for setup period 111: 7 cycles for setup period
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6 to 4	READCS HOLD CYCLE	111	R/W	These bits specify the $\overline{CS}$ and address hold cycles with respect to the $\overline{RD}$ signal during reading from area 1. 000: 0 cycle for hold period 001: 1 cycle for hold period 010: 2 cycles for hold period 011: 3 cycles for hold period 100: 4 cycles for hold period 101: 5 cycles for hold period 110: 6 cycles for hold period 111: 7 cycles for hold period

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	READ PULSE CYCLE	1111	R/W	These bits specify the $\overline{RD}$ pulse cycles during reading from area 1. 0000: Setting prohibited 0001: 1-cycle pulse 0010: 2-cycle pulse 0011: 3-cycle pulse 0100: 4-cycle pulse 0101: 5-cycle pulse 0110: 6-cycle pulse 0111: 7-cycle pulse 1000: 8-cycle pulse 1001: 9-cycle pulse 1010: 10-cycle pulse 1011: 11-cycle pulse 1100: 12-cycle pulse 1101: 13-cycle pulse 1110: 14-cycle pulse 1111: 15-cycle pulse

Notes: 1. A minimum of two clock cycles are required for one EX_BUS access cycle and therefore, the setting must satisfy this lower limit.

Setting less than two clock cycles for one access cycle is not expected in the design of this LSI, and if attempted correct operation is not guaranteed.

Example: When CSSetupCycle = B'000, CSHoldCycle = B'000, and PulseCycle = B'0001, correct operation is not guaranteed.

2. When controlling wait insertion through LSI external pins (EX_WAIT[2:0]), set PulseCycle to B'0010 or a larger value. If B'0001 or a smaller value is specified, wait insertion through an external pin is disabled.
3. DMA transfer access to area 1 is performed according to the settings in the RD/WE pulse control register (EXDMAWCRy (y = 0 to 2)) for the respective LBSC-DMAC channel.
4. For details, refer to section 6B.6.1, SRAM Interface (Basic Functionality).

6B.5.7 Expansion Area x RD/WE Pulse Control Register (ECSWCRx (x = 0 to 5))

Function: ECSWCRx specifies the $\overline{RD}/\overline{WE}$ pulse cycles and setup and hold cycles for the $\overline{CS}$ signal and address during access to expansion area x (x = 0 to 5) (EX_BUS).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	WRITE CS SETUP CYCLE			—	WRITE CS HOLD CYCLE			WRITE PULSE CYCLE			
Initial value:	0	0	0	0	0	1	1	1	0	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	READ CS SETUP CYCLE			—	READ CS HOLD CYCLE			READ PULSE CYCLE			
Initial value:	0	0	0	0	0	1	1	1	0	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 24	WRITE CS SETUP CYCLE	111	R/W	These bits specify the $\overline{CS}$ and address setup cycles with respect to the $\overline{WE}$ signal during writing to expansion area x. 000: 0 cycle for setup period 001: 1 cycle for setup period 010: 2 cycles for setup period 011: 3 cycles for setup period 100: 4 cycles for setup period 101: 5 cycles for setup period 110: 6 cycles for setup period 111: 7 cycles for setup period
23	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
22 to 20	WRITE CS HOLD CYCLE	111	R/W	These bits specify the $\overline{CS}$ and address hold cycles with respect to the $\overline{WE}$ signal during writing to expansion area x. 000: 0 cycle for hold period 001: 1 cycle for hold period 010: 2 cycles for hold period 011: 3 cycles for hold period 100: 4 cycles for hold period 101: 5 cycles for hold period 110: 6 cycles for hold period 111: 7 cycles for hold period
19 to 16	WRITE PULSE CYCLE	1111	R/W	These bits specify the $\overline{WE}$ pulse cycles during writing to expansion area x. 0000: Setting prohibited 0001: 1-cycle pulse 0010: 2-cycle pulse 0011: 3-cycle pulse 0100: 4-cycle pulse 0101: 5-cycle pulse 0110: 6-cycle pulse 0111: 7-cycle pulse 1000: 8-cycle pulse 1001: 9-cycle pulse 1010: 10-cycle pulse 1011: 11-cycle pulse 1100: 12-cycle pulse 1101: 13-cycle pulse 1110: 14-cycle pulse 1111: 15-cycle pulse
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
10 to 8	READ CS SETUP CYCLE	111	R/W	These bits specify the $\overline{CS}$ and address setup cycles with respect to the $\overline{RD}$ signal during reading from expansion area x. 000: 0 cycle for setup period 001: 1 cycle for setup period 010: 2 cycles for setup period 011: 3 cycles for setup period 100: 4 cycles for setup period 101: 5 cycles for setup period 110: 6 cycles for setup period 111: 7 cycles for setup period
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6 to 4	READ CS HOLD CYCLE	111	R/W	These bits specify the $\overline{CS}$ and address hold cycles with respect to the $\overline{RD}$ signal during reading from expansion area x. 000: 0 cycle for hold period 001: 1 cycle for hold period 010: 2 cycles for hold period 011: 3 cycles for hold period 100: 4 cycles for hold period 101: 5 cycles for hold period 110: 6 cycles for hold period 111: 7 cycles for hold period

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	READ PULSE CYCLE	1111	R/W	These bits specify the $\overline{RD}$ pulse cycles during reading from expansion area x. 0000: Setting prohibited 0001: 1-cycle pulse 0010: 2-cycle pulse 0011: 3-cycle pulse 0100: 4-cycle pulse 0101: 5-cycle pulse 0110: 6-cycle pulse 0111: 7-cycle pulse 1000: 8-cycle pulse 1001: 9-cycle pulse 1010: 10-cycle pulse 1011: 11-cycle pulse 1100: 12-cycle pulse 1101: 13-cycle pulse 1110: 14-cycle pulse 1111: 15-cycle pulse

Notes: 1. A minimum of two clock cycles are required for one EX_BUS access cycle and therefore, the setting must satisfy this lower limit.

Setting less than two clock cycles for one access cycle is not expected in the design of this LSI, and if attempted correct operation is not guaranteed.

Example: When CSSetupCycle = B'000, CSHoldCycle = B'000, and PulseCycle = B'0001, correct operation is not guaranteed.

2. When controlling wait insertion through LSI external pins (EX_WAIT[2:0]), set PulseCycle to B'0010 or a larger value. If B'0001 or a smaller value is specified, wait insertion through an external pin is disabled.
3. DMA transfer access to expansion area x is performed according to the settings in the RD/WE pulse control register (EXDMAWCRy (y = 0 to 2)) for the respective LBSC-DMAC channel.
4. For details, refer to section 6B.6.1, SRAM Interface (Basic Functionality).

6B.5.8 LBSC-DMAC Channel y RD/WE Pulse Control Register (EXDMAWCRy (y = 0 to 2))

Function: EXDMAWCRy specifies the $\overline{RD}/\overline{WE}$ pulse cycles and setup and hold cycles for the $\overline{CS}$ signal and address during access to EX_BUS in LBSC-DMAC channel y (y = 0 to 5).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	WRITE CS SETUP CYCLE			—	WRITE CS HOLD CYCLE			WRITE PULSE CYCLE			
Initial value:	0	0	0	0	0	1	1	1	0	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	READ CS SETUP CYCLE			—	READ CS HOLD CYCLE			READ PULSE CYCLE			
Initial value:	0	0	0	0	0	1	1	1	0	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 24	WRITE CS SETUP CYCLE	111	R/W	These bits specify the $\overline{CS}$ and address setup cycles with respect to the $\overline{WE}$ signal during write in channel y. 000: 0 cycle for setup period 001: 1 cycle for setup period 010: 2 cycles for setup period 011: 3 cycles for setup period 100: 4 cycles for setup period 101: 5 cycles for setup period 110: 6 cycles for setup period 111: 7 cycles for setup period
23	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
22 to 20	WRITE CS HOLD CYCLE	111	R/W	These bits specify the $\overline{CS}$ and address hold cycles with respect to the $\overline{WE}$ signal during write in channel y. 000: 0 cycle for hold period 001: 1 cycle for hold period 010: 2 cycles for hold period 011: 3 cycles for hold period 100: 4 cycles for hold period 101: 5 cycles for hold period 110: 6 cycles for hold period 111: 7 cycles for hold period
19 to 16	WRITE PULSE CYCLE	1111	R/W	These bits specify the $\overline{WE}$ pulse cycles during write in channel y. 0000: Setting prohibited 0001: 1-cycle pulse 0010: 2-cycle pulse 0011: 3-cycle pulse 0100: 4-cycle pulse 0101: 5-cycle pulse 0110: 6-cycle pulse 0111: 7-cycle pulse 1000: 8-cycle pulse 1001: 9-cycle pulse 1010: 10-cycle pulse 1011: 11-cycle pulse 1100: 12-cycle pulse 1101: 13-cycle pulse 1110: 14-cycle pulse 1111: 15-cycle pulse
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
10 to 8	READ CS SETUP CYCLE	111	R/W	These bits specify the $\overline{CS}$ and address setup cycles with respect to the $\overline{RD}$ signal during read in channel y. 000: 0 cycle for setup period 001: 1 cycle for setup period 010: 2 cycles for setup period 011: 3 cycles for setup period 100: 4 cycles for setup period 101: 5 cycles for setup period 110: 6 cycles for setup period 111: 7 cycles for setup period
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6 to 4	READ CS HOLD CYCLE	111	R/W	These bits specify the $\overline{CS}$ and address hold cycles with respect to the $\overline{RD}$ signal during read in channel y. 000: 0 cycle for hold period 001: 1 cycle for hold period 010: 2 cycles for hold period 011: 3 cycles for hold period 100: 4 cycles for hold period 101: 5 cycles for hold period 110: 6 cycles for hold period 111: 7 cycles for hold period

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	READ PULSE CYCLE	1111	R/W	These bits specify the $\overline{RD}$ pulse cycles during read in channel y. 0000: Setting prohibited 0001: 1-cycle pulse 0010: 2-cycle pulse 0011: 3-cycle pulse 0100: 4-cycle pulse 0101: 5-cycle pulse 0110: 6-cycle pulse 0111: 7-cycle pulse 1000: 8-cycle pulse 1001: 9-cycle pulse 1010: 10-cycle pulse 1011: 11-cycle pulse 1100: 12-cycle pulse 1101: 13-cycle pulse 1110: 14-cycle pulse 1111: 15-cycle pulse

Notes: 1. A minimum of two clock cycles are required for one EX_BUS access cycle and therefore, the setting must satisfy this lower limit.

Setting less than two clock cycles for one access cycle is not expected in the design of this LSI, and if attempted correct operation is not guaranteed.

Example: When CSSetupCycle = B'000, CSHoldCycle = B'000, and PulseCycle = B'0001, correct operation is not guaranteed.

2. When controlling wait insertion through LSI external pins (EX_WAIT[2:0]), set PulseCycle to B'0010 or a larger value. If B'0001 or a smaller value is specified, wait insertion through an external pin is disabled.
3. External wait insertion is controlled according to the external wait control register for the area where the DMAC channel (0 to 2) is assigned.
4. For details, refer to section 6B.6.3, LBSC-DMAC → DMA Interface.

6B.5.9 Area 0 External Wait Control Register (CSPWCR0)

Function: CSPWCR0 makes settings for external wait signal during access to area 0 (EX_BUS) (the settings are ignored when the burst ROM interface is selected).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	V	RB	WINV	EXWT2	EXWT1	EXWT0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	V	0	R/W	Area 0 External Wait Signal Enable/Disable 0: Disabled 1: Enabled
4	RB	0	R/W	Area 0 READY/BUSY Logic Selection 0: BUSY logic 1: READY logic
3	WINV	0	R/W	Area 0 External Wait Signal Polarity 0: Does not invert the polarity of the area 0 external wait signal. 1: Inverts the polarity of the area 0 external wait signal.
2	EXWT2	0	R/W	Area 0 EX_WAIT2 Enable 0: Disables EX_WAIT2 for area 0. 1: Enables EX_WAIT2 for area 0.
1	EXWT1	0	R/W	Area 0 EX_WAIT1 Enable 0: Disables EX_WAIT1 for area 0. 1: Enables EX_WAIT1 for area 0.

Bit	Bit Name	Initial Value	R/W	Description
0	EXWT0	0	R/W	Area 0 EX_WAIT0 Enable 0: Disables EX_WAIT0 for area 0. 1: Enables EX_WAIT0 for area 0.

Notes: 1. When this register setting is made valid (bit V = 1), any one of bits EXWT0 to EXWT2 must be set to 1. Simultaneously setting more than one of EXWT0 to EXWT2 to 1 is not expected in the design of this LSI, and if attempted correct operation is not guaranteed.

2. When bit V = 0, the settings in EXWT0 to EXWT2 are ignored. In area 0, this register setting is ignored in read access when the burst ROM interface is selected. For details on wait control, refer to section 6B.6.1, SRAM Interface (Basic Functionality).

6B.5.10 Area 1 External Wait Control Register (CSPWCR1)

Function: CSPWCR1 makes settings for external wait input pins during access to area 1 (EX_BUS).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	V	RB	WINV	EXWT2	EXWT1	EXWT0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	V	0	R/W	Area 1 External Wait Signal Enable/Disable 0: Disabled 1: Enabled
4	RB	0	R/W	Area 1 READY/BUSY Logic Selection 0: BUSY logic 1: READY logic
3	WINV	0	R/W	Area 1 External Wait Signal Polarity 0: Does not invert the polarity of the area 1 external wait signal. 1: Inverts the polarity of the area 1 external wait signal.
2	EXWT2	0	R/W	Area 1 EX_WAIT2 Enable 0: Disables EX_WAIT2 for area 1. 1: Enables EX_WAIT2 for area 1.
1	EXWT1	0	R/W	Area 1 EX_WAIT1 Enable 0: Disables EX_WAIT1 for area 1. 1: Enables EX_WAIT1 for area 1.

Bit	Bit Name	Initial Value	R/W	Description
0	EXWT0	0	R/W	Area 1 EX_WAIT0 Enable 0: Disables EX_WAIT0 for area 1. 1: Enables EX_WAIT0 for area 1.

Notes: 1. When this register setting is made valid (bit V = 1), any one of bits EXWT0 to EXWT2 must be set to 1. Simultaneously setting more than one of EXWT0 to EXWT2 to 1 is not expected in the design of this LSI, and if attempted correct operation is not guaranteed.

2. When bit V = 0, the settings in EXWT0 to EXWT2 are ignored. For details on wait control, refer to section 6B.6.1, SRAM Interface (Basic Functionality).

6B.5.11 Expansion Area x External Wait Control Register (ECSPWCRx (x = 0 to 5))

Function: ECSPWCRx makes settings for external wait input pins during access to expansion area x (EX_BUS).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	V	RB	WINV	EXWT2	EXWT1	EXWT0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	V	0	R/W	Expansion Area x External Wait Signal Enable/Disable 0: Disabled 1: Enabled
4	RB	0	R/W	Expansion Area x READY/BUSY Logic Selection 0: BUSY logic 1: READY logic
3	WINV	0	R/W	Expansion Area x External Wait Signal Polarity 0: Does not invert the polarity of the expansion area x external wait signal. 1: Inverts the polarity of the expansion area x external wait signal.
2	EXWT2	0	R/W	Expansion Area x EX_WAIT2 Enable 0: Disables EX_WAIT2 for expansion area x. 1: Enables EX_WAIT2 for expansion area x.
1	EXWT1	0	R/W	Expansion Area x EX_WAIT1 Enable 0: Disables EX_WAIT1 for expansion area x. 1: Enables EX_WAIT1 for expansion area x.

Bit	Bit Name	Initial Value	R/W	Description
0	EXWT0	0	R/W	Expansion Area x EX_WAIT0 Enable 0: Disables EX_WAIT0 for expansion area x. 1: Enables EX_WAIT0 for expansion area x.

Notes: 1. When this register setting is made valid (bit V = 1), any one of bits EXWT0 to EXWT2 must be set to 1. Simultaneously setting more than one of EXWT0 to EXWT2 to 1 is not expected in the design of this LSI, and if attempted correct operation is not guaranteed.

2. When bit V = 0, the settings in EXWT0 to EXWT2 are ignored. For details on wait control, refer to section 6B.6.1, SRAM Interface (Basic Functionality).

6B.5.12 External Wait Input Control Register (EXWTSYNC)

Function: EXWTSYNC controls whether or not to synchronize the external wait pins EX_WAIT[2:0].

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	EXWT SYNC2	EXWT SYNC1	EXWT SYNC0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	EXWTSYNC2	0	R/W	0: Does not synchronize EX_WAIT2 (the original EX_WAIT2 is synchronous with CLKOUT). 1: Synchronizes EX_WAIT2 (the original EX_WAIT2 is asynchronous with CLKOUT).
1	EXWTSYNC1	0	R/W	0: Does not synchronize EX_WAIT1 (the original EX_WAIT1 is synchronous with CLKOUT). 1: Synchronizes EX_WAIT1 (the original EX_WAIT1 is asynchronous with CLKOUT).
0	EXWTSYNC0	0	R/W	0: Does not synchronize EX_WAIT0 (the original EX_WAIT0 is synchronous with CLKOUT). 1: Synchronizes EX_WAIT0 (the original EX_WAIT0 is asynchronous with CLKOUT).

Note: For details on wait control, refer to section 6B.6.1, SRAM Interface (Basic Functionality).

6B.5.13 Area 0 Burst Control Register (CS0BSTCTL)

Function: CS0BSTCTL specifies the burst length for area 0 when the burst ROM interface is selected.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	A0BST[2:0]			—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13 to 11	A0BST[2:0]	000	R/W	Area 0 Burst Length for Burst ROM Interface 001: 4 access cycles 010: 8 access cycles 011: 16 access cycles 100: 32 access cycles Others: No burst transfer
10 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Notes: 1. This register is valid only when the burst ROM interface is selected (CS0IF = B'01 in CS0CTRL).
2. Set bits A0BST2 to A0BST0 to an appropriate value so that (area 0 bus size) x (burst length set in this register) becomes 32 bytes or less.
3. For details, refer to section 6B.6.2, CPU (SuperHyway Bus) → Burst ROM Interface.

6B.5.14 Area 0 Burst Pitch Set Register (CS0BTPH)

Function: CS0BTPH specifies the burst pitches for the first access cycle and the second and later cycles for area 0 when the burst ROM interface is selected.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	A0H	A0W[3:0]				—	A0B[2:0]		
Initial value:	0	0	0	0	0	0	0	0	1	1	1	1	0	1	1	1
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	A0H	0	R/W	Specifies the $\overline{CS}$ and address hold cycles with respect to the $\overline{RD}$ signal for area 0 in the burst ROM interface. 0: 0 cycle for hold period 1: 1 cycle for hold period
7 to 4	A0W[3:0]	1111	R/W	These bits specify the burst pitch (wait cycles to be inserted) after the first burst cycle for area 0 in the burst ROM interface. 0000: Setting prohibited 0001: Setting prohibited 0010: 2 cycles 0011: 3 cycles 0100: 4 cycles 0101: 5 cycles 0110: 6 cycles 0111: 7 cycles

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	A0W[3:0]	1111	R/W	1000: 8 cycles 1001: 9 cycles 1010: 10 cycles 1011: 11 cycles 1100: 12 cycles 1101: 13 cycles 1110: 14 cycles 1111: 15 cycles
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2 to 0	A0B[2:0]	111	R/W	These bits specify the burst pitch (wait cycles to be inserted) after the second burst cycle for area 0 in the burst ROM interface. 000: Setting prohibited 001: 1 cycle 010: 2 cycles 011: 3 cycles 100: 4 cycles 101: 5 cycles 110: 6 cycles 111: 7 cycles

Notes: 1. Be sure to specify this register before setting CS0BSTCTL.
2. For details, refer to section 6B.6.2, CPU (SuperHyway Bus) → Burst ROM Interface.

6B.5.15 Area 1 Guard Setting Register (CS1GDST)

Function: CS1GDST specifies the guard interval (period of access prohibition) between sequential access cycles in area 1.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	CS1GD	TIMER_SET			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	CS1GD	0	R/W	0: Makes the TIMER_SET setting invalid. 1: Makes the TIMER_SET setting valid.
3 to 0	TIMER_ SET	0000	R/W	Guard Interval (Period of Access Prohibition) between Sequential Access Cycles for Area 1 0000: 0 clock cycle 0001: 1 clock cycle 0010: 2 clock cycles 0011: 3 clock cycles 0100: 4 clock cycles 0101: 5 clock cycles 0110: 6 clock cycles 0111: 7 clock cycles 1000: 8 clock cycles 1001: 9 clock cycles 1010: 10 clock cycles 1011: 11 clock cycles

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	TIMER_SET	0000	R/W	1100: 12 clock cycles 1101: 13 clock cycles 1110: 14 clock cycles 1111: 15 clock cycles

Notes: 1. The TIMER_SET setting is ignored when CS1GD = 0.
2. This register must not be dynamically modified regardless of whether area 1 is being accessed.
3. The actual guard interval between sequential access cycles on the EX_BUS is (register setting) + (idle cycles due to conflicts).
4. For details, refer to section 6B.6.1 (4), Controlling Guard Intervals.

6B.5.16 Expansion Area x Guard Setting Register (ECSxGDST (x = 0 to 5))

Function: ECSxGDST specifies the guard interval (period of access prohibition) between sequential access cycles in expansion area 0 to 5.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	ECSx GD	TIMER_SET			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	ECSxGD	0	R/W	0: Makes the TIMER_SET setting invalid. 1: Makes the TIMER_SET setting valid.

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	TIMER_SET	0000	R/W	Guard Interval (Period of Access Prohibition) between Sequential Access Cycles for Expansion Area x 0000: 0 clock cycle 0001: 1 clock cycle 0010: 2 clock cycles 0011: 3 clock cycles 0100: 4 clock cycles 0101: 5 clock cycles 0110: 6 clock cycles 0111: 7 clock cycles 1000: 8 clock cycles 1001: 9 clock cycles 1010: 10 clock cycles 1011: 11 clock cycles 1100: 12 clock cycles 1101: 13 clock cycles 1110: 14 clock cycles 1111: 15 clock cycles

Notes: 1. The TIMER_SET setting is ignored when ECSxGD = 0.
2. This register must not be dynamically modified regardless of whether the expansion area is being accessed.
3. The actual guard interval between sequential access cycles on the EX_BUS is (register setting) + (idle cycles due to conflicts).
4. For details, refer to section 6B.6.1 (4), Controlling Guard Intervals.

6B.5.17 LBSC-DMAC Channel y Area Assignment Register (EXDMASETy (y = 0 to 2))

Function: EXDMASETy specifies the area where LBSC-DMAC channels 0 to 2 is assigned.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	DMy ECS5	DMy ECS4	DMy ECS3	DMy ECS2	DMy ECS1	DMy ECS0	DMy CS1	DMy CS0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	DMyECS5	0	R/W	0: Does not assign LBSC-DMAC channel y to expansion area 5. 1: Assigns LBSC-DMAC channel y to expansion area 5.
6	DMyECS4	0	R/W	0: Does not assign LBSC-DMAC channel y to expansion area 4. 1: Assigns LBSC-DMAC channel y to expansion area 4.
5	DMyECS3	0	R/W	0: Does not assign LBSC-DMAC channel y to expansion area 3. 1: Assigns LBSC-DMAC channel y to expansion area 3.
4	DMyECS2	0	R/W	0: Does not assign LBSC-DMAC channel y to expansion area 2. 1: Assigns LBSC-DMAC channel y to expansion area 2.

Bit	Bit Name	Initial Value	R/W	Description
3	DMyECS1	0	R/W	0: Does not assign LBSC-DMAC channel y to expansion area 1. 1: Assigns LBSC-DMAC channel y to expansion area 1.
2	DMyECS0	0	R/W	0: Does not assign LBSC-DMAC channel y to expansion area 0. 1: Assigns LBSC-DMAC channel y to expansion area 0.
1	DMyCS1	0	R/W	0: Does not assign LBSC-DMAC channel y to area 1. 1: Assigns LBSC-DMAC channel y to area 1.
0	DMyCS0	0	R/W	0: Does not assign LBSC-DMAC channel y to area 0. 1: Assigns LBSC-DMAC channel y to area 0.

Notes:
1. Setting more than one bit to 1 in this register is prohibited. Such a setting is not expected in the design of this LSI, and if attempted correct operation is not guaranteed.
2. For an expansion area whose capacity is 0 Mbytes, DMAC channel must not be assigned.
3. Be sure to specify this register before starting LBSC-DMAC access. If an LBSC-DMAC channel starts access before specifying this register, correct operation is not guaranteed (such access is not expected in the design of this LSI).
4. When CS0 area is used as a 128-Mbyte space, DMAC channel must not be assigned to CS1 area.
5. For details, refer to section 6B.6.3, LBSC-DMAC → DMA Interface.

6B.5.18 LBSC-DMAC Channel y Control Register (EXDMCRy (y = 0 to 2))

Function: EXDMCRy specifies conversion of DREQ[y], DACK[y], and DRACK[0] in the area where LBSC-DMAC channel y is assigned.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DRST	—	DSTS	DBST	—	EXQL	EXDY	EXDS	—	—	EXRS	EXRL	—	EXAL	DAKCTL	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	R	R	R/W	R	R/W	R/W	R/W	R	R	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15	DRST	0	W	DACK Signal Forcible Negation (Enabled when DBST = 1) 0: Writing 0 is ignored. 1: Forcibly negates the DACK signal being continuously asserted
14	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
13	DSTS	0	R	DACK Signal Assert State Indication 0: The DACK signal is not currently asserted. 1: The DACK signal is currently asserted.
12	DBST	0	R/W	Specifies whether or not to continuously assert the DACK signal if DREQ is continuously asserted during the intervals between DMA bus transfers (in ATA mode only). 0: Negates DACK after each bus transfer. 1: Continuously asserts DACK even during the intervals between bus transfers if DREQ is continuously asserted. (When negation of DREQ is detected, DACK is also negated.)

Bit	Bit Name	Initial Value	R/W	Description
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	EXQL	0	R/W	0: Receives DREQ[y] signal at a low level. 1: Receives DREQ[y] signal at a high level.
9	EXDY	0	R/W	0: Does not synchronize the DREQ[y] signal. 1: Synchronizes the DREQ[y] signal.
8	EXDS	0	R/W	0: Detects DREQ[y] signal at a level. 1: Detects DREQ[y] signal at an edge.
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	EXRS	0	R/W	0: Asserts DRACK[0] one clock cycle before $\overline{CS}$/DACK[0] is asserted. 1: Asserts DRACK[0] two clock cycles before $\overline{CS}$/DACK[0] is asserted.
4	EXRL	0	R/W	0: Outputs DRACK[0] as a high-active signal. 1: Outputs DRACK[0] as a low-active signal.
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	EXAL	0	R/W	0: Outputs DACK[y] as a high-active signal. 1: Outputs DACK[y] as a low-active signal.
1, 0	DAKCTL	00	R/W	Signals asserted for area where LBSC-DMAC channel y is assigned 00 and 11: Asserts the $\overline{CS}$ signal and DACK[y] signal together for the area. 01: Asserts only the $\overline{CS}$ signal for the area. 10: Asserts only the DACK[y] signal for the area.

Notes: 1. This register must not be dynamically modified regardless of whether the respective area is being accessed.

2. The EXRL and EXRS settings are valid only for LBSC-DMAC channel 0 (DRACK[0]). LBSC-DMAC channels 1 and 2 do not have the DRACK signal.

3. For details on the DMA interface, refer to section 6B.6.3, LBSC-DMAC → DMA Interface.

6B.5.19 BSC Interrupt Source Status Register (BCINTSR)

Function: BCINTSR indicates the state of the BSC interrupt source.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ATTE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	ATTE	0	R	ATA Wait Timeout Error Status 0: The ATA interface is working correctly. 1: A timeout error has occurred in the ATA interface. (A timeout error occurs when 100 EX_BUS clock (CLKOUT) cycles of wait time have elapsed.)

Notes: 1. No interrupt signal is output with only the ATTE bit being set to 1 in this register. To output an interrupt signal, BCINTMR must be appropriately specified.

2. For details on the ATA interface, refer to section 6B.6.5, CPU (SuperHyway Bus) → ATA Device Interface.

6B.5.20 BSC Interrupt Source Clear Register (BCINTCR)

Function: BCINTCR clears the state of the BSC interrupt indicator.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ATTEC
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	—/W*

Note: Write-only. The read value is undefined.

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	ATTEC	0	—/W*[1]	ATA Wait Timeout Error Status Clear 0: Writing 0 is ignored. 1: Clears the ATA wait timeout error status.

Notes: 1. Write-only. The read value is undefined.
2. This register is always read as 0.
3. For details on the ATA interface, refer to section 6B.6.5, CPU (SuperHyway Bus) → ATA Device Interface.

6B.5.21 BSC Interrupt Enable Register (BCINTMR)

Function: BCINTMR enables or disables the BSC interrupt.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ATTEM
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	ATTEM	0	R/W	ATA Wait Timeout Error Interrupt Enable 0: Disables output of an interrupt signal for this interrupt source. 1: Enables output of an interrupt signal for this interrupt source.

6B.5.22 EX_BUS Priority Level Set Register (EXBATLV)

Function: Specifies the priority levels for EX_BUS arbitration.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	EX-BLV
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	EX-BLV	0	R/W	Priority Level Setting for EX_BUS Arbitration (Access Selection) 0: Higher priority: PIO (SuperHyway access) Lower priority: LBSC-DMAC 1: Higher priority: LBSC-DMAC Lower priority: PIO (SuperHyway access)

Notes: 1. EX-BLV sets a fixed priority between PIO and LBSC-DMAC.
2. This register must not be dynamically modified except for initial setting.
3. For details on external bus arbitration, refer to section 6B.6.6, EX_BUS Arbitration.

6B.5.23 External Wait Status Register (EXWTSTS)

Function: EXWTSTS indicates the state of the external wait pin EX_WAIT.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	EXWT2 STS	EXWT1 STS	EXWT0 STS
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	EXWT2 STS	—	R	Indicates the EX_WAIT2 pin state.
1	EXWT1 STS	—	R	Indicates the EX_WAIT1 pin state.
0	EXWT0 STS	—	R	Indicates the EX_WAIT0 pin state.

6B.5.24 ATACS Control Register (ATACSCTRL)

Function: ATACSCTRL specifies the $\overline{\text{ATACS0}}$ and $\overline{\text{ATACS1}}$ signal settings.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	ATAECS5_EN	ATAECS4_EN	ATAECS3_EN	ATAECS2_EN	ATAECS1_EN	ATAECS0_EN	ATAECS1_EN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6	ATAECS5_EN		R/W	Selects assertion or deassertion of the ATACS signal when expansion area 5 is accessed in ATA mode 0: The ATACS signal is deasserted. 1: The ATACS signal is asserted.
5	ATAECS4_EN	0	R/W	Selects assertion or deassertion of the ATACS signal when expansion area 4 is accessed in ATA mode 0: The ATACS signal is deasserted. 1: The ATACS signal is asserted.
4	ATAECS3_EN	0	R/W	Selects assertion or deassertion of the ATACS signal when expansion area 3 is accessed in ATA mode 0: The ATACS signal is deasserted. 1: The ATACS signal is asserted.
3	ATAECS2_EN	0	R/W	Selects assertion or deassertion of the ATACS signal when expansion area 2 is accessed in ATA mode 0: The ATACS signal is deasserted. 1: The ATACS signal is asserted.
2	ATAECS1_EN	0	R/W	Selects assertion or deassertion of the ATACS signal when expansion area 1 is accessed in ATA mode 0: The ATACS signal is deasserted. 1: The ATACS signal is asserted.

Bit	Bit Name	Initial Value	R/W	Description
1	ATAECS0_EN	0	R/W	Selects assertion or deassertion of the ATACS signal when expansion area 0 is accessed in ATA mode 0: The ATACS signal is deasserted. 1: The ATACS signal is asserted.
0	ATACS1_EN	0	R/W	Asserts or deasserts the ATACS signal when area 1 is accessed in ATA mode 0: The ATACS signal is deasserted. 1: The ATACS signal is asserted.

Note: If the ATA mode is selected with the CS1IF bit of the CS1CTRL register and the ECSxIF bit of the ECSxCTRL register, and the ATACS1_EN and ATAECSx_EN bits of this register are set to 1, $\overline{\text{ATACS0}}$ or $\overline{\text{ATACS1}}$ is asserted in accord with the value being output as address bit 4 on the external bus. For details, refer to section 6B.6.5, CPU (SuperHyway Bus) → ATA Device Interface.

6B.6 Operation

6B.6.1 SRAM Interface (Basic Functionality)

The BSC reads access requests from the CPU stored in the FIFO in the SuperHyway target port control block and writes them to the EX_BUS. By default, all spaces, areas 0 and 1, and expansion areas 0 to 5 are all set for SRAM interface. The pulse width for access signals in this SRAM access interface can be varied according to register settings. Thus, the SRAM interface has functionality for easily accommodating devices with various access specifications, connected on the EX_BUS. In addition, to support low-speed external devices, the guard-interval control functionality is provided for insertion of the appropriate interval for each bus access; and the SRAM interface receives wait-for-response requests (or access complete signals) from external devices on a synchronous/asynchronous-selectable and polarity-selectable basis, thus ensuring flexibility in bus design. Figures 6B.3 and 6B.4 show SRAM interface timing charts for SuperHyway → EX_BUS conversions.

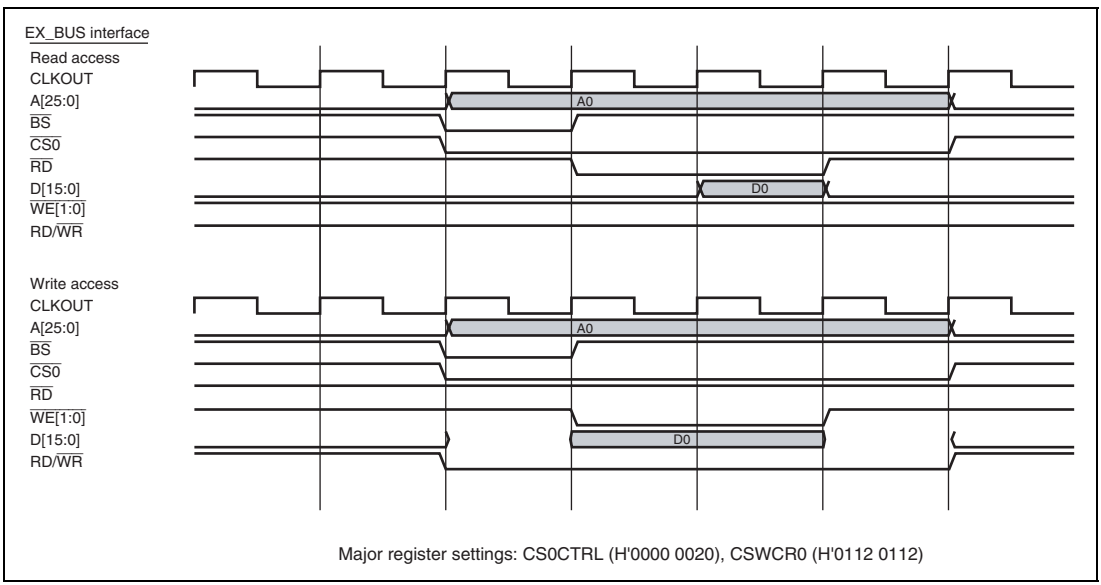


Figure 6B.3 Basic Timing Chart for Access from SuperHyway to SRAM (Area 0)

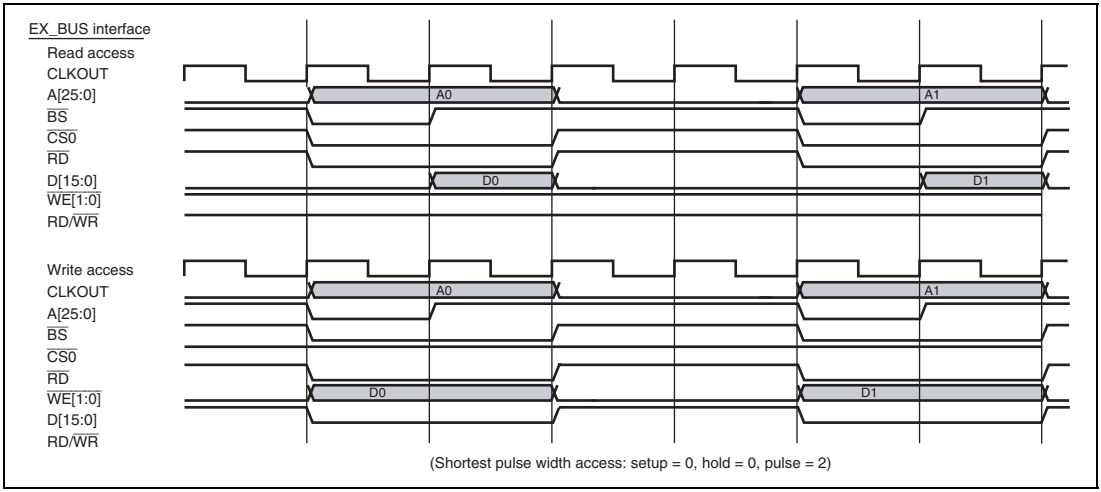


Figure 6B.4 Basic Timing Chart for Access from SuperHyway to SRAM (Shortest Pulse Width × Shortest PIO Consecutive Accesses)

Figure 6B.4 is an example of the basic SRAM interface waveform with the shortest pulse width, wherein the same waveform occurs twice in succession.

If consecutive PIO access requests are made from the CPU (SuperHyway bus), the access interval will be 2 clock cycles, as indicated in the above waveforms, regardless of whether the preceding or succeeding access request is for the same area or for different areas. If a switching occurs, such as PIO → DMA or DMA → PIO, however, the access interval will be 1 clock cycle.

The access interval can be extended by means of the guard setting registers.

(1) Address Generation/Alignment

When making access to the EX_BUS as an SRAM interface, the LBSC generates addresses and performs data alignment conversions in addition to determining which area is to be accessed according to how the expansion area space is partitioned by register settings. Figure 6B.5 shows the method by which the expansion area is partitioned. Figures 6B.6 and 6B.7 provide an overview of address generation and data alignment/write enable conversions.

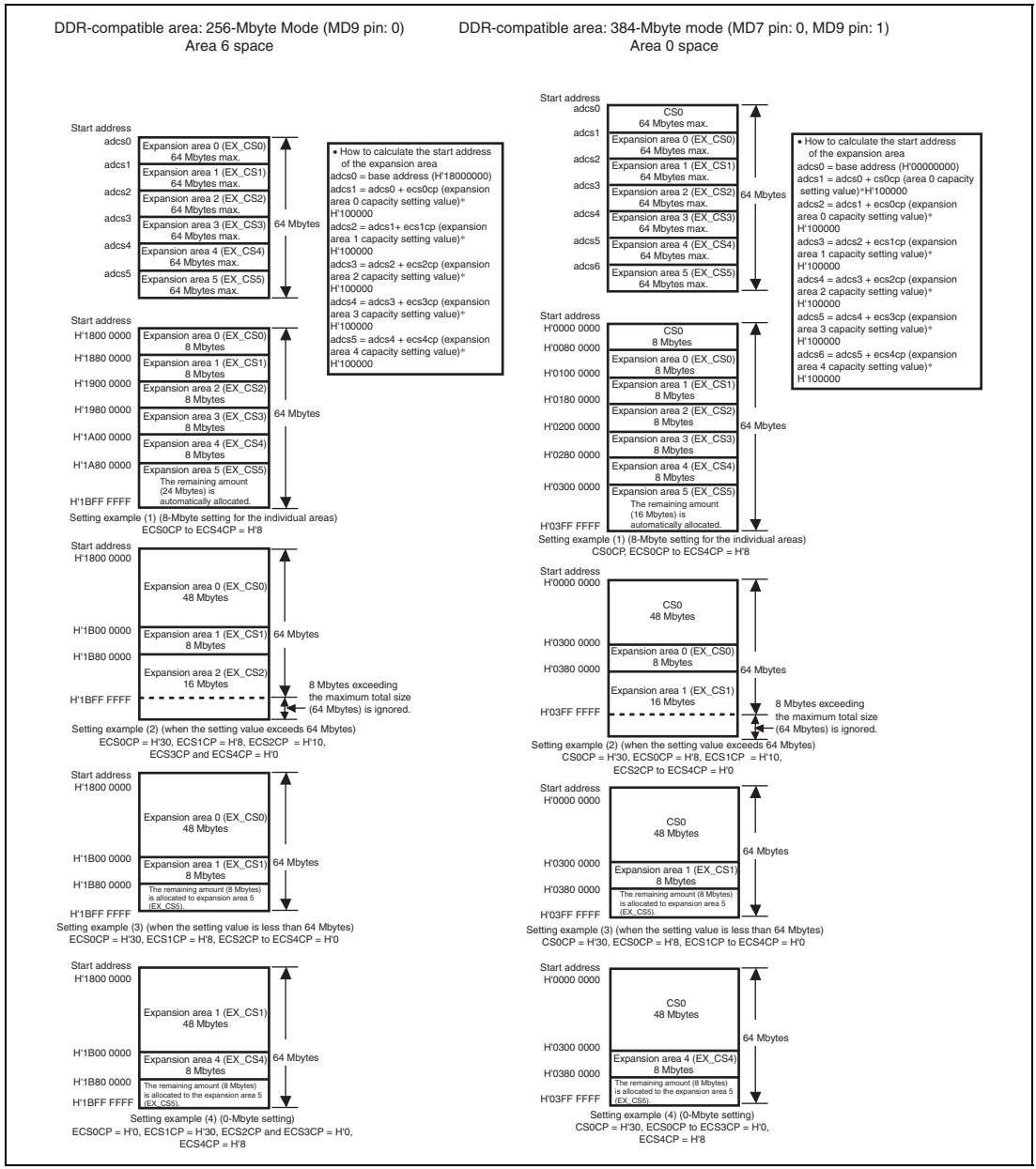
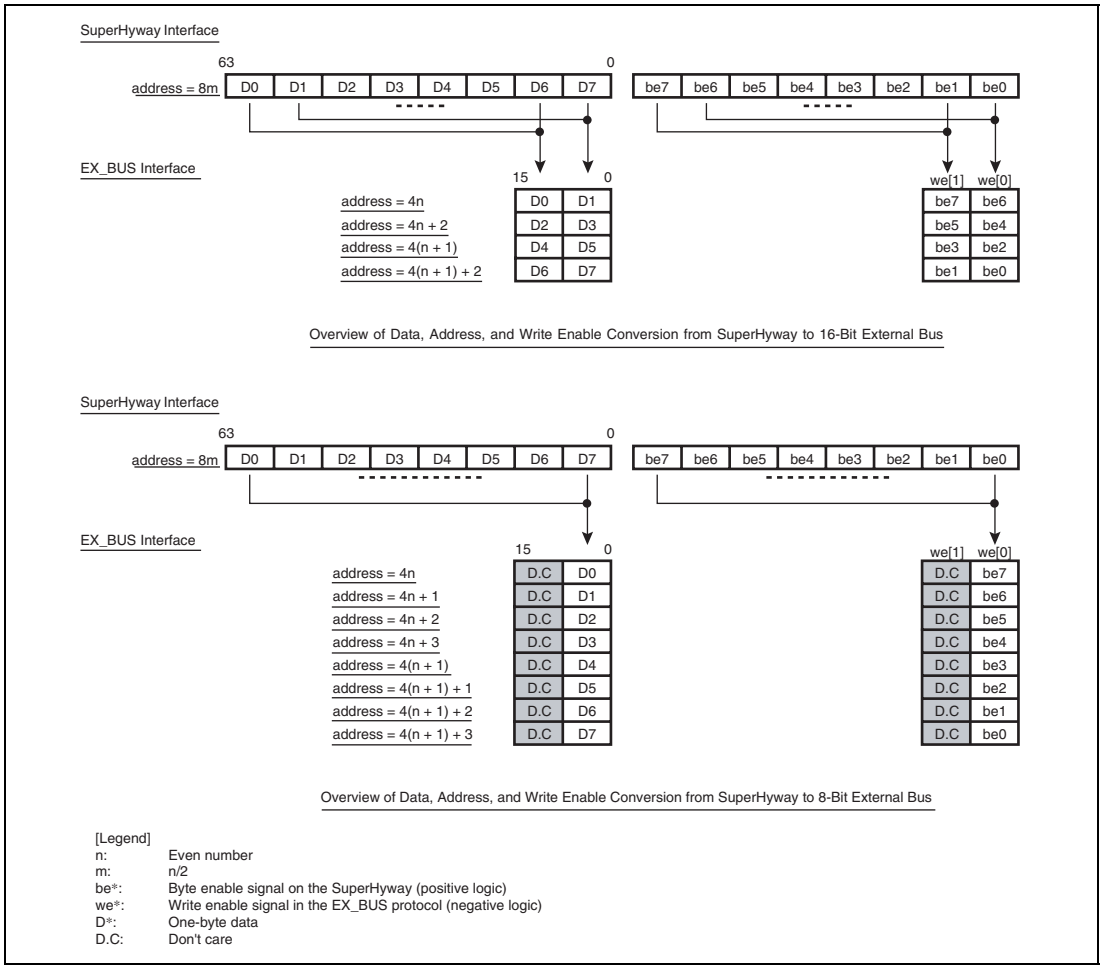
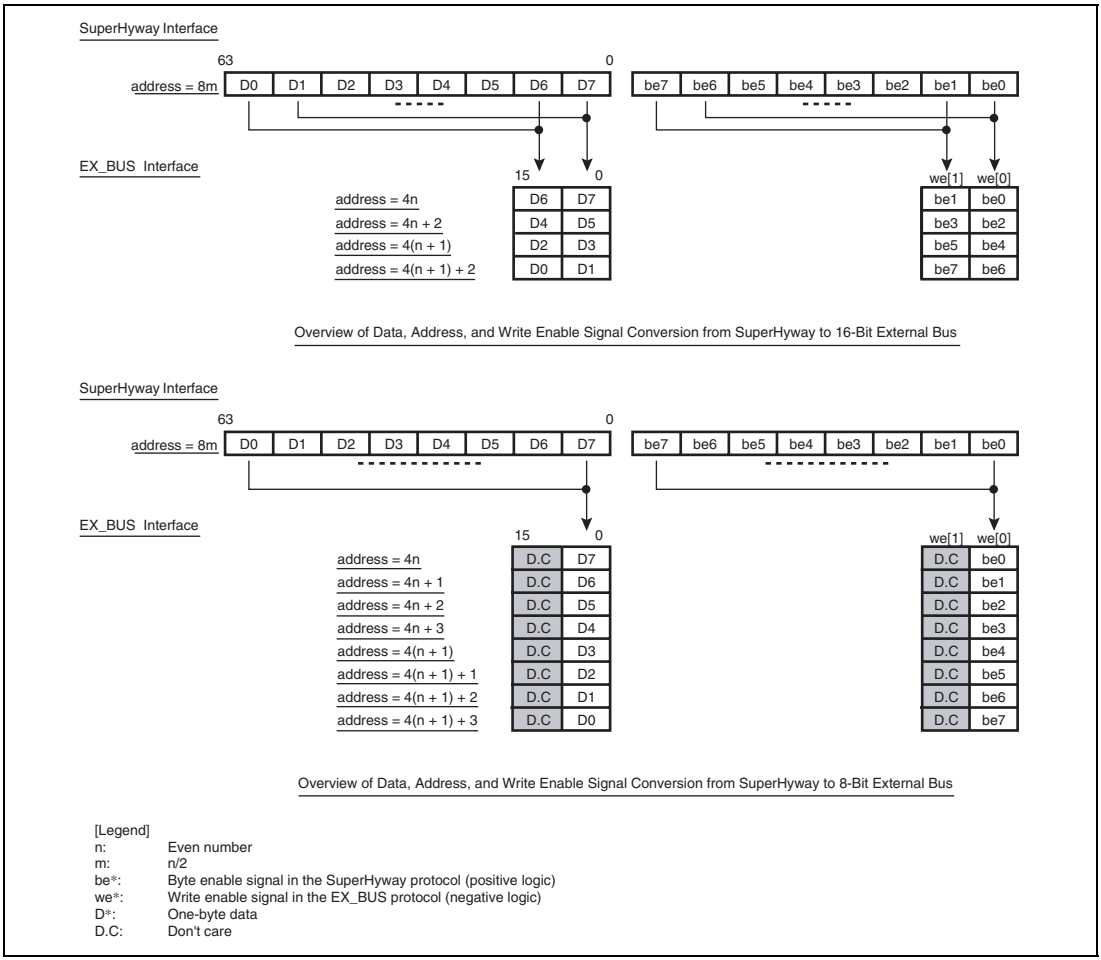


Figure 6B.5 Expansion Area Partition



[Legend]
n: Even number
m: n/2
be*: Byte enable signal on the SuperHyway (positive logic)
we*: Write enable signal in the EX_BUS protocol (negative logic)
D*: One-byte data
D.C: Don't care

Figure 6B.6 Overview of Data, Address, and Write Enable Signal Conversion from CPU (64-Bit SuperHyway Bus) to External Bus (little = 0)



[Legend]
n: Even number
m: n/2
be*: Byte enable signal in the SuperHyway protocol (positive logic)
we*: Write enable signal in the EX_BUS protocol (negative logic)
D*: One-byte data
D.C: Don't care

Figure 6B.7 Overview of Data, Address, and Write Enable Signal Conversion from CPU (64-Bit SuperHyway Bus) to External Bus (little = 1)

(2) Setting the Pulse Width for an Access Signal

When making access to the EX_BUS, the LBSC can set the setup or hold time for addresses, RD/$\overline{WR}$, and $\overline{CS}$ signals based on $\overline{WE}$ signals and $\overline{RD}$ signals, and the pulse width for $\overline{WE}$ and $\overline{RD}$ signals in units of clock cycles, according to the values that are set for each area by means of the CSWCR0, CSWCR1, or ECSWCR0 to ECSWCR5 registers. (During a burst ROM read operation, however, the LBSC conforms to CS0BTPH settings rather than CSWCR0 settings. Similarly, during a DMA operation, it conforms to EXDMAWCR0 to EXDMAWCR2 settings rather than CSWCR0, CSWCR1, or ECSWCR0 to ECSWCR5 settings.) The pulse width for $\overline{WE}$ or $\overline{RD}$ signal that is stored in a register can be extended by an externally supplied EX_WAIT signal. The minimum total value that is set should be two clock cycles.

(3) External Wait Control

The LBSC controls the external wait signal (EX_WAIT) from a device connected to the EX_BUS based on settings that are provided on external wait control registers (CSPWCR0, CSPWCR1, or ECSPWCR0 to ECSPWCR5) and the external wait input control register (EXWTSYNC). The external wait control registers permit the selection of the four types of interfaces to accommodate various types of specifications, whether the wait signal input from an external device is based on the READY logic (posting a READY status) or BUSY logic (posting a BUSY status), or which signal polarity is in effect.

Figure 6B.8 shows waveforms for the four wait signal patterns that can be input and the waveforms for the wait signals internal to the LBSC after conversion according to the register settings.

The external wait input control register allows the switching between the synchronous/asynchronous handling of external wait input signals. The default is to treat such signals on a synchronous basis. Figure 6B.9 shows external wait input timings for synchronization/asynchronization. In the figure, the position indicated by the symbol * represents the point at which the LBSC determines whether or not external wait is in effect. Specifically, for synchronization, the position is one clock cycle before the point at which the $\overline{WE}$ and $\overline{RD}$ signals would normally be negated by pulse width settings. If external wait is in effect at this position, the pulse width for the $\overline{WE}$ or $\overline{RD}$ signal continues to extend until the wait status becomes the ready status. If the pulse width for the $\overline{WE}$ or $\overline{RD}$ signal is extended by the EX_WAIT signal, the address and the RD/$\overline{WR}$ and $\overline{CS}$ signals are negated when the hold time is satisfied. If the synchronization setting is switched to the asynchronization setting, any external wait is nullified unless there is an external wait two clock cycles before the * position, or three clock cycles before the point in which the $\overline{WE}$ and $\overline{RD}$ signals would normally be negated.

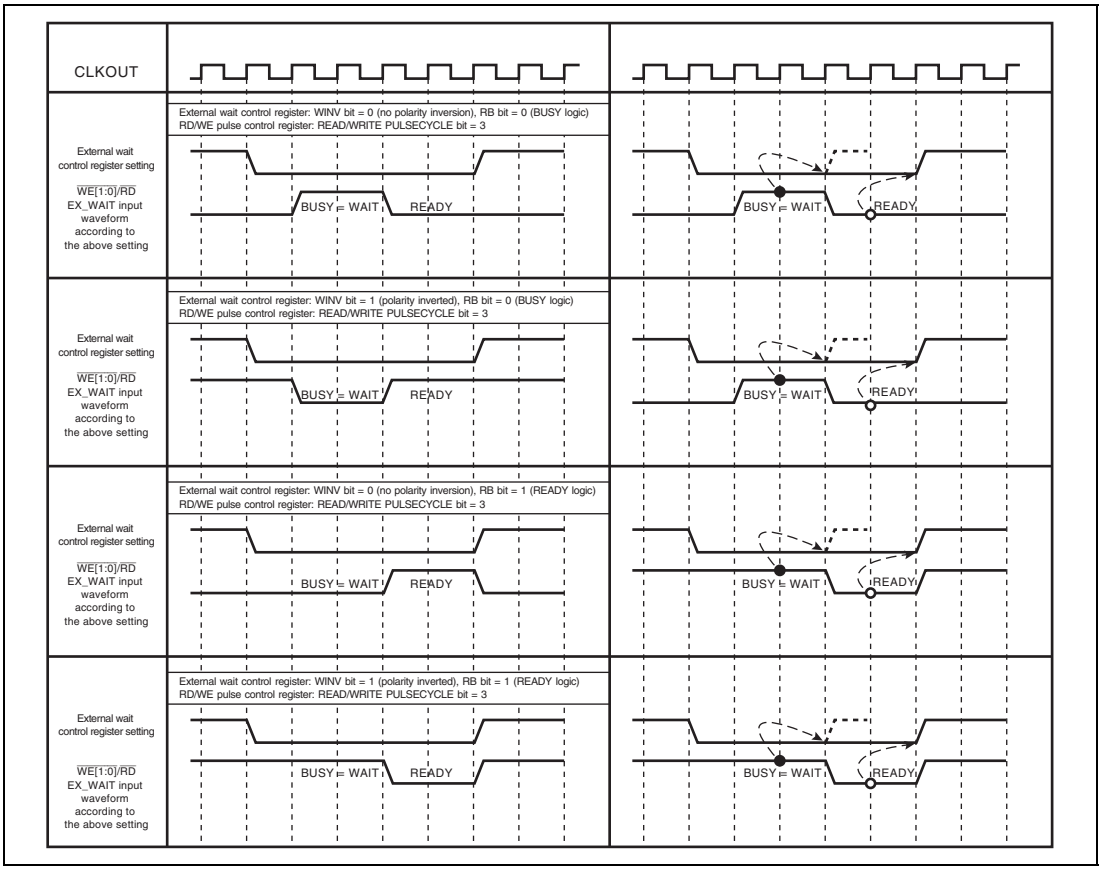


Figure 6B.8 Waveforms of Converted External Wait Interface

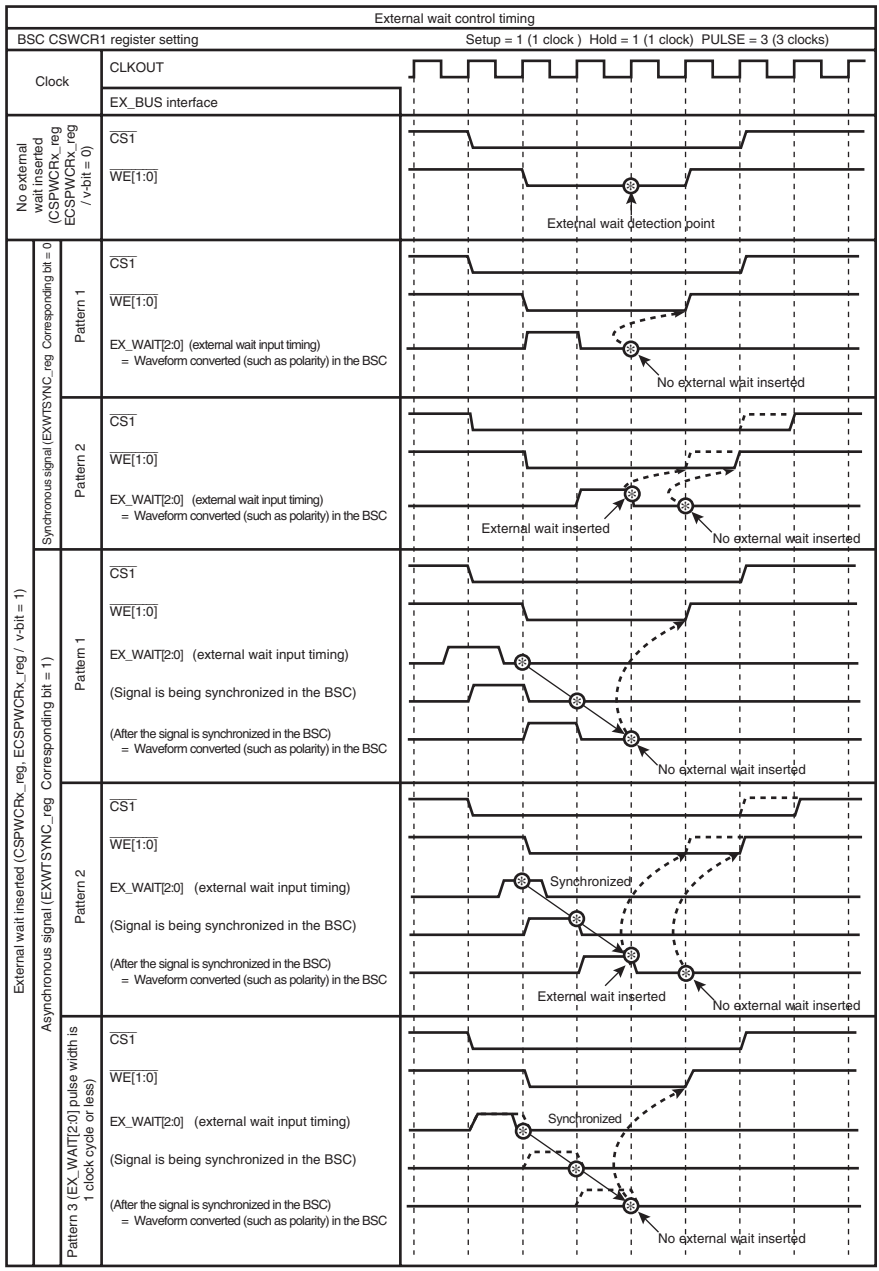


Figure 6B.9 Waveforms of External Wait Input Signals

(4) Controlling Guard Intervals

For some external devices, the transition of the data lines to the high-Z state after completion of read access takes a relatively long time. To prevent any contention for the data bus when further access over the EX_BUS proceeds immediately after a read operation for such a device, the LBSC supports a function to guard against further access over a given interval following each bus access. The function is also effective for external devices that have difficulty coping with consecutive reception of access requests. Specifically, set the number of clock cycles over which guarding against access applies in the guard setting register (CS1GDST, ECS0GDST to ECS5GDST) for the given area. (Even if the value assigned to the guard setting register is 0, hardware processing before a further bus access request is issued requires at least 2 clock cycles between PIO and PIO, and at least 1 clock cycle between PIO and DMA. Therefore, the actual interval on the EX_BUS is the time equivalent to either 1 or 2 clock cycles plus the setting for number of clock cycles in the guard setting register.)

After access to a given area, the guard interval is assigned corresponding to the register for that area, regardless of the kind of access or target space of the next access request. Note, however, that there is no guard setting register for area 0. Therefore, guard intervals are not set up for access that immediately follows access to this area. Since area 0 is supposed to be for the connection of general-purpose memory such as ROM, SRAM, or flash-ROM, guard intervals should not be necessary for this area. Figure 6B.10 is a basic timing chart to illustrate the concept of guard-interval control.

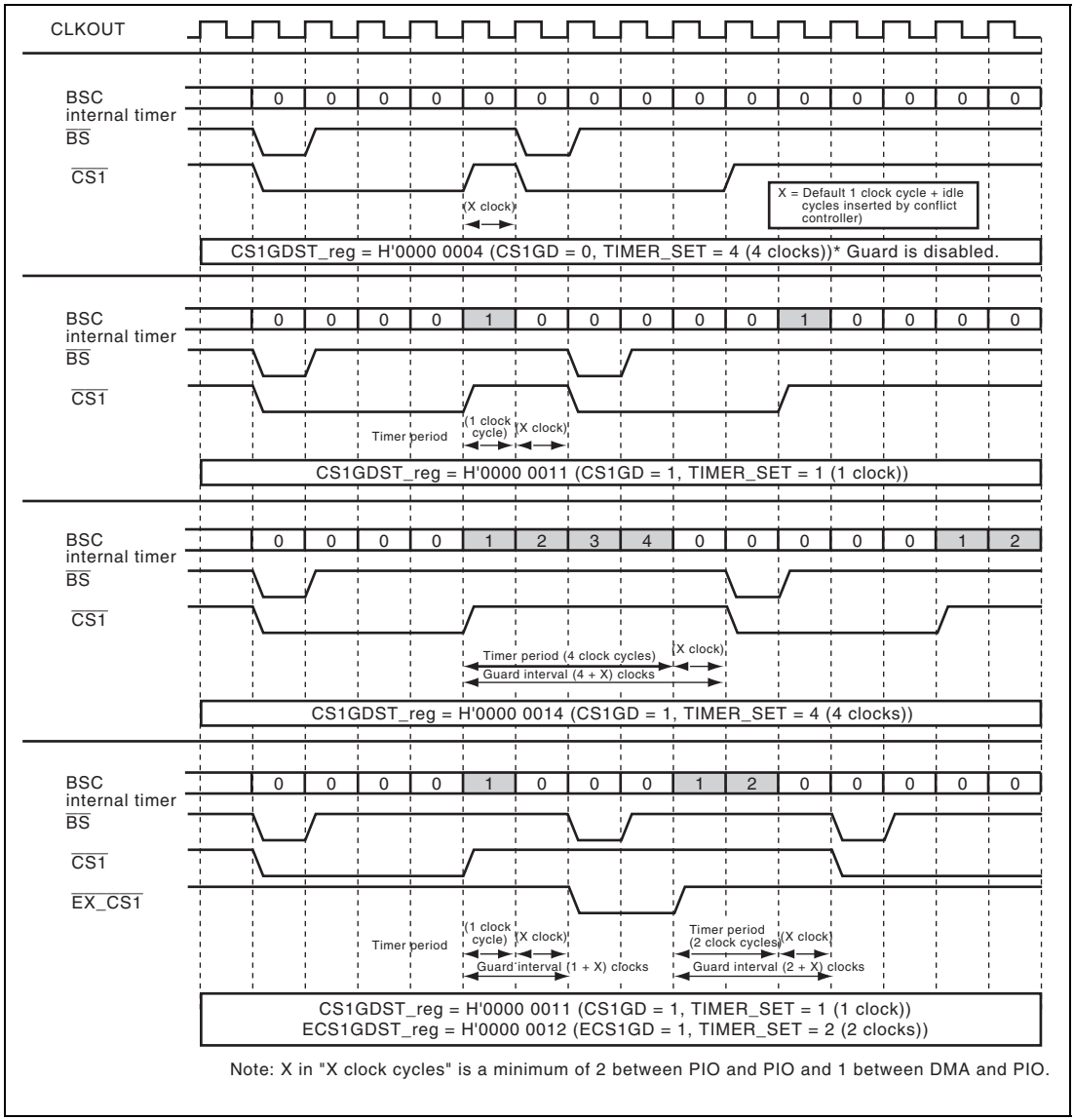


Figure 6B.10 Concept of Guard-Interval Control

6B.6.2 CPU (SuperHyway Bus) → Burst ROM Interface

For area 0 on the EX_BUS, the BSC supports a page-mode read burst ROM interface. Switching to the burst ROM interface is performed by CS0CTRL settings. Because burst ROM access operations begin immediately after setting CS0CTRL, values must be pre-assigned to the CS0BTPH and CS0BSTCTL. In CS0BTPH, burst ROM access waveforms can be assigned, such as a first-cycle burst pitch, second and subsequent cycle burst pitches, and $\overline{CS}$ signal hold cycles on the $\overline{RD}$ signal. A burst access count (burst length) can be assigned to CS0BSTCTL. A burst access count should be assigned according to the component specifications for the actual external ROM that is connected.

In two kinds of exceptional cases, however, the BSC handles access with a burst count less than the specified burst count. The first are those cases where the CPU requests access to fewer data than the specified burst count. In such cases, access is terminated on completion of processing for the number of bytes requested by the CPU, even though the specified burst count has not been reached. The second are those cases where a variable burst address boundary is crossed in the midst of a specified burst count. In such cases, access is terminated immediately before the boundary is crossed, and the burst access is divided up, with the remainder of the access executed in a second round. The reason for this behavior is that the variable address boundaries for burst ROM devices, by which the LBSC must abide, are defined. Table 6B.3 shows the LBSC's methodology for breaking off burst access.

Table 6B.3 Methodology for Breaking Off Bust Access

Burst Count Assigned in CS0BSTCTL	Bus Width	
	8-Bit Bus Width	16-Bit Bus Width
4	Break at a change of A2	Break at a change of A3 (A0 not connected)
8	Break at a change of A3	Break at a change of A4 (A0 not connected)
16	Break at a change of A4	Break at a change of A5* (A0 not connected)
32	Break at a change of A5*	(Setting prohibited)

Note: * Since A5 never changes in the midst of a 32-byte burst operation (a 32-byte boundary), access is not broken off in the case indicated by *, and the specified number of access operations proceed.

As described above, in burst ROM access, the LBSC performs a maximum of 32-byte access to accommodate CPU cache fill while performing access breakup. However, because this operation involves access on 32-byte boundaries, for setting a burst ROM device, the user should specify wrap-around settings instead of continuous burst operations.

Depending on the mode of burst ROM connection, burst operations can also be accommodated in the synchronous burst mode. When using the LBSC in synchronous burst mode, the user should not receive the Wait signals that are output by the burst ROM. Instead, a first-word data output latency should be assigned to the configuration register of the burst ROM, and CS0BTPH should be set as the pitch for the first word and this latency. For the second and subsequent words, a pitch count of 2 should be assigned to both the configuration register of the burst ROM and CS0BTPH. Subject to the AC characteristics of the burst ROM and given frequency for the EX_BUS, if a pitch count of 1 is assigned to, data reception timing can lag behind due to the fact that the burst ROM has a data output delay of approximately 13ns. Figure 6B.11 shows a basic timing chart for the burst ROM interface.

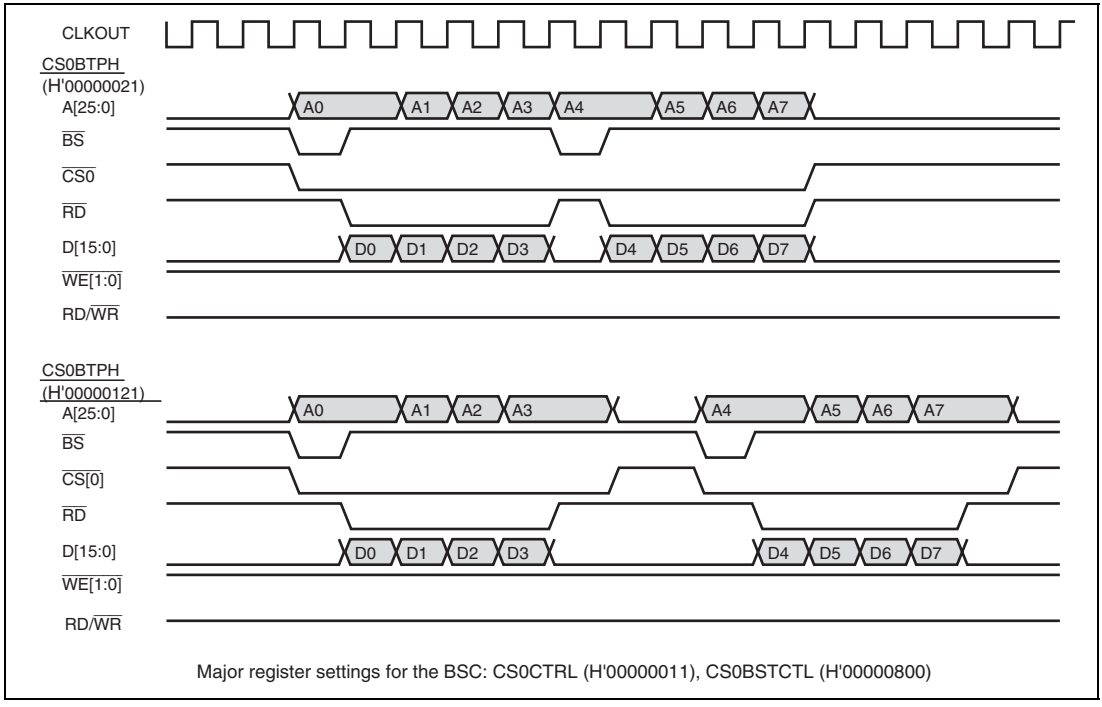


Figure 6B.11 Timing Chart for Burst ROM Interface

6B.6.3 LBSC-DMAC → DMA Interface

On the EX_BUS, three LBSC-DMAC channels (0 to 2) are assigned as DMACs that have external pins DREQ, DRACK, and DACK to support DMA transfer operations. Among these, channels 0 and 1 also support DMA with a burst length of 8.

All DMA channels operate in dual address mode. Table 6B.4 provides a list of DMA pins by channel.

When operating a DMAC channel, first each DMAC should be assigned to the applicable area by using area assignment registers (EXDMASET0 to EXDMASET2) for the LBSC-DMAC channels. Also, in RD/WE pulse control registers (EXDMAWCR0 to EXDMAWCR2) for LBSC-DMAC channels, waveform pulse widths to be used during DMA bus access should be assigned (LBSC-DMAC channels 0 to 2 operate by the settings provided in EXDMAWCR0 to EXDMAWCR2, not by pulse width settings that are provided in CSWCR0, CSWCR1 or ECSWCR0 to ECSWCR5. In addition, methods by which DREQ, DACK, and DRACK signals are to be sent and received should be specified with the LBSC-DMAC channel control registers (EXDMCR0 to EXDMCR2).

Table 6B.4 List of DMA Pins by Channel

DMAC Channel Number	Operating Mode	Pins			Remarks
		DREQ	DRACK	DACK	
Channel 0	Dual address mode (with 8-burst function)	DREQ0 (polarity and edge/level detection selectable)	DRACK0 (output timing adjustable and polarity selectable)	DACK0 (polarity selectable)	The burst count (8) does not depend on the access size; 8 consecutive bus accesses are executed for 8-bit and 16-bit bus widths equally.
Channel 1	Dual address mode (with 8-burst function)	DREQ1 (polarity and edge/level detection selectable)	—	DACK1 (polarity selectable)	DRACK1 does not exist as an LSI pin. The burst count (8) does not depend on the access size; 8 consecutive bus accesses are executed for 8-bit and 16-bit bus widths equally.
Channel 2	Dual address mode	DREQ2 (polarity and edge/level detection selectable)	—	DACK2 (polarity selectable)	DRACK2 does not exist as an LSI pin.

(1) Data Alignment during LBSC-DMAC Access

When receiving an access request from the LBSC-DMAC, the BSC translates the LBSC-DMAC interface into the EX_BUS protocol. During this operation, the BSC determines the area to be accessed based on the settings provided in the LBSC-DMAC area assignment registers (EXDMASET0 to EXDMASET2), and generates a chip select signal. For the data alignment of DMAC channels, one of two modes can be selected based on DMAC register settings.

- Fixed Alignment Mode

  In this operating mode, when performing a DMA transfer with a device on the EX_BUS, the LBSC determines that the access size from the LBSC-DMAC is equal to the data width used by the external device (not bound by the bus width specified in the CS0CTRL, CS1CTRL or ECS0CTRL to ECS5CTRL). In the fixed alignment mode, no alignment translation is performed in the BSC. Instead, data is output using the alignment received from the LBSC-DMAC. The LBSC, however, generates the $\overline{WE1}$ to $\overline{WE0}$ signals, depending on the access size received from the LBSC-DMAC. In the initial state, the LBSC operates in this mode. Table 6B.5 shows the relationship between data widths and access sizes.

Table 6B.5 Relationship between Data Widths and Access Sizes in Fixed Alignment Mode

Access Size Set in DMAC (Assumed to be Bus Width)	Data Bus Read/Write Position (Common to Big and Little Endians)		Write Enable Signal Output Position (Common to Big and Little Endians)	
	D15 to D8	D7 to D0	$\overline{WE1}$	$\overline{WE0}$
16 bits	Data 15 to 8	Data 7 to 0	Assert	Assert
8 bits	—	Data 7 to 0	—	Assert

- Variable Alignment Mode

  If the access size set in the DMAC is smaller than the bus width set in the BSC, variable alignment mode can be set up by DMAC register settings. In this case, the BSC changes byte lanes according to the specific access address to be serviced. Tables 6B.6 and 6B.7 show the relationship between data widths and access sizes in variable alignment mode. If the bus width set in the LBSC is equal to the DMAC access size, variable alignment mode operates in the same way as fixed mode.

Table 6B.6 Relationship between Data Widths and Access Sizes in Variable Alignment Mode (Big Endian)

Bus Width Set in BSC	Access Size Set in DMAC	Data Bus Read/Write Position (Big Endian)		Write Enable Signal Output Position (Big Endian)	
		D15 to D8	D7 to D0	$\overline{WE1}$	$\overline{WE0}$
16 bits	16 bits	2n address		2n address	
	8 bits	2n address	2n+1 address	2n address	2n+1 address
8 bits	8 bits	—	n address	—	n address

Table 6B.7 Relationship between Data Widths and Access Sizes in Variable Alignment Mode (Little Endian)

Bus Width Set in BSC	Access Size Set in DMAC	Data Bus Read/Write Position (Little Endian)		Write Enable Signal Output Position (Little Endian)	
		D15 to D8	D7 to D0	$\overline{WE1}$	$\overline{WE0}$
16 bits	16 bits	2n address		2n address	
	8 bits	2n+1 address	2n address	2n+1 address	2n address
8 bits	8 bits	—	n address	—	n address

(2) External Device DMA Transfer Request Detection Function

To provide generality, the LBSC supports a polarity and level/edge selection function as a method for the reception of external device DMA transfer request signals (DREQ). The selections are made by setting the relevant values in registers EXDMCR0 to EXDMCR2. In compliance with the settings, the BSC internally receives a DREQ signal and outputs the DMA transfer request signal from the BSC to the LBSC-DMAC. If level detection is selected, the BSC continues to output the DMA transfer request signal to the LBSC-DMAC as long as the DREQ signal from the external device is asserted. Conversely, if edge detection of the DREQ signal is selected, the BSC will not output a DMA transfer request signal to the LBSC-DMAC until an edge is detected, even if the DREQ signal from the external device continues to be asserted. In addition, even if more than one edge is detected during the period from the first edge detection to the start of DMA transfer, only one transfer request is issued.

(3) Role of the BSC between the LBSC-DMAC and an External Device

Figure 6B.12 shows a basic timing chart for LBSC-DMAC-to-EX_BUS output translation. Upon receipt of an access request from the DMAC, the BSC first performs bus contention arbitration. When the LBSC-DMAC acquires the bus mastership, the BSC controls the pulse width of $\overline{RD}$, $\overline{WE1}$ and $\overline{WE0}$ that are output on the EX_BUS according to the settings provided in the LBSC-DMAC channel RD/WE pulse control registers (EXDMAWCR0 to EXDMAWCR2) that the BSC possesses, for the LBSC-DMAC access period.

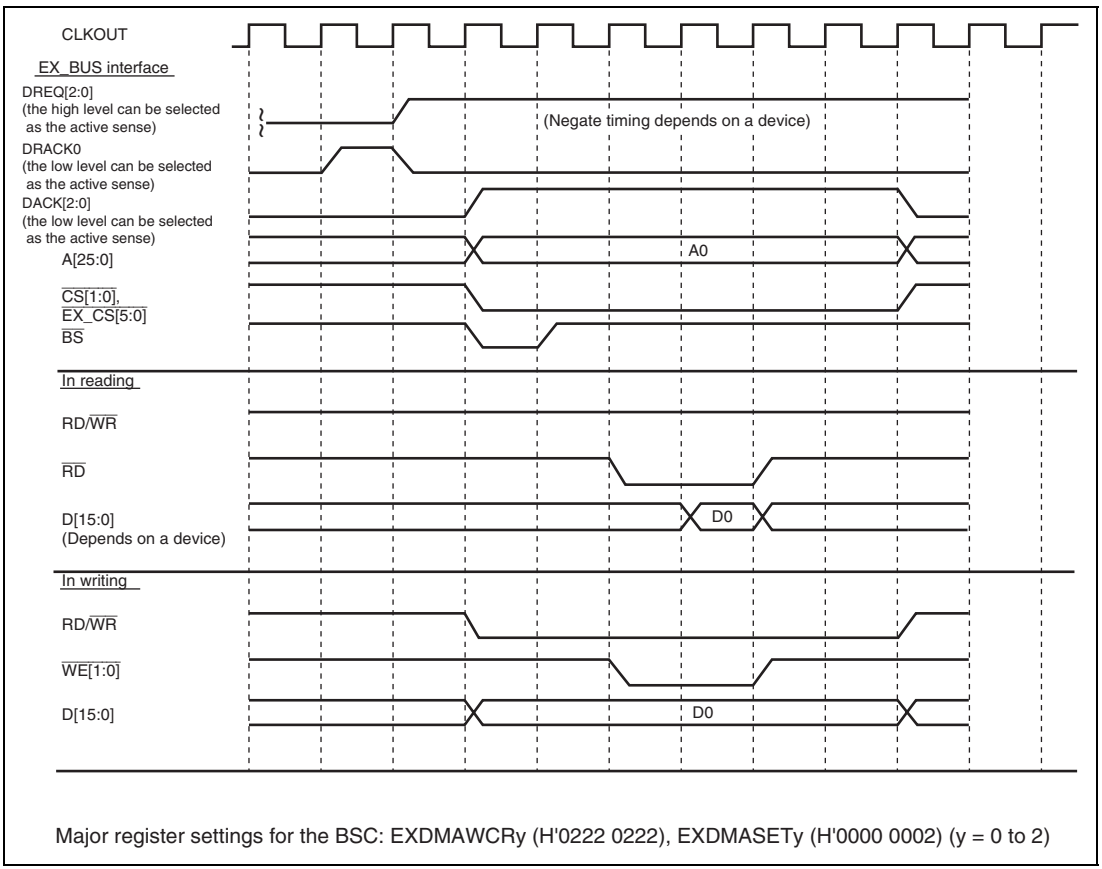


Figure 6B.12 Basic Timing Chart for Access from LBSC-DMAC to EX_BUS

(4) LBSC-DMAC → EX_BUS Burst Access

The BSC supports the EX_BUS output function for burst access operations from the LBSC-DMAC (eight consecutive transfers irrespective of the access size). Figure 6B.13 shows a timing chart on LBSC-DMAC burst access. The waveform for LBSC-DMAC burst access basically takes the form of repeated single-access waveforms. However, because external DMAC bus access does not release the bus mastership until the burst access is finished, PIO access from the CPU (SuperHyway) does not result in a cycle stealing. In addition, transfer performance is enhanced because the DMAC treats data involving 8 transfer operations as a single packet and performs transfer operations with the destination.

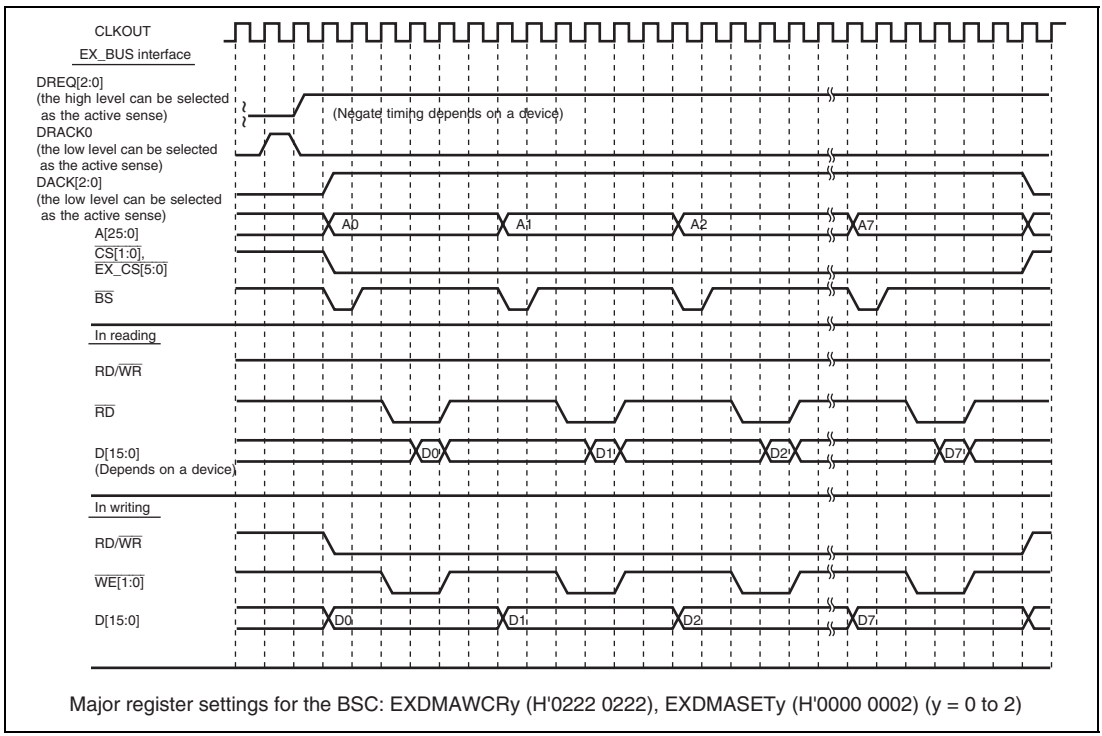
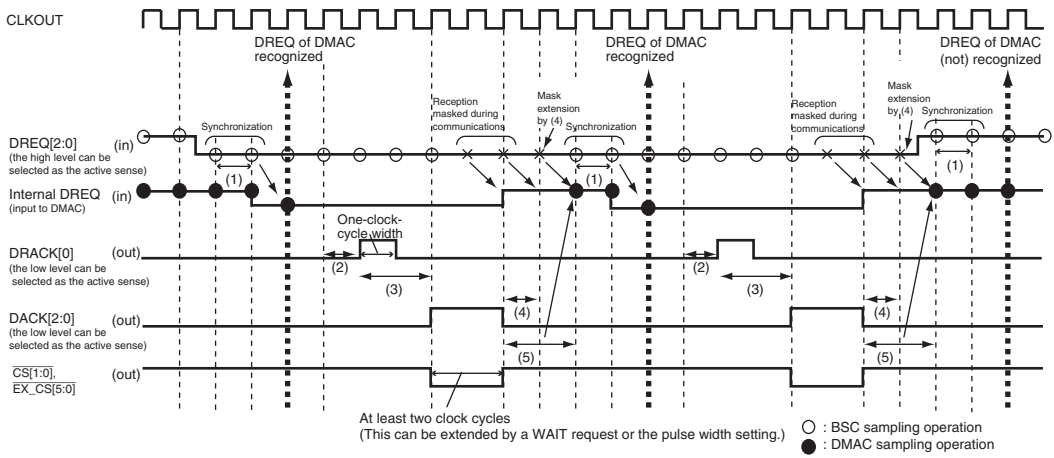


Figure 6B.13 Basic Timing Chart for Burst Access from LBSC- DMAC to EX_BUS

(5) DREQ Reception Timing

Figure 6B.14 shows the timing at which a DMA request (DREQ) signal is set in the level reception mode and any continuation of an asserted state is regarded as another DMA request. The figure also shows DREQ negate timing that prevents the above condition from being mistaken as another DREQ request.



(1) Synchronization time for asynchronous DREQ
By setting an appropriate value in EXDMCR0 of the BSC within bus bridge, DREQ can also be treated as a synchronous signal.
The default is to treat it as a synchronous signal, in which case the above 1 clock cycle becomes 0 clock.

(2) EX_BUS arbitration time
The above 1 clock cycle represents the shortest time. In the case of contention and the time is extended until the access right is acquired.

(3) DRACK → DACK time
Either one or two clock cycles can be selected by means of EXDMCR0 of the BSC within bus bridge.
The above diagram is a case where the DRACK-to-DACK time is specified as two clock cycles.
Because IO-DMAC channels other than channel 0 do not have a DRACK, the interval in (3) will be 1 clock cycle for the channels.

(4) Next DREQ recognition masking time (not available in the edge mode)
In the clock points following the $\overline{CS}$/$\overline{EX\_CS}$ negate, the reception of the next DREQ can be masked.
The above diagram represents the case where a 1 clock cycle interval is masked.
This varies in a 0 to 15 clock-cycle range in DRMSKR of the DMAC within bus bridge.

Note: Setting (4) is not allowed in the edge mode, in which exists no such period of reception masking during communications as shown in the diagram above. Therefore, DREQ is detected on an edge basis even during the interval indicated as "Reception masked during communications" in the diagram above.

(5) Next DREQ for DMAC sampling resumption timing
This is the timing at which the DMAC resumes the sampling of another DREQ (a synchronized signal) from the $\overline{CS}$ and $\overline{EX\_CS}$ negate points.
The timing at which DREQ sampling is resumed, shown in the table below, varies with access mode.

Access Mode	Clock Cycle Count
Write access	Sampling resumed after two clock cycles
Read access	Sampling resumed after three clock cycles

According to the above diagram, in the case where setting (4) is 0, DREQ is detected as another DMA request unless DREQ is negated between the $\overline{CS}$ and $\overline{EX\_CS}$ negate points and the next clock cycle.

Figure 6B.14 Next DREQ Signal Recognition Timing on EX_BUS

6B.6.4 CPU (SuperHyway Bus) → Byte-Control SRAM Interface

With respect to EX_BUS area 1 and expansion areas 0 to 5, the LBSC supports byte-control SRAM interface from the CPU (SuperHyway bus) or LBSC-DMAC (area 0 is excluded). Byte-control SRAM interface is a memory interface in which byte select strobe signal WE is output in both read and write cycles. Write timing for byte-control SRAM interface is the same as that for SRAM interface. In read access, however, $\overline{WE}$ is output at a different timing. In read access, WE is output for a read byte only. The assertion timing of $\overline{WE}$ can be selected with the CS1BRM bit in the CS1CTRL register or the ECSxBRM bit in the ECSxCTRL register to be the same as that of $\overline{CS}$ or $\overline{RD}$. In default setting, SRAM operating mode is set for each area. Accordingly, along with the setting of assertion timing, byte-control SRAM mode should be set in the registers corresponding to a given area (CS1CTRL or ECS0CTRL to ECS5CTRL).

(1) Byte-control SRAM Interface

- The $\overline{WE}$ assertion period can be selected by the CS1BRM bit or the ECS0BRM to ECS5BRM bit to be the same as that of $\overline{CS}$ or $\overline{RD}$.
- In read access, $\overline{WE}$ is asserted only for a valid access byte. (For example, when a byte is read from a device with the byte-control SRAM interface and a bus width of 16 bits, any one of the $\overline{WE}$ bits is asserted.)
- The waveform in write access is the same as that for SRAM interface.

(2) Basic Timing Charts

Figure 6B.15 shows the basic timing chart for byte-control SRAM interface.

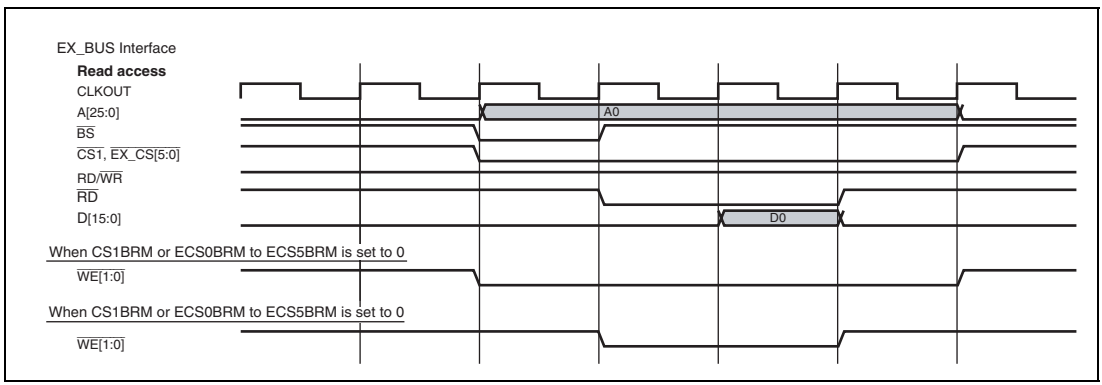


Figure 6B.15 Waveforms for Byte-Control SRAM Interface

6B.6.5 CPU (SuperHyway Bus) → ATA Device Interface

With respect to EX_BUS area 1 and expansion areas 0 to 5, the LBSC supports an ATA device interface from the CPU (SuperHyway bus) or LBSC-DMAC (area 0 is excluded). Since in each space the default is the SRAM operating mode, register settings for a given area from among CS1CTRL and ECS0CTRL to ECS5CTRL should be set for the ATA space. To perform multi-word DMA transfer in the ATA interface, set the DBST bit in the EXDMCRy (y = 0 to 2) register to 1.

$\overline{\text{ATACS0}}$ and $\overline{\text{ATACS1}}$ can be asserted in synchronization with the access space $\overline{\text{CS}}$ signal depending on the ATACS control register (ATACSCTRL) settings.

(1) $\overline{\text{ATACS0}}$/$\overline{\text{ATACS1}}$ Signal

If ATACS1_EN and ATAECSx_EN (x = 0 to 5) of the ATACS control register are set to 1 and the ATA mode is selected with the CS1IF bit of the CS1CTRL register and ECSxIF bit of the ECSxCTRL register, $\overline{\text{ATACS0}}$ or $\overline{\text{ATACS1}}$ is asserted in synchronization with the relevant space $\overline{\text{CS}}$ signal when the relevant space is accessed (the relevant space $\overline{\text{CS}}$ signal is also asserted at the same time).

$\overline{\text{ATACS0}}$ or $\overline{\text{ATACS1}}$ is asserted in accord with the value being output as address bit 4 on the external bus (i.e., the A4 pin).

If, for example, the CS1 space is ATA interface, the conditions below should apply to $\overline{\text{ATACS0}}$ and $\overline{\text{ATACS1}}$ assertion.

Table 6B.8 Conditions of $\overline{\text{ATACS0}}$ and $\overline{\text{ATACS1}}$ Assertion (CS1 Space: ATA Interface)

ATACS1_EN	A4 Pin	$\overline{\text{CS1}}$	$\overline{\text{ATACS0}}$	$\overline{\text{ATACS1}}$
0	*	√	—	—
1	0	√	—	√
1	1	√	√	—

√: Assert, —: Deassert

Note: Don't care.

(2) ATA Interface Signals

Table 6B.9 shows the LBSC output signals when the area is set for ATA device interface.

Table 6B.9 List of ATA Interface Signals

No.	ATA I/F Signals	LBSC I/O Signal	I/O	Function	Remarks
1	$\overline{CS0}$	$\overline{ATACS0}$	Output	Chip select (command block)	
2	$\overline{CS1}$	$\overline{ATACS1}$	Output	Chip select (control block)	
3	DMARQ	DREQ[2:0] (the low level can be selected as the active sense)	Input	DMA request	Polarity selectable. Only pins DREQ[1:0] are used in Ultra DMA operation.
4	$\overline{DMACK}$	$\overline{DACK[2:0]}$ (the low level can be selected as the active sense)	Output	DMA acknowledge	Polarity selectable. Only pins DACK[1:0] are used in Ultra DMA operation.
5	A[2:0]	A[3:1]	Output	Address	
6	DD[15:0]	D[15:0]	I/O	Data	
7	DIOW/ STOP	$\overline{ATAWR}$	Output	Write enable/ STOP signal (for Ultra DMA operation)	
8	DIOR/$\overline{HDMARDY}$/ HSTROBE	$\overline{ATARD}$	Output	Read enable/ host DMA ready (for Ultra DMA read)/ host data strobe (for Ultra DMA write)	
9	IORDY/DDMARDY/ DSTROBE	Selected from WAIT[2:0] (the low level can be selected as the active sense)	Input	I/O ready/ device DMA ready (for Ultra DMA write)/ device data strobe (for Ultra DMA read)	
10	—	$\overline{ATAG}$	Output	External buffer enable	
11	—	$\overline{ATADIR}$	Output	External buffer direction control	Low output for write accesses

Note: When $\overline{ATACS0}$ and $\overline{ATACS1}$ are used, the ATACSCTRL register should be set.

(3) ATA Interface Connection Configuration

Figure 6B.16 shows a sample connection between the ATA device and the LBSC when EX_BUS is set for ATA interface.

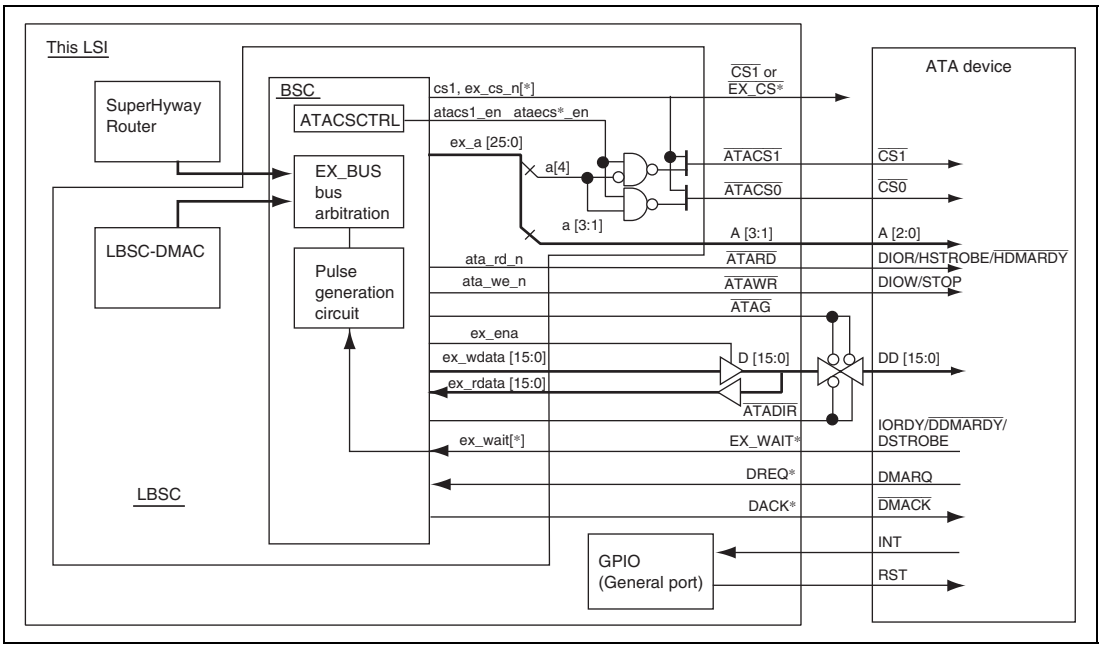


Figure 6B.16 EX_BUS ATA Device Configuration Example

(4) ATA Interface (PIO Mode, Multi-word Mode)

The BSC provides the following ATA interface support functions:

- Based on the IORDY signal received from an ATA device, the BSC performs wait control on the ATA interface.
- With an ATA interface, allocating the LBSC-DMAC makes multi-word DMA transfer possible. However, use the CPU to handle access to the ATA registers and the DMA controller to handle data transfer.
- For the detection of I/O ready timeout, the BSC monitors the status of the IORDY signal from ATA devices, and if an ATA device wait state lasting beyond a certain length of time is detected, the BSC indicates an ATA wait timeout error in a register inside the BSC, and forces the termination of the ATA interface access.
  The required detection time depends on the EX_BUS operating frequency according to the following formula:

  Detection time [ns] = EX_BUS operating frequency (ns) × 100 clock cycles

(5) Basic Timing Charts (PIO Mode, Multi-word Mode)

Figure 6B.17 shows the basic timing chart for PIO transfer in the ATA interface.

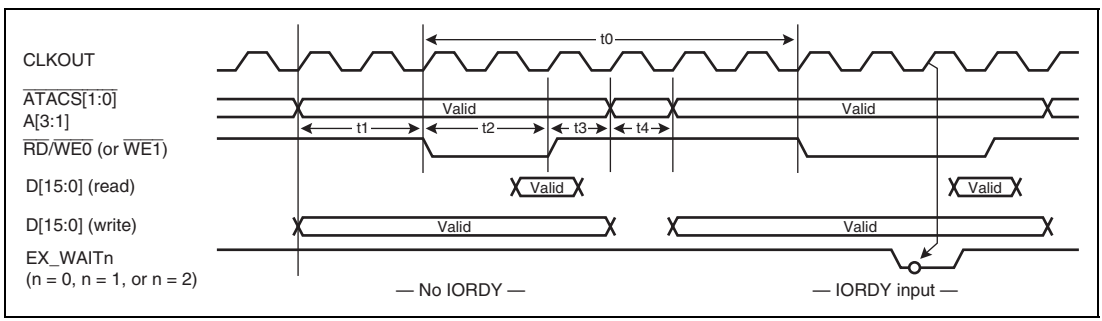


Figure 6B.17 Waveforms for ATA Interface

Table 6B.10 shows sample settings for the RD/WE pulse control register for PIO transfers in the ATA interface (when the EX_BUS operating frequency is 50 MHz).

Table 6B.10 Sample Settings for RD/WE Pulse Control Register for PIO Transfers in ATA Interface (EX_BUS Operating Frequency: e.g. 50 MHz)

Mode	CSWCR Setting	Cycle Time (t0)	Address Setup (t1)	DIOR/DIOW Pulse Width (t2)	Address Hold (t3)	Idle Cycle (t4)
0	H'077F077F	600 (600)	140 (70)	300 (290)*	140 (20)	20 (–)
1	H'031F031F	400 (383)	60 (50)	300 (290)*	20 (15)	20 (–)
2	H'021F021F	380 (330)	40 (30)	300 (290)*	20 (10)	20 (–)
3	H'02150215	180 (180)	40 (30)	100 (80)	20 (10)	20 (–)
4	H'02150215	180 (120)	40 (25)	100 (70)	20 (10)	20 (–)

Units: ns

Values in parentheses indicate ATA prescribed values.

Note: The CSWCR setting must be such that the following condition is satisfied: ATA prescribed value ≤ CSWCR set value × EX_BUS clock (CLKOUT) width

Figure 6B.18 shows the basic timing chart for multi-word DMA transfer in the ATA interface.

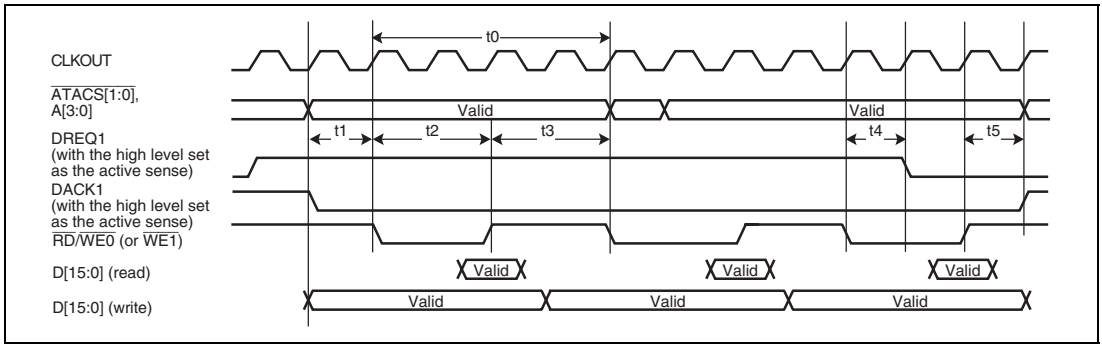


Figure 6B.18 Multi-Word DMA Waveforms for ATA Interface

Table 6B.11 shows sample settings for the RD/WE pulse control register for multi-word DMA transfers in the ATA interface (when the EX_BUS operating frequency is 50 MHz).

Table 6B.11 Sample Settings for RD/WE Pulse Control Register for Multi-Word DMA Transfers in ATA Interface (EX_BUS Operating Frequency: 50 MHz)

Mode	CSWCR Setting	Cycle Time (t0)	CS[1:0] Setup (t1)	DIOR/ DIOW Pulse Width (t2)	DIOW Negate Hold/DIOR Negate Pulse Width (t3)	DIOW/ DIOR DREQ Delay Time (t4)	DACK Hold (t5)
0	H'067B067B	480 (480)	120 (50)	220 (215)	260 (215/50)	– (120/40)	140 (20)
1	H'02240224	160 (150)	40 (30)	80 (80)	80 (50)	– (40)	40 (5)
2	H'02140214	140 (120)	40 (25)	80 (70)	60 (25)	– (35)	20 (5)

Units: ns

Values in parentheses indicate ATA prescribed values.

Note: The EXDMAWCR setting must be such that the following condition is satisfied: ATA prescribed value ≤ EXDMAWCR set value × EX_BUS clock (CLKOUT) width

(6) UltraATA DMA Mode

The UltraATA DMA mode differs from the PIO mode or multi-word mode with respect to the function of each signal interfacing with the ATA device. The LBSC operates with changing each interface signal function in UltraATA DMA mode as shown in table 6B.9. The connections between the LBSC and an ATA device on a board need not be changed from those shown in figure 6B.16.

For operation in UltraATA DMA mode, specify the ATA mode as the operating mode for the external bus area where an ATA device is connected. Specify LBSC-DMAC channel 0 or 1 as the DMAC assigned to the target area. After that, select the UltraATA DMA mode through UATMR in the LBSC-DMAC, make the necessary settings in the same way as for normal DMAC activation, and activate the DMA operation.

The following is a list of the features of the UltraATA DMA operation.

- Access to the registers in the ATA device such as initial settings, activation, or status read access can be done directly through PIO.
- Up to UltraDMA mode 4 is supported.
- LBSC-DMAC channel 0 or 1 supports UltraDMA.
- The A, D, CS, DIOR or DIOW, and EX_WAIT signals are used for PIO R/W access to the area set to the ATA mode.
- For DMA write operation, the setup and hold periods of data relative to the HSTROBE (DIOR) signal edge can be specified in units of output clock cycles.
- For DMA read operation, data is received in synchronization with the DSTROBE (IORDY) signal input. The HDMARDY signal is temporarily negated while reception is busy.
- Cycle-stealing is done for PIO transfer in other areas or transfer in other LBSC-DMAC channels even during UltraDMA bus operation.
- Cycle-stealing occurs only when an external access request is generated within the LSI. It is executed while HDMARDY or DDMARDY is temporarily negated.
- The DMA transfer size is set to one sector, and a desired sector size can be specified. However, note that the base unit of transfer is 2 bytes × 16 burst, and this unit should be repeated an integral number of times (n) to obtain the desired transfer size.
- Cycle-stealing for PIO transfer in the current area is not allowed while one-sector DMA transfer is in progress.
- The CRC check function is supported.
- After one-sector DMA transfer, a CRC code (generation polynomial: X^16 + X^12 + X^5 + 1) is output to the device.

- The output CRC code can be read from an LBSC-DMAC register (for debugging and evaluation use).
- The LBSC-DMAC provides a function for monitoring timeout during DMA transfer. When an overflow occurs, a timeout interrupt notification signal can be issued.

Figure 6B.19 shows the UltraATA DMA operation timing.

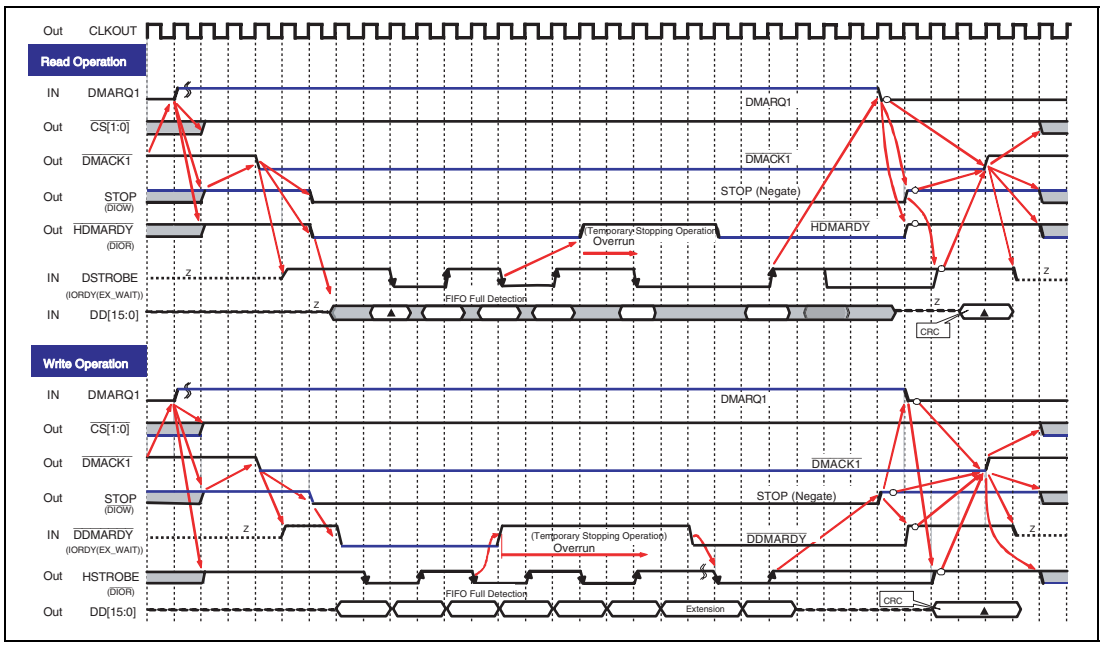


Figure 6B.19 UltraATA DMA Operating Waveform

Figure 6B.20 shows a setting procedure for the UltraATA DMA transfer.

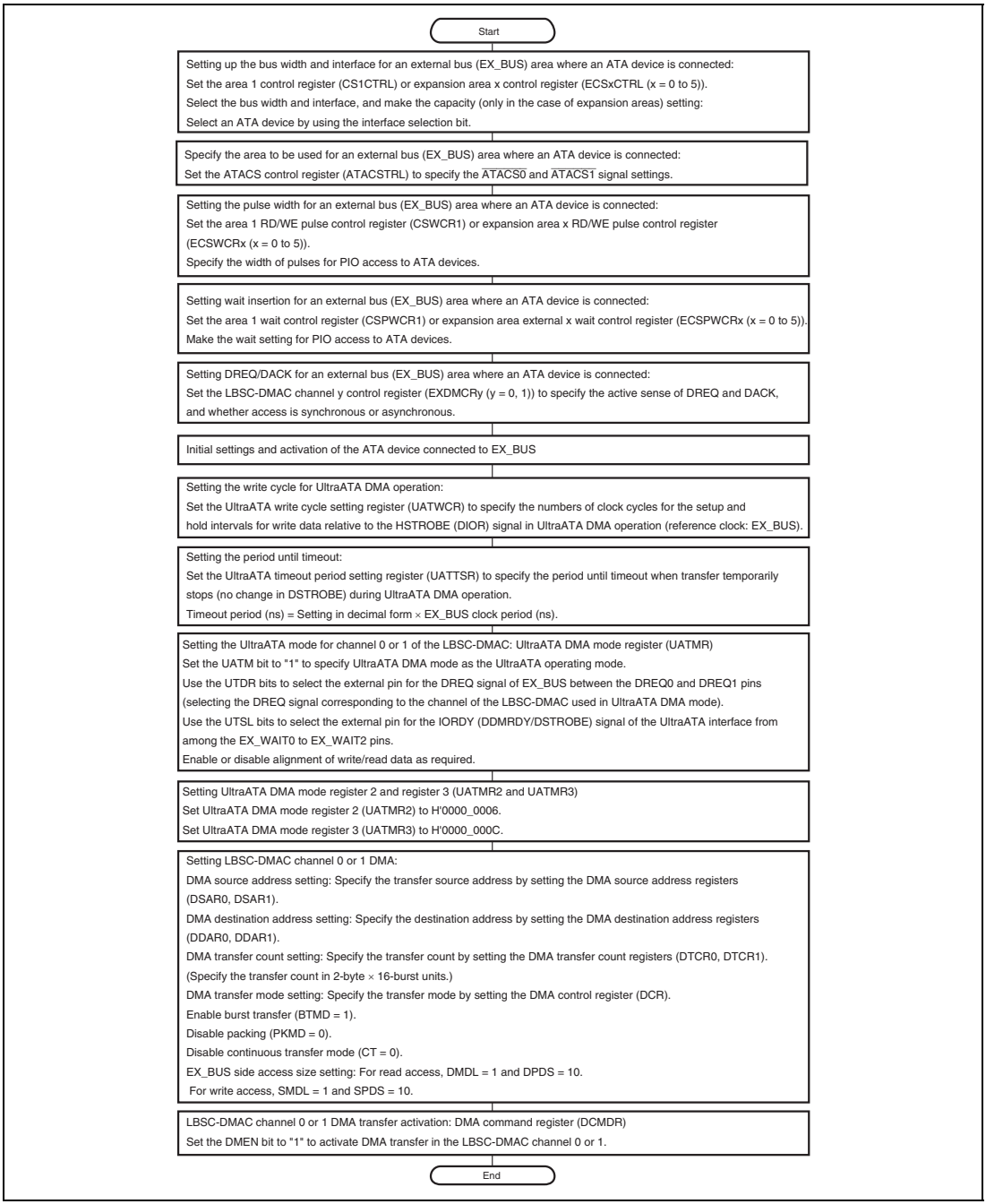


Figure 6B.20 Setting Procedure for the UltraATA DMA Transfer

6B.6.6 EX_BUS Arbitration

To deal with the conflicts between the access requests from the CPU (SuperHyway bus) and LBSC-DMAC channels 0 to 2, the EXBATLV register in the LBSC can set the priority levels for these accesses. Furthermore, to deal with the conflicts among the LBSC-DMAC channels 0 to 2, the three channels are grouped into two groups by the DMA memory access priority level control register (DMLVLR) in the LBSC-DMAC, and the priority levels are determined according to the round-robin scheme in each group.

Figure 6B.21 shows a concept diagram for EX_BUS arbitration.

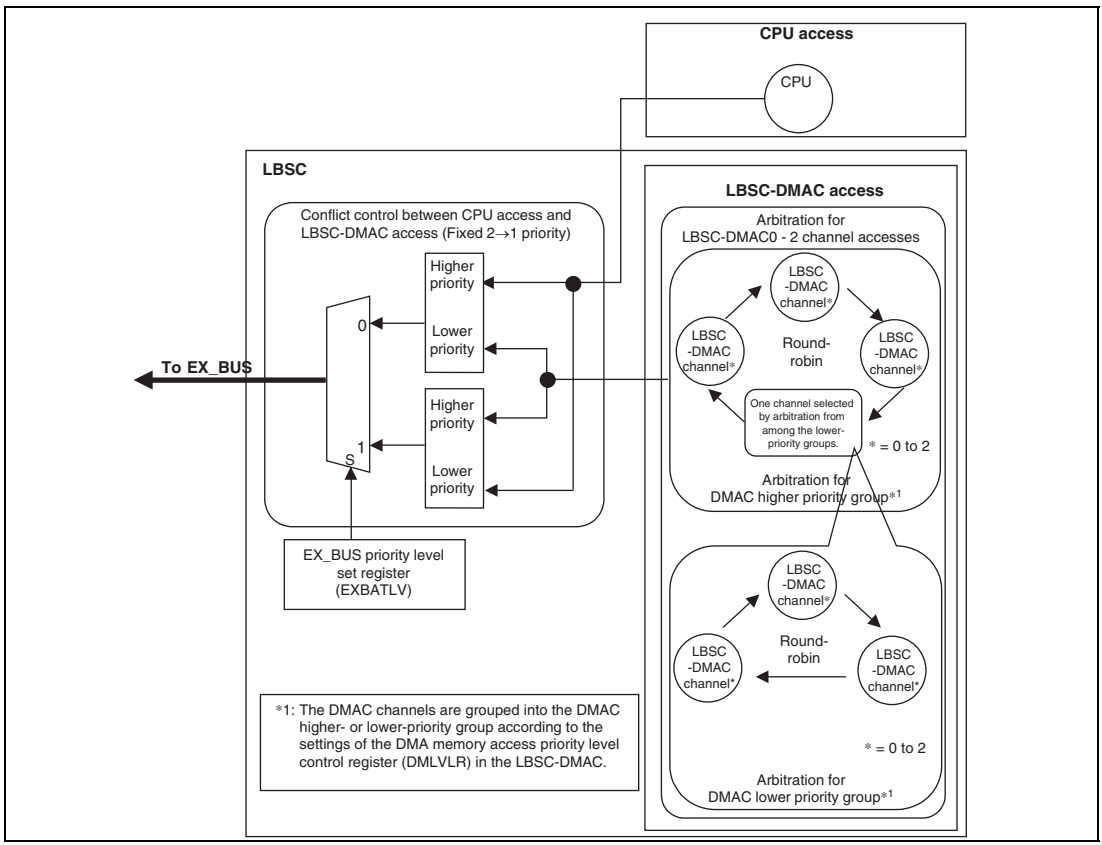


Figure 6B.21 Concept Diagram for EX_BUS Arbitration

6B.7 Usage Notes

Please keep the following notes in mind when using this LBSC.

6B.7.1 Pin Multiplexing

When starting this LSI, be sure to set multiplexed pins appropriately using the LSI pin multiplexing set register.

Also, make pull-up resistor settings using the LSI pin pull-up control registers.

For details on the specific set values, refer to section 37, Pin Function Controller (PFC).

6B.7.2 Operations for Read Access by UltraDMA

The LBSC and LBSC-DMAC handle UltraDMA transfer to and from the ATA devices connected to EX_BUS.

Since the EX_BUS is an external bus for the connection of external devices, transfer over the bus is also performed while controlling conflict with access to external devices (transfer is conducted by acquiring and releasing bus masterships). The operation is thus as follows.

(1) Operations for Read Access by UltraDMA

1. The HDMARDY signal is negated (the bus mastership is acquired and released per 32-byte unit of transfer).

   So that transfer can proceed while controlling conflicts with access to external devices, the bus mastership is released (the HDMARDY signal is negated) per 32-byte unit of transfer.

   Figure 6B.22 shows the operation of the $\overline{\text{HDMARDY}}$ and DSTROBE signals during operations for read access.

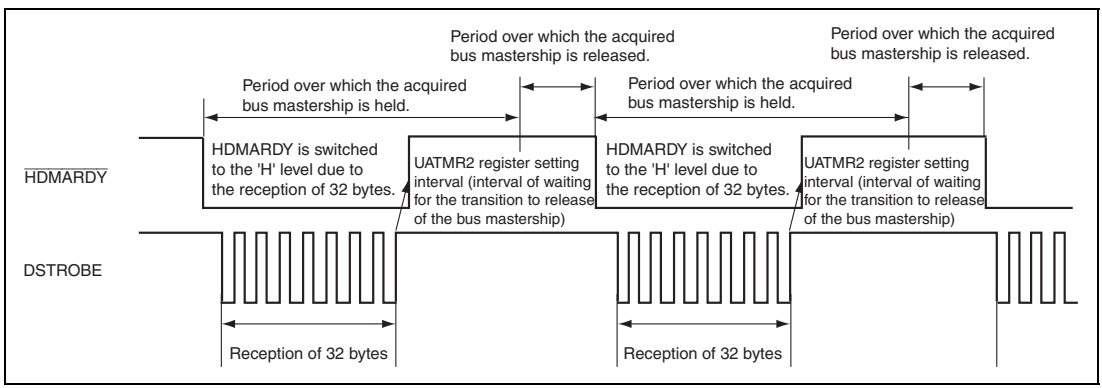


Figure 6B.22 Operation of the $\overline{\text{HDMARDY}}$ and DSTROBE Signals during Operations for Read Access

Section 7 INTC/INTC2

7.1 Overview

The interrupt controller (INTC) notifies the CPU of an NMI input from the external pin. It also controls interrupt requests to the CPU by determining the priority of the source of the external interrupts IRL and IRQ and a representative interrupt from the INTC2 that controls the interrupts from on-chip peripheral modules. The INTC has registers for setting the priority of each of the IRQ interrupts. Based on the priority order set in these registers by the user, interrupt requests are processed. The registers to control interrupts from on-chip peripheral modules are supported by the INTC2.

7.2 Features

7.2.1 Features of INTC

The INTC is an interrupt controller that is placed under the LBSC on the SuperHyway bus to improve the response of registers accessed by software. The INTC is connected to the register interface that is extended and output to the LBSC, and it responses to accesses made by the CPU at high speed. The INTC has the following features:

- Notifies the CPU of the external interrupts, NMI, IRQ and IRL, and interrupts from the INTC2 according to the specified priority order.
- By setting the interrupt priority registers, the priorities of external interrupts can be selected from among 15 levels for individual pins.
- An NMI input-level bit indicates the NMI pin state. The bit can be read within the interrupt exception handling routine to confirm the pin state and thus achieve a form of noise cancellation.
- Masking or non-masking of NMI requests when the BL bit in SR is set to 1 can be selected.
- Automatically updates the IMASK bit in SR according to the accepted interrupt level
- An interrupt mask level in the user interrupt mask level register (USERIMASK) can be specified to disable interrupts which do not have higher priority than the specified mask level. This setting can be made in user mode.

- For the IRQ and IRL interrupts when the level sensing is set, the following two modes are available: a mode in which the source of interrupt is temporarily held inside of the INTC even if the input level of the external pin is not retained (ICR0.LVLMODE = 0), and a mode in which the source of interrupt is not held inside of the INTC (ICR0.LVLMODE = 1). The initial value of the ICR0.LVLMODE is 0; however, it is recommended to set ICR0.LVLMODE to 1 by setting the interrupt control register 0 (ICR0) by the initialization routine.

7.2.2 Features of INTC2

The INTC2 is an interrupt controller, similarly to the INTC, that is placed under the LBSC on the SuperHyway bus to improve the response of registers accessed by software. The INTC2 is connected to the register interface that is extended and output to the LBSC, and it response to accesses made by the CPU at high speed. The INTC2 has the following features:

- Receives interrupts from on-chip peripheral modules, GPIO and IO chips, and notifies the CPU of the interrupts according to the specified priority order via the INTC.
- Any of 30 priority levels can be assigned to the individual requesting sources.
- The high-speed masking functionality that accelerates negation of the interrupt signals can prevent erroneous detection of the same interrupt by the CPU during the period from clearing, by software, of the interrupt for the on-chip peripheral module to the reception of the interrupt signal negation by the CPU.

7.3 Block Diagram

Figure 7.1 is a block diagram of the INTC and INTC2. The details of the INTC's input control circuit of figure 7.1 are shown in figure 7.2.

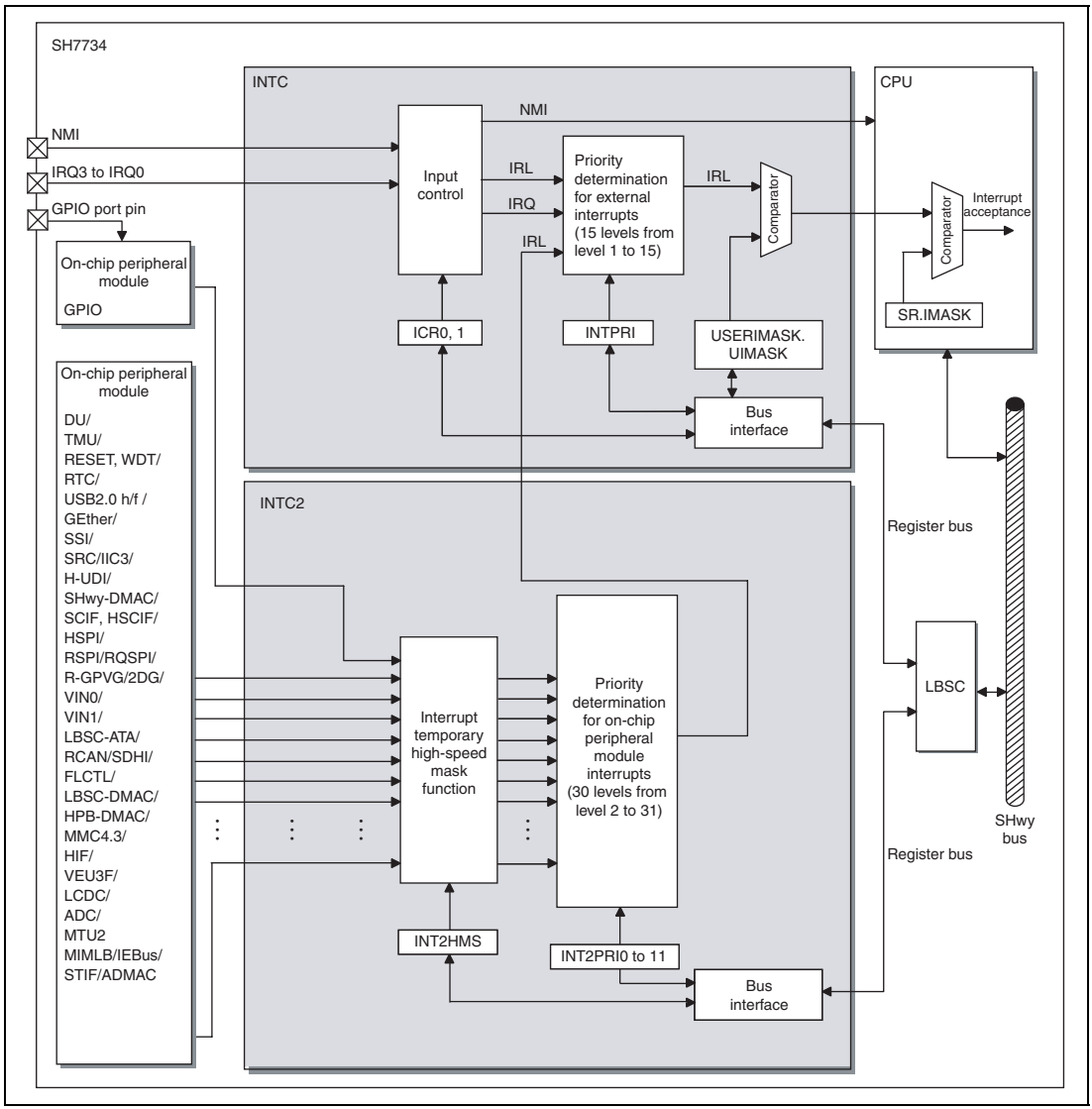
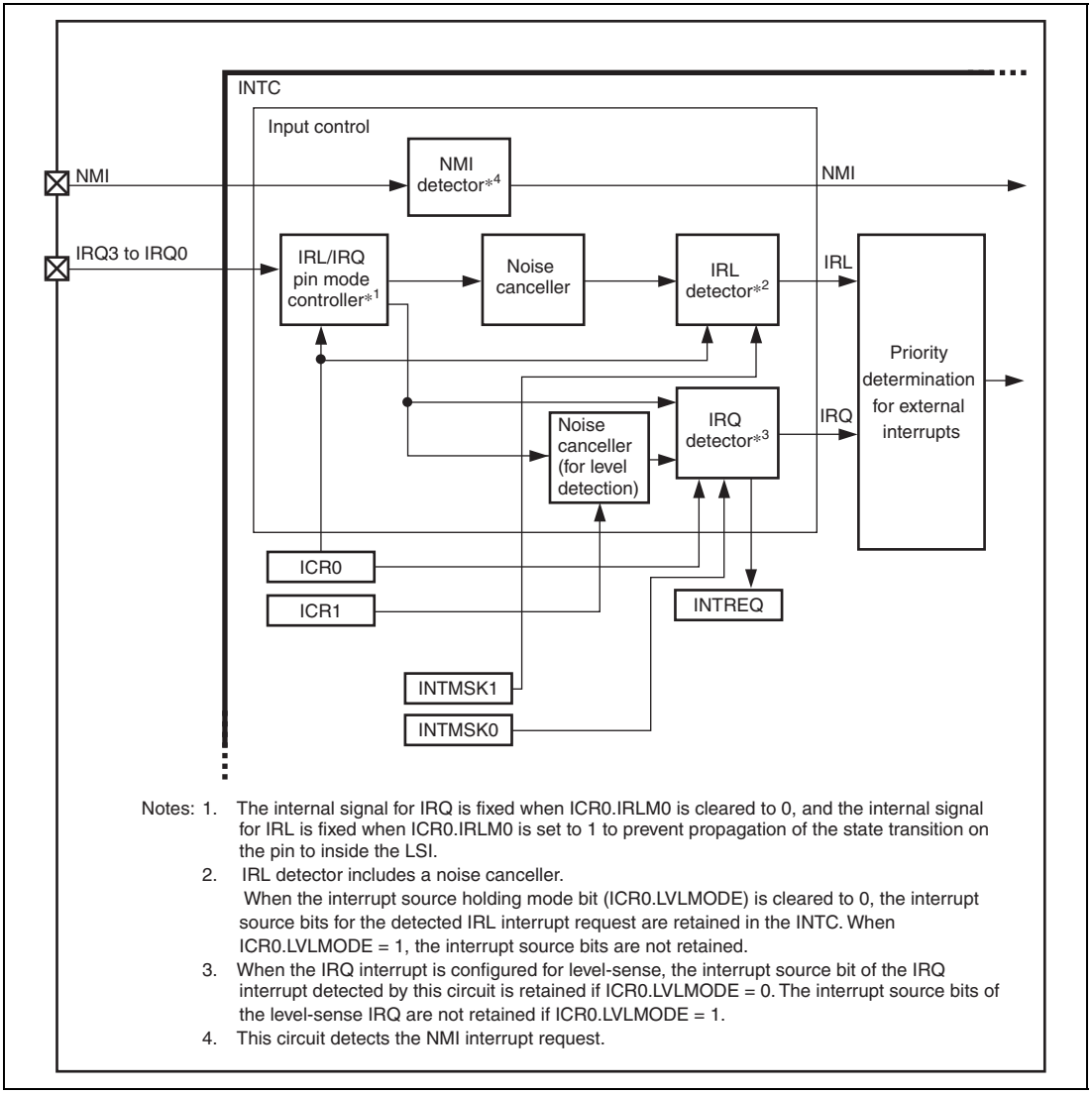


Figure 7.1 Block Diagram of INTC and INTC2



Notes: 1. The internal signal for IRQ is fixed when ICR0.IRLM0 is cleared to 0, and the internal signal for IRL is fixed when ICR0.IRLM0 is set to 1 to prevent propagation of the state transition on the pin to inside the LSI.

2. IRL detector includes a noise canceller.
When the interrupt source holding mode bit (ICR0.LVLMODE) is cleared to 0, the interrupt source bits for the detected IRL interrupt request are retained in the INTC. When ICR0.LVLMODE = 1, the interrupt source bits are not retained.

3. When the IRQ interrupt is configured for level-sense, the interrupt source bit of the IRQ interrupt detected by this circuit is retained if ICR0.LVLMODE = 0. The interrupt source bits of the level-sense IRQ are not retained if ICR0.LVLMODE = 1.

4. This circuit detects the NMI interrupt request.

Figure 7.2 Details of Input Control Circuit

7.3.1 Interrupt Method

The basic flow of exception handling for interrupts is as follows.

In interrupt exception handling, the contents of the program counter (PC), status register (SR), and general register 15 (R15) are saved in the saved program counter (SPC), saved status register (SSR), and saved general register15 (SGR), and the CPU starts execution of the interrupt exception handling routine at the corresponding vector address. An interrupt exception handling routine is a program written by the user to handle a specific exception. The interrupt exception handling routine is terminated and control returned to the original program by executing a return-from-exception instruction (RTE). This instruction restores the contents of PC and SR and returns control to the normal processing routine at the point at which the exception occurred. The contents of SGR are not written back to R15 by the RTE instruction.

1. The contents of the PC, SR and R15 are saved in SPC, SSR and SGR, respectively.
2. The block (BL) bit in SR is set to 1.
3. The mode (MD) bit in SR is set to 1.
4. The register bank (RB) bit in SR is set to 1.
5. In a reset, the FPU disable (FD) bit in SR is set to 0.
6. The exception code is written to bits 13 to 0 of the interrupt event register (INTEVT).
   Processing jumps to the start address of the interrupt exception handling routine, vector base register (VBR) + H'600.
7. The processing branches to the corresponding exception handling vector address and the exception handling routine starts.

7.3.2 Interrupt Sources

Table 7.1 shows an example of the interrupt sources.

The INTC selects an interrupt from among external interrupts and the interrupts from the INTC2 that controls on-chip peripheral module interrupts, according to the set priority order. Then, it notifies the CPU of the selected interrupt, with the corresponding 4-bit interrupt priority level code and interrupt source code (INTEVT code). The INTC2 also outputs the interrupt priority level code and source code (INTEVT code) to the CPU via the INTC. When the interrupt is accepted by the CPU, the appropriate INTEVT code is indicated in the INTEVT register. The interrupt handler can identify the interrupt source by reading the INTEVT code register of the CPU without reading the source indication registers of the INTC and INTC2.

The priority order of the IRQ interrupt and on-chip peripheral module interrupts can be set via the INTC and INTC2 registers. If multiple interrupts with the same priority occur at the same time, the priority order is determined as shown in table 7.1.

Table 7.1 Interrupt Source Code and Priority Order (1)

Source		Number of Sources (Max.)	Priority	INTEVT		Remarks
External interrupts (INTC)	NMI	1	—	1C0		
	IRL	1	Inverse of values on the pin inputs (because the pins have negative sense.)	200	IRQ[3:0] pin = H'0	If the same priority is given to multiple interrupts, the priority is determined according to this table.
				220	IRQ[3:0] pin = H'1	
				240	IRQ[3:0] pin = H'2	
				260	IRQ[3:0] pin = H'3	
				280	IRQ[3:0] pin = H'4	(High)
				2A0	IRQ[3:0] pin = H'5	
				2C0	IRQ[3:0] pin = H'6	
				2E0	IRQ[3:0] pin = H'7	
				300	IRQ[3:0] pin = H'8	
				320	IRQ[3:0] pin = H'9	
				340	IRQ[3:0] pin = H'A	
				360	IRQ[3:0] pin = H'B	

Source		Number of Sources (Max.)	Priority	INTEVT	Remarks	
External interrupts (INTC)	IRL	1	Inverse values on the pin inputs (because of the pins' negative sense)	380	RQ[3:0] pin = H'C	(Continued from the previous page)
				3A0	IRQ[3:0] pin = H'D	
				3C0	IRQ[3:0] pin = H'E	
	IRQ	4	Values set in INTPRI (A larger value has a higher priority.)	240	IRQ[0]	(High) 
				280	IRQ[1]	
				2C0	IRQ[2]	
				300	IRQ[3]	

Table 7.1 Interrupt Source Code and Priority Order (2)

	Source	Number of Detailed Sources	Detailed Source Indicate Register	INTEVT Code	Priority
On-chip peripheral module interrupts (INTC2)	DU	14	INT2B0	3E0	If the same priority is given to multiple interrupts, the priority is determined according to this table. (High)
	TMU00	1	INT2B1	400	
	TMU10	1		420	
	TMU20	1		440	
	TMU21 (input capture)	1		460	
	TMU30	1	INT2B2	480	
	TMU40	1		4A0	
	TMU50	1		4C0	
	TMU51 (input capture)	1		4E0	
	TMU60	1		500	
	TMU70	1		520	
	TMU80	1		540	
	Reset, WDT	1	INT2B3	560	
	USB2.0-h 0, 1/f 0	3	INT2B4	580	
	(Reserved)	—	—	5A0	
	(Reserved)	—	—	5C0	
	(Reserved)	—	—	5E0	
	DEBUG (H-UDI)	1	INT2B8	600	
	SHwy-DMAC 0, 1	6	INT2B9	620	
	(Reserved)	—	—	640	
	(Reserved)	—	—	660	
	(Reserved)	—	—	680	
	(Reserved)	—	—	6A0	
	SSI0	4	INT2B14	6C0	
	SSI1	4	INT2B15	6E0	
	SSI2	4	INT2B16	700	
	SSI3	4	INT2B17	720	
	VIN0	8	INT2B18	740	
	R-GPVG	4	INT2B10	760	
	2DG	4	INT2B20	780	
	MMC4.3	2	INT2B7	7A0	
	(Reserved)	—	INT2B22	7C0	

	Source	Number of Detailed Sources	Detailed Source Indicate Register	INTEVT Code	Priority
On-chip peripheral module interrupts (INTC2)	HSPI	1	INT2B23	7E0	(Continued from the previous page)
	LBSC-ATA	1	INT2B24	840	
	IIC3 0	5	INT2B25	860	
	RCAN0	18	INT2B26	880	(High)
	MIMSLB	5	INT2B50	8A0	
	SCIF0	8	INT2B28	8C0	
	SCIF1	8	INT2B29	8E0	
	SCIF2	8	INT2B30	900	
	SCIF3	8	INT2B31	920	
	SCIF4	8	INT2B32	940	
	SCIF5	8	INT2B33	960	
	(Reserved)	—	—	980	
	(Reserved)	—	—	9A0	
	IIC3 1	5	INT2B36	9C0	
	LBDC-DMAC0	1	INT2B37	9E0	
	LBDC-DMAC1	1		A00	
	LBDC-DMAC2	1		A20	
	(Reserved)	—	—	A40	
	RCAN1	18	INT2B39	A60	
	(Reserved)	—	—	A80	
	(Reserved)	—	—	AA0	
	(Reserved)	—	—	AC0	
	SDHI0	3	INT2B43	AE0	
	SDHI1	3	INT2B44	B00	
	IEBus	14	INT2B52	B20	
	(Reserved)	—	—	B40	
	HPB-DMAC0 to HPB-DMAC3	4	INT2B47	B60	
	HPB-DMAC4 to HPB-DMAC10	7		B80	
	HPB-DMAC11 to HPB-DMAC18	8		BA0	
	HPB-DMAC19 to HPB-DMAC22	4		BC0	
	HPB-DMAC 23 to 25, HPB-DMAC27, HPB-DMAC28	5		BE0	

	Source	Number of Detailed Sources	Detailed Source Indicate Register	INTEVT Code	Priority
On-chip peripheral module interrupts (INTC2)	RTC	3	INT2B5	C00	(Continued from the previous page)
	VIN1	1	INT2B19	C20	
	LCDC	6	INT2B6	C40	
	SRC 0	3	INT2B41	C60	(High)
	SRC 1	3	INT2B42	C80	
	GEther	1	INT2B35	CA0	
	SDHI 2	3	INT2B45	CC0	
	GPIO 0 to GPIO 3	128	INT2B48	CE0	
	GPIO 4, GPIO 5	42	INT2B49	D00	
	STIF0	1	INT2B21	D20	
	STIF1	1	INT2B22	D40	
	(Reserved)	—	—	D60	
	(Reserved)	—	—	D80	
	ADMAC	4	INT2B13	DA0	
	HIF	2	INT2B27	DC0	
	FLCTL	4	INT2B34	DE0	
	ADC	1	INT2B12	E00	
	MTU2	25	INT2B38	E20	
	RSPI	3	INT2B40	E40	
	RQSPI	3	INT2B11	E60	
	(Reserved)	—	—	E80	
	(Reserved)	—	—	EA0	
	HSCIF	8	INT2B51	EC0	
	(Reserved)	—	—	EE0	
	(Reserved)	—	—	F00	
	(Reserved)	—	—	F20	
	VEU3F (VE3)	1	INT2B46	F40	
	(Reserved)	—	—	F60	
	(Reserved)	—	—	F80	
	(Reserved)	—	—	FA0	
	(Reserved)	—	—	FC0	
	(Reserved)	—	—	FE0	

Note: Since the INTEVT codes H'800 and H'820 are used as CPU internal source code, they cannot be used by the INTC2.

7.4 Register Descriptions

Table 7.2 shows the register configuration. Each register is initialized by a power-on reset or a manual reset.

[Legend]

Initial value:	Register value after a reset
—:	Undefined value
R/W:	Readable/writable. The written value can be read.
R/WC1:	Readable/writable. Writing 1 initializes the bit. Writing 0 is ignored.
R:	Read-only. The write value should always be 0.
—/W:	Write-only. The read value is undefined.

Table 7.2 Register Configuration

Name	Abbreviation	R/W	Address	Access Size	Allocated in
Interrupt control register 0	ICR0	R/W	H'FF80 2000	32	INTC
Interrupt control register 1	ICR1	R/W	H'FF80 201C	32	INTC
Interrupt priority register	INTPRI	R/W	H'FF80 2010	32	INTC
Interrupt source register	INTREQ	R/(W)	H'FF80 2024	32	INTC
Interrupt mask register 0	INTMSK0	R/W	H'FF80 2044	32	INTC
Interrupt mask register 1	INTMSK1	R/W	H'FF80 2048	32	INTC
Interrupt mask clear register 0	INTMSKCLR0	–/W	H'FF80 2064	32	INTC
Interrupt mask clear register 1	INTMSKCLR1	–/W	H'FF80 2068	32	INTC
NMI flag control register	NMIFCR	R/(W)	H'FF80 20C0	32	INTC
User interrupt mask level register	USERIMASK	R/W	H'FF80 3000	32	INTC
Interrupt priority registers	INT2PRI0	R/W	H'FF80 4000	32	INTC2
	INT2PRI1	R/W	H'FF80 4004	32	INTC2
	INT2PRI2	R/W	H'FF80 4008	32	INTC2
	INT2PRI3	R/W	H'FF80 400C	32	INTC2
	INT2PRI4	R/W	H'FF80 4010	32	INTC2
	INT2PRI5	R/W	H'FF80 4014	32	INTC2
	INT2PRI6	R/W	H'FF80 4018	32	INTC2
	INT2PRI7	R/W	H'FF80 401C	32	INTC2
	INT2PRI8	R/W	H'FF80 4020	32	INTC2
	INT2PRI9	R/W	H'FF80 4024	32	INTC2

Name	Abbreviation	R/W	Address	Access Size	Allocated in
Interrupt priority registers	INT2PRI10	R/W	H'FF80 4028	32	INTC2
	INT2PRI11	R/W	H'FF80 402C	32	INTC2
Interrupt source register (not affected by masking state)	INT2A0	R	H'FF80 4038	32	INTC2
Interrupt source register (affected by masking state)	INT2A1	R	H'FF80 403C	32	INTC2
Interrupt mask register	INT2MSKRG	R/W	H'FF80 4040	32	INTC2
Interrupt mask clear register	INT2MSKCR	–/W	H'FF80 4044	32	INTC2
Interrupt detailed source registers	INT2B0	R	H'FF80 4048	32	INTC2
	INT2B1	R	H'FF80 404C	32	INTC2
	INT2B2	R	H'FF80 4050	32	INTC2
	INT2B3	R	H'FF80 4054	32	INTC2
	INT2B4	R	H'FF80 4058	32	INTC2
	INT2B5	R	H'FF80 405C	32	INTC2
	INT2B6	R	H'FF80 4060	32	INTC2
	INT2B7	R	H'FF80 4064	32	INTC2
	INT2B8	R	H'FF80 4068	32	INTC2
	INT2B9	R	H'FF80 406C	32	INTC2
	INT2B10	R	H'FF80 4070	32	INTC2
	INT2B11	R	H'FF80 4074	32	INTC2
	INT2B12	R	H'FF80 4078	32	INTC2
	INT2B13	R	H'FF80 407C	32	INTC2
	INT2B14	R	H'FF80 4080	32	INTC2
	INT2B15	R	H'FF80 4084	32	INTC2
	INT2B16	R	H'FF80 4088	32	INTC2
	INT2B17	R	H'FF80 408C	32	INTC2
	INT2B18	R	H'FF80 4090	32	INTC2
	INT2B19	R	H'FF80 4094	32	INTC2
	INT2B20	R	H'FF80 4098	32	INTC2
	INT2B21	R	H'FF80 409C	32	INTC2
	INT2B22	R	H'FF80 40A0	32	INTC2
	INT2B23	R	H'FF80 40A4	32	INTC2
	INT2B24	R	H'FF80 40A8	32	INTC2

Name	Abbreviation	R/W	Address	Access Size	Allocated in
Interrupt detailed source registers	INT2B25	R	H'FF80 40AC	32	INTC2
	INT2B26	R	H'FF80 40B0	32	INTC2
	INT2B27	R	H'FF80 40B4	32	INTC2
	INT2B28	R	H'FF80 40B8	32	INTC2
	INT2B29	R	H'FF80 40BC	32	INTC2
	INT2B30	R	H'FF80 40C0	32	INTC2
	INT2B31	R	H'FF80 40C4	32	INTC2
	INT2B32	R	H'FF80 40C8	32	INTC2
	INT2B33	R	H'FF80 40CC	32	INTC2
	INT2B34	R	H'FF80 40D0	32	INTC2
	INT2B35	R	H'FF80 40D4	32	INTC2
	INT2B36	R	H'FF80 40D8	32	INTC2
	INT2B37	R	H'FF80 40DC	32	INTC2
	INT2B38	R	H'FF80 40E0	32	INTC2
	INT2B39	R	H'FF80 40E4	32	INTC2
	INT2B40	R	H'FF80 40E8	32	INTC2
	INT2B41	R	H'FF80 40EC	32	INTC2
	INT2B42	R	H'FF80 40F0	32	INTC2
	INT2B43	R	H'FF80 40F4	32	INTC2
	INT2B44	R	H'FF80 40F8	32	INTC2
	INT2B45	R	H'FF80 40FC	32	INTC2
	INT2B46	R	H'FF80 4100	32	INTC2
	INT2B47	R	H'FF80 4104	32	INTC2
	INT2B48	R	H'FF80 4108	32	INTC2
	INT2B49	R	H'FF80 410C	32	INTC2
	INT2B50	R	H'FF80 4110	32	INTC2
	INT2B51	R	H'FF80 4114	32	INTC2
	INT2B52	R	H'FF804118	32	INTC2
Interrupt temporary high-speed mask register	INT2HMS	R/W	H'FF80 4200	32	INTC2
Interrupt temporary high-speed mask clear mode setting register	INT2HMCMS	R/W	H'FF80 4204	32	INTC2

Name	Abbreviation	R/W	Address	Access Size	Allocated in
Interrupt temporary high-speed mask clear register	INT2HMCR	–/W	H'FF80 4208	32	INTC2
Interrupt temporary high-speed mask auto clear status register	INT2HMCRS	R/W	H'FF80 420C	32	INTC2
Interrupt submask register 0	INT2SMSKRG0	R/W	H'FF80 4280	32	INTC2
Interrupt submask register 1	INT2SMSKRG1	R/W	H'FF80 4284	32	INTC2
Interrupt submask register 2	INT2SMSKRG2	R/W	H'FF80 4288	32	INTC2
Interrupt submask register 3	INT2SMSKRG3	R/W	H'FF80 428C	32	INTC2
Interrupt submask register 4	INT2SMSKRG4	R/W	H'FF80 4290	32	INTC2
Interrupt submask register 5	INT2SMSKRG5	R/W	H'FF80 4294	32	INTC2
Interrupt submask register 6	INT2SMSKRG6	R/W	H'FF80 4298	32	INTC2
Interrupt submask clear register 0	INT2SMSKCR0	–/W	H'FF80 42A0	32	INTC2
Interrupt submask clear register 1	INT2SMSKCR1	–/W	H'FF80 42A4	32	INTC2
Interrupt submask clear register 2	INT2SMSKCR2	W	H'FF80 42A8	32	INTC2
Interrupt submask clear register 3	INT2SMSKCR3	W	H'FF80 42AC	32	INTC2
Interrupt submask clear register 4	INT2SMSKCR4	W	H'FF80 42B0	32	INTC2
Interrupt submask clear register 5	INT2SMSKCR5	W	H'FF80 42B4	32	INTC2
Interrupt submask clear register 6	INT2SMSKCR6	W	H'FF80 42B8	32	INTC2

Note: Addresses other than the above must not be write-accessed to. If written to, operation is not guaranteed. If read, an undefined value is read.

Table 7.3 Register States in Each Operating Mode

Abbreviation	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
ICR0	Undefined	Undefined	Retained	Retained	—	Initialized
ICR1	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
INTPRI	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
INTREQ	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
INTMSK0	H'FF00_0000	H'FF00_0000	Retained	Retained	—	Initialized
INTMSK1	H'FF00_0000	H'FF00_0000	Retained	Retained	—	Initialized
INTMSKCLR0	Undefined	Undefined	Retained	Retained	—	Initialized
INTMSKCLR1	Undefined	Undefined	Retained	Retained	—	Initialized
NMIFCR	Undefined	Undefined	Retained	Retained	—	Initialized
USERIMASK	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
INT2PRI0 to INT2PRI11	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
INT2A0	Undefined	Undefined	Retained	Retained	—	Initialized
INT2A1	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
INT2MSKRG	H'FFFF_FFFF	H'FFFF_FFFF	Retained	Retained	—	Initialized
INT2MSKCR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
INT2B0 to INT2B52	Undefined	Undefined	Retained	Retained	—	Initialized
INT2HMS	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
INT2HMCMS	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
INT2HMCR	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
INT2HMCRS	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized
INT2SMSKRG0 to INT2SMSKRG6	H'FFFF_FFFF	H'FFFF_FFFF	Retained	Retained	—	Initialized
INT2SMSKCR0 to INT2SMSKCR6	H'0000_0000	H'0000_0000	Retained	Retained	—	Initialized

Note: Initialized: The value is the one written in "Power-on Reset".

7.4.1 INTC/Interrupt Control Register 0 (ICR0)

Function: ICR0 is a 32-bit readable and partially writable register that sets the input signal detection mode for the external interrupt input pins and NMI pin, and indicates the level being input on the NMI pin.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NMIL	MAI	—	—	—	—	NMIB	NMIE	IRLM0	—	LVL MODE	—	—	—	—	—
Initial value:	—	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
R/W:	R	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Name	Initial Value	R/W	Description
31	NMIL	Undefined	R	NMI Input Level Indicates the signal level being input on the NMI pin. Reading this bit allows the user to know the NMI pin level. 0: Low level is being input on the NMI pin 1: High level is being input on the NMI pin
30	MAI	0	R/W	MAI Interrupt Mask Specifies whether all interrupts are masked while the NMI pin is at the low level regardless of the setting of the BL bit in SR of the CPU. 0: Interrupts remain enabled even when the NMI pin goes low 1: Interrupts are disabled when the NMI pin goes low
29 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Name	Initial Value	R/W	Description
25	NMIB	0	R/W	NMI Block Mode Selects whether an NMI interrupt is held until the BL bit in SR is cleared to 0 or detected immediately when the BL bit in SR of the CPU is set to 1. 0: An NMI interrupt is held when the BL bit in SR is set to 1 (initial value) 1: An NMI interrupt is not held when the BL bit in SR is set to 1 Note: If interrupts are accepted with the BL bit in SR set to 1, information saved for any previous exception (SSR, SPC, SGR, and INTEVT) is lost.
24	NMIE	0	R/W	NMI Edge Selection Selects whether an interrupt request signal to the NMI pin is detected at the rising edge or the falling edge. 0: An interrupt request is detected at the falling edge of NMI input (initial value) 1: An interrupt request is detected at the rising edge of NMI input
23	IRLM0	0	R/W	IRL Pin Mode 0 Selects whether IRQ3 to IRQ0 are used as encoded interrupt requests or as four independent interrupts. 0: Used as the encoded interrupt requests (initial value) 1: Used for independent interrupt inputs
22	—	1	R/W	Reserved Do not change the initial value.
21	LVLMODE	0	R/W	Source Holding Mode Selects operation for the case when an interrupt is canceled before it is accepted by the CPU. 0: Interrupt request is held even if the level-input interrupt (IRQS[1] = 1 for IRL or IRQ) is canceled before it is accepted by the CPU. (initial value) 1: Interrupt request is not held if the level-input interrupt (IRQS[1] = 1 for IRL or IRQ) is canceled before it is accepted by the CPU. (recommended setting value)
20 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Notes: 1. When IRLM0 is changed from 0 to 1, the IRL interrupt source that has been detected or held is cleared. When IRLM0 is changed from 1 to 0, the IRL interrupt source that has been detected or held is not cleared.

2. When using the IRQ [3:0] pins as encoded IRL interrupt inputs, write 1 to IM00 to IM03 in INTMSK0, respectively.

3. Rewriting the LVLMODE bit should be performed by the initialization routine before canceling the masking (INTMASK0 and INTMASK1) for the IRQ interrupts and IRL interrupt. After this, do not rewrite this bit until power-on reset or manual reset is made. The initial value is 0, however, it is recommended to use the INTC after this bit has been set to 1 by the initialization routine.

   For the details of the operation when this bit is set to 0, refer to sections 7.5.2, IRQ Interrupts, 7.5.3, IRL Interrupts, 7.8.1, Example of Handling Routine of IRL Interrupts and Level Detection IRQ Interrupts when ICR0.LVLMODE = 0, and 7.8.3, Clearing IRQ and IRL Interrupt Requests.

7.4.2 INTC/Interrupt Control Register 1 (ICR1)

Function: ICR1 is a 32-bit readable/writable register that specifies the individual input signal detection modes, detection on rising or falling edges, or at the high or low level, for the external interrupt input pins IRQ3 to IRQ0. These settings are only valid for IRQ3 to IRQ0 when set up as individual interrupt inputs by setting the IRLM0 bit in ICR0 to 1.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IRQ0S1	IRQ0S0	IRQ1S1	IRQ1S0	IRQ2S1	IRQ2S0	IRQ3S1	IRQ3S0	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Name	Initial Value	R/W	Description
31	IRQ0S1	0	R/W	IRQn Sense Selection
30	IRQ0S0	0	R/W	Selects whether the corresponding individual pin interrupt signal on the IRQ3 to IRQ0 pins is detected on rising or falling edges, or at the high or low level.
29	IRQ1S1	0	R/W	
28	IRQ1S0	0	R/W	
27	IRQ2S1	0	R/W	
26	IRQ2S0	0	R/W	
25	IRQ3S1	0	R/W	
24	IRQ3S0	0	R/W	

Bit 31 - 2n: IRQnS1	Bit 30 - 2n: IRQnS0	
0	0	The interrupt request is detected on falling edges of the IRQn input.
0	1	The interrupt request is detected on rising edges of the IRQn input.
1	0	The interrupt request is detected at the low level of the IRQn input.
1	1	The interrupt request is detected at the high level of the IRQn input.

Note: n= 0 to 3

Bit	Name	Initial Value	R/W	Description
23 to 16	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Notes: 1. When an IRQ pin is set for level input (IRQnS1 = 1) and the source holding mode (LVLMODE of ICR0) of the interrupt control register 0 (ICR0) is 0, the interrupt source is held until the CPU accepts the interrupt (this is also true for other interrupts). Therefore, even if an interrupt source is disabled before this LSI returns from sleep mode, branching of processing to the interrupt handler when this LSI returns from sleep mode is guaranteed. A held interrupt can be cleared by setting the corresponding interrupt mask bit (the IM bit in the interrupt mask register) to 1 (refer to section 7.8.3, Clearing IRQ and IRL Interrupt Requests).

2. When the IRQnS setting is changed from edge sense (IRQnS is 00 or 01) to level sense (IRQnS is 10 or 11), the IRQ interrupt source that has been edge sensed is cleared. When the IRQnS setting is changed from level sense (IRQnS is 10 or 11) to edge sense (IRQnS is 00 or 01), the IRQ interrupt source that has been sensed or held is cleared. When the IRQnS setting is changed form falling-edge sense (IRQnS is 00) to rising edge sense (IRQnS is 01), or changed from rising edge sense (IRQnS is 01) to the falling edge sense (IRQnS is 00), the IRQ interrupt source that has been sensed before changing the setting is not cleared. Likewise, when IRQnS setting is changed from low-level sense (IRQnS is 10) to high-level sense (IRQnS is 11), or changed from high-level sense (IRQnS is 11) to the low-level sense (IRQnS is 10), the IRQ interrupt source that has been sensed before changing the setting is not cleared.

7.4.3 INTC/Interrupt Priority Register (INTPRI)

Function: INTPRI is a 32-bit readable/writable register used to set the priorities of IRQ3 to IRQ0 (as levels from 15 to 0). These settings are only valid for IRQ3 to IRQ0 when set up as individual interrupt inputs by setting the IRLM0 bit in ICR0 to 1.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IP03	IP02	IP01	IP00	IP13	IP12	IP11	IP10	IP23	IP22	IP21	IP20	IP33	IP32	IP31	IP30
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	Initial Value	R/W	Description
31 to 28	IP03 to IP00	H'0	R/W	These bits set the priority of IRQ0 as an individual pin interrupt request.
27 to 24	IP13 to IP10	H'0	R/W	These bits set the priority of IRQ1 as an individual pin interrupt request.
23 to 20	IP23 to IP20	H'0	R/W	These bits set the priority of IRQ2 as an individual pin interrupt request.
19 to 16	IP33 to IP30	H'0	R/W	These bits set the priority of IRQ3 as an individual pin interrupt request.
15 to 0	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.

Note: Interrupt priorities are established by setting values from H'F to H'1 in each of the 4-bit fields. A larger value corresponds to a higher priority. When the value H'0 is set in a field, the corresponding interrupt request is masked (initial value).

7.4.4 INTC/Interrupt Source Register (INTREQ)

Function: INTREQ is a 32-bit readable and conditionally writable register that indicates which of the individual interrupt request signals, IRQ3 to IRQ0, is currently asserting a request for the INTC. Even if an interrupt is masked by the setting in INTPRI or INTMSK0, operation of the corresponding INTREQ bit is not affected.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IR0	IR1	IR2	IR3	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/(W)	R/(W)	R/(W)	R/(W)	R/(W)	R/(W)	R/(W)	R/(W)	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

				Description	
Bit	**Name**	**Initial Value**	**R/W**	**Edge Detection (IRQ3S to IRQ0S = b'00 or b'01)**	**Level Detection (IRQ3S to IRQ0S = b'10 or b'11)**
31	IR0	0	R/(W)	[When read] 0: The corresponding IRQ interrupt request has not been detected. 1: The corresponding IRQ interrupt request has been detected. [When written] 0: Only bits read as 1 can be cleared to 0. 1: No effect	[When read] (ICR0.LVLMODE = 0) 0: The corresponding interrupt source has not been detected. 1: The corresponding interrupt source has been detected. (ICR0.LVLMODE = 1) 0: The corresponding IRQ interrupt pin is not asserted. 1: The corresponding IRQ interrupt pin is asserted, but the CPU has not accepted the interrupt request yet. Writing has no effect.
30	IR1	0	R/(W)		
29	IR2	0	R/(W)		
28	IR3	0	R/(W)		
27 to 24	—	All 0	R/(W)	Reserved These bits are always read as 0. The write value should always be 0.	
23 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.	

Notes: 1. Write 1 to the bit if it should not be cleared yet.
2. For the method of clearing the IRQ interrupt request that has been detected by level sensing, refer to section 7.8.3, Clearing IRQ and IRL Interrupt Requests.

7.4.5 INTC/Interrupt Mask Registers (INTMSK0, INTMSK1)

Function: INTMSK0 and INTMSK1 are 32-bit readable and conditionally writable registers that set masking for each of the interrupt requests IRQ3 to IRQ0. To clear the mask setting for an interrupt, write 1 to the corresponding bit in INTMSKCLR0/1. Writing 0 to the bits in INTMSK0/1 has no effect. By reading this register once after writing to this register or after clearing the mask by writing INTMSKCLR0/1, the time length necessary for reflecting the register value can be assured (the value read is reflected to the mask status).

When using IRQ3 to IRQ0 pins as encoded IRL interrupt inputs, write 1 to IM00 to IM03, respectively.

(1) INTC/Interrupt Mask Register 0 (INTMSK0)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IM00	IM01	IM02	IM03	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Name	Initial Value	R/W	Description
31	IM00	1	R/W	Sets masking of individual interrupt source on IRQ0.
30	IM01	1	R/W	Sets masking of individual interrupt source on IRQ1.
29	IM02	1	R/W	Sets masking of individual interrupt source on IRQ2.
28	IM03	1	R/W	Sets masking of individual interrupt source on IRQ3.
				[When read] 0: The interrupts are accepted. 1: The interrupts are masked. [When written] 0: No effect 1: Masks the interrupt.
27 to 24	—	All 1	R/W	Reserved These bits are always read as 1. The write value should always be 1.
23 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

(2) INTC/Interrupt Mask Register 1 (INTMSK1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IM10	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Name	Initial Value	R/W	Description
31	IM10	1	R/W	Mask setting for IRQ3 to IRQ0 interrupt sources when pins IRQ3 to IRQ0 are used as encoded interrupt inputs. [When read] 0: The interrupts are accepted. 1: The interrupts are masked. [When written] 0: No effect 1: Masks the interrupt
30 to 24	—	All 1	R/W	Reserved These bits are always read as 1. The write value should always be 1.
23 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

7.4.6 INTC/Interrupt Mask Clear Registers (INTMSKCLR0, INTMSKCLR1)

Function: INTMSKCLR0 and INTMSKCLR1 are 32-bit write-only registers that clear the mask settings for each of the interrupt requests IRQ3 to IRQ0.

(1) INTC/Interrupt Mask Clear Register 0 (INTMSKCLR0)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IC00	IC01	IC02	IC03	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Name	Initial Value	R/W	Description
31	IC00	Undefined	W	Clears masking of individual interrupt source on IRQ0. [When read] Undefined values are read. [When written] 0: No effect 1: Clears the corresponding interrupt mask (enables the interrupt)
30	IC01	Undefined	W	Clears masking of individual interrupt source on IRQ1.
29	IC02	Undefined	W	Clears masking of individual interrupt source on IRQ2.
28	IC03	Undefined	W	Clears masking of individual interrupt source on IRQ3.
27 to 24	—	Undefined	W	Reserved These bits always return undefined values for read. The write value should always be 0.
23 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

(2) INTC/Interrupt Mask Clear Register 1 (INTMSKCLR1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IC10	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Name	Initial Value	R/W	Description
31	IC10	Undefined	W	Clears masking of IRQ3 to IRQ0 interrupt sources when pins IRQ3 to IRQ0 are used as encoded interrupt inputs. [When read] Undefined values are read. [When written] 0: No effect 1: Clears the corresponding interrupt mask (enables the interrupt)
30	—	Undefined	W	Reserved This bit always returns an undefined value for read. The write value should always be 0.
29 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

7.4.7 INTC/NMI Flag Control Register (NMIFCR)

Function: NMIFCR is a register with a flag for the NMI (NMIFL bit) that can be read or cleared by software. The NMIFL bit is automatically set to 1 when an NMI interrupt is detected by the INTC. The bit is cleared by writing 0 by software.

The value of the NMIFL bit does not affect acceptance of the NMI by the CPU. Although an NMI request detected by the INTC is cleared when the CPU accepts the NMI, the NMIFL bit is not cleared automatically. Even if 0 is written to the NMIFL bit before the NMI request is accepted by the CPU, the NMI request is not canceled.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NMIL	—	—	—	—	—	—	—	—	—	—	—	—	—	—	NMIFL
Initial value:	—	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/(W)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Name	Initial Value	R/W	Description
31	NMIL	Undefined	R	NMI Input Level Indicates the level of the signal input to the NMI pin; that is, this bit is read to determine the level on the NMI pin. This bit cannot be modified. 0: The low level is being input to the NMI pin 1: The high level is being input to the NMI pin
30 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	NMIFL	0	R/(W)	NMI Flag (NMI Interrupt Request Detection) Indicates whether an NMI interrupt request signal has been detected. This bit is automatically set to 1 when the INTC detects an NMI interrupt request. Write 0 to clear the bit. Writing 1 to this bit is ignored. [When read] 0: NMI has not been detected 1: NMI has been detected [When written] 0: Clears the NMI flag 1: No effect
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

7.4.8 INTC/User Interrupt Mask Level Register (USERIMASK)

Function: USERIMASK is a register that sets the acceptable interrupt level. This register is allocated to the 64-kbyte page in which the other registers in the INTC are not allocated. Therefore, only this register can be set to be accessible in user mode by changing the address to area 7 address through the MMU.

The interrupts whose level is lower than the level set in the UIMASK bits are masked. When H'F is set in the UIMASK bit, all interrupts other than the NMI are masked.

The interrupts whose level is higher than the level set in the UIMASK bits are accepted under the following conditions. The corresponding interrupt mask bit in the interrupt mask register is cleared to 0 (the interrupt is enabled). The IMASK bit in SR is set lower than its interrupt level. The value of the UIMASK bit does not change even if an interrupt is accepted.

USERIMASK is initialized to 32'H0000 0000 (all interrupts are enabled) by a power-on reset or manual reset.

To prevent incorrect writing caused by software bug, this register should not be written to unless bits 31 to 24 are set to H'A5.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	WKEY								—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	UIMASK				—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Name	Initial Value	R/W	Description
31 to 24	WKEY	H'00	W	These bits should be set to H'A5 when writing to the UIMASK bits. These bits are always read as 0.
23 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 4	UIMASK	H'0	R/W	Interrupt Mask Level The interrupts whose level is equal to or lower than the value set in the UIMASK bits are masked.
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Procedure for Using the User Interrupt Mask Level Register: By setting the interrupt mask level in USERIMASK, the interrupts whose level is equal to or lower than the value set in USERIMASK are disabled. This function is used to disable less urgent interrupts when more urgent processing is performed by the tasks such as device drivers operating in user mode to reduce the processing time.

USERIMASK is allocated in a different 64-kbyte space apart from the one where other INTC registers are allocated. Access to this register in user mode involves address translation by the MMU. In a multitasking OS, the memory-protection functions of the MMU should be used to control the processes that can access USERIMASK. Clear UIMASK to 0 before completing a task or switching to another task. If a task is completed with the UIMASK bits set to a value other than 0, the interrupts whose level is equal to or lower than UIMASK remain disabled. This can lead to problems, for example, the OS may not be able to switch between tasks.

An example procedure for using USERIMASK is described below.

1. Classify interrupts into A and B, as described below. In that case, set the interrupt level of A-type interrupts higher than that of the B-priority interrupts.
   A. Interrupts to be accepted by device drivers (interrupts used in the OS, such as a timer interrupt)
   B. Interrupts that should not be accepted by device drivers
2. Set the MMU so that the access to the address space containing USERIMASK is only allowed for the device drivers that need to disable the interrupts.
3. Branch to the device driver.
4. In the device driver operating in user mode, set the UIMASK bits to mask the B-type interrupts.
5. Process more urgent interrupts in the device driver.
6. Clear the UIMASK bit to 0 and return from the processing by the device driver.

7.4.9 INTC2/Interrupt Priority Registers (INT2PRI0 to INT2PRI11)

Function: INT2PRI0 to INT2PRI11 are 32-bit readable/writable registers that set priority levels (31 to 0) of the on-chip peripheral module interrupts. These registers are initialized to 0 by a reset.

These registers can set the priority of each interrupt source in 32 levels (H'00 and H'01 mask the interrupt request) using the 5-bit field. The CPU interrupt reception interface supports 16 levels by the 4-bit field, while the INTC2 extends the field by 1 bit to select the interrupt source according to the priority level. After selection, each field is converted to a 4-bit value by truncating the lowest bit and is notified. For instance, two interrupt sources with priority levels set to H'1A and H'1B will both be output as the 4-bit priority level H'D. However, if these interrupts occur at the same time, the INTEVT code for the interrupt with the priority level H'1B has priority for output when the 5-bit field is based on.

If multiple interrupts with the same priority level occur at the same time, the INTEVT code to be notified is determined according to the priority order shown in table 7.1.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—						—	—	—						—	—	—						—	—	—					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Table 7.4 shows the correspondence between interrupt request sources and bits in INT2PRI0 to INT2PRI11.

Table 7.4 Interrupt Request Sources and INT2PRI0 to INT2PRI11

	Bits			
Register	**28 to 24**	**20 to 16**	**12 to 8**	**4 to 0**
INT2PRI0	DU	TMU00	TMU10	TMU20, TMU21
INT2PRI1	TMU30, TMU40, TMU50, TMU51	TMU60, TMU70, TMU80	RTC	SDHI0 to SDHI2
INT2PRI2	DEBUG (H-UDI)	SHwy-DMAC0, SHwy-DMAC1	USB2.0 Host/Function	SSI0 to SSI3
INT2PRI3	VIN0	HSPI/RSPI/RQSPI	2DG	LBSC-ATA
INT2PRI4	SCIF0 to SCIF2	SCIF3 to SCIF5	HSCIF	LCDC
INT2PRI5	RCAN0, RCAN1	LBSC-DMAC0, LBSC-DMAC1	LBSC-DMAC2	MMC4.3
INT2PRI6	HPB-DMAC0 to HPB-DMAC3	HPB-DMAC4 to HPB-DMAC10	HPB-DMAC11 to HPB-DMAC18	HPB-DMAC19 to HPB-DMAC22
INT2PRI7	HPB-DMAC23 to HPB-DMAC25, HPB-DMAC27, HPB-DMAC28	I2C0, I2C1	SRC0	SRC1
INT2PRI8	ADC	VIN1	RESET/WDT	HIF
INT2PRI9	ADMAC	FLCTL	GPIO0 to GPIO3	GPIO4, GPIO5
INT2PRI10	STIF0	STIF1	VEU3F (VE30)	GEther
INT2PRI11	MTU2	R-GPVG	MIMLB	IEBus

Note: The larger the value is, the higher the priority is. If the value is set to H'00 or H'01, the request is masked. For details, see the description above.

7.4.10 INTC2/Interrupt Source Register (Not affected by Masking State) (INT2A0)

Function: INT2A0 is a 32-bit read-only register that indicates the interrupt sources of on-chip peripheral modules. Even if an interrupt is masked by the interrupt mask register, the corresponding bit in INT2A0 is set (further interrupt operation is not performed for the corresponding bit). Use INT2A1 instead if the bits for the interrupt sources masked by the interrupt mask registers should not be set.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	VEU3F	SDHI0 to SDHI2	ADMAC	FLCTL	RESET/WDT	HIF	ADC	MTU2	STIF0, STIF1	GPIO0 to GPIO5	GEther	HPB-DMAC0 to HPB-DMAC25, HPB-DMAC27, HPB-DMAC28	LBSC-DMAC0 to LBSC-DMAC2	RCAN0, RCAN1/IEBus	SRC0, SRC1	LBSC-ATA	SCIF0 to SCIF5, HSCIF	LCDC/MIMLB	2DG/R-GPVG	HSPI/RSPI/RQSPI	VIN0, VIN1	SSI0 to SSI3	USB2.0 Host/Function	SHwy-DMAC0, SHwy-DMAC1	DEBUG(H-UDI)	MMC4.3	RTC	IIC3 0, IIC3 1	TMU 30 to TMU 80	TMU 00 to TMU 21	DU
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

When read
0: No interrupt occurred
1: An interrupt occurred

When written
0: nop
1: nop

Note: Interrupt sources can also be identified by directly reading the INTEVT code that was notified to the CPU. In this case, there is no need to read this register.

7.4.11 INTC2/Interrupt Source Register (Affected by Masking State) (INT2A1)

Function: INT2A1 is a 32-bit read-only register that indicates the interrupt sources of on-chip peripheral modules. If an interrupt is masked by the interrupt mask register, the corresponding bit in INT2A1 is not set to 1. Use INT2A0 to check whether interrupts have been generated, regardless of the state of the interrupt mask register.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	VEU3F	SDHI0 to SDHI2	ADMAC	FLCTL	RESET/WDT	HIF	ADC	MTU2	STIF0, STIF1	GPIO0 to GPIO5	GEther	HPB-DMAC0 to HPB-DMAC25, HPB-DMAC27, HPB-DMAC28	LBSC-DMAC0 to LBSC-DMAC2	RCAN0, RCAN1/IEBus	SRC0, SRC1	LBSC-ATA	SCIF0 to SCIF5, HSCIF	LCDC/MIMLB	2DG/R-GPVG	HSPI/RSPI/RQSPI	VIN0, VIN1	SSI0 to SSI3	USB2.0 Host/Function	Shwy-DMAC0, Shwy-DMAC1	DEBUG(H-UDI)	MMC4.3	RTC	IIC3 0, IIC3 1	TMU 30 to TMU 80	TMU 00 to TMU 21	DU
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

When read
0: No interrupt occurred
1: An interrupt occurred

When written
0: nop
1: nop

Note: Interrupt sources can also be identified by directly reading the INTEVT code that was notified to the CPU. In this case, there is no need to read this register.

7.4.12 INTC2/Interrupt Mask Register (INT2MSKRG)

Function: INT2MSKRG is a 32-bit readable/writable register that can set masks for the interrupts indicated in the interrupt source register for each source module. When a bit in this register is set to 1, interrupts from the corresponding module are not notified to the CPU. INT2MSKRG is initialized to 1 (all masked) by a reset.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	VEU3F	SDHI0 to SDHI2	ADMAC	FLCTL	RESET/WDT	HIF	ADC	MTU2	STIF0, STIF1	GPIO0 to GPIO5	GEther	HPB-DMAC0 to HPB-DMAC25, HPB-DMAC27, HPB-DMAC28	LBSC-DMAC0 to LBSC-DMAC2	RCAN0, RCAN1/IEBus	SRC0, SRC1	LBSC-ATA	SCIF0 to SCIF5, HSCIF	LCDC/MIMLB	2DG/R-GPVG	HSPI/RSPI/RQSPI	VIN0, VIN1	SSI0 to SSI3	USB2.0 Host/Function	SHwy-DMAC0, SHwy-DMAC1	DEBUG (H-UDI)	MMC4.3	RTC	IIC3 0, IIC31	TMU 30 to TMU 80	TMU 00 to TMU 21	DU
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

When read
0: Not masked
1: Masked

When written
0: nop
1: Sets the interrupt mask.

Note: With the INTC2, interrupt masks cannot be set or canceled for each of detailed sources of the interrupts generated by each module. To set masks for each detailed source, the registers in the corresponding module should be used.

7.4.13 INTC2/Interrupt Mask Clear Register (INT2MSKCR)

Function: INT2MSKCR is a 32-bit write-only register that clears the mask bits set in the interrupt mask register. When the bit in this register is set to 1, the corresponding interrupt source masking is cleared. The read data is undefined.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	VEU3F	SDHI0 to SDHI2	ADMAC	FLCTL	RESET/WDT	HIF	ADC	MTU2	STIF0, STIF1	GPIO0 to GPIO5	GEther	HPB-DMAC0 to HPB-DMAC25, HPB-DMAC27, HPB-DMAC28	LBSC-DMAC0 to LBSC-DMAC2	RCAN0, RCAN1/IEBus	SRC0, SRC1	LBSC-ATA	SCIF0 to SCIF5, HSCIF	LCDC/MIMLB	2DG/R-GPVG	HSPI/RSPI/RQSPI	VIN0, VIN1	SSI0 to SSI3	USB2.0 Host/Function	SHwy-DMAC0, SHwy-DMAC1	DEBUG (H-UDI)	MMC4.3	RTC	IIC3 0, IIC3 1	TMU 30 to TMU 80	TMU 00 to TMU 21	DU
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

When read
0: Don't care
1: Don't care

When written
0: nop
1: Clears the mask.

Note: With the INTC2, interrupt masks cannot be set or cleared for individual detailed source of interrupts generated by each module. In order to set masks for each detailed source, registers in the corresponding module should be used.

7.4.14 INTC2/Interrupt Detailed Source Registers (INT2B0 to INT2B52)

Function: INT2B0 to INT2B52 are 32-bit read-only registers that indicate more detailed interrupt sources for the interrupt sources of the on-chip peripherals indicated by the interrupt source register. These registers are not affected by the masked state in the interrupt mask register (If interrupt output mask is set in the interrupt source module, the source is not indicated in the INTC2 detailed source register). To mask detailed sources individually, the interrupt mask register or the interrupt enable register of the corresponding module should be set. For details of detailed source in each register, see the specification of the on-chip peripheral module that generates the interrupt.

(0) INT2B0: DU INTEVT = H'3E0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TVR	FRM	—	—	VBK	CMINT	RINT	HBK	ADC 8	ADC 7	ADC 6	ADC 5	ADC 4	ADC 3	ADC 2	ADC 1
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

(1) INT2B1: TMU00 to TMU21 INTEVT = H'400, 420, 440, 460 (00 - 21)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TICPI 21	TUNI 20	TUNI 10	TUNI 00
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

(2) INT2B2: TMU30 to TMU80 (30 - 80) INTEVT = H'480, 4A0, 4C0, 4E0, 500, 520, 540

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TUNI 80	TUNI 70	TUNI 60	TICPI 51	TUNI 50	TUNI 40	TUNI 30
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

(3) INT2B3: RESET, WDT INTEVT = H'560

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ITI
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

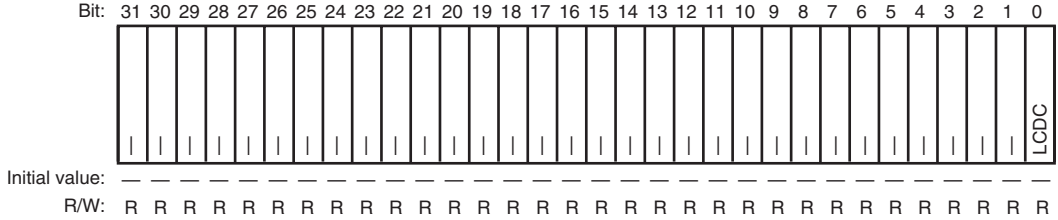
(4) INT2B4: USB2.0 Host/Function INTEVT = H'580

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	USB-f0	USB-h0-EHCI	USB-h0-OHCI
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

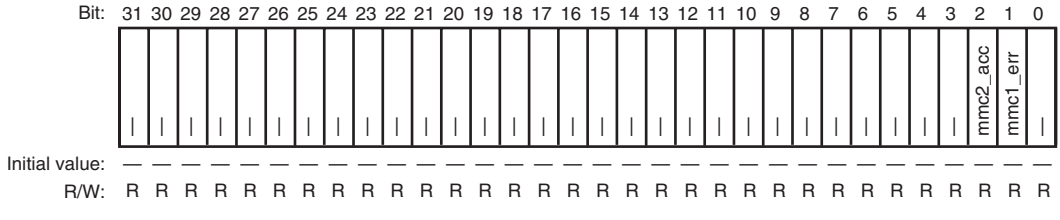
(5) INT2B5: RTC INTEVT = H'C00

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RTC-PRD	RTC-CUP	RTC-ARM
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

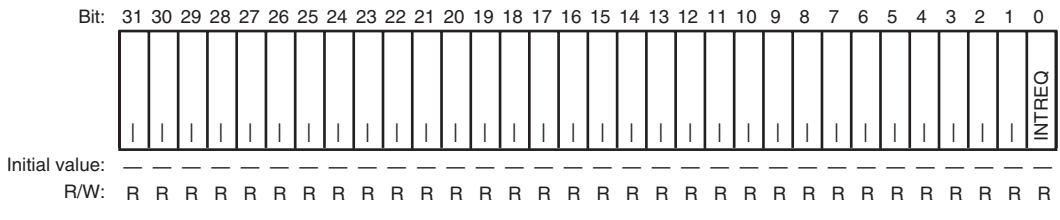
(6) INT2B6: LCDC INTEVT = H'C40



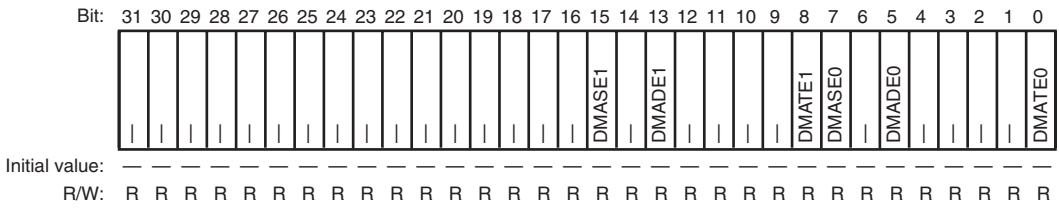
(7) INT2B7: MMC4.3 INTEVT = H'7A0



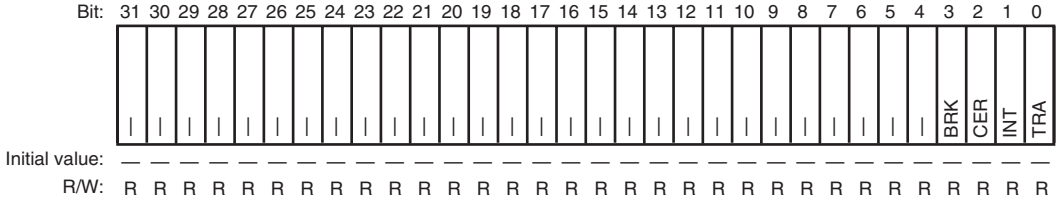
(8) INT2B8: DEBUG (H-UDI) INTEVT = H'600



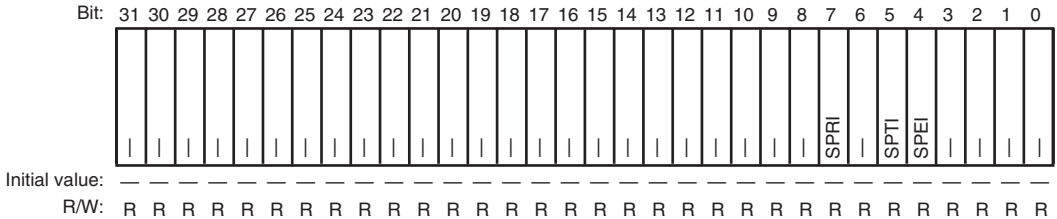
(9) INT2B9: SHwy-DMAC0, SHwy-DMAC1 INTEVT = H'620



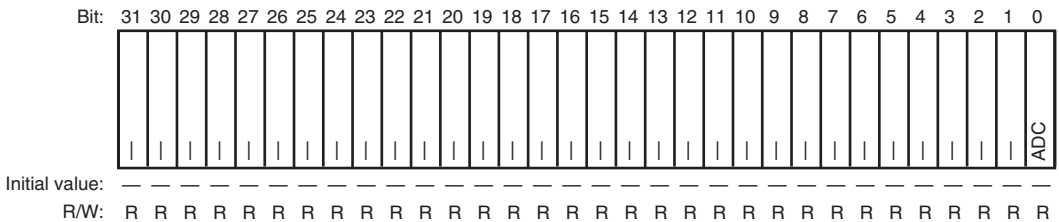
(10) INT2B10: R-GPVG INTEVT = H'760



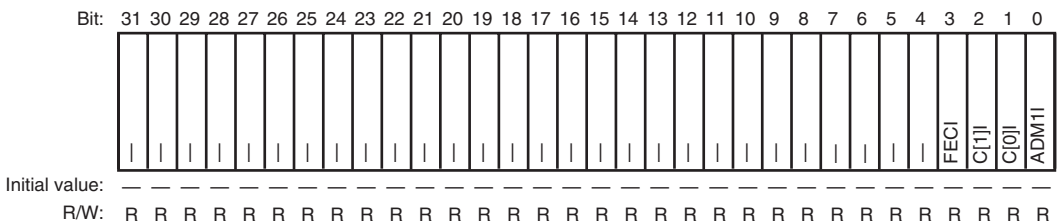
(11) INT2B11: RQSPI INTEVT = H'E60



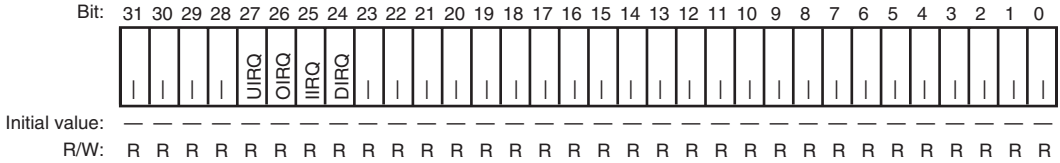
(12) INT2B12: ADC INTEVT = H'E00



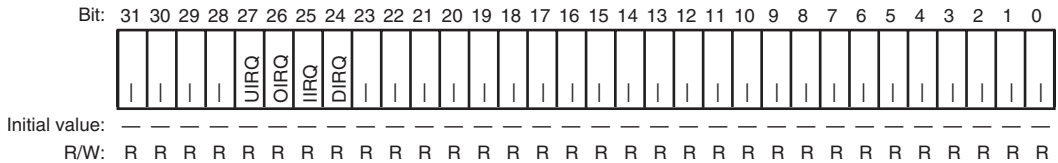
(13) INT2B13: ADMAC INTEVT = H'DA0



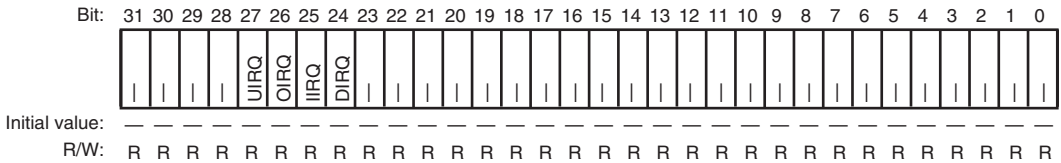
(14) INT2B14: SSI0 INTEVT = H'6C0



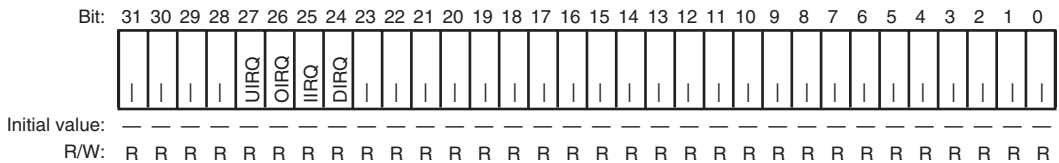
(15) INT2B15: SSI1 INTEVT = H'6E0



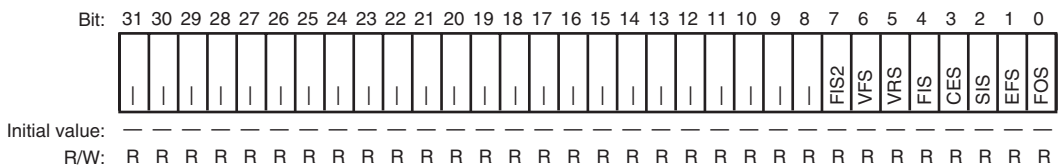
(16) INT2B16: SSI2 INTEVT = H'700



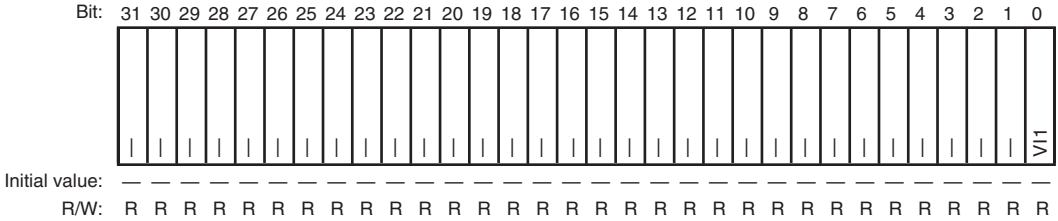
(17) INT2B17: SSI3 INTEVT = H'720



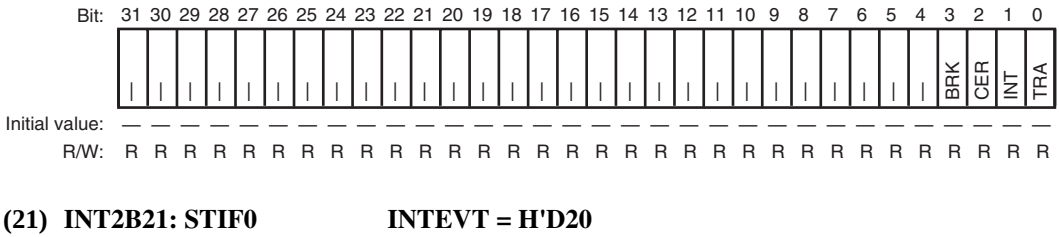
(18) INT2B18: VIN0 INTEVT = H'740



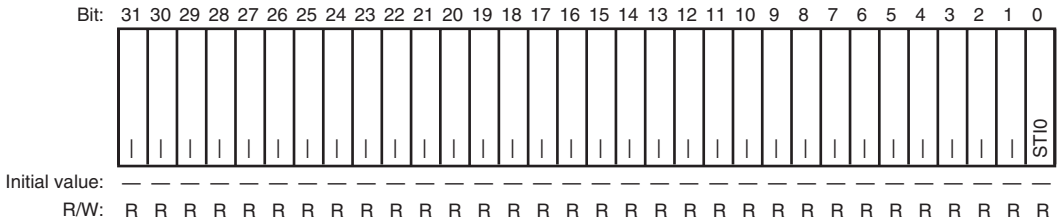
(19) INT2B19:VIN1 INTEVT = H'C20



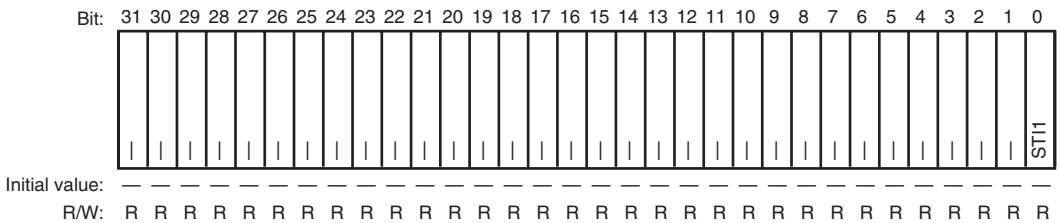
(20) INT2B20: 2DG INTEVT = H'780



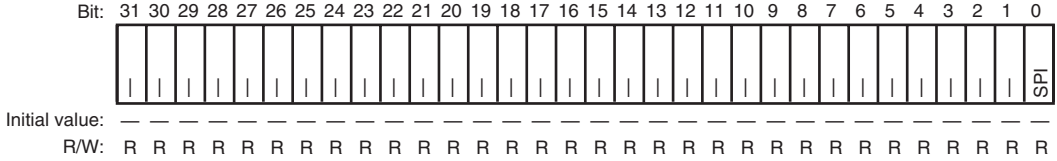
(21) INT2B21: STIF0 INTEVT = H'D20



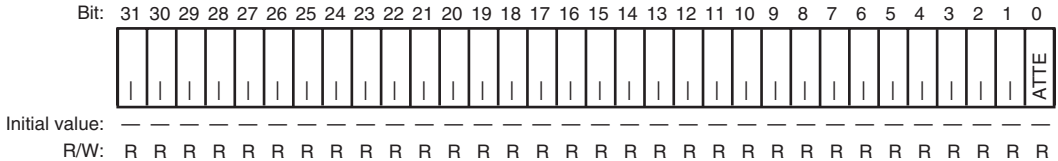
(22) INT2B22: STIF1 INTEVT = H'D40



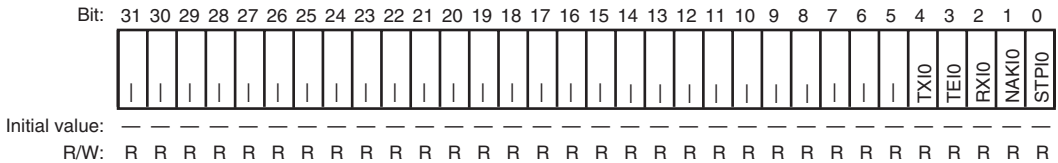
(23) INT2B23: HSPI INTEVT = H'7E0



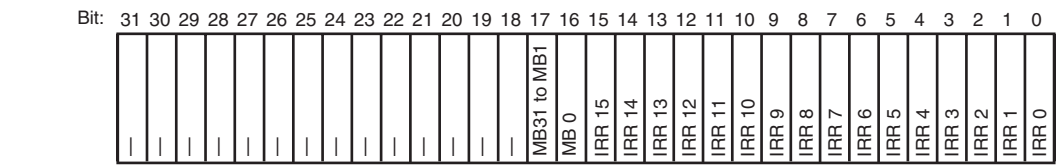
(24) INT2B24: LBSC-ATA INTEVT = H'840



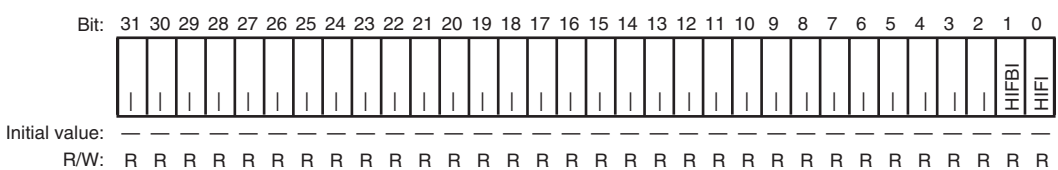
(25) INT2B25: IIC3 0 INTEVT = H'860



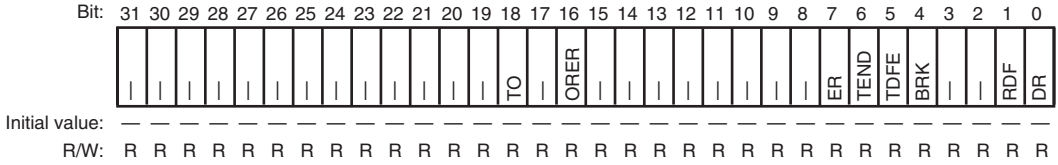
(26) INT2B26: RCAN0 INTEVT = H'880



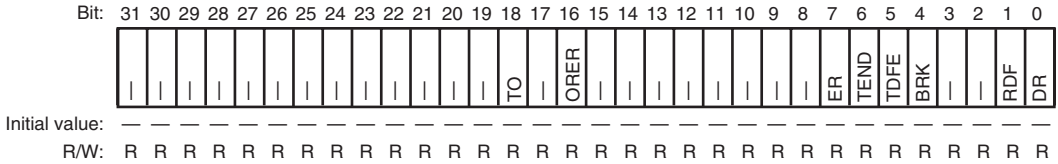
(27) INT2B27: HIF INTEVT = H'DC0



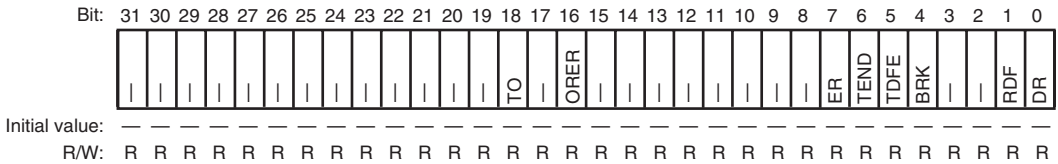
(28) INT2B28: SCIF0 INTEVT = H'8C0



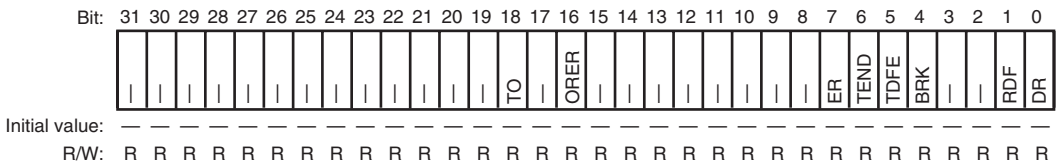
(29) INT2B29: SCIF1 INTEVT = H'8E0



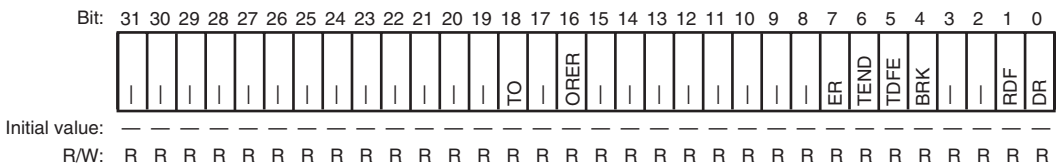
(30) INT2B30: SCIF2 INTEVT = H'900



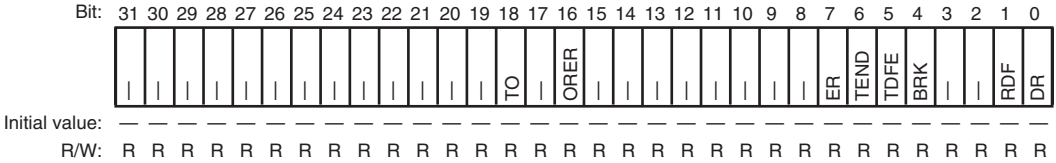
(31) INT2B31: SCIF3 INTEVT = H'920



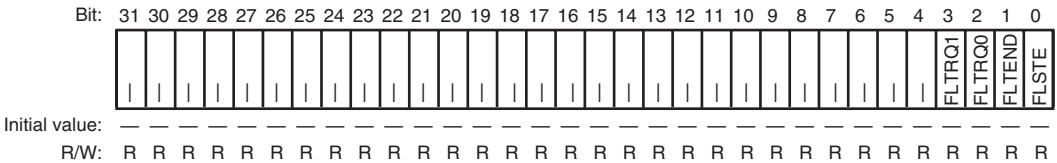
(32) INT2B32: SCIF4 INTEVT = H'940



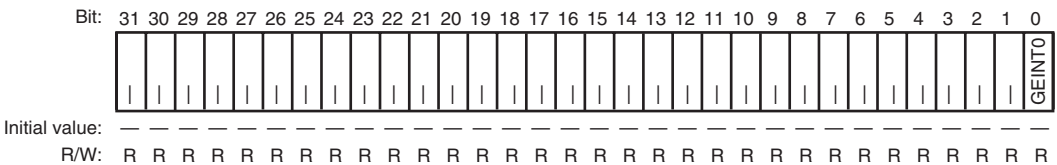
(33) INT2B33: SCIF5 INTEVT = H'960



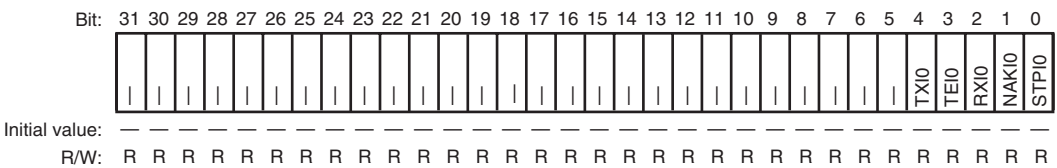
(34) INT2B34: FLCTL INTEVT = H'DE0



(35) INT2B35: GEther INTEVT = H'CA0

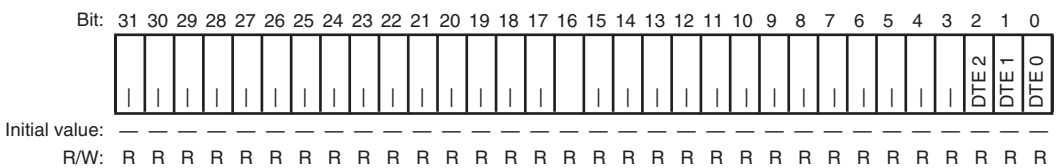


(36) INT2B36: IIC3 1 INTEVT = H'9C0



(37) INT2B37: LBSC-DMAC0 to LBSC-DMAC2

INTEVT = H'9E0, A00, A20 (channel 0 to channel 2)



(38) INT2B38: MTU2 INTEVT = H'E20

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TGI4V	TGI4D	TGI4C	TGI4B	TGI4A	TGI3V	TGI3D	TGI3C	TGI3B	TGI3A	TCI2U	TCI2V	TGI2B	TGI2A	TCI1U	TCI1V	TGI1B	TGI1A	TGI0F	TGI0E	TCI0V	TGI0D	TGI0C	TGI0B	TGI0A
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

(39) INT2B39: RCAN 1 INTEVT = H'A60

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MB31 to MB1	MB 0	IRR 15	IRR 14	IRR 13	IRR 12	IRR 11	IRR 10	IRR 9	IRR 8	IRR 7	IRR 6	IRR 5	IRR 4	IRR 3	IRR 2	IRR 1	IRR 0
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

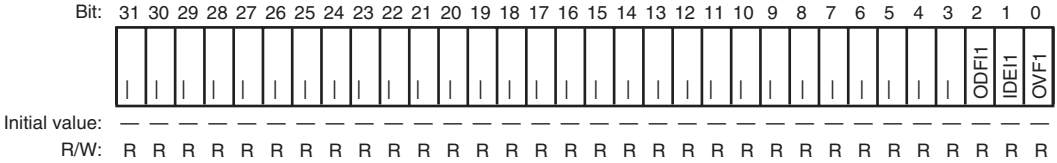
(40) INT2B40: RSPI INTEVT = H'E40

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SPRI	—	SPTI	SPEI	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

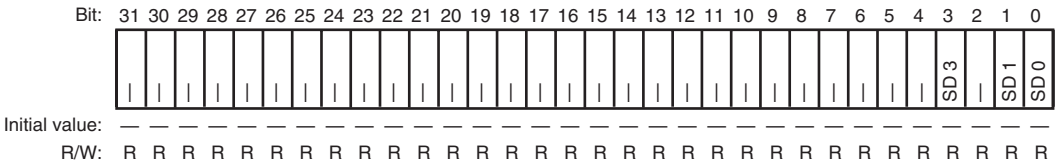
(41) INT2B41: SRC0 INTEVT = H'C60

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ODFI0	IDEI0	OVF0
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

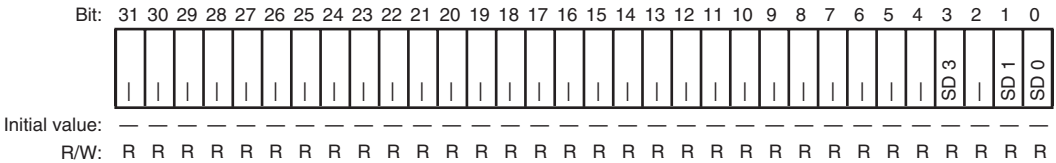
(42) INT2B42: SRC1 INTEVT = H'C80



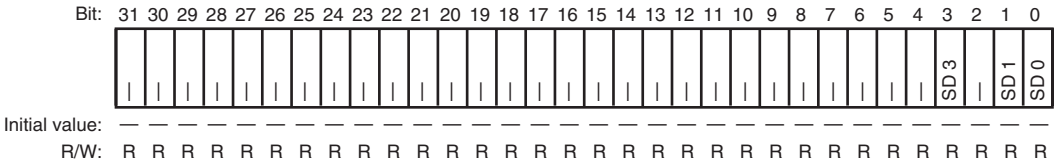
(43) INT2B43: SDHI0 INTEVT = H'AE0



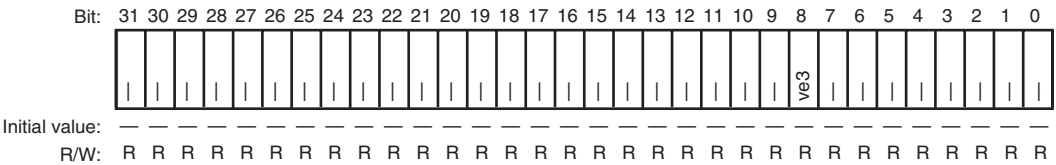
(44) INT2B44: SDHI1 INTEVT = H'B00



(45) INT2B45: SDHI2 INTEVT = H'CC0



(46) INT2B46: VEU3F INTEVT = H'F40



(47) INT2B47: HPB-DMAC0 to HPB-DMAC25, HPB-DMAC27, HPB-DMAC28

INTEVT = H'B60 to H'BE0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	DTE28	DTE27	—	DTE25	DTE24	DTE23	DTE22	DTE21	DTE20	DTE19	DTE18	DTE17	DTE16	DTE15	DTE14	DTE13	DTE12	DTE11	DTE10	DTE9	DTE8	DTE7	DTE6	DTE5	DTE4	DTE3	DTE2	DTE1	DTE0
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

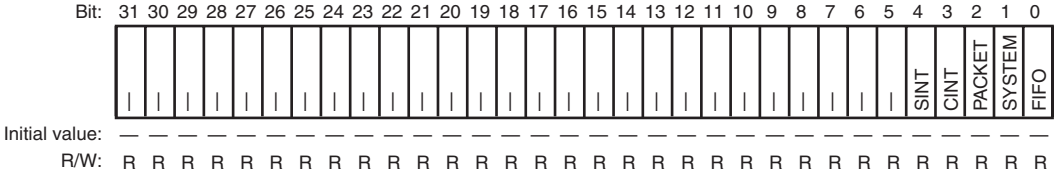
(48) INT2B48: GPIO0 to GPIO3 INTEVT = H'CE0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	3_INTDT31 to 3_INTDT28	3_INTDT27 to 3_INTDT24	3_INTDT23 to 3_INTDT20	3_INTDT19 to 3_INTDT16	3_INTDT15 to 3_INTDT12	3_INTDT11 to 3_INTDT08	3_INTDT07 to 3_INTDT04	3_INTDT03 to 3_INTDT00	2_INTDT31 to 2_INTDT28	2_INTDT27 to 2_INTDT24	2_INTDT23 to 2_INTDT20	2_INTDT19 to 2_INTDT16	2_INTDT15 to 2_INTDT12	2_INTDT11 to 2_INTDT08	2_INTDT07 to 2_INTDT04	2_INTDT03 to 2_INTDT00	1_INTDT31 to 1_INTDT28	1_INTDT27 to 1_INTDT24	1_INTDT23 to 1_INTDT20	1_INTDT19 to 1_INTDT16	1_INTDT15 to 1_INTDT12	1_INTDT11 to 1_INTDT08	1_INTDT07 to 1_INTDT04	1_INTDT03 to 1_INTDT00	0_INTDT31 to 0_INTDT28	0_INTDT27 to 0_INTDT24	0_INTDT23 to 0_INTDT20	0_INTDT19 to 0_INTDT16	0_INTDT15 to 0_INTDT12	0_INTDT11 to 0_INTDT08	0_INTDT07 to 0_INTDT04	0_INTDT03 to 0_INTDT00
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

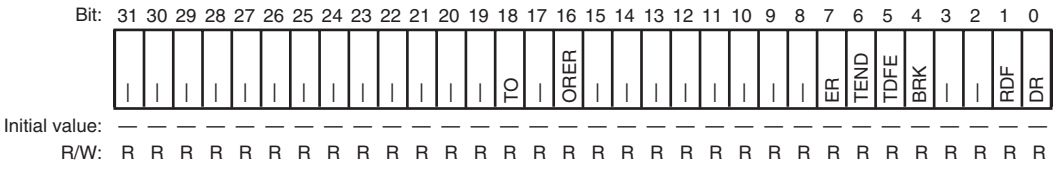
(49) INT2B49: GPIO4, GPIO5 INTEVT = H'D00

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	5_INTDT11 to 5_INTDT08	5_INTDT07 to 5_INTDT04	5_INTDT03, 5_INTDT02, 5_INTDT00	4_INTDT31 to 4_INTDT28	4_INTDT27 to 4_INTDT24	4_INTDT23 to 4_INTDT20	4_INTDT19 to 4_INTDT16	4_INTDT15 to 4_INTDT12	4_INTDT11 to 4_INTDT08	4_INTDT07 to 4_INTDT04	4_INTDT03 to 4_INTDT00
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

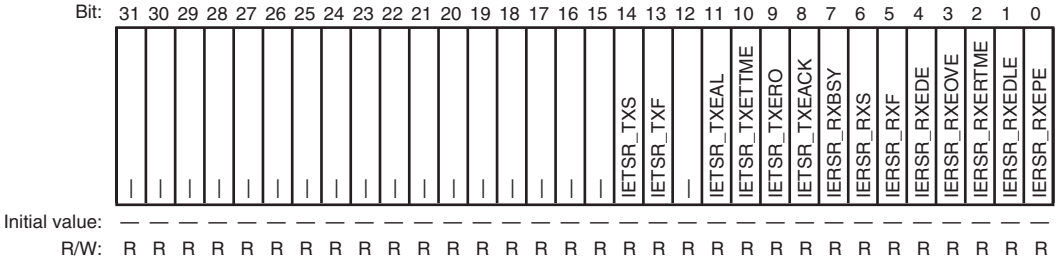
(50) INT2B50: MIMSLB INTEVT = H'8A0



(51) INT2B51: HSCIF INTEVT = H'EC0



(52) INT2B52: IEBus INTEVT = H'B20



7.4.15 INTC2/Interrupt Temporary High-Speed Mask Register (INT2HMS)

Function: INT2HMS is a 32-bit readable/writable register that sets temporary interrupt masks to immediately negate CPU interrupt input. This register is used to prevent the CPU from erroneously receiving the source that was already cleared. For details, see section 7.6.4, Interrupt Temporary High-Speed Mask Function.

This register is initialized to 0 (not masked) by a reset.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	VEU3F	SDHI0 to SDHI2	ADMAC	FLCTL	RESET/WDT	HIF	ADC	MTU2	STIF0, STIF1	GPIO0 to GPIO5	GEther	HPB-DMAC0 to HPB-DMAC25, HPB-DMAC27, HPB-DMAC28	LBSC-DMAC0 to LBSC-DMAC2	RCAN0, RCAN1/IEBus	SRC0, SRC1	LBSC-ATA	SCIF0 to SCIF5, HSCIF	LCDC/MIMLB	2DG/R-GPVG	HSPI/RSPI/RQSPI	VIN0, VIN1	SSI0 to SSI3	USB2.0 Host/Function	SHwy-DMAC0, SHwy-DMAC1	DEBUG (H-UDI)	MMC4.3	RTC	IIC3 0, IIC3 1	TMU 30 to TMU 80	TMU 00 to TMU 21	DU
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

When read
0: Temporary high-speed mask not set
1: Temporary high-speed mask set

When written
0: No effect
1: Sets the temporary high-speed mask.

7.4.16 INTC2/Interrupt Temporary High-Speed Mask Clear Mode Setting Register (INT2HMCMS)

Function: INT2HMCMS is a 32-bit readable/writable register that selects the method of clearing temporary interrupt masks set by INT2HMS. Selection can be made between two methods to clear masks for each module type. First, clearing automatically when the interrupt signal from an interrupt module is negated. Second, clearing by writing INT2HMCR using software. For details, see section 7.6.4, Interrupt Temporary High-Speed Mask Function.

This register is initialized to 0 (the state in which masking is to be disabled using software) by a reset.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	VEU3F	SDHI0 to SDHI2	ADMAC	FLCTL	RESET/WDT	HIF	ADC	MTU2	STIF0, STIF1	GPIO0 to GPIO5	GEther	HPB-DMAC0 to HPB-DMAC25, HPB-DMAC27, HPB-DMAC28	LBSC-DMAC0 to LBSC-DMAC2	RCAN0, RCAN1/IEBus	SRC0, SRC1	LBSC-ATA	SCIF0 to SCIF5, HSCIF	LCDC/MIMLB	2DG/R-GPVG	HSPI/RSPI/RQSPI	VIN0, VIN1	SSI0 to SSI3	USB2.0 Host/Function	SHwy-DMAC0, SHwy-DMAC1	DEBUG (H-UDI)	MMC4.3	RTC	IIC3 0, IIC3 1	TMU 30 to TMU 80	TMU 00 to TMU 21	DU
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
	/W	/W	/W	/W	/W	/W	/W	/W	/W	/W	/W	/W	/W	/W	/W	/W	/W	/W	/W	/W	/W	/W	/W	/W	/W	/W	/W	/W	/W	/W	/W	/W

When read

0: The mode in which masks are cleared by writing to INT2HMCMS using software is set.

1: The mode in which masks are automatically cleared when an interrupt signal from a module is negated is set.

When written

0: Clears masks by writing to INT2HMCR using software.

1: Automatically clears masks when an interrupt signal from a module is negated

7.4.17 INTC2/Interrupt Temporary High-Speed Mask Clear Register (INT2HMCR)

Function: INT2HMCR is a 32-bit register that clears temporary high-speed interrupt masks set by INT2HMS when the method in which interrupt masks are clear by writing this register using software has been selected with INT2HMCMS. For details, see section 7.6.4, Interrupt Temporary High-Speed Mask Function. Writing 0 to this register has no effect. The read values are undefined.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	VEU3F	SDHI0 to SDHI2	ADMAC	FLCTL	RESET/WDT	HIF	ADC	MTU2	STIF0, STIF1	GPIO0 to GPIO5	GEther	HPB-DMAC0 to HPB-DMAC25, HPB-DMAC27, HPB-DMAC28	LBSC-DMAC0 to LBSC-DMAC2	RCAN0, RCAN1/IEBus	SRC0, SRC1	LBSC-ATA	SCIF0 to SCIF5, HSCIF	LCDC/MIMLB	2DG/R-GPVG	HSPI/RSPI/RQSPI	VIN0, VIN1	SSI0 to SSI3	USB2.0 Host/Function	SHwy-DMAC0, SHwy-DMAC1	DEBUG (H-UDI)	MMC4:3	RTC	IIC3 0, IIC3 1	TMU 30 to TMU 80	TMU 00 to TMU 21	DU
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

When read
0: Don't care.
1: Don't care.

When written
0: No effect
1: Clears the temporary high-speed mask state.

Note: This register is valid only when the mode in which masks are cleared by writing to INT2HMCR using software has been set with INT2HMCMS.

7.4.18 INTC2/Interrupt Temporary High-Speed Mask Auto Clear Status Register (INT2HMCRS)

Function: INT2HMCRS indicates that mask setting status specified by INT2HMS was auto-cleared, as a history record. When the mask setting status is auto-cleared, this register bit is set to 1. This history record can be used to identify that individual masks are cleared by software or by interrupt auto clear function. This facilitates software debugging in multitasking operations. History record in this register is cleared by writing 0. Even if clearing the history is omitted, high-speed masking for the subsequent interrupt is not affected. (Once set so however, it is no longer possible to know the exact time of an auto-cleared interrupt from the history record indicated when this register is read next time.)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OR of bits 0 to 30	VEU3F	SDHI0 to SDHI2	ADMAC	FLCTL	RESET/WDT	HIF	ADC	MTU2	STIF0, STIF1	GPIO0 to GPIO5	GEther	HPB-DMAC0 to HPB-DMAC25, HPB-DMAC27, HPB-DMAC28	LBSC-DMAC0 to LBSC-DMAC2	RCAN0, RCAN1/IEBus	SRC0, SRC1	LBSC-ATA	SCIF0 to SCIF5, HSCIF	LCDC/MIMLB	2DG/R-GPVG	HSPI/RSPI/RQSPI	VIN0, VIN1	SSI0 to SSI3	USB2.0 Host/Function	SHwy-DMAC0, SHwy-DMAC1	DEBUG (H-UDI)	MMC4.3	RTC	IIC3 0, IIC3 1	TMU 30 to TMU 80	TMU 00 to TMU 21	DU
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

When read
0: Auto-clear not generated
1: Auto-clear generated

When written
0: Clears history record on auto-clear generation (bit 31 is invalid.)
1: No effect

7.4.19 INTC2/Interrupt Submask Register 0 (INT2SMSKRG0)

Function: INT2SMSKRG0 is a 32-bit readable/writable register that can individually set masks for the interrupts, indicated in the interrupt source register (INT2B40), from the RSPI. When a mask is set, interrupts from the corresponding module are not notified to the CPU. INT2SMSKRG0 is initialized to 1 (all masked) by a reset.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SPRI_smsk	—	SPTI_smsk	SPEI_smsk	—	—	—	—
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

When read
0: Mask not set
1: Mask set

When written
0: No effect
1: Sets a mask

7.4.20 INTC2/Interrupt Submask Register 1 (INT2SMSKRG1)

Function: INT2SMSKRG1 is a 32-bit readable/writable register that can individually set masks for the interrupts, indicated in the interrupt source register (INT2B46), from the VEU3F. When a mask is set, interrupts from the corresponding module are not notified to the CPU. INT2SMSKRG1 is initialized to 1 (all masked) by a reset.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ve3_smsk	—	—	—	—	—	—	—	—
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

When read
0: Mask not set
1: Mask set

When written
0: No effect
1: Sets a mask

7.4.21 INTC2/Interrupt Submask Register 2 (INT2SMSKRG2)

Function: INT2SMSKRG2 is a 32-bit readable/writable register that can individually set masks for the interrupts, indicated in the interrupt source register (INT2B25), from the IIC3 0. When a mask is set, interrupts from the corresponding module are not notified to the CPU. INT2SMSKRG2 is initialized to 1 (all masked) by a reset.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TXI0_smsk	TEI0_smsk	RXI0_smsk	NAKI0_smsk	STPI0_smsk
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

When read
0: Mask not set
1: Mask set

When written
0: No effect
1: Sets a mask

7.4.22 INTC2/Interrupt Submask Register 3 (INT2SMSKRG3)

Function: INT2SMSKRG3 is a 32-bit readable/writable register that can individually set masks for the interrupts, indicated in the interrupt source register (INT2B36), from the IIC3 1. When a mask is set, interrupts from the corresponding module are not notified to the CPU. INT2SMSKRG3 is initialized to 1 (all masked) by a reset.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TXI0_smsk	TEI0_smsk	RXI0_smsk	NAKI0_smsk	STPI0_smsk
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

When read
0: Mask not set
1: Mask set

When written
0: No effect
1: Sets a mask

7.4.23 INTC2/Interrupt Submask Register 4 (INT2SMSKRG4)

Function: INT2SMSKRG4 is a 32-bit readable/writable register that can individually set masks for the interrupts, indicated in the interrupt source register (INT2B12), from the ADC. When a mask is set, interrupts from the corresponding module are not notified to the CPU. INT2SMSKRG4 is initialized to 1 (all masked) by a reset.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ADC_smsk
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

When read
0: Mask not set
1: Mask set

When written
0: No effect
1: Sets a mask

7.4.24 INTC2/Interrupt Submask Register 5 (INT2SMSKRG5)

Function: INT2SMSKRG5 is a 32-bit readable/writable register that can individually set masks for the interrupts, indicated in the interrupt source register (INT2B38), from the MTU2. When a mask is set, interrupts from the corresponding module are not notified to the CPU. INT2SMSKRG5 is initialized to 1 (all masked) by a reset.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TGI4V_smsk	TGI4D_smsk	TGI4C_smsk	TGI4B_smsk	TGI4A_smsk	TGI3V_smsk	TGI3D_smsk	TGI3C_smsk	TGI3B_smsk	TGI3A_smsk	TCI2U_smsk	TCI2V_smsk	TGI2B_smsk	TGI2A_smsk	TCI1U_smsk	TCI1V_smsk	TGI1B_smsk	TGI1A_smsk	TGI0F_smsk	TGI0E_smsk	TGI0V_smsk	TGI0D_smsk	TGI0C_smsk	TGI0B_smsk	TGI0A_smsk
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

When read
0: Mask not set
1: Mask set

When written
0: No effect
1: Sets a mask

7.4.25 INTC2/Interrupt Submask Register 6 (INT2SMSKRG6)

Function: INT2SMSKRG6 is a 32-bit readable/writable register that can individually set masks for the interrupts, indicated in the interrupt source register (INT2B11), from the RQSPI. When a mask is set, interrupts from the corresponding module are not notified to the CPU. INT2SMSKRG6 is initialized to 1 (all masked) by a reset.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																									SPRI_smsk		SPTI_smsk	SPEI_smsk				
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

When read
0: Mask not set
1: Mask set

When written
0: No effect
1: Sets a mask

7.4.26 INTC2/Interrupt Submask Clear Register 0 (INT2SMSKCR0)

Function: INT2SMSKCR0 is a 32-bit write-only register that clears the mask bits set in the interrupt submask register 0. When the bit in this register is set to 1, the corresponding interrupt source masking is cleared. The read data is undefined.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																									SPRI_smsk		SPTI_smsk	SPEI_smsk				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

When read
0: Don't care
1: Don't care

When written
0: No effect
1: Clears a mask

Note: INTC2 does not permit setting or clearing a mask for each of detailed factors which cause interrupts from each module.
If a mask is necessary for each of such detailed factors, set the mask by using a register in the relevant module.

7.4.27 INTC2/Interrupt Submask Clear Register 1 (INT2SMSKCR1)

Function: INT2SMSKCR1 is a 32-bit write-only register that clears the mask bits set in the interrupt submask register 1. When the bit in this register is set to 1, the corresponding interrupt source masking is cleared. The read data is undefined.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ve3_smsk	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

When read
0: Don't care
1: Don't care

When written
0: No effect
1: Clears a mask

7.4.28 INTC2/Interrupt Submask Clear Register 2 (INT2SMSKCR2)

Function: INT2SMSKCR2 is a 32-bit write-only register that clears the mask bits set in the interrupt submask register 2. When the bit in this register is set to 1, the corresponding interrupt source masking is cleared. The read data is undefined.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TXI0_smsk	TEI0_smsk	RXI0_smsk	NAKI0_smsk	STPI0_smsk
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

When read
0: Don't care
1: Don't care

When written
0: No effect
1: Clears a mask

7.4.29 INTC2/Interrupt Submask Clear Register 3 (INT2SMSKCR3)

Function: INT2SMSKCR3 is a 32-bit write-only register that clears the mask bits set in the interrupt submask register 3. When the bit in this register is set to 1, the corresponding interrupt source masking is cleared. The read data is undefined.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																												TXI0_smsk	TEI0_smsk	RXI0_smsk	NAKI0_smsk	STPI0_smsk
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

When read
0: Don't care
1: Don't care

When written
0: No effect
1: Clears a mask

7.4.30 INTC2/Interrupt Submask Clear Register 4 (INT2SMSKCR4)

Function: INT2SMSKCR4 is a 32-bit write-only register that clears the mask bits set in the interrupt submask register 4. When the bit in this register is set to 1, the corresponding interrupt source masking is cleared. The read data is undefined.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
																																ADC_smsk
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

When read
0: Don't care
1: Don't care

When written
0: No effect
1: Clears a mask

7.4.31 INTC2/Interrupt Submask Clear Register 5 (INT2SMSKCR5)

Function: INT2SMSKCR5 is a 32-bit write-only register that clears the mask bits set in the interrupt submask register 5. When the bit in this register is set to 1, the corresponding interrupt source masking is cleared. The read data is undefined.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TGI4V_smsk	TGI4D_smsk	TGI4C_smsk	TGI4B_smsk	TGI4A_smsk	TGI3V_smsk	TGI3D_smsk	TGI3C_smsk	TGI3B_smsk	TGI3A_smsk	TCI2U_smsk	TCI2V_smsk	TGI2B_smsk	TGI2A_smsk	TCI1U_smsk	TCI1V_smsk	TGI1B_smsk	TGI1A_smsk	TGI0F_smsk	TGI0E_smsk	TCI0V_smsk	TGI0D_smsk	TGI0C_smsk	TGI0B_smsk	TGI0A_smsk
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

When read
0: Don't care
1: Don't care

When written
0: No effect
1: Clears a mask

7.4.32 INTC2/Interrupt Submask Clear Register 6 (INT2SMSKCR6)

Function: INT2SMSKCR6 is a 32-bit write-only register that clears the mask bits set in the interrupt submask register 6. When the bit in this register is set to 1, the corresponding interrupt source masking is cleared. The read data is undefined.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SPRI_smsk	—	SPTI_smsk	SPEI_smsk	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

When read
0: Don't care
1: Don't care

When written
0: No effect
1: Clears a mask

7.5 Interrupt Source Descriptions

There are four types of interrupt sources, NMI, IRQ, IRL, and on-chip module interrupts. Each interrupt has a priority level (16 to 0). Level 16 is the highest and level 1 is the lowest. When the level is set to 0, the interrupt is masked and interrupt requests are ignored.

7.5.1 NMI Interrupts

The NMI interrupt has the highest priority of level 16. The interrupt is always accepted unless the BL bit in SR of the CPU is set to 1. In sleep mode, the interrupt is accepted even if the BL bit is set to 1. According to a setting, the NMI interrupt can be accepted even if the BL bit is set to 1.

Input from the NMI pin is detected at the edge. The NMI edge select bit (NMIE) in ICR0 is used to select from the rising or falling edge. After the NMIE bit in ICR0 is modified, the NMI interrupt is not detected for up to six bus clock cycles.

When the INTMU bit in the CPUOPM is set to 1, the interrupt mask level (IMASK) in SR is automatically set to level 15. When the INTMU bit in CPUOPM is set to 0, the IMASK value in SR is not affected by accepting an NMI interrupt.

7.5.2 IRQ Interrupts

(1) Independence from Source Retention Mode (ICR0.LVLMODE) Setting

The IRQ interrupt is valid when 1 is written to the IRLM0 bits in ICR0 and pins IRQ3 to IRQ0 are used for independent interrupts. The rising edge, falling edge, low level, and high level detections are enabled by setting the IRQnS1 and IRQnS0 bits (n = 0 to 3). The priority of interrupts is set by INTPRI.

If an IRQ interrupt request is detected by the low level or high level detection, the pin state in IRQ interrupt should be retained until interrupt handling starts after interrupts are accepted.

When the INTMU bit in CPUOPM is set to 1, the interrupt mask level (IMASK) in SR is automatically set to the level of the accepted interrupt. When the INTMU bit in CPUOPM is set to 0, the IMASK value in SR is not affected by accepting an IRQ interrupt.

(2) Dependence on Source Retention Mode (ICR0.LVLMODE) Setting

For the IRQ interrupt at level detection, there are the following features according to the setting of the source retention mode (ICR0.LVLMODE). The initial value of the ICR0 bit in LVLMODE is 0. Set the bit to 1 before using the INTC.

(a) When ICR0.LVLMODE = 0

After an IRQ interrupt request is detected at the level detection, the source is retained in INTREQ even if the pin state of IRQ interrupts is changed and the request is turned down before the request is accepted by the CPU. The source is retained until the CPU accepts an interrupt (including other interrupts), or the correspondence interrupt mask bit is set to 1.

To clear the IRQ interrupt source retained in the INTC, change the pin state of IRQ interrupts by interrupt routine and withdraw the request. Then, clear the source retained in INTREQ to 0. For details of clearing, see section 7.8.3, Clearing IRQ and IRL Interrupt Requests.

(b) When ICR0.LVLMODE = 1

The INTC does not retain the IRQ interrupt source detected at the level detection.

7.5.3 IRL Interrupts

(1) Independence from Source Retention Mode (ICR0.LVLMODE) Setting

The IRL interrupt is an interrupt input as level from pins IRQ3 to IRQ0.

The priority level is indicated by pins IRQ3 to IRQ0. Pins IRQ3 to IRQ0 indicate the interrupt request with the highest priority (level 15) when these pins are all low. When these pins are all high, these pins indicate no interrupt requests (level 0). Table 7.5 shows the correspondence between the levels on the IRL pins and interrupt priority.

Table 7.5 IRL Interrupt Pins ($\overline{\text{IRL3}}$ to $\overline{\text{IRL0}}$) and Interrupt Levels

IRL3	IRL2	IRL1	IRL0	Interrupt Priority Level		Interrupt Request
Low	Low	Low	Low	15	High	Level 15 interrupt request
Low	Low	Low	High	14		Level 14 interrupt request
Low	Low	High	Low	13		Level 13 interrupt request
Low	Low	High	High	12		Level 12 interrupt request
Low	High	Low	Low	11		Level 11 interrupt request
Low	High	Low	High	10		Level 10 interrupt request
Low	High	High	Low	9		Level 9 interrupt request
Low	High	High	High	8		Level 8 interrupt request
High	Low	Low	Low	7		Level 7 interrupt request
High	Low	Low	High	6		Level 6 interrupt request
High	Low	High	Low	5		Level 5 interrupt request
High	Low	High	High	4		Level 4 interrupt request
High	High	Low	Low	3		Level 3 interrupt request
High	High	Low	High	2		Level 2 interrupt request
High	High	High	Low	1	Low	Level 1 interrupt request
High	High	High	High	0		No interrupt request

IRL interrupt detection requires an on-chip noise-cancellation feature. This detection is performed when the level sampled at every bus clock is the same for three consecutive cycles. This detection can prevent the incorrect level from being taken in when the IRL interrupt pin state changes.

The priority of IRL interrupts should be retained from when an interrupt is accepted to when interrupt handling starts, except the case where it is changed to a higher priority.

When the INTMU bit in CPUOPM is set to 1, the interrupt mask level (IMASK) bit in SR is automatically set to the level of the accepted interrupt. When the INTMU bit in CPUOPM is cleared to 0, the IMASK bit in SR is not affected.

(2) Dependence on Source Retention Mode (ICR0.LVLMODE) Setting

(a) When ICR0.LVLMODE = 0

After an IRL interrupt request is detected, the IRL interrupt with the highest priority is retained in the following cases until the CPU accepts an interrupt (including other interrupts). The cases are where the interrupt request is withdrawn or where the priority is set lower by changing the IRL interrupt pin state.

To clear the retained IRL interrupt source, change the pin state of IRL interrupts by interrupt routine and withdraw the interrupt request. Then, clear the corresponding interrupt mask bit to 1 (To clear the IRL interrupt request by pins IRQ3 to IRQ0, write 1 to the IM10 bit in INTMSK1.).

(b) When ICR0.LVLMODE = 1

The INTC does not retain the IRL interrupt source.

7.5.4 On-Chip Peripheral Module Interrupts

On-chip module interrupts are interrupts generated in the on-chip modules.

The interrupt vectors are not allocated to each source, but the source is reflected on INTEVT. Therefore, the source can be easily identified by branching with the value of INTEVT as offset during exception handling routine. A priority ranging from 31 to 0 can be set for each module by INT2PRI0 to INT2PRI11 of the INTC2. To notify the CPU of the priority, round down the least 1-bit and change to 4 bits. For details, see section 7.5.4 (1), Priority of On-Chip Peripheral Module Interrupts.

When the INTMU bit in CPUOPM is set to 1, the interrupt mask level (IMASK) bit in SR is automatically set to the level of the accepted interrupt. On the other hand, when the INTMU bit in CPUOPM is cleared to 0, the IMASK bit in SR is not affected.

An on-chip peripheral module interrupt source flag or an interrupt enable flag should be updated when the BL bit in SR is set to 1 or when the corresponding interrupt is not generated by the setting of interrupt masking. To prevent the acceptance of incorrect interrupts by the updated interrupt source, read the on-chip peripheral module register that has the corresponding flag. Then, clear the BL bit to 0, use the interrupt temporary high-speed mask function, or update the interrupt masking setting so that the corresponding interrupt can be accepted, after waiting for the on-chip peripheral module priority determination time specified by table 7.7, Interrupt Response Time (for example, read the INTC register operating with the peripheral module clock once). These procedures can guarantee the required timing. When updating multiple flags, read the register that includes the flag updated last.

If a flag is updated when the BL bit is 0, the processing may jump to interrupt processing routine with the INTEVT value cleared to 0. This is because interrupt processing starts, relative to the timing that the flag is updated and the interrupt request is identified in this LSI. In this case, the processing can be continued successfully by executing the RTE instruction.

Note that the GPIO interrupt is a low-active interrupts. Unlike the IRL interrupt and the IRQ interrupt that is detected by the level detection, the GPIO interrupt source cannot be retained by hardware if the pin state is changed and the interrupt request is withdrawn.

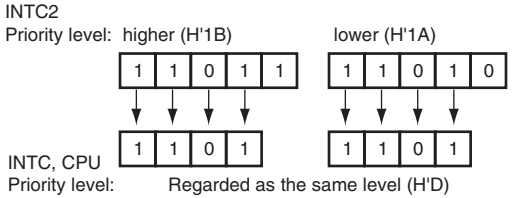
(1) Priority of On-Chip Peripheral Module Interrupts

When an on-chip peripheral module has raised an interrupt to the CPU via the INTC, the INTC2 outputs the exception code specific to each source to the CPU via the INTC. Accepting the interrupt, the CPU indicates the corresponding INTEVT code in INTEVT. An interrupt handler can identify the source by reading from INTEVT in the CPU, without reading the source indicate register in the INTC2. For the correspondence between on-chip peripheral module interrupt sources and exception codes, see table 7.1.

As shown in figure 7.3, an on-chip module interrupt source can be set to 30 levels (H'00 and H'01 mask interrupt requests) by the 5-bit field. The CPU interrupt reception interface supports 15 levels by the 4-bit field and 15 levels are set (H'0 masks interrupt request). Accordingly, the field is extended by one bit to select the interrupt source according to the priority level. After an interrupt with the highest priority is selected, 5-bit data is changed to 4-bit data by truncating the least significant bit, and then is output to the CPU. For example, two interrupt sources with priority levels H'1A and H'1B will be both output to the CPU as H'D, the 4-bit priority level. Although the two interrupt sources have the same priority value, with regard to the INTEVT code that is notified in the case of a contention between two interrupts with the same priority level, priority is given to the INTEVT code corresponding to the H'1B level interrupt because H'1B is higher than H'1A in 5-bit priority selection.

If the interrupts of the same priority coincide, the INTEVT code is notified according to the priority shown in table 7.1.

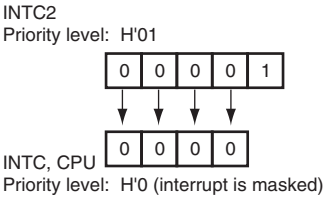
INTC2 distinguishes between priority levels H'1A and H'1B, although both become the same level after truncating the least significant bit for the CPU.



If multiple interrupt requests from on-chip modules occur simultaneously, the INTC2 processes the interrupt with the higher priority level (H'1B in the above case). However, if an external interrupt request is also generated at the same time, the external interrupt request will have higher priority in the following cases:

- an NMI interrupt request is generated.
- an IRQ or IRL interrupt request that has the same or higher priority level (H'D or greater in the above case) is generated.

An interrupt request is masked if priority level H'01 is set.



Priority level H'01 becomes H'00 after the least significant bit is truncated, so the CPU is not notified of the corresponding interrupt. The range of priority levels in the interrupt priority register is H'02 to H'1F (30 priority levels).

Figure 7.3 Priority of On-Chip Peripheral Module Interrupts

(2) On-Chip Peripheral Module Interrupts

Table 7.6 shows the on-chip peripheral modules that generate interrupts to the INTC2. The INTC2 first aggregates interrupts from these modules, and then notifies the CPU of them via the INTC.

Table 7.6 On-Chip Peripheral Modules that Notify INTC2 of Interrupts and Number of Detailed Source Indicate Registers

On-Chip Peripheral Module	Number of Detailed-Source Indicate Registers	Description	On-Chip Peripheral Module	Number of Detailed-Source Indicate Registers	Description
DU	1		LBSC-ATA	1	
TMU	2	One for TMU00 to TMU21, one for TMU30 to TMU80	IIC3 0, IIC3 1	2	Two registers for two channels in the IIC3. (INTEVT is independent.)
RESET, WDT	1		RCAN 0, RCAN 1	2	Two registers for two channels in the RCAN. (INTEVT is independent.)
USB2.0 Host 0, 1/ Function 0	1		HIF	1	
RTC	1		SCIF0 to SCIF5, HSCIF	7	Seven registers for seven channels in the SCIF. (INTEVT is independent.)
LCDC	1		LBSC-DMAC0 to LBSC-DMAC2	1	One register for three channels in the DMAC. (INTEVT is independent.)
MMC4.3	1		FLCTL	1	
DEBUG (H-UDI)	1		GEther	1	—
SHwy-DMAC0, SHwy-DMAC1	1		MTU2	1	
STIF0, STIF 1	2	Two registers for two channels in the STIF. (INTEVT is independent.)	RSPI	1	
ADC	1		SDHI0 to SDHI 2	3	Three registers for three channels in the SDHI. (INTEVT is independent.)
SSI0 to SSI3	4	Four registers for four channels in the SSI. (INTEVT is independent.)	SRC0, SRC1	2	Two registers for Two channels in the SRC. (INTEVT is independent.)
VIN0	1		VEU3F	1	

On-Chip Peripheral Module	Number of Detailed-Source Indicate Registers	Description	On-Chip Peripheral Module	Number of Detailed-Source Indicate Registers	Description
VIN1	1		HPB-DMAC 0 to HPB-DMAC 25, HPB-DMAC 27, HPB-DMAC28	1	One register for 28 channels in the DMAC. (INTEVT cases are merged into 5.)
R-GPVG	1		GPIO0 to GPIO5	2	For every 4 bits corresponding to all 171 port pins for 6 GPIO channels, one bit is allocated within a register.
2DG	1		HSPI	1	
IEBus	1		RQSPI	1	
MIMLB	1		ADMAC	1	

1. The interrupt detailed sources of the on-chip peripheral modules are indicated in the INTC2 internal registers.

   The interrupt detailed sources of the various on-chip modules are indicated in the INTC2 internal registers. Consequently, if the registers of the INTC2 are read without reading the interrupt source registers of the on-chip peripheral modules, it is possible to shorten the bus access latency in order to analyze the sources, and thus to shorten the analysis time.

   (Note: The detailed sources are also indicated by the on-chip peripheral module in which the interrupt was generated, and reading the registers in the interrupt source module does not cause any problems, but reading the registers in the INTC2 is recommended to reduce the time required for interrupt source analysis. However, mask settings and source clearing for the individual detailed sources have to be done with respect to the interrupt source module.)
2. Unification of the detailed source indicate registers in relation to the INTEVT codes

   The INTEVT codes that are reported to the CPU for interrupts are allocated individually to the detailed source indicate registers in a one-to-one correspondence.

   Consequently, if the software reads one detailed source indicate register in the INTC2 that corresponds to an INTEVT code, the detailed source analysis can be completed.

7.6 Operation

7.6.1 Interrupt Sequence

The sequence of interrupt operations is described below.

1. Interrupt request sources send interrupt request signals to the INTC.
2. The INTC selects the interrupt with the highest priority among the interrupts that have been sent, according to the priority set in INTPRI and INT2PRI0 to INT2PRI11. Lower priority interrupts are held as pending interrupts. If two of the interrupts have the same priority level or multiple interrupts by a single module are generated, the interrupt with the highest priority is selected according to table 7.1.
3. The priority level of the interrupt selected by the INTC is compared with the interrupt mask level (IMASK) set in SR of the CPU. Only the interrupt with a higher priority than the IMASK bit is accepted, and an interrupt request signal is sent to the CPU.
4. The CPU accepts an interrupt at the next break between instructions.
5. The interrupt source code is set in the interrupt event register (INTEVT).
6. The status register (SR) and the program counter (PC) are saved in SSR and SPC, respectively. At that time, R15 is saved in SGR.
7. The BL, MD, and RB bits in SR are set to 1.
8. Execution jumps to the start address of the interrupt exception handling routine (the sum of the value set in the vector base register (VBR) and H'0000 0600).

In the exception handling routine, the program branches using the INTEVT value as an offset to identify the interrupt source. This easily enables to branch the exception handling routine to handling routine for each interrupt source.

Notes: 1. When the INTMU bit in CPUOPM is set to 1, all the interrupt mask level (IMASK) in SR is automatically set to the level of the accepted interrupt. When the INTMU bit in CPUOPM is set to 0, the value of the IMASK in SR is not affected by the accepted interrupt.

2. The interrupt source flag should be cleared during exception handling routine. To ensure that an interrupt source which should have been cleared is not erroneously accepted again, read the interrupt source flag after it has been cleared, and wait for the time shown in table 7.7, or use the interrupt temporary high-speed mask function. Then, clear the BL bit or execute an RTE instruction.

3. The IRQ interrupts, IRL interrupts, and on-chip peripheral module interrupts are initialized to the interrupt masking state by a power-on reset. Therefore, clear the interrupt masking for each interrupt (INTMSK0, INTMSK1, INT2MSKRG, or INT2SMSKRG) by using INTMSKCLR0, INTMSKCLR1, INT2MSKCLR, or INTC2IOMSKCLR.

7.6.2 Multiple Interrupts

To handle multiple interrupts, the procedure for the interrupt handling routine should be as follows.

1. To identify the interrupt source, the program branches to the interrupt handling routine for each interrupt source using the INTEVT value as an offset.
2. Clear the corresponding interrupt source in the interrupt handling routine.
3. Save SPC and SSR on the stack.
4. Clear the BL bit in SR. When the INTMU bit in CPUOPM is set to 1, the interrupt mask level (IMASK) in SR is automatically set to the level of the accepted interrupt. On the other hand, when the INTMU bit in CPUOPM is set to 0, software should be used to set the IMASK bit in SR to the priority level of the accepted interrupt.
5. Execute processing as required in response to the interrupt.
6. Set the BL bit in SR to 1.
7. Release SPC and SSR from the stack.
8. Execute the RTE instruction.

By following the above procedure, if further interrupts are generated right after step 4, an interrupt with higher priority than the one currently being handled can be accepted after step 4. This reduces the interrupt response time for urgent processing.

7.6.3 Interrupt Masking by MAI Bit

When the MAI bit in ICR0 is set to 1, all interrupts can be masked while the NMI pin is low regardless of the settings of the BL and IMASK bits in SR. However, only the NMI interrupts are issued whenever the NMI pin state has been changed.

7.6.4 Interrupt Temporary High-Speed Mask Function

(1) Our Approach to the Interrupt Temporary High-Speed Mask Function

The INTC2 has a “interrupt temporary high-speed mask function” that carries out high-speed negation of interrupt signals notified to the CPU for interrupts from the on-chip peripheral modules. With ordinary software, sources are cleared with respect to the module from which the interrupt was generated as part of the interrupt processing routine, and then the BL bit of the CPU SR register is cleared (so that reception of the next interrupt is enabled), but if the BL bit is cleared before the negated state of the pertinent interrupt signal has been propagated to the CPU, there is a possibility that the source that has been cleared is erroneously received again. Conventionally, in order to avoid this, the software had to confirm whether or not the sources of the module that generated the interrupt had been cleared, through processing such as polling, before clearing the BL bit. The interrupt temporary high-speed mask setting function provided for the INTC2 is aimed at improving this processing method.

The software first sets the INTC2 interrupt high-speed mask in the interrupt processing routine. This causes the INTC2 to negate the interrupt notification signal of the INTC and disable the interrupt input to the CPU (if another interrupt exists, it is immediately updated). After setting the temporary high-speed mask, the software clears the sources for the interrupt source module, and then clears the BL bit of SR. This eliminates the possibility of a cleared source being erroneously received again. To cancel the interrupt temporary high-speed mask setting, there are two selectable modes; one is the mode in which the mask setting is automatically canceled when software has cleared the sources for the interrupt source module and the interrupt signal has been negated; the other is the mode in which the mask is canceled directly by software. Software selects the mode in advance for each source. Figure 7.4 shows a conceptual diagram of the interrupt temporary high-speed mask function.

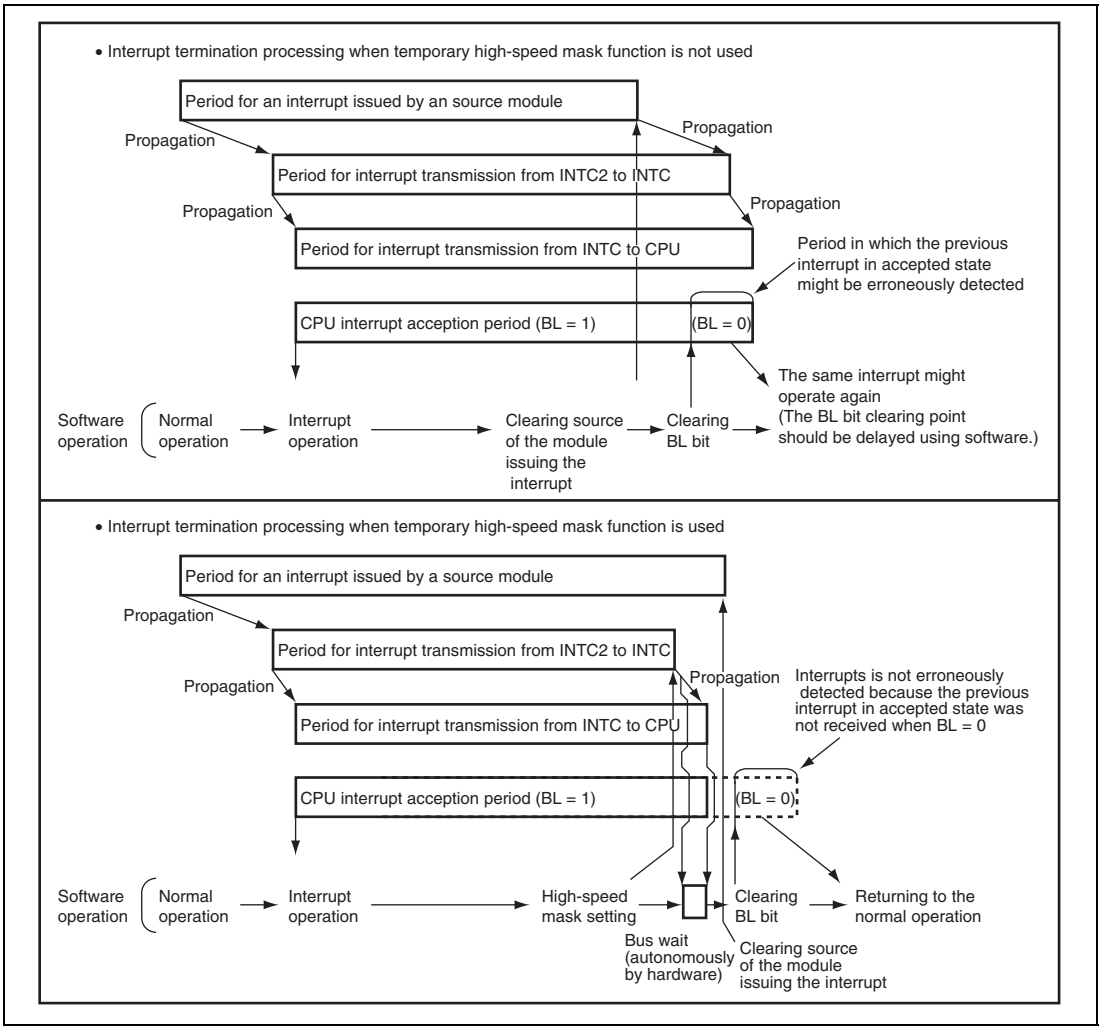
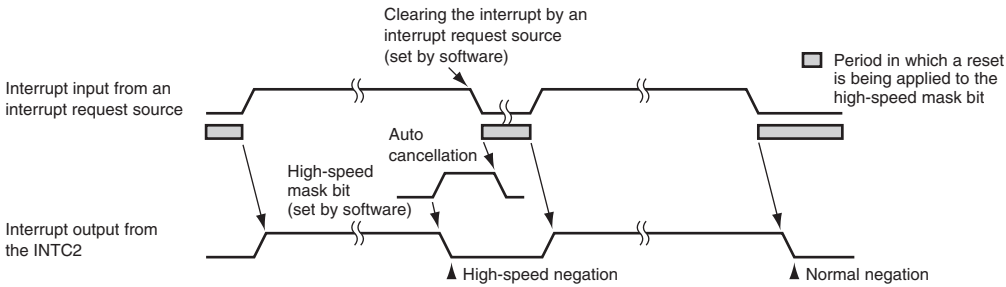


Figure 7.4 Conceptual Diagram of Temporary High-Speed Mask Function

(2) Setting and Cancellation Conditions for Interrupt Temporary High-Speed Mask Register

Setting a mask for the interrupt temporary high-speed mask register is an effective means for carrying out the interrupt routine termination processing smoothly, but caution is required in cases where interrupts are being generated consecutively from the module requesting the interrupts, because the interrupt temporary high-speed mask register has to be manipulated in such a way that the interrupts can be processed individually. The following shows the configuration of the interrupt temporary high-speed mask register, high-speed mask setting conditions, high-speed mask clearing conditions, and invalid access conditions.



[High-speed mask setting conditions]
- In the interrupt temporary high-speed mask auto cancel mode, "1" is set only if the pertinent interrupt has been generated.
- In the interrupt temporary high-speed mask cancel mode by software, as well, "1" is set only if the pertinent interrupt has been generated.

[High-speed mask clearing conditions]
- In the interrupt temporary high-speed mask auto cancel mode, interrupt mask is canceled automatically if an interrupt negated state is detected for an interrupt request source.
  Subsequently, a reset applies to the interrupt temporary high-speed mask bit as long as the negate state is in effect for the interrupt request source.
  Also, in the temporary high-speed mask auto cancel mode, interrupt mask cannot be canceled by writing 0 by software.
- In the interrupt temporary high-speed mask cancel mode by software, interrupt mask can be canceled regardless of the state of the interrupt from an interrupt request source.
  Normally, however, it is assumed that the interrupt temporary high-speed mask bit will be cleared after an interrupt generated by a request source is cleared.

[Invalid access conditions]
- In both the temporary high-speed mask auto cancel mode and the cancel mode by software, the "1" setting for the interrupt temporary high-speed mask register is invalid when no interrupts have been generated by a request source.
- Cancellation access by software is invalid in the interrupt temporary high-speed mask auto cancel mode.
- In the mode in which the interrupt temporary high-speed mask is canceled by software, interrupt temporary high-speed mask is not canceled automatically, even if an interrupt generated by a request source is negated.

Figure 7.5 Configuration of Temporary High-Speed Mask Register and Conditions for Entry to and Exit from High-Speed Masking

In the auto cancel mode, a reset is applied to the interrupt temporary high-speed mask register autonomously by the hardware during an interval with no interrupts between two intervals with interrupts. Consequently, if the interrupt signal is negated before the software sets an interrupt temporary high-speed mask register, no setting will be made for the interrupt temporary high-speed mask register. However, since there is no needs to set an interrupt temporary high-speed mask register for interrupt sources that have already been negated, this is the correct operation.

For modules in which there is no negation period whatsoever between one interrupt and the next for the same interrupt signal, and in the way in which such modules are used, the interrupt temporary high-speed mask for an interrupt bit is not cleared automatically. In such cases, masks must be canceled in interrupt temporary high-speed mask clear mode by software.

7.7 Interrupt Response Time

Table 7.7 shows the interval from generation of an interrupt request until the start of interrupt exception handling and until fetching of the first instruction of the exception handling routine (interrupt response time).

Table 7.7 Interrupt Response Time

	Number of States					
				Peripheral Modules		
Item	NMI	IRL	IRQ	Other than GPIO	GPIO	Remarks
Priority determination time	6Bcyc + 2Pcyc	8Bcyc + 2Pcyc	4Bcyc + 2Pcyc	5Pcyc	7Pcyc	
Wait time until the CPU finishes the current sequence			S-1 (≥ 0) × Icyc			
Interval from the start of interrupt exception handling (saving SR and PC) until a SHwy bus request is issued to fetch the first instruction of the exception handling routine			11Icyc+ 1Scyc			
Response Total time	(S + 10) Icyc + 1Scyc + 6Bcyc + 2Pcyc	(S + 10) Icyc + 1Scyc + 8Bcyc + 2Pcyc	(S + 10) Icyc + 1Scyc + 4Bcyc + 2Pcyc	(S + 10) Icyc + 1Scyc + 5Pcyc	(S + 10) Icyc + 1Scyc + 7Pcyc	

[Legend]

Icyc: Period of one clock cycle of an internal clock pulse that is supplied to the CPU

Scyc: Period of one SHwy clock cycle

Bcyc: Period of one bus clock cycle

Pcyc: Period of one peripheral module clock cycle

S: Number of instruction execution states

Table 7.8 shows the interval from the start of an interrupt exception handling until the start of fetching of the first instruction in exception handling routine (interrupt response time) in the case where the setting values of the registers that enable or disable interrupt, INTMSK0, NTMSK1, and INT2MSKRG are changed from the interrupt disable state to the interrupt enable state.

Table 7.8 Response Time after Changing the Value of Interrupt Enable/Disable Registers (Interrupt Disabled → Interrupt Enabled)

		Number of States			
		IRL	IRQ	Peripheral Modules	Remarks
Item		INTMSK1	INTMSK0	INT2MSKRG	Registers that enable/disable interrupts
Priority determination time*		1Pcyc	1Pcyc	4Pcyc	
Wait time until the CPU finishes the current sequence			S-1 (≥ 0) × Icyc		
Interval from the start of interrupt exception handling (saving SR and PC) until a SuperHyway bus request is issued to fetch the first instruction of the exception handling routine			11Icyc + 1Scyc		
Response time	Total	(S + 10) Icyc + 1Scyc + 1Pcyc	(S + 10) Icyc + 1Scyc + 1Pcyc	(S + 10) Icyc + 1Scyc + 4Pcyc	

[Legend]

Icyc: Period of one clock cycle of an internal clock pulse that is supplied to the CPU

Scyc: Period of one SHwy clock cycle

Bcyc: Period of one bus clock cycle

Pcyc: Period of one peripheral module clock cycle

S: Number of instruction execution states

Note: * When INTMSKCLR0, INTMSKCLR1, and INT2MSKCR are written to, INTMSK0, INTMSK1, and INT2MSKRG enable an interrupt by clearing the mask bits in INTMSK0, INTMSK1, and INT2MSKRG. The priority determination times in table 7.8 are the values after the values of INTMSK0, INTMSK1, and INT2MSKRG are changed.

Table 7.9 shows the time until when the interrupt request signal from the INTC to the CPU is negated in the case where the setting values of the registers, INTMSK0, INTMSK1, and INT2MSKRG are changed from the interrupt enable state to the interrupt disable state.

Table 7.9 Response Time after Changing the Value of Interrupt Enable/Disable Registers (Interrupt Enabled → Interrupt Disabled)

	Number of States			
	IRL	IRQ	Peripheral Modules	Remarks
Item	INTMSK1	INTMSK0	INT2MSKRG	Registers that enable/disable interrupts
Response time	1Pcyc	1Pcyc	4Pcyc	

7.8 Usage Notes

7.8.1 Example of Handling Routine of IRL Interrupts and Level Detection IRQ Interrupts when ICR0.LVLMODE = 0

When the LVLMODE bit in ICR0 is 0, the IRL interrupt request and the level detection IRQ interrupt request are detected even after the CPU has accepted the interrupts. Accordingly, these requests retained in the INTC should be cleared in the interrupt handling routine.

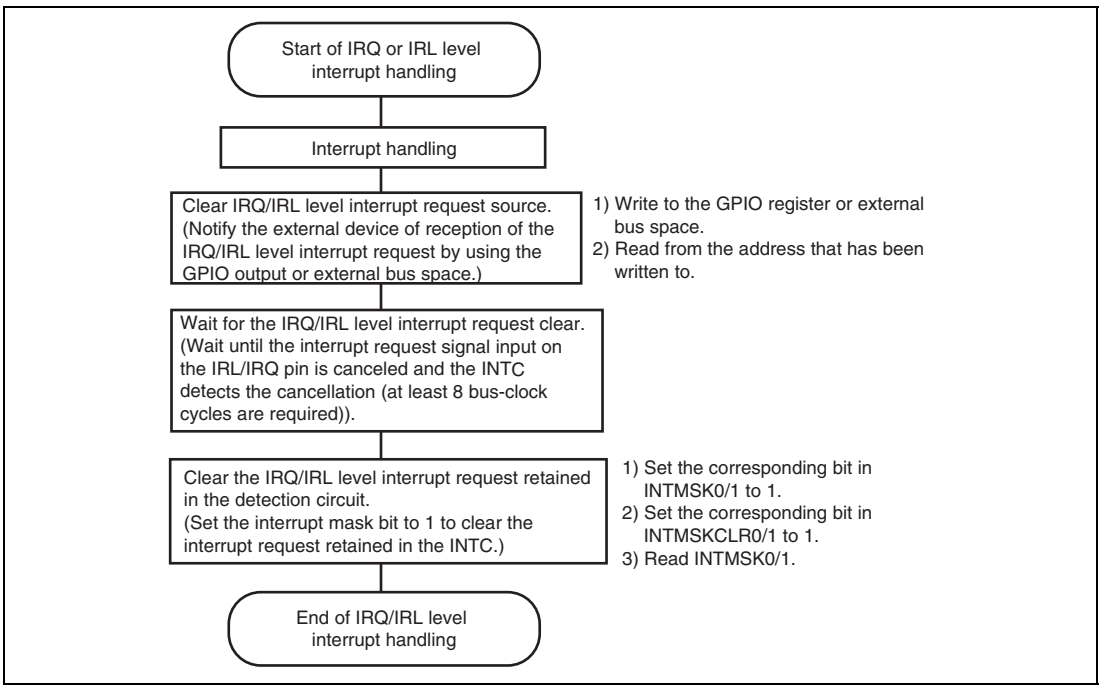


Figure 7.6 Example of Interrupt Handling Routine

For canceling the IRL interrupt request and level detection IRQ interrupt request that the CPU has accepted, the interrupt acceptance by the CPU should be notified to the external device in the interrupt handling routine. The acceptance can be notified to the external device by, for example, outputting data by which acceptance level and pins can be identified, to the GPIO pin, or by writing it to the specific address on the external bus space. At this time, writing to the GPIO register or the external bus space and reading from the address that has been written to should be performed continuously.

After data has been written to the INTC to clear an interrupt request retained in the INTC, a wait time is required until the CPU detects the cancellation of an interrupt request. To secure the wait time, writing to INTMSK0/1 and INTMSKCLR0/1, and reading from INTMSK0/1 should be performed continuously.

7.8.2 Notes on Setting IRQ[3:0] Pin Function

At the mode switching to IRQ or IRL, the INTC may retain an incorrectly detected interrupt request. Therefore, the IRL and IRQ interrupt requests should be masked before the mode switching.

Table 7.10 Switching Sequence of IRQ[3:0] Pin Function

Sequence	Item	Procedure
1	Masking IRL interrupt request and IRQ interrupt request	Write 1 to the valid bits in INTMSK0 and INTMSK1
2	Setting IRQ[3:0] pins to IRL or IRQ	Set the IRLM0 bit in ICR0 to 0 or 1.
3	Starting detection of IRL and IRQ interrupt requests	Write 1 to the corresponding bit in INTMSKCLR0 and INTMSKCLR1

7.8.3 Clearing IRQ and IRL Interrupt Requests

To clear the interrupt request retained in the INTC, follow the procedure below.

(1) Clearing Interrupt Request Independent from Source Retention Mode (ICR0.LVLMODE) Setting

- Clearing IRQ interrupt requests when edge detection is set

  To clear the IRQ3 to IRQ0 interrupt requests for which edge detection is set, read 1 from the IR3 to IR0 bits corresponding to INTREQ and write 0 to the bits. The IRQ interrupt request being detected cannot be cleared even if 1 is written to the corresponding bit in INTMSK0.

(2) Clearing Interrupt Request Dependent on Source Retention Mode (ICR0.LVLMODE) Setting

(a) When ICR0.LVLMODE = 0

- Clearing IRL interrupt requests

  To clear the IRL interrupt requests from the IRQ[3:0] pins, write 1 to the IM10 bit in INTMSK1.
- Clearing IRQ interrupt requests when level detection is set

  To clear the IRQ3 to IRQ0 interrupt requests for which level detection is set, write 1 to the corresponding IM03 to IM00 bits in INTMSK0. The IRQ interrupt request being detected cannot be cleared even if 0 is written to the corresponding bit in INTPRI. The IRQ interrupt request being detected (cleared when the CPU accepts the interrupt) can be checked by reading from INTREQ.

(b) When ICR0.LVLMODE = 1

The INTC does not retain the interrupt source even if IRQ interrupts are detected at level detection or IRL interrupt requests are detected.

Section 8 Clock Pulse Generator (CPG)

8.1 Overview

The clock pulse generator (CPG) module generates clocks to be supplied in the processor and controls the chip states such as reset and sleep.

8.1.1 Features

- Generates internal clocks
  Generates the CPU clock (clki), SHwy clocks (clks and clks1), bus clock (clkb), peripheral clock (clkp).
- Generates external clocks
  Generates the external bus clock (CLKOUT) and DDR clocks (MCK0, $\overline{\text{MCK0}}$, MCK1, and $\overline{\text{MCK1}}$).
- Clock operating mode
  Selects either crystal resonator input or external clock input as a clock input to the CPG.

Note: For details on the clock used in each module, see section for each module.

8.1.2 Block Diagram

Figure 8.1 shows the block diagram of the CPG.

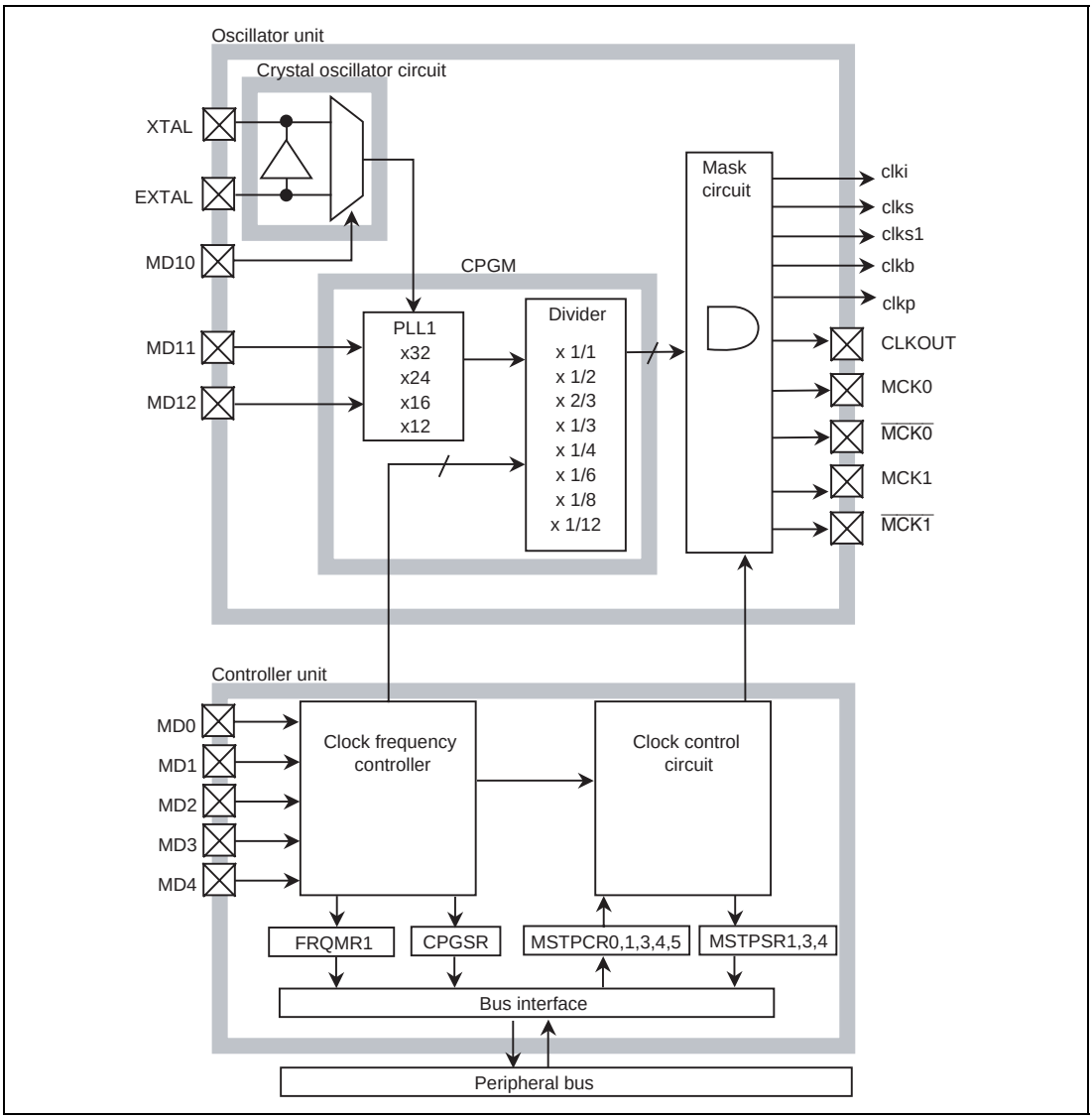


Figure 8.1 Block Diagram

8.1.3 External Pins

Table 8.1 shows the pin configuration of the CPG.

Table 8.1 I/O Pins

Pin Name	I/O	Function
MD0	Input	Switches free-running mode or step-up mode. (For details, see table 8.8.)
MD1	Input	These pins set external bus operating frequency. (For details, see table 8.2.)
MD2	Input	MD3 and MD4 are fixed to 0 and 0, respectively.
MD3	Input	
MD4	Input	
MD10	Input	Selects the crystal resonator or oscillator. (For details, see table 8.7.)
MD11	Input	These pins set the PLL1 multiplication ratio. (For details, see tables 8.3 to 8.5.)
MD12	Input	
XTAL	Output	Used as crystal resonator or external clock input pins. (For details, see table 8.7.)
EXTAL	Input	
CLKOUT	Output	External bus clock (The frequency can be changed by pins MD2 and MD1.)
MCK0	Output	DDR bus clocks
$\overline{\text{MCK0}}$	Output	
MCK1	Output	
$\overline{\text{MCK1}}$	Output	

(1) External Pin Functions

- Clock mode control

  The MD1 to MD4 pins, and MD11 and MD12 pins set multiplication ratio and division ratio of PLL1 in combination.

Table 8.2 MD2 and MD1 Settings

MD2	MD1	
0	0	400-MHz mode (2)
0	1	533-MHz mode
1	0	400-MHz mode (1)
1	1	Setting prohibited

Table 8.3 PLL1 Multiplication Ratio (400-MHz Mode (2))

MD12	MD11	PLL1 Multiplication Ratio	EXTAL Input Minimum Frequency	EXTAL Input Recommended Frequency*	EXTAL Input Maximum Frequency*
0	0	× 12	27.77 MHz (clki = 333.33 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 250.00 MHz, CLKOUT = 41.66 MHz)	33.33 MHz (clki = 400.00 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 266.66 MHz, CLKOUT = 50.00 MHz)	33.33 MHz (clki = 400.00 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 266.66 MHz, CLKOUT = 50.00 MHz)
0	1	× 16	20.83 MHz (clki = 333.33 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 250.00 MHz, CLKOUT = 41.66 MHz)	25.00 MHz (clki = 400.00 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 266.66 MHz, CLKOUT = 50.00 MHz)	25.00 MHz (clki = 400.00 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 266.66 MHz, CLKOUT = 50.00 MHz)
1	0	× 24	13.88 MHz (clki = 333.33 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 250.00 MHz, CLKOUT = 41.66 MHz)	16.66 MHz (clki = 400.00 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 266.66 MHz, CLKOUT = 50.00 MHz)	16.66 MHz (clki = 400.00 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 266.66 MHz, CLKOUT = 50.00 MHz)

MD12	MD11	PLL1 Multiplication Ratio	EXTAL Input Minimum Frequency	EXTAL Input Recommended Frequency*	EXTAL Input Maximum Frequency*
1	1	× 32	10.41 MHz (clki = 333.33 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 166.66 MHz, CLKOUT = 41.66 MHz)	12.50 MHz (clki = 400.00 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 200.00 MHz, CLKOUT = 50.00 MHz)	12.50 MHz (clki = 400.00 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 200.00 MHz, CLKOUT = 50.00 MHz)

Table 8.4 PLL1 Multiplication Ratio (533-MHz Mode)

MD12	MD11	PLL1 Multiplication Ratio	EXTAL Input Minimum Frequency	EXTAL Input Recommended Frequency	EXTAL Input Maximum Frequency
0	0	× 12	41.66 MHz (clki = 500.00 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 250.00 MHz, CLKOUT = 41.66 MHz)	44.44 MHz (clki = 533.33 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 266.66 MHz, CLKOUT = 44.44 MHz)	44.44 MHz (clki = 533.33 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 266.66 MHz, CLKOUT = 44.44 MHz)
0	1	× 16	31.25 MHz (clki = 500.00 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 250.00 MHz, CLKOUT = 41.66 MHz)	33.33 MHz (clki = 533.33 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 266.66 MHz, CLKOUT = 44.44 MHz)	33.33 MHz (clki = 533.33 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 266.66 MHz, CLKOUT = 44.44 MHz)
1	0	× 24	20.83 MHz (clki = 500.00 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 250.00 MHz, CLKOUT = 41.66 MHz)	22.22 MHz (clki = 533.33 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 266.66 MHz, CLKOUT = 44.44 MHz)	22.22 MHz (clki = 533.33 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 266.66 MHz, CLKOUT = 44.44 MHz)
1	1	× 32	15.62 MHz (clki = 500.00 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 250.00 MHz, CLKOUT = 41.66 MHz)	16.66 MHz (clki = 533.33 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 266.66 MHz, CLKOUT = 44.44 MHz)	16.66 MHz (clki = 533.33 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 266.66 MHz, CLKOUT = 44.44 MHz)

Table 8.5 PLL1 Multiplication Ratio (400-MHz Mode (1))

MD12	MD11	PLL1 Multiplication Ratio	EXTAL Input Minimum Frequency	EXTAL Input Recommended Frequency	EXTAL Input Maximum Frequency
0	0	× 12	50.00 MHz (clki = 400.00 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 300.00 MHz, CLKOUT = 50.00 MHz)	51.00 MHz (clki = 408.00 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 306.00 MHz, CLKOUT = 51.00 MHz)	51.00 MHz (clki = 408.00 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 306.00 MHz, CLKOUT = 51.00 MHz)
0	1	× 16	37.50 MHz (clki = 400.00 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 300.00 MHz, CLKOUT = 50.00 MHz)	38.25 MHz (clki = 408.00 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 306.00 MHz, CLKOUT = 51.00 MHz)	38.25 MHz (clki = 408.00 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 306.00 MHz, CLKOUT = 51.00 MHz)
1	0	× 24	25.00 MHz (clki = 400.00 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 300.00 MHz, CLKOUT = 50.00 MHz)	25.50 MHz (clki = 408.00 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 306.00 MHz, CLKOUT = 51.00 MHz)	25.50 MHz (clki = 408.00 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 306.00 MHz, CLKOUT = 51.00 MHz)
1	1	× 32	18.75 MHz (clki = 400.00 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 300.00 MHz, CLKOUT = 50.00 MHz)	19.12 MHz (clki = 408.00 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 306.00 MHz, CLKOUT = 51.00 MHz)	19.12 MHz (clki = 408.00 MHz, MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$ = 306.00 MHz, CLKOUT = 51.00 MHz)

Table 8.6 MD4 and MD3 Settings

MD4	MD3	MD4 and MD3 settings
0	0	Be sure to set both MD4 and MD3 to 0.
0	1	Setting prohibited
1	0	Setting prohibited
1	1	Setting prohibited

- Clock input mode control

  The MD10 pin select either external input or crystal resonator.

Table 8.7 MD10 Settings

MD10	EXTAL/XTAL Pin Description
0	Inputs an external clock to the EXTAL pin.
1	Connects the crystal resonator to the EXTAL and XTAL pins.

- Free-running and step-up mode control

  The MD0 pin selects either free-running mode or step-up mode. In both free-running and step-up modes, the phase-locked loop (PLL1) starts oscillating while the low level is still on the $\overline{\text{PRESET}}$ pin (the PLL1 oscillation settling time must be secured while the $\overline{\text{PRESET}}$ pin is low).

  In step-up mode, the clock signals listed in tables 8.9 to 8.11 are generated by dividing the clock signal from the EXTAL and XTAL pins (input clock signal for PLL1) by (division ratio shown under Step 1) × (1/2) while the $\overline{\text{PRESET}}$ pin is low. After the level on the $\overline{\text{PRESET}}$ pin becomes high, the source of the clock signal for the divider changes from the EXTAL and XTAL pins to PLL1. (While this change is glitch-free, the frequency suddenly rises because of the change to the input clock signal for the divider. The period or duty cycle of the clock signal may also be temporarily perturbed by the change.) After that, the individual clock signals are generated by dividing the PLL1 output clock by the division ratios under Step 1 in tables 8.9 to 8.11. Step 1 is followed by Step 2, and finally each of the clocks operates in the state given under Step 3.

  In free-running mode, on the other hand, each of the clock signals always starts operating in the state shown under Step 3 in tables 8.9 to 8.11 while the level on the $\overline{\text{PRESET}}$ pin is still low.

  (In step-up mode, the LSI frequency is increased step by step to increase LSI power consumption gradually.)

Table 8.8 MD0 Settings

MD0	Description
0	Free-running mode
1	Step-up mode

Table 8.9 Frequency Change in Step-up Mode (Condition: MD2 = 0, MD1 = 0)

	clki	clks	clks1	clkb	MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$	clkp0, clkp1	CLKOUT
Step 1	1/4	1/4	1/8	1/8	1/2	1/8	1/8
Step 2	1/2	1/4	1/4	1/8	1/2	1/8	1/8
Step 3	1/1	1/2	1/4	1/8	1/2	1/8	1/8

Table 8.10 Frequency Change in Step-up Mode (Condition: MD2 = 0, MD1 = 1)

	clki	clks	clks1	clkb	MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$	clkp0, clkp1	CLKOUT
Step 1	1/4	1/4	1/12	1/12	1/2	1/12	1/12
Step 2	1/2	1/4	1/6	1/12	1/2	1/12	1/12
Step 3	1/1	1/3	1/6	1/12	1/2	1/12	1/12

Table 8.11 Frequency Change in Step-up Mode (Condition: MD2 = 1, MD1 = 0)

	clki	clks	clks1	clkb	MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$	clkp0, clkp1	CLKOUT
Step1	1/6	1/6	1/12	1/12	1/2	1/12	1/12
Step2	1/3	1/6	1/6	1/12	1/2	1/12	1/12
Step3	2/3	1/3	1/6	1/12	1/2	1/12	1/12

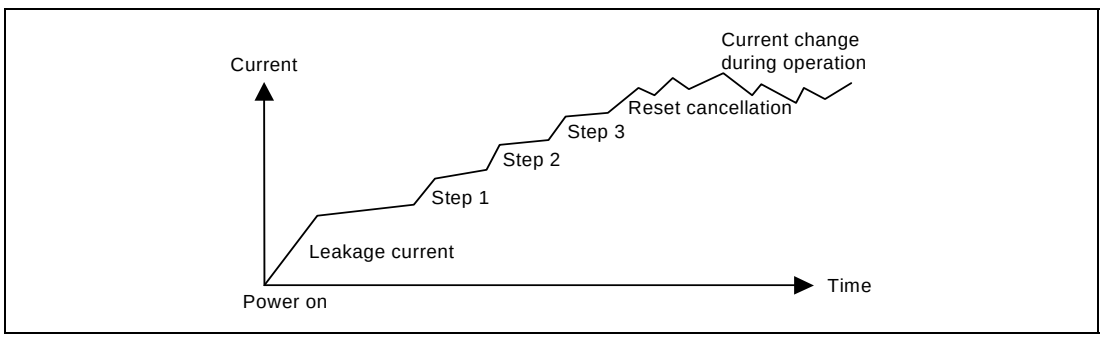


Figure 8.2 Current Change in Step-Up Mode

Table 8.12 Clock Division Ratio in Free-Running Mode

MD2	MD1	clki	clks	clks1	clkb	MCK0, $\overline{\text{MCK0}}$, MCK1, $\overline{\text{MCK1}}$	clkp0, clkp1	CLKOUT
0	0	1/1	1/2	1/4	1/8	1/2	1/8	1/8
0	1	1/1	1/3	1/6	1/12	1/2	1/12	1/12
1	0	2/3	1/3	1/6	1/12	1/2	1/12	1/12

8.2 Register Descriptions

Tables 8.13 and 8.14 show the CPG registers and the states of the registers in each of the processing modes, respectively.

Table 8.13 CPG Registers

Name	Abbr.	R/W	Address for P4	Address for Area 7	Access Size
Frequency control register 0	FRQCR0	R/W	H'FFC8 0000	H'1FC8 0000	32
Frequency control register 2	FRQCR2	R/W	H'FFC8 0008	H'1FC8 0008	32
Frequency display register 1	FRQMR1	R	H'FFC8 0014	H'1FC8 0014	32
Frequency display register 2	FRQMR2	R	H'FFC8 0018	H'1FC8 0018	32

Table 8.14 States of Registers in Each Processing Mode

Name	Abbr.	Power-On Reset $\overline{\text{PRESET}}$ Pin/ WDT/H-UDI	Manual Reset Overflow of WDTCNT/ Multiple Exceptions	Sleep SLEEP Instruction	Software Standby SLEEP Instruction	Module Standby —	Deep Standby SLEEP Instruction
Frequency control register 0	FRQCR0	H'0000 0000	Retained	Retained	Retained	—	H'0000 0000
Frequency control register 2	FRQCR2	H'0000 0000	Retained	Retained	Retained	—	H'0000 0000
Frequency display register 1	FRQMR1	H'x4xx 21xx	Retained	Retained	Retained	—	H'x4xx 21xx
Frequency display register 2	FRQMR2	H'0000 BFFF	Retained	Retained	Retained	—	H'0000 BFFF

Note: Addresses other than the above must not be write-accessed to. If written to, operation is not guaranteed. If read, an undefined value is read.

8.2.1 Frequency Control Register 0 (FRQCR0)

FRQCR0 is a 32-bit readable/writable register that initiates the sequence of changing the frequencies of the various clock signals. This register is automatically cleared to 0 on completion of the sequence. Only longword access to FRQCR0 is possible.

The upper byte of the write value should always be the code value H'CF. Writing values other than H'CF is invalid. If code value is read, it is always read as 0.

This register is reset only by a power-on reset by a $\overline{\text{PRESET}}$ signal, power-on reset caused by a WDT overflow, and power-on reset by the H-UDI.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CODE (H'CF)								—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FRQE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	H'00	R/W	Code Value (H'CF) These bits are always read as 0. The write value should always be H'CF when writing to these bits.
23 to 1	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
0	FRQE	0	R/W	Frequency Change Enable Writing 1 to this bit initiates the sequence of changing the frequencies. This bit is cleared to 0 on completion of the sequence.

8.2.2 Frequency Control Register 2 (FRQCR2)

FRQCR2 is set to change the division rate for the STIF clock (clks2). This register is automatically cleared to 0 on completion of the sequence. The STIF clock frequency should only be changed as part of initializing this LSI.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	S2FC3	S2FC2	S2FC1	S2FC0	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
15	S2FC3	0	R/W	Set the division rate for the STIF clock (clks2)
14	S2FC2	0	R/W	0000: No change
13	S2FC1	0	R/W	0101: 1/6 of the PLL1 output in the CPGM block
12	S2FC0	0	R/W	0110: 1/8 of the PLL1 output in the CPGM block 0111: 1/9 of the PLL1 output in the CPGM block 1000: 1/12 of the PLL1 output in the CPGM block 1001: 1/16 of the PLL1 output in the CPGM block 1010: 1/18 of the PLL1 output in the CPGM block 1011: 1/24 of the PLL1 output in the CPGM block
11 to 0	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.

8.2.3 Frequency Display Register 1 (FRQMR1)

FRQMR1 displays the division rate for the various internal clocks. The base frequency is that of the PLL1 output clock shown in tables 8.3 to 8.5.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IFS3	IFS2	IFS1	IFS0	—	—	—	—	SFS3	SFS2	SFS1	SFS0	BFS3	BFS2	BFS1	BFS0
Initial value:	—	—	0	—	0	1	0	0	0	0	1	—	—	—	—	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MFS3	MFS2	MFS1	MFS0	—	—	—	—	S1FS3	S1FS2	S1FS1	S1FS0	PFS3	PFS2	PFS1	PFS0
Initial value:	0	0	1	0	0	0	0	1	0	1	0	*	—	—	—	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	IFS3	0	R	Display the division rate of the CPU clock (clki)
30	IFS2	0	R	0001: 1/1 1100: 2/3
29	IFS1	0	R	
28	IFS0	1	R	
27	—	0	R	Reserved
26	—	1	R	Reserved
25, 24	—	All 0	R	Reserved
23	SFS3	0	R	Display the division rate of the SHwy clock (clks)
22	SFS2	0	R	0010: 1/2 0011: 1/3
21	SFS1	1	R	
20	SFS0	—	R	
19	BFS3	—	R	Display the division rate of the external bus clock (CLKOUT)
18	BFS2	—	R	
17	BFS1	—	R	0110: 1/8 1000: 1/12
16	BFS0	0	R	
15	MFS3	0	R	Display the division rate of the DDR clock (MCK0/$\overline{\text{MCK0}}$/MCK0/$\overline{\text{MCK0}}$)
14	MFS2	0	R	
13	MFS1	1	R	0010: 1/2
12	MFS0	0	R	
11 to 9	—	All 0	R	Reserved

Bit	Bit Name	Initial Value	R/W	Description
8	—	1	R	Reserved
7	S1FS3	0	R	Display the division rate of the SHwy clock (clks1)
6	S1FS2	1	R	0100: 1/4
5	S1FS1	0	R	0101: 1/6
4	S1FS0	—	R	
3	PFS3	—	R	Display the division rate of the peripheral clock (clkp)
2	PFS2	—	R	0110: 1/8
1	PFS1	—	R	1000: 1/12
0	PFS0	0	R	

8.2.4 Frequency Display Register 2 (FRQMR2)

FRQMR2 displays the division rate of the STIF clock (clks2). The base frequency is that of the PLL1 output clock shown in tables 8.3 to 8.5.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	S2FS3	S2FS2	S2FS1	S2FS0	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0.
15	S2FS3	1	R	Set the division rate of the STIF clock (clks2)
14	S2FS2	0	R	0101: 1/6
13	S2FS1	1	R	0110: 1/8
12	S2FS0	1	R	0111: 1/9 1000: 1/12 1001: 1/16 1010: 1/18 1011: 1/24
11 to 0	—	All 1	R	Reserved These bits are always read as 1.

8.3 Description of Operation

8.3.1 Procedure for Changing the Frequency

The STIF clock (clks2) frequency can be changed by following the procedure below.

1. Set FRQCR2 to a desired value.
2. Set FRQCR0 to H'CF00 0001.
3. Check to ensure that the value written to FRQCR2 matches the current value of FRQMR2.

Note that the STIF clock (clks2) frequency should only be changed as part of initializing this LSI.

[Program Example] Change the divisor for the STIF clock to 8.

```
MOV.L   #FRQCR1,R1
MOV.L   #H'00006000,R0
MOV.L   R0,@R1
```

Corresponds to step 1. The value FRQCR1 is the address of the given register.

```
MOV.L   #FRQCR0,R1
MOV.L   #H'CF000001,R0
MOV.L   R0,@R1
```

Corresponds to step 2. The value FRQCR0 is the address of the given register.

Step 3 in the procedure above is omitted since it merely consists of comparison for confirmation.

8.4 Board Design Consideration

8.4.1 Notes on Using Crystal Resonator

The crystal resonator and capacitors should be placed as close to the XTAL and EXTAL pins as possible. In addition, these pins signal lines should not cross over any other signal lines. Otherwise, induction may interfere with correct oscillation.

The values of the load capacitor and dumping resistance around the crystal resonator should be consulted with the crystal resonator manufacturer.

8.4.2 Notes on Supplying External Clock from EXTAL Pin

Leave the XTAL pin open.

8.4.3 Notes on Using PLL Oscillator Circuit

The VDD-PLL and VSS-PLL lines must be separated from the VDD and VSS lines for the board's power supply. Likewise, the VCCQ-PLL and VSSQ-PLL must be separated from the VCCQ and VSS lines for the board's power supply.

Furthermore, a noise filter should be set up by inserting resister RCB and bypass capacitors CPB and CB as close to these pins as is possible.

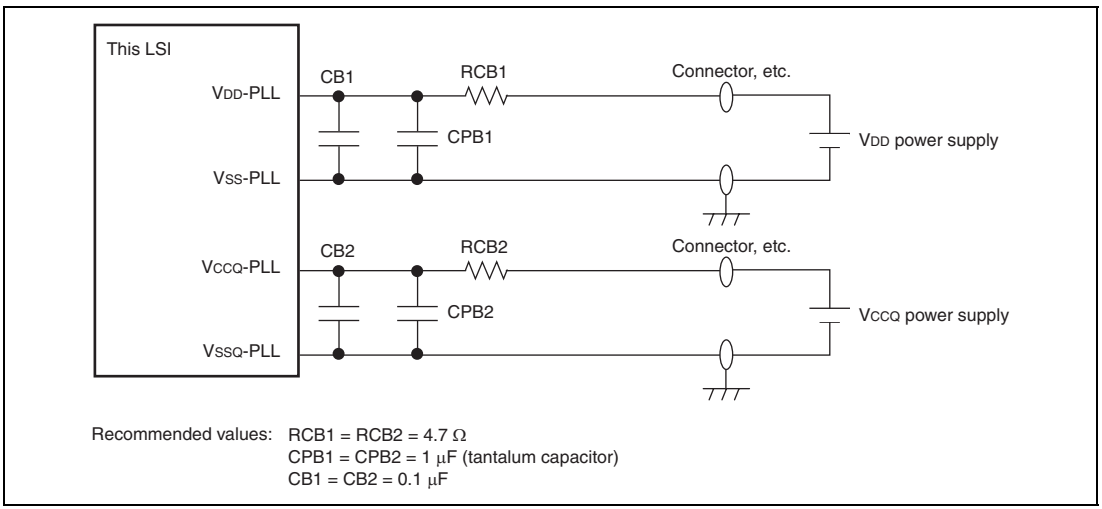


Figure 8.3 Notes on Using PLL Oscillator Circuit

Section 9 Operating Modes and Power-Down Modes

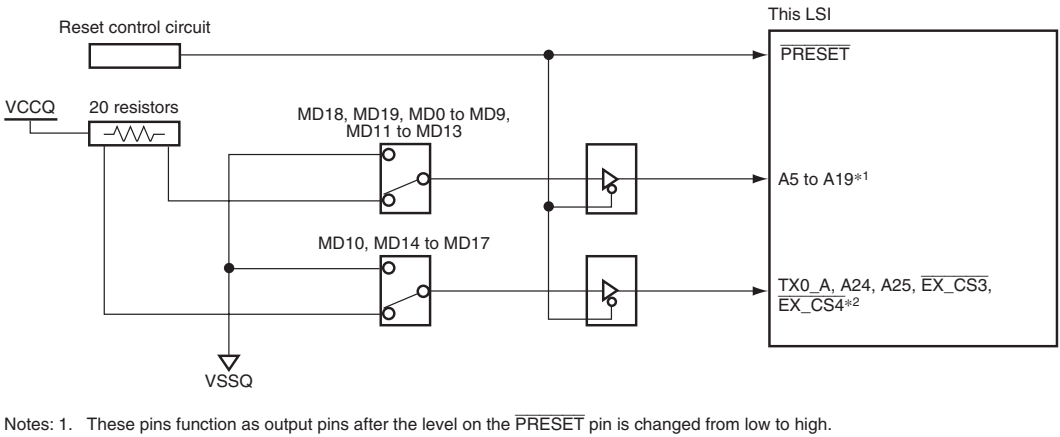
9.1 Overview of Operating Modes

In this LSI, the operating mode is determined by setting the BSMODE, MPMD, MD0 to MD18, and HIFMD (MD19)* pins while the $\overline{\text{PRESET}}$ pin is low.

The clock mode, data bus width of the EX-BUS area 0, area division, endian, addressing mode, and boot mode are determined by the operating mode.

In addition, this LSI supports four power-down modes: sleep mode, software standby mode, deep standby mode, and module standby function.

Note: * The MD0 to MD18 and HIFMD (MD19) pins are multiplexed with peripheral module functional pins. A function that conflicts with the set value of a mode pin should be handled by an external circuit.



Notes: 1. These pins function as output pins after the level on the $\overline{\text{PRESET}}$ pin is changed from low to high.
2. These pins still function as input pins after the level on the $\overline{\text{PRESET}}$ pin is changed from low to high.

Figure 9.1 Reference Diagram of Circuits Used to Control the Operating Mode

9.1.1 Clock Mode

For clock modes, refer to section 8, Clock Pulse Generator (CPG).

9.1.2 Data Bus Width of EX-BUS Area 0

The data bus width of the EX-BUS area 0 is determined by the MD5 and MD6 settings.

For the data bus width of the EX-BUS area 0, refer to section 6B, LBSC within Bus Bridge.

9.1.3 Endian

The endian of the CPU is selected by the MD8 setting.

9.1.4 Addressing Mode

The 29-bit addressing mode and 32-bit addressing mode are switched by the MD13 setting. The division method of area 0 to area 6 is selected by the MD9 and MD7 settings. For details on area division, refer to section 6B, LBSC within Bus Bridge.

9.1.5 Boot Mode

The boot mode is selected by the MD14, MD16, MD17, MD18, and MD19 settings.

For the boot mode settings and operations in each boot mode, refer to section 9.5, Boot Mode.

9.2 Overview of Power-Down Modes

This LSI supports sleep mode, software standby mode, deep standby mode, and module standby function as power-down modes. In power-down modes, functions of the CPU, clocks, on-chip memory, or some on-chip peripheral modules are halted or the power supply is turned off, through which low power consumption is achieved. These modes are canceled by a reset or interrupt.

9.2.1 Power-Down Modes

This LSI has the following power-down modes and function:

1. Sleep mode
2. Software standby mode
3. Deep standby mode
4. Module standby function

Table 9.1 shows the transition conditions for entering the modes from the program execution state, as well as the CPU and peripheral module states in each mode and the procedures for canceling each mode.

Table 9.1 States of Power-Down Modes

		State									
Power-Down Mode	**Transition Condition**	**CPG**	**CPU**	**CPU Register**	**ILRAM/ Cache Memory**	**HIFRAM**	**On-Chip Peripheral Modules**	**Realtime Clock**	**Power Supply**	**External Memory**	**Canceling Methods**
Sleep mode	Execute SLEEP instruction with STBY bit in STBCR cleared to 0	Running	Halted	Held	Halted (contents are held)	Running	Running	Running	Running	Auto-refresh	• Interrupt • Power-on reset • Manual reset • DMA address error
Software standby mode	Execute SLEEP instruction with STBY bit in STBCR set to 1 and DEEP bit cleared to 0	Halted	Halted	Held	Halted (contents are held)	Halted (contents are held)	Halted	Running	Running	Self-refresh	• NMI interrupt • IRQ interrupt • Power-on reset
Deep standby mode	Execute SLEEP instruction with STBY and DEEP bits in STBCR set to 1	Halted	Halted	Halted	Halted (contents are not held)	Halted (RRAMKP specifies whether contents are held or not)	Halted	Running	Halted	Self-refresh	• NMI interrupt • Power-on reset • Realtime clock alarm interrupt • Change on the pins for canceling • Magic-packet detection interrupt from the GETHER module
Module standby function	Set MSTP bit in MSTPCR0, MSTPCR1, MSTPCR3, MSTPCR4, MSTPCR5 to 1	Running	Running	Held	Running	Running	Specified module halted	Halted	Running	Auto-refresh	• Clear MSTP bit to 0 • Power-on reset (except for modules that are initialized to the halted state at a power-on reset)

9.3 Register Configuration

Table 9.2 shows a list of registers. Table 9.3 shows the register states in each processing mode.

Table 9.2 List of Registers

Name	Abbreviation	R/W	P4 Address	Area 7 Address	Access Size
Module standby control register 0	MSTPCR0	R/W	H'FFC8 0030	H'1FC8 0030	32
Module standby control register 1	MSTPCR1	R/W	H'FFC8 0034	H'1FC8 0034	32
Module standby control register 3	MSTPCR3	R/W	H'FFC8 003C	H'1FC8 003C	32
Module standby status register 1	MSTPSR1	R	H'FFC8 0044	H'1FC8 0044	32
Module standby status register 3	MSTPSR3	R	H'FFC8 004C	H'1FC8 004C	32
Module standby status register 4	MSTPSR4	R	H'FFC8 0048	H'1FC8 0048	32
Module standby control register 4	MSTPCR4	R/W	H'FFC8 0050	H'1FC8 0050	32
Module standby control register 5	MSTPCR5	R/W	H'FFC8 0054	H'1FC8 0054	32
Standby control register	STBCR	R/W	H'FFC8 0020	H'1FC8 0020	32
On-chip data-retention RAM area setting register	RRAMKP	R/W	H'FFC7 F000	H'1FC7 F000	8
Deep standby control register	DSCTR	R/W	H'FFC7 F002	H'1FC7 F002	8
Deep standby cancel source select register	DSSSR	R/W	H'FFC7 F004	H'1FC7 F004	16
Deep standby cancel edge select register	DSESR	R/W	H'FFC7 F006	H'1FC7 F006	16
Deep standby cancel source flag register	DSFR	R/(W)*	H'FFC7 F008	H'1FC7 F008	16
Data-Retention RAM boot address register	HIAR	R/W	H'FFC7 F012	H'1FC7 F012	16
Area 0 boot address register low	BARL	R/W	H'FFC7 F014	H'1FC7 F014	16
Area 0 boot address register high	BARH	R/W	H'FFC7 F016	H'1FC7 F016	16

Table 9.3 Register States in Each Processing Mode

Abbreviation	Initial Value	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
MSTPCR0	H'0000 0000	Initialized	Retained	Retained	Retained	—	Initialized
MSTPCR1	H'0000 C3C9	Initialized	Retained	Retained	Retained	—	Initialized
MSTPCR3	H'C0FF B013	Initialized	Retained	Retained	Retained	—	Initialized
MSTPSR1	H'0080 C3C9	Initialized	Retained	Retained	Retained	—	Initialized
MSTPSR3	H'C0FF B013	Initialized	Retained	Retained	Retained	—	Initialized
MSTPSR4	H'0000 0028	Initialized	Retained	Retained	Retained	—	Initialized
MSTPCR4	H'0000 0028	Initialized	Retained	Retained	Retained	—	Initialized
MSTPCR5	H'0000 0004	Initialized	Retained	Retained	Retained	—	Initialized
STBCR	H'0000 0000	Initialized	Retained	Retained	Retained	—	Initialized
RRAMKP	H'00	Initialized	Retained	Retained	Retained	—	Initialized
DSCTR	H'00	Initialized	Retained	Retained	Retained	—	Initialized
DSSSR	H'0000	Initialized	Retained	Retained	Retained	—	Initialized
DSESR	H'0000	Initialized	Retained	Retained	Retained	—	Initialized
DSFR	H'0000	Initialized*[1]	Retained*[2]	Retained	Retained	—	Retained
HIAR	H'0000	Initialized	Retained	Retained	Retained	—	Initialized
BARL	H'0000	Initialized	Retained	Retained	Retained	—	Initialized
BARH	H'0000	Initialized	Retained	Retained	Retained	—	Initialized

Notes: Addresses other than the above must not be write-accessed to. If written to, correct operation is not guaranteed. If read, an undefined value is read.

Registers except for DSFR are initialized when returning from deep standby mode.

1. Retained pin states are released.
2. Retention of pin states continues.

9.3.1 Module Standby Control Register 0 (MSTPCR0)

Before handling a bit in MSTPCR0, check that the module corresponding to the bit is not operating. If a bit in MSTPCR0 is handled while the module corresponding to the bit is operating, correct operation of the module cannot be guaranteed. Setting a bit in this register to 1 stops supply of the clock signal to the corresponding module and clearing a bit to 0 enables clock supply to the corresponding module.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	MSTP030	MSTP029	—	—	MSTP026	MSTP025	MSTP024	MSTP023	MSTP022	MSTP021	—	MSTP019	—	—	MSTP016
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MSTP015	MSTP014	—	MSTP012	MSTP011	MSTP010	MSTP009	MSTP008	MSTP007	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30	MSTP030	0	R/W	Controls stopping of the IIC3_0 clock.
29	MSTP029	0	R/W	Controls stopping of the IIC3_1 clock.
28, 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26	MSTP026	0	R/W	Controls stopping of the SCIF0 clock.
25	MSTP025	0	R/W	Controls stopping of the SCIF1 clock.
24	MSTP024	0	R/W	Controls stopping of the SCIF2 clock.
23	MSTP023	0	R/W	Controls stopping of the SCIF3 clock.
22	MSTP022	0	R/W	Controls stopping of the SCIF4 clock.
21	MSTP021	0	R/W	Controls stopping of the SCIF5 clock.
20	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
19	MSTP019	0	R/W	Controls stopping of the HSCIF clock.

Bit	Bit Name	Initial Value	R/W	Description
18, 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	MSTP016	0	R/W	Controls stopping of the TIMER0 clock.
15	MSTP015	0	R/W	Controls stopping of the TIMER1 clock.
14	MSTP014	0	R/W	Controls stopping of the TIMER2 clock.
13	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
12	MSTP012	0	R/W	Controls stopping of the SSI0 clock.
11	MSTP011	0	R/W	Controls stopping of the SSI1 clock.
10	MSTP010	0	R/W	Controls stopping of the SSI2 clock.
9	MSTP009	0	R/W	Controls stopping of the SSI3 clock.
8	MSTP008	0	R/W	Controls stopping of the SSS clock.
7	MSTP007	0	R/W	Controls stopping of the HSPI clock.
6 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: Writing should be such that the reserved bits are not changed from their initial values, i.e. only the target bits should be modified (read-modify-write).

9.3.2 Module Standby Control Register 1 (MSTPCR1)

MSTPCR1 requests a module stop (a module standby). Whether the module operation is stopped can be checked by confirming that the corresponding bit in module standby status register 1 (MSTPSR1) is set to 1 indicating that module operation is stopped. Setting a bit in this register to 1 requests that the module be placed on standby and setting a bit to 0 requests that the module operate.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MSTP115	MSTP114	—	—	MSTP111	—	MSTP109	MSTP108	MSTP107	MSTP106	—	—	MSTP103	—	—	MSTP100
Initial value:	1	1	0	0	0	0	1	1	1	1	0	0	1	0	0	1
R/W:	R/W	R/W	R	R	R/W	R	R/W	R/W	R/W	R/W	R	R	R/W	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15	MSTP115	1	R/W	Controls module standby of ADMAC.
14	MSTP114	1	R/W	Controls module standby of GETHER.
13, 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11	MSTP111	0	R/W	Controls module standby of DMAC.
10	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
9	MSTP109	1	R/W	Controls module standby of VIN1.
8	MSTP108	1	R/W	Controls module standby of VIN0.
7	MSTP107	1	R/W	Controls module standby of R-GPVG.
6	MSTP106	1	R/W	Controls module standby of 2DG.
5, 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
3	MSTP103	1	R/W	Controls module standby of DU.
2, 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	MSTP100	1	R/W	Controls module standby of USB.

Note: Writing should be such that the reserved bits are not changed from their initial values, i.e. only the target bits should be modified (read-modify-write).

9.3.3 Module Standby Control Register 3 (MSTPCR3)

Before handling a bit in MSTPCR3, check that the module corresponding to the bit is not operating. If a bit in MSTPCR3 is handled while the module corresponding to the bit is operating, correct operation of the module cannot be guaranteed. Setting a bit in this register to 1 stops supply of the clock signal to the corresponding module and clearing a bit to 0 enables clock supply to the corresponding module.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MSTP331	MSTP330	—	—	—	—	—	—	MSTP323	MSTP322	MSTP321	MSTP320	MSTP319	MSTP318	MSTP317	MSTP316
Initial value:	1	1	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MSTP315	MSTP314	MSTP313	MSTP312	—	—	—	—	—	—	—	MSTP304	MSTP303	MSTP302	MSTP301	MSTP300
Initial value:	1	0	1	1	0	0	0	0	0	0	0	1	0	0	1	1
R/W:	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	MSTP331	1	R/W	Controls stopping of the MMC clock.
30	MSTP330	1	R/W	Controls stopping of the MIMLB clock.
29 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23	MSTP323	1	R/W	Controls stopping of the SDHI0 clock.
22	MSTP322	1	R/W	Controls stopping of the SDHI1 clock.
21	MSTP321	1	R/W	Controls stopping of the SDHI2 clock.
20	MSTP320	1	R/W	Controls stopping of the RQSPI clock.
19	MSTP319	1	R/W	Controls stopping of the SRC0 clock.
18	MSTP318	1	R/W	Controls stopping of the SRC1 clock.
17	MSTP317	1	R/W	Controls stopping of the RSPI clock.
16	MSTP316	1	R/W	Controls stopping of the RCAN0 clock.
15	MSTP315	1	R/W	Controls stopping of the RCAN1 clock.
14	MSTP314	0	R/W	Controls stopping of the FLTCL clock.
13	MSTP313	1	R/W	Controls stopping of the ADC clock.
12	MSTP312	1	R/W	Controls stopping of the MTU clock.
11 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	MSTP304	1	R/W	Controls stopping of the IEBus clock.
3	MSTP303	0	R/W	Controls stopping of the RTC clock. Note: This function stops the clock that is supplied to the internal RTC. To stop the external clock, such a function should be implemented in the customer's system.
2	MSTP302	0	R/W	Controls stopping of the HIF clock.
1	MSTP301	1	R/W	Controls stopping of the STIF0 clock.
0	MSTP300	1	R/W	Controls stopping of the STIF1 clock.

Note: Writing should be such that the reserved bits are not changed from their initial values, i.e. only the target bits should be modified (read-modify-write).

9.3.4 Module Standby Control Register 4 (MSTPCR4)

MSTPCR4 requests a module stop (a module standby). Whether the module operation is stopped can be checked by confirming that the corresponding bit in module standby status register 4 (MSTPSR4) is set to 1 indicating that module operation is stopped. Setting a bit in this register to 1 requests that the module be placed on standby and setting a bit to 0 requests that the module operate.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	MSTP405	—	MSTP403	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	MSTP405	1	R/W	Controls stopping of the LCDC clock.
4	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
3	MSTP403	1	R/W	Controls stopping of the LMB clock.
2 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: Writing should be such that the reserved bits are not changed from their initial values, i.e. only the target bits should be modified (read-modify-write).

9.3.5 Module Standby Control Register 5 (MSTPCR5)

Before handling a bit in MSTPCR5, check that the module corresponding to the bit is not operating. If a bit in MSTPCR5 is handled while the module corresponding to the bit is operating, correct operation of the module cannot be guaranteed. Setting a bit in this register to 1 stops supply of the clock signal to the corresponding module and clearing a bit to 0 enables clock supply to the corresponding module.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	MSTP502	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	MSTP502	1	R/W	Controls stopping of the VEU3F0 clock.
1, 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: Writing should be such that the reserved bits are not changed from their initial values, i.e. only the target bits should be modified (read-modify-write).

9.3.6 Module Standby Status Register 1 (MSTPSR1)

MSTPSR1 indicates the module operation state. "1" indicates module operation is stopped, and "0" indicates the module is in operation.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	STBY 120	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	STBY 115	STBY 114	—	—	STBY 111	—	STBY 109	STBY 108	STBY 107	STBY 106	—	—	STBY 103	—	—	STBY 100
Initial value:	1	1	0	0	0	0	1	1	1	1	0	0	1	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are always read as 0.
20	STBY120	0	R	Indicates the TMU module operation state. (Set to 1 when all channels of TMU0 to TMU2 and the TMU internal bus interface are in the standby state.)
19 to 16	—	All 0	R	Reserved These bits are always read as 0.
15	STBY115	1	R	Indicates the ADMAC module operation state.
14	STBY114	1	R	Indicates the GETHER module operation state.
13, 12	—	All 0	R	Reserved These bits are always read as 0.
11	STBY111	0	R	Indicates the DMAC module operation state.
10	—	0	R	Reserved This bit is always read as 0.
9	STBY109	1	R	Indicates the VIN1 module operation state.
8	STBY108	1	R	Indicates the VIN0 module operation state.
7	STBY107	1	R	Indicates the R-GPVG module operation state.
6	STBY106	1	R	Indicates the 2DG module operation state.
5, 4	—	All 0	R	Reserved These bits are always read as 0.
3	STBY103	1	R	Indicates the DU module operation state.

Bit	Bit Name	Initial Value	R/W	Description
2, 1	—	All 0	R	Reserved These bits are always read as 0.
0	STBY100	1	R	Indicates the USB module operation state.

9.3.7 Module Standby Status Register 3 (MSTPSR3)

MSTPSR3 indicates the module operation state. "1" indicates module operation is stopped, and "0" indicates the module is in operation. Since the MSTPCR3 setting is immediately reflected in the clock operating/stopped state of each module, except for the SDHI0 to SDHI2, RCAN0, RCAN1, STIF0, and STIF1, the value set in MSTPCR3 is reflected in MSTPSR3 without change.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	STBY331	STBY330	—	—	—	—	—	—	STBY323	STBY322	STBY321	STBY320	STBY319	STBY318	STBY317	STBY316
Initial value:	1	1	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	STBY315	STBY314	STBY313	STBY312	—	—	—	—	—	—	—	STBY304	STBY303	STBY302	STBY301	STBY300
Initial value:	1	0	1	1	0	0	0	0	0	0	0	1	0	0	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	STBY331	1	R	Indicates the MMC module operation state.
30	STBY330	1	R	Indicates the MIMLB module operation state.
29 to 24	—	All 0	R	Reserved These bits are always read as 0.
23	STBY323	1	R	Indicates the SDHI0 module operation state.
22	STBY322	1	R	Indicates the SDHI1 module operation state.
21	STBY321	1	R	Indicates the SDHI2 module operation state.
20	STBY320	1	R	Indicates the RQSPI module operation state.
19	STBY319	1	R	Indicates the SRC0 module operation state.
18	STBY318	1	R	Indicates the SRC1 module operation state.
17	STBY317	1	R	Indicates the RSPI module operation state.
16	STBY316	1	R	Indicates the RCAN0 module operation state.

Bit	Bit Name	Initial Value	R/W	Description
15	STBY315	1	R	Indicates the RCAN1 module operation state.
14	STBY314	0	R	Indicates the FLTCL module operation state.
13	STBY313	1	R	Indicates the ADC module operation state.
12	STBY312	1	R	Indicates the MTU module operation state.
11 to 5	—	All 0	R	Reserved These bits are always read as 0.
4	STBY304	1	R	Indicates the IEBus module operation state.
3	STBY303	0	R	Indicates the RTC module operation state.
2	STBY302	0	R	Indicates the HIF module operation state.
1	STBY301	1	R	Indicates the STIF0 module operation state.
0	STBY300	1	R	Indicates the STIF1 module operation state.

9.3.8 Module Standby Status Register 4 (MSTPSR4)

MSTPSR4 indicates the module operation state. "1" indicates module operation is stopped, and "0" indicates the module is in operation.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	STBY405	—	STBY403	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved These bits are always read as 0.
5	STBY405	1	R	Indicates the LCDC module operation state.
4	—	0	R	Reserved This bit is always read as 0.
3	STBY403	1	R	Indicates the LMB module operation state.

Bit	Bit Name	Initial Value	R/W	Description
2 to 0	—	All 0	R	Reserved These bits are always read as 0.

9.3.9 Standby Control Register (STBCR)

STBCR is a 32-bit readable/writable register that specifies sleep mode, software standby mode, or deep standby mode. STBCR can be accessed only in longword units.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	STBY	DEEP	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	STBY	0	R/W	Software Standby, Deep Standby
6	DEEP	0	R/W	Specifies transition to software standby mode or deep standby mode. 0x: Executing the SLEEP instruction puts the LSI into sleep mode. 10: Executing the SLEEP instruction puts the LSI into software standby mode. 11: Executing the SLEEP instruction puts the LSI into deep standby mode.
5 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

9.3.10 On-Chip Data-Retention RAM Area Setting Register (RRAMKP)

RRAMKP is an 8-bit readable/writable register that selects whether the contents of the corresponding area of the on-chip data-retention RAM are retained or not in deep standby mode.

When the RRAMKP bit is set to 1, the contents of the HIFRAM are retained in deep standby mode.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	RRAMKP
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	RRAMKP	0	R/W	On-Chip Data-Retention RAM Storage Area 0: The contents of the on-chip data-retention RAM are not retained in deep standby mode. 1: The contents of the on-chip data-retention RAM are retained in deep standby mode.

9.3.11 Deep Standby Control Register (DSCTR)

DSCTR is an 8-bit readable/writable register that controls the state of the external memory control pins and the activation method when returning from deep standby mode.

Bit:	7	6	5	4	3	2	1	0
	EBUSKEEPE	RAMBOOT	CKOKEEPE	PRSTOKEEPE	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	EBUSKEEPE	0	R/W	Retention of External Memory Control Pin State 0: When returning from deep standby mode, the state of the external memory control pins is not retained. 1: When returning from deep standby mode, the state of the external memory control pins is retained until IOKEEP is cleared.*
6	RAMBOOT	0	R/W	Selection of Activation Method after Returning from Deep Standby Mode Selects the activation method used after returning from deep standby mode. 0: Activated according to the boot mode specified for a reset. 1: The program is read from the on-chip data-retention RAM. (data-retention RAM boot) Program counter value (PC): HIFRAM start address + HIAR
5	CKOKEEPE	0	R/W	Retention of CLKOUT Pin State 0: When returning from deep standby mode, the state of the CLKOUT pin is not retained. 1: When returning from deep standby mode, the state of the CLKOUT pin is retained until IOKEEP is cleared.
4	PRSTOKEEPE	0	R/W	Retention of $\overline{\text{PRESETOUT}}$ Pin State 0: When returning from deep standby mode, the state of the $\overline{\text{PRESETOUT}}$ pin is not retained. 1: When returning from deep standby mode, the state of the $\overline{\text{PRESETOUT}}$ pin is retained until IOKEEP is cleared.
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: * The state of the DDR2-SDRAM and DDR3-SDRAM pins are retained by DDR-PHY. For initialization of the DDR2-SDRAM and DDR3-SDRAM pins after returning from deep standby mode, refer to section 4, Memory Controller (DBSC3).

9.3.12 Deep Standby Cancel Source Select Register (DSSSR)

DSSSR is a 16-bit readable/writable register that selects a source to cancel deep standby mode. DSSSR can be accessed only in word units.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	GET	—	RTCAR	—	—	IRQ3	IRQ2	IRQ1	IRQ0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R/W	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	GET	0	R/W	Cancel by Magic Packet Detection of GETHER 0: Deep standby mode is not canceled by Magic Packet detection of GETHER. 1: Deep standby mode is canceled by Magic Packet detection of GETHER. Note: If this bit is 1 when deep standby mode is entered, the GETHER power remains turned on and standby current increases. If the GETHER power is not needed in deep standby mode, clear this bit to 0. For usable function pins, refer to section 9.4.3 (8), Note on Return from Deep Standby Mode Due to a Magic-Packet Detection Interrupt from the GETHER Module.
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	RTCAR	0	R/W	Cancel by Realtime Clock Alarm Interrupt 0: Deep standby mode is not canceled by a realtime clock alarm interrupt. 1: Deep standby mode is canceled by a realtime clock alarm interrupt.
5, 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
3	IRQ3	0	R/W	Cancel by Change on IRQ3 0: Deep standby mode is not canceled by change on the IRQ3 pin. 1: Deep standby mode is canceled by change on the IRQ3 pin. Note: When the CAN0_RX_A and CAN1_RX_A pins are used as the IRQ0 and IRQ1 pins, respectively, the IRQ3 pin cannot be used as a cancel source.
2	IRQ2	0	R/W	Cancel by Change on IRQ2 0: Deep standby mode is not canceled by change on the IRQ2 pin. 1: Deep standby mode is canceled by change on the IRQ2 pin. Note: When the CAN0_RX_A and CAN1_RX_A pins are used as the IRQ0 and IRQ1 pins, respectively, the IRQ2 pin cannot be used as a cancel source.
1	IRQ1	0	R/W	Cancel by Change on IRQ1 0: Deep standby mode is not canceled by change on the IRQ1 pin. 1: Deep standby mode is canceled by change on the IRQ1 pin. Note: When the CAN1_RX_A pin is used as the IRQ1 pin, set the PFC pin to be used as the IRQ1_B pin.
0	IRQ0	0	R/W	Cancel by Change on IRQ0 0: Deep standby mode is not canceled by change on the IRQ0 pin. 1: Deep standby mode is canceled by change on the IRQ0 pin. Note: When the CAN0_RX_A pin is used as the IRQ0 pin, set the PFC pin to be used as the IRQ0_B pin.

9.3.13 Deep Standby Cancel Edge Select Register (DSESR)

DSESR is a 16-bit readable/writable register that selects an edge to be detected for the function pin specified as a deep standby cancel source. DSESR can be accessed only in word units.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	NMIE	—	—	—	—	IRQ3E	IRQ2E	IRQ1E	IRQ0E
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	NMIE	0	R/W	NMI Edge Detection 0: Falling edge of NMI is detected. 1: Rising edge of NMI is detected.
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	IRQ3E	0	R/W	IRQ3 Edge Detection 0: Falling edge of IRQ3 is detected. 1: Rising edge of IRQ3 is detected.
2	IRQ2E	0	R/W	IRQ2 Edge Detection 0: Falling edge of IRQ2 is detected. 1: Rising edge of IRQ2 is detected.
1	IRQ1E	0	R/W	IRQ1 Edge Detection 0: Falling edge of IRQ1 is detected. 1: Rising edge of IRQ1 is detected.
0	IRQ0E	0	R/W	IRQ0 Edge Detection 0: Falling edge of IRQ0 is detected. 1: Rising edge of IRQ0 is detected.

9.3.14 Deep Standby Cancel Source Flag Register (DSFR)

DSFR is a 16-bit readable/writable register composed of two types of bits. One is the flag that confirms which source canceled deep standby mode. The other is the bit that releases the state of pins after canceling deep standby mode. When deep standby mode is canceled by an interrupt (NMI or realtime clock alarm interrupt) or changes on the pins for canceling, DSFR retains the previous data although power-on reset exception handling is executed. When deep standby mode is canceled by a power-on reset, this register is initialized to H'0000.

All flags must be cleared immediately before transition to deep standby mode.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IO KEEP	—	—	—	—	—	GETF	NMIF	—	RTC ARF	—	—	IRQ3F	IRQ2F	IRQ1F	IRQ0F
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/(W)*	R	R	R	R	R	R/(W)*	R/(W)*	R	R/(W)*	R	R	R/(W)*	R/(W)*	R/(W)*	R/(W)*

Note: * Only 0 can be written after reading 1 in order to clear the flag.

Bit	Bit Name	Initial Value	R/W	Description
15	IOKEEP	0	R/(W)*	Release of Pin State Retention Releases the retention of the pin state after canceling deep standby mode. 0: Pin state is not retained. 1: Pin state is retained. [Clearing condition] • Writing 0 after reading 1 [Setting condition] • When deep standby mode is entered
14 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	GETF	0	R/(W)*	GETHER Flag 0: No GETHER Magic Packet is detected. 1: GETHER Magic Packet is detected.
8	NMIF	0	R/(W)*	NMI Flag 0: No interrupt on NMI pin. 1: Interrupt on NMI pin.

Bit	Bit Name	Initial Value	R/W	Description
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	RTCARF	0	R/(W)*	RTCAR Flag 0: No realtime clock alarm interrupt is generated. 1: Realtime clock alarm interrupt is generated.
5, 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	IRQ3F	0	R/(W)*	IRQ3 Flag 0: No change on the IRQ3 pin. 1: Change on the IRQ3 pin.
2	IRQ2F	0	R/(W)*	IRQ2 Flag 0: No change on the IRQ2 pin. 1: Change on the IRQ2 pin.
1	IRQ1F	0	R/(W)*	IRQ1 Flag 0: No change on the IRQ1 pin. 1: Change on the IRQ1 pin.
0	IRQ0F	0	R/(W)*	IRQ0 Flag 0: No change on the IRQ0 pin. 1: Change on the IRQ0 pin.

9.3.15 HIFRAM Returning Address Register (HIAR)

HIAR sets the offset value from the area start address when booting from the HIFRAM for recovering the contents of on-chip data-retention RAM (1 is written to the RRAMKP bit), after returning from deep standby mode. HIAR can be accessed only in word units. This register is initialized after returning from deep standby mode.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	HIAR[10]	HIAR[9]	HIAR[8]	HIAR[7]	HIAR[6]	HIAR[5]	HIAR[4]	HIAR[3]	HIAR[2]	HIAR[1]	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 1	HIAR[10:1]	All 0	R/W	These bits specify the value of the 10th to 1st bits of the offset for the boot start address after returning from deep standby mode. The setting of this register is invalid when data-retention RAM boot is not performed.
0	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

9.3.16 Boot Address Register Low (BARL)

BARL is used to set the lower 16 bits of the offset value from the start address of the area containing the address at which to start boot after returning from deep standby mode. This register sets the offset value from the area start address when performing boot from the CS0 area. This register is initialized when returning from deep standby mode.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BAR[15]	BAR[14]	BAR[13]	BAR[12]	BAR[11]	BAR[10]	BAR[9]	BAR[8]	BAR[7]	BAR[6]	BAR[5]	BAR[4]	BAR[3]	BAR[2]	BAR[1]	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 1	BAR[15:1]	All 0	R/W	These bits specify the value of the 15th to 1st bits of the offset for the boot start address after returning from deep standby mode. The setting of this register is invalid when booting is from data-retention RAM or from an external device other than CS0.
0	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

9.3.17 Boot Address Register High (BARH)

BARH is used to set the upper 16 bits of the offset value from the start address of the area containing the address at which to start boot after returning from deep standby mode. This register sets the offset value from the area start address when performing boot from the CS0 area. This register is initialized when returning from deep standby mode.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	BAR[23]	BAR[22]	BAR[21]	BAR[20]	BAR[19]	BAR[18]	BAR[17]	BAR[16]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	BAR[23:16]	All 0	R/W	These bits specify the value of the 23rd to 16th bits of the address offset when starting boot from the CS0 area after returning from deep standby mode. The setting of this register is invalid when booting is from data-retention RAM or from an external device other than CS0.

9.4 Operation

9.4.1 Sleep Mode

(1) Transition to Sleep Mode

Executing the SLEEP instruction when the STBY bit in STBCR is 0 causes a transition from the program execution state to sleep mode. Although the CPU halts immediately after executing the SLEEP instruction, the contents of its internal registers remain unchanged. The on-chip peripheral modules continue to run in sleep mode. The outputs of the external bus clock (CLKOUT) and DDR clocks (MCK0, $\overline{\text{MCK0}}$) are continued. CPU target access is available even in sleep mode.

(2) Canceling Sleep Mode

Sleep mode is canceled by an interrupt (NMI, IRQ, or on-chip peripheral module), a DMA address error, or a reset (manual reset or power-on reset).

(a) Canceling by an interrupt

When an NMI, IRQ, or on-chip peripheral module interrupt occurs, sleep mode is canceled and interrupt exception handling is executed. When the priority level of the generated interrupt is equal to or lower than the interrupt mask level that is set in the status register (SR) of the CPU, or the interrupt by the on-chip peripheral module is disabled on the module side, the interrupt request is not accepted and sleep mode is not canceled.

(b) Canceling by a DMA address error

When a DMA address error occurs, sleep mode is canceled and DMA address error exception handling is executed.

(c) Canceling by a reset

Sleep mode is canceled by a power-on reset or a manual reset.

9.4.2 Software Standby Mode

(1) Transition to Software Standby Mode

The LSI switches from a program execution state to software standby mode by executing the SLEEP instruction when the STBY bit and DEEP bit in STBCR are 1 and 0 respectively. In software standby mode, not only the CPU but also the clock supply and on-chip peripheral modules halt. The external bus clock (CLKOUT) and DDR clocks (MCK0, $\overline{\text{MCK0}}$) also stop.

The contents of the CPU and cache registers remain unchanged. Some registers of on-chip peripheral modules are, however, initialized. As for the states of on-chip peripheral module registers in software standby mode, see the register descriptions in the relevant module sections.

The CPU takes one cycle to finish writing to STBCR, and then executes processing for the next instruction. However, it takes one or more cycles to actually write. Therefore, execute a SLEEP instruction after reading STBCR to have the values written to STBCR by the CPU to be reflected in the SLEEP instruction without fail.

The procedure for switching to software standby mode is as follows:

1. Write 0 to the TME bit in WTCSR to stop the watchdog timer (WDT).
2. Set WDTBST to the time needed before this LSI restarts operation after oscillation of the PLL circuit settles. Writing H'55000001 sets the minimum specifiable time and writing H'55000000 sets the maximum specifiable time.
3. After setting the STBY and DEEP bits in STBCR to 1 and 0 respectively, read STBCR. After that, execute a SLEEP instruction.

(2) Canceling Software Standby Mode

Software standby mode is canceled by an interrupt (NMI or IRQ) or a reset (power-on reset). The external bus clock (CLKOUT) and DDR clocks (MCK0, $\overline{\text{MCK0}}$) are output.

(a) Canceling by an interrupt

When the falling edge or rising edge of the NMI pin (selected by the NMI edge selection bit (NMIE) in interrupt control register 0 (ICR0) of the interrupt controller) or the falling edge or rising edge of an IRQ pin (IRQ3 to IRQ0) (selected by the IRQn sense selection bits (IRQnS1 and IRQnS0) in interrupt control register 1 (ICR1)) is detected*, clock oscillation is started. This clock pulse is supplied only to the oscillation settling counter (watchdog timer) used to count the oscillation settling time.

Software standby mode is thus cleared and NMI interrupt exception handling (IRQ interrupt exception handling in case of IRQ) is started. If the priority level of the generated interrupt is equal to or lower than the interrupt mask level specified in the status register (SR) of the CPU, the interrupt request is not accepted and software standby mode is not canceled.

When software standby mode is canceled by an NMI interrupt or an IRQ interrupt, the clock output phase of the CLKOUT pin may be unstable immediately after detecting an interrupt and until software standby mode is canceled. When software standby mode is canceled by the signal from the NMI pin and the NMI falling edge is selected as the cancel source, make sure that the NMI pin level is low when software standby mode is entered (clock is stopped). When the rising edge is selected as the cancel source, make sure that the NMI pin level is low when software standby mode is entered (the clock is stopped) and high when software standby mode is canceled (the clock is started after oscillation settling). This should also be satisfied when the IRQ signal is used for cancellation.

Note: * Edges to trigger release are detected as independent interrupt inputs regardless of the setting of the IRLM0 bit in the ICR0 register.

(b) Canceling by a reset

Driving the $\overline{\text{PRESET}}$ pin low cancels software standby mode and causes a transition to the power-on reset state. After that, driving the $\overline{\text{PRESET}}$ pin high initiates power-on reset exception handling.

Keep the $\overline{\text{PRESET}}$ pin low until the clock oscillation settles. The internal clock will continue to be output to the CLKOUT pin.

9.4.3 Deep Standby Mode

(1) Transition to Deep Standby Mode

The LSI switches from a program execution state to deep standby mode by executing the SLEEP instruction when the STBY bit and DEEP bit in STBCR are set to 1. In deep standby mode, not only the CPU, clocks, and on-chip peripheral modules but also power supply is turned off excluding the on-chip data-retention RAM area (HIFRAM) specified by the RRAMKP bit in the RRAMKP register and realtime clock. This can significantly reduce power consumption. Therefore, data in the registers of the CPU, cache, and on-chip peripheral modules are not retained. Pin state values immediately before the transition to deep standby mode are retained except for some analog function pins and ET0_GTX_CLK output.

The CPU takes one cycle to finish writing to STBCR, and then executes processing for the next instruction. However, it actually takes one or more cycles to write. Therefore, execute a SLEEP instruction after reading STBCR to reflect the values written to STBCR by the CPU in the SLEEP instruction without fail.

The procedure for switching to deep standby mode is as follows. Figure 9.2 shows its flowchart.

1. When a program needs to be retained in the data-retention RAM area, set up the RRAMKP bit in the RRAMKP register and transfer the program to be retained to the on-chip data-retention RAM area.
2. Set the RAMBOOT and EBUSKEEPE bits in DSCTR to specify the activation method for returning from deep standby mode and to select whether the external memory control pin status is retained or not.
3. When canceling deep standby mode by an interrupt, set the corresponding bit in DSSSR to select the pin or source to cancel deep standby mode. In this case, specify the input signal detection mode for the selected pin with the corresponding bit in DSESR.
4. Execute read and write of an arbitrary but the same address for each page in the on-chip data-retention RAM area. When this is not executed, data last written may not be written to the on-chip data-retention RAM. If there is a write to the on-chip data-retention RAM after this time, execute this processing after the last write to the on-chip data-retention RAM.
5. Set the STBY and DEEP bits in STBCR to 1.
6. Read out the DSFR register after clearing the flag in the DSFR register. After that, execute the SLEEP instruction.

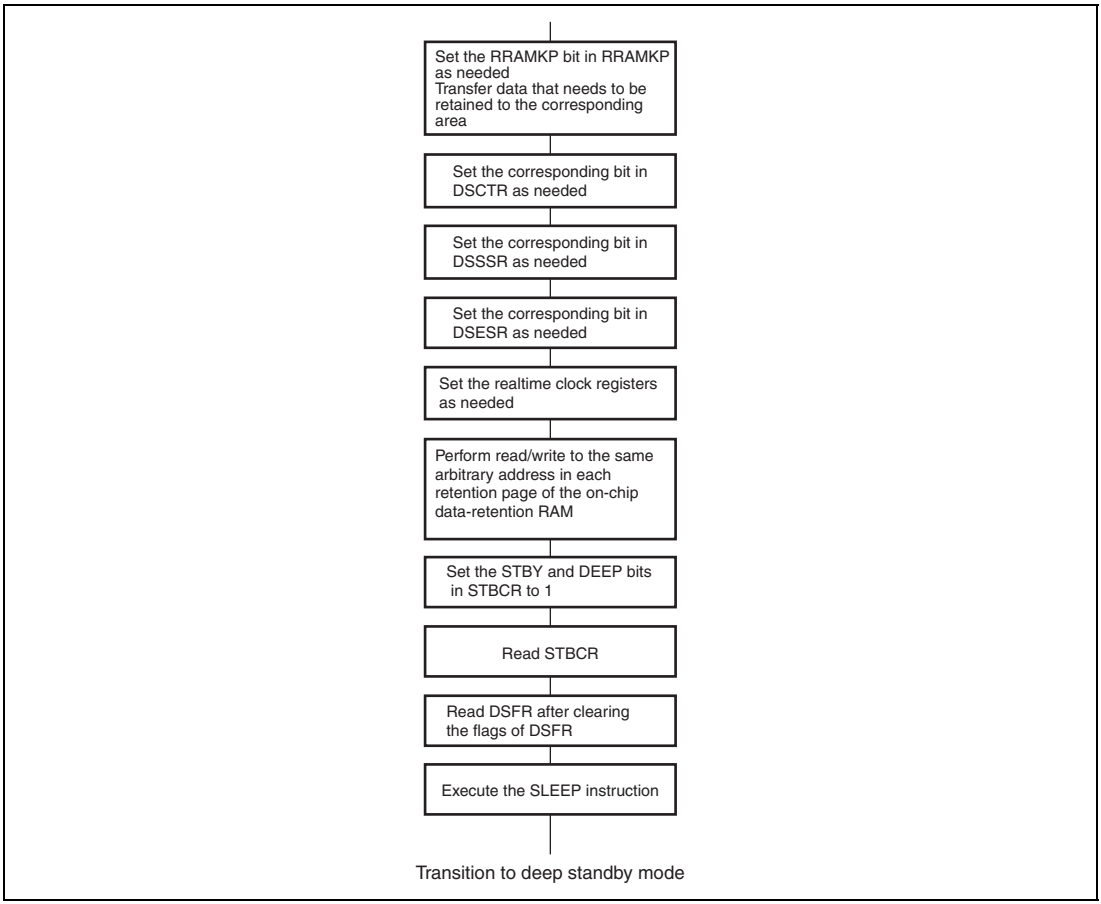


Figure 9.2 Flowchart of Transition to Deep Standby Mode

(2) Canceling Deep Standby Mode

Deep standby mode is canceled by an interrupt (NMI, realtime clock alarm interrupt, or magic-packet detection interrupt from the GETHER module), change on the pin for canceling, or a reset (power-on reset). The realtime clock alarm interrupt and magic-packet detection interrupt from the GETHER module can always cancel deep standby mode regardless of the interrupt priority level or the status register (SR) setting in the CPU. When canceling the mode by a source other than a reset, power-on reset exception handling is executed instead of interrupt exception handling.

Figure 9.3 shows the flowchart of canceling deep standby mode.

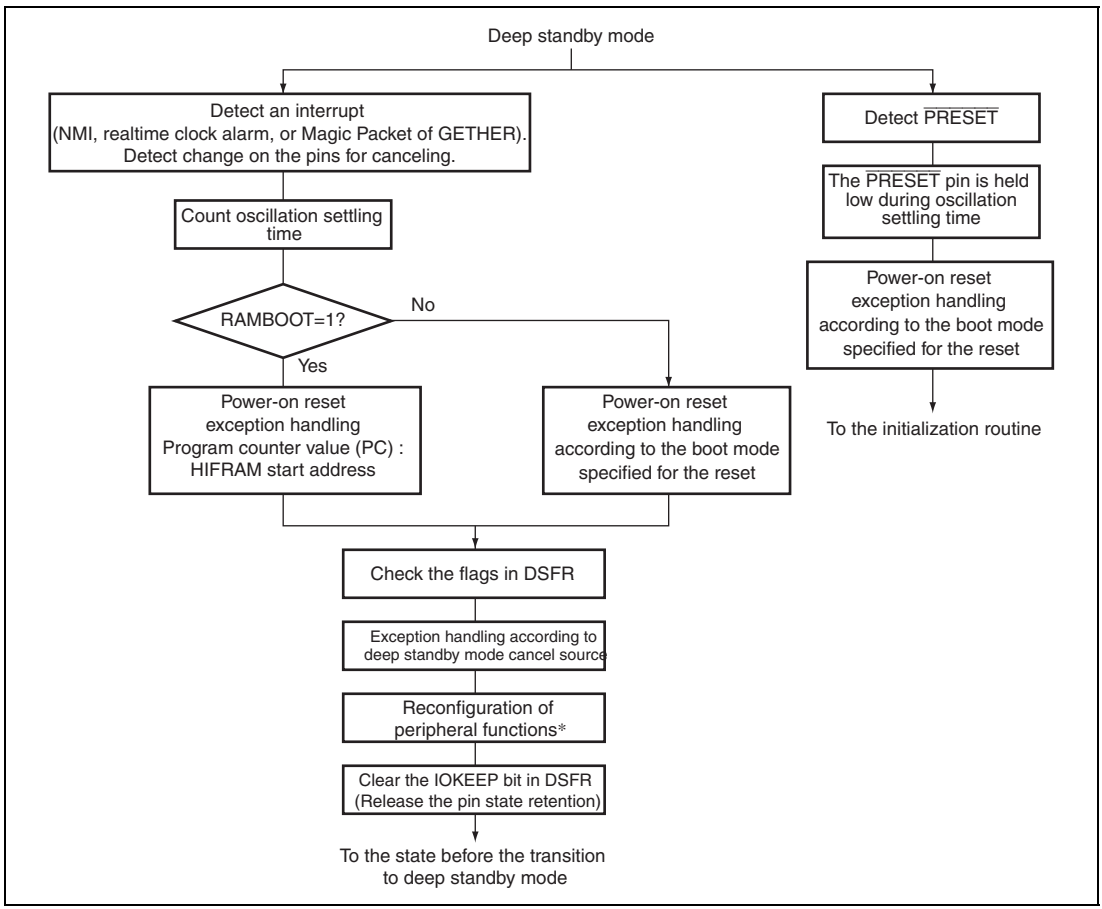


Figure 9.3 Flowchart of Canceling Deep Standby Mode

(a) Canceling by a source other than a reset

When the falling or rising edge of the NMI pin (selected by a corresponding bit in DSESR) or falling or rising edge of the pin for canceling (selected by a corresponding bit in DSESR) is detected or the realtime clock alarm interrupt (see section 26.4.4, Alarm Function, in section 26, Realtime Clock) is generated, clock oscillation is started after the wait time for the oscillation settling time. After the oscillation settling time has elapsed, deep standby mode is cancelled and the power-on reset exception handling is executed.

The clock output phase of the CLKOUT pin may be unstable immediately after detecting a cancel source and until deep standby mode is canceled. When deep standby mode is canceled by the signal from the NMI pin and the NMI falling edge is selected as the cancel source, make sure that the NMI pin level is high when deep standby mode is entered (clock is stopped) and low when deep standby mode is canceled (the clock is started after oscillation settling). When the NMI rising edge is selected as the cancel source, make sure that the NMI pin level is low when deep standby mode is entered (clock is stopped) and high when deep standby mode is canceled (the clock is started after oscillation settling). This should also be satisfied when another cancel source pin is used. When CKOKEEPE is set to 1, negate NMI and IRQ for at least 1 ms after issuing the SLEEP instruction to enter deep standby mode, and assert them for at least 1 ms.

(b) Canceling by a reset

Driving the $\overline{\text{PRESET}}$ pin low cancels deep standby mode and causes a transition to the power-on reset state. After this, driving the $\overline{\text{PRESET}}$ pin high initiates power-on reset exception handling. Output of the internal clock from the CLKOUT pin also starts by driving the $\overline{\text{PRESET}}$ pin low.

Keep the $\overline{\text{PRESET}}$ pin low until the clock oscillation settles.

(3) Operation after Canceling Deep Standby Mode

After canceling deep standby mode, the LSI can be activated through the external memory or from the on-chip data-retention RAM, which can be selected by setting the RAMBOOT bit in DSCTR. By setting the EBUSKEEPE bit, the states of the external memory control pins can be retained regardless of the boot mode setting even after cancellation of deep standby mode. Table 9.4 shows the pin states after cancellation of deep standby mode according to the setting of each bit. Table 9.5 lists the external memory control pins.

Table 9.4 Pin States after Cancellation of Deep Standby Mode and System Activation Method by the DSCTR Settings

EBUSKEEPE Bit	RAMBOOT Bit	Activation Method	Pin States After Cancellation of Deep Standby Mode
0	0	External memory	The states of the external memory control pins are not retained. For other pins, the retention of their states is cancelled when the IOKEEP bit is cleared.
0	1	On-chip data-retention RAM	In CS0 boot, the states of the external memory control pins are not retained. After cancellation of deep standby mode, the retention of the external memory control pin states is cancelled. For other pins, the retention of their states is cancelled when the IOKEEP bit is cleared.
1	0	—	Setting prohibited.
1	1	On-chip data-retention RAM	The states of the external memory control pin are retained. The retention of the states of the external memory control pins and other pins is cancelled when the IOKEEP bit is cleared.

Table 9.5 External Memory Control Pins in Different Modes

CS0 Boot (CS0 Space)	NAND Flash Boot (NAND Flash Memory)	Serial Boot (Serial Flash Memory)	MMC Boot (MMC Flash Memory)	eSD Boot (eSD Device)
A[23:0], D[15:0], $\overline{BS}$, $\overline{CS0}$, $\overline{RD}$	NAF[7:0], $\overline{FRE}$, FCLE, $\overline{FALE}$, $\overline{FEW}$, $\overline{FCE}$, FRB	RSPCK, SSL, MOSI, MISO	MMC_D[7:0], MMC_CLK, MMC_CMD	SD0_CLK, SD0_CMD, SD0_DAT[3:0], SD0_WP, SD0_CD

When deep standby mode is canceled by an interrupt (NMI or realtime clock alarm) or change on the pin for canceling, the deep standby cancel source flag register (DSFR) can be used to confirm which source has canceled the mode.

Pins retain the state immediately before the transition to deep standby mode. However, in system activation through the external memory, the retention of the states of the external memory control pins is cancelled so that programs can be fetched after cancellation of deep standby mode. Other pins, after cancellation of deep standby mode, continue to retain the pin states until 0 is written to the IOKEEP bit in DSFR after 1 is read from the same bit. When any of NAND flash boot, serial boot, MMC boot, and eSD boot is selected while $\overline{\text{PRESET}}$ is low and the LSI is activated from the on-chip data-retention RAM (data-retention RAM boot), after cancellation of deep standby mode the external memory control pins also continue to retain the pin states until 0 is written to the IOKEEP bit in DSFR after 1 is read from the same bit. Likewise, when deep standby mode is cancelled after the LSI is activated with HIF boot being specified while $\overline{\text{PRESET}}$ is low, the external memory control pins continue to retain the pin states until 0 is written to the IOKEEP bit in DSFR after 1 is read from the same bit.

Reconfiguration of peripheral functions is required to return to the previous state of deep standby mode. The retention of the pin state for peripheral functions (such as the clock pulse generator, interrupt controller, general I/O ports, and peripheral modules) are canceled and the LSI returns to the state prior to the transition to deep standby mode.

(4) Notes on Transition to Deep Standby Mode

After deep standby mode is specified, interrupts other than that selected with the deep standby cancel source select register are masked. That is, only an interrupt selected with the deep standby cancel source select register will be accepted. If, however, multiple canceling sources have been specified and multiple canceling sources are input, multiple cancel source flags will be set.

In addition, if a SLEEP instruction to initiate the transition to deep standby mode coincides with the NMI, an IRQ interrupt, or a manual reset, acceptance of the interrupt may initiate the cancellation of deep standby mode.

After the return from deep standby mode, the power and PLL settling wait times are secured by counting the EXTAL pulses. Therefore, EXTAL should continue operation after transition to deep standby mode.

When the pins used for booting are set to the functions of modules other than the boot module through the PFC settings, the boot module functions should be selected for the pins before transition to deep standby mode.

(5) Notes on HIFRAM Usage in Deep Standby Mode

When deep standby mode is entered while RAMBOOT is set to 0, the power to the HIFRAM is turned off and external devices cannot write to the HIFRAM after that. When RRAMKP is set to 1, the power to the HIFRAM continues to be supplied but the clock stops, so external devices cannot write to the HIFRAM as when RRAMKP is set to 0.

To enter deep standby mode while enabling write access to the HIFRAM from external devices through the HIF, use the following transition and return procedures.

1. Write 1 to the MASK bit* in HIFRDYCR to mask HIFRDY.
2. Read 1 from the MASK bit in HIFRDYCR to check that HIFRDY is masked.
3. Follow the procedure shown in figure 9.2 to enter deep standby mode. HIFRDY is fixed to be masked during deep standby mode as a result of steps 1 and 2.
4. After clearing IOKEEP, write 1 to the MASK bit in the HIFRDYCR and then read 1 from the bit so that HIFRDY is masked after the return from deep standby mode.
5. Follow the procedure shown in figure 9.3 to cancel deep standby mode
6. Write 0 to the MASK bit in the HIFRDYCR to enable access to HIFRDY, and make external devices write to the HIFRAM.

When activating the LSI in HIF boot mode, do not execute data-retention RAM boot (do not set RAMBOOT to 1).

Note: * For the function of the MASK bit in the HIFRDYCR, refer to section 21.4.12, HIFRDY Control Register (HIFRDYCR) in section 21, Host Interface (HIF).

(6) Note on Deep Standby Mode in User Debugging Mode

When transition to deep standby mode is specified while a user debugging mode such as ASE mode is used, standby mode is entered but the internal power continues to be supplied to enable debugging function operation.

(7) Note on Operation after Return from Deep Standby Mode

When IOKEEP is cleared after the return from deep standby mode, the waveforms of the signals output from the peripheral functions may be incomplete (output may start from the middle of the waveform patterns). Likewise, when CKOKEEPE is set to 1, the CLKOUT waveform may also be incomplete (output may start from the middle of the waveform pattern) when IOKEEP is cleared after the return from deep standby mode.

The on-chip PLL for generating CLKOUT starts oscillation immediately after the return from deep standby mode and the oscillation settles before the reset is canceled. When CKOKEEPE = 0, the waveform before oscillation settles is temporarily output from the CLKOUT pin during the reset period.

(8) Note on Return from Deep Standby Mode Due to a Magic-Packet Detection Interrupt from the GETHER Module

In this LSI chip, selecting Group B in GETHER (the control section) switches on the structure for sending the GETF bit on acceptance of magic packet detection interrupts from GETHER, and this interrupt generation can be used as the trigger for return from deep standby mode. Reception of packet data is enabled by selecting GMII Group B#, MII Group B#, or RMII Group A#.

Magic-packet detection interrupts from the GETHER module are automatically initialized by the reset on return from deep standby mode. Therefore, if a magic-packet detection interrupt from the GETHER module conflicts with another source for return from deep standby mode, the GETF bit may not be set to 1 if the reset is applied before acceptance of the magic-packet detection interrupt. For the same reason, the level on the ET0_MAGIC pin for this function may still be low by the time of a reset on detection of a magic packet. To prevent this, the ET0_MAGIC pin must be kept at the high level by using general IO port functions and so on over the required period after return from the deep standby mode.

9.4.4 Module Standby Functions

This function is used to stop supply of the clock signal to specified modules. For stopping or restarting the supply of clock signals by using the module standby function, refer to the procedures for stopping and restarting clock supply in the following descriptions of the individual modules.

(1) Module Standby Control Register 0 (MSTPCR0)

This register controls the modules that have no power-down control signals. Accordingly, if clock stop and restart is requested by this register, the clock supply is controlled regardless of the module conditions.

(2) Module Standby Control Register 1 (MSTPCR1)

This register controls the modules that have power-down control signals. Accordingly, when clock stop is requested by this register, the corresponding module is stopped according to the power-down procedure before the clock is actually stopped. When clock restart is requested by this register, the clock is first supplied, then the power-down state is cancelled, and finally the module is restarted.

(3) Module Standby Control Register 3 (MSTPCR3)

This register controls the modules that have no power-down control signals. Accordingly, if clock stop and restart is requested by this register, the clock supply is controlled regardless of the module conditions.

(4) Module Standby Control Register 4 (MSTPCR4)

This register controls the modules that have power-down control signals. Accordingly, when clock stop is requested by this register, the corresponding module is stopped according to the power-down procedure before the clock is actually stopped. When clock restart is requested by this register, the clock is first supplied, then the power-down state is cancelled, and finally the module is restarted.

(5) Module Standby Control Register 5 (MSTPCR5)

This register controls the modules that have no power-down control signals. Accordingly, if clock stop and restart is requested by this register, the clock supply is controlled regardless of the module conditions.

When individual modules are in module standby mode, the manual reset is effective for some modules and ineffective for others as summarized in the table 9.6. Thus, if initialization is required for modules for which the manual reset is ineffective, restart the clock supply to the modules before issuing the reset.

Table 9.6 Manual Reset in Module Standby Mode

Reset is not Effective in Module Standby Mode	Reset is Effective in Module Standby Mode
SCIF, HSCIF, TMU, SSI*, SSS*, HSPI*, ADMAC*, SHwy-DMAC, VIN0*, VIN1*, USB, MIMLB*, SRC, RCAN*, FLCTL, IEBus, HIF, LCDC*, LMB*, VEU3F0*	IIC3, DU, MMC, SDHI, RQSPI, RSPI, ADC, MTU, GPIO, RTC, STIF, PFC

Note: * If a manual-reset is issued in module standby mode, some of the circuitry within this module is initialized.

(a) Transition to Module Standby Mode

To make a transition to module standby mode by the individual module clock stop function, refer to the following transition procedure for each module.

1. IIC3

(1) Set the MSTP029 and MSTP030 bits (corresponding to IIC3_1 and IIC3_0, respectively) in module standby control register 0 (MSTPCR0) to 1.

2. SCIF

(1) Complete the current serial communication supported by the SCIF. Then check the completion of the communication as follows. During transmission, check that the transmit data stored in the transmit FIFO data register has been transmitted to the serial communication pin by monitoring the serial status register. During reception, check that the receive data, which is provided from serial communication pins and stored in the receive FIFO data register, has been popped by monitoring the serial status register.

(2) Set the MSTP026 to MSTP021 bits (corresponding to SCIF0 to SCIF5, respectively) in module standby control register 0 (MSTPCR0) to 1.

3. HSCIF

(1) Set the MSTP019 bit in module standby control register 0 (MSTPCR0) to 1.

4. TIMER

(1) Stop the current timer counting operation by setting the bit in the timer start register.

(2) Set the MSTP016 to MSTP014 bits (corresponding to TIMER0 to TIMER2, respectively) in module standby control register 0 (MSTPCR0) to 1.

5. SSI (0 to 3)

(1) Check that all the DMEN, UIEN, OIEN, IIEN, DIEN, and EN bits in the SSI0 to SSI3 control register (SSICR) are cleared to 0 (disabled).

(2) Check that the IDST bit in SSI0 to SSI3 status register (SSISR) is set to 1.

(3) Check that the CONT bit in SSI0 to SSI3 TDM mode register (SSITDM) is cleared to 0 (disabled).

(4) Set the MSTP009 to MSTP012 bits in module standby control register 0 (MSTPCR0) to 1.

Note: The MSTP009 to MSTP012 bits should be set while the SSI module operation has been completed and is placed in the idle state in which the SSI module cannot be activated by external pins or other modules.

6. SSS

(1) Check that all the SDOA_CTL, SDIA_CTL, SDOB_CTL, and SDIB_CTL bits in the SSS interface control register (IFCTL) are cleared to 0.

(2) Check that all the DMA_OB, DMA_OA, DMA_IB, DMA_IA, FCL_DMA_OB, FCL_DMA_OA, FCL_DMA_IB, FCL_DMA_IA, EVENT_OMASKA, EVENT_OMASKB, EVENT_IMASKA, and EVENT_IMASKB bits in the SSS data transfer method setting register (TRDAT) are cleared to 0.

(3) Check that all the ST_OFIFORB, ST_OFIFORA, ST_IFIFORB, ST_IFIFORA, ST_OUTPB, ST_OUTPA, ST_INPB, ST_INPA, WT_OFIFORB, WT_OFIFORA, WT_IFIFORB, WT_IFIFORA, WT_OUTPB, WT_OUTPA, WT_INPB, and WT_INPA bits in the SSS FIFO operation enable register (STFIFO) are cleared to 0.

(4) Stop the clock supply to the SSS module.

(5) Set the MSTP008 bit in module standby control register 0 (MSTPCR0) to 1.

Notes: 1. Before making a transition to module standby mode, check that the SSS module is stopped after the SSS operation has been completed. The MSTP008 bit in MSTPCR0 should be set to 1 while the SSS module cannot be activated by external pins or other modules. Otherwise, correct operation cannot be guaranteed.

2. After entering into module standby mode, the SSS should be set so that it does not affect other modules and external pins. For example, DMA should not be activated and interrupts should not be generated.

7. SPI

(1) Check whether the data transfer has been completed. In other words, the transmit buffer (or FIFO) must be empty, and the receive buffer (or FIFO) must be read out to be empty.

(2) Disable all interrupt requests and disable FIFO mode.

(3) Set the MSTP007 bit in module standby control register 0 (MSTPCR0) to 1.

8. ADMAC

(1) Check that the ADMAC operations have all been stopped.

(2) Set the MSTP115 bit in module standby control register 1 (MSTPCR1) to 1.

(3) Check that the STBY115 bit in module standby status register 1 (MSTPSR1) is set to 1.

9. GETHER

(1) Set the MSTP114 bit in module standby control register 1 (MSTPCR1) to 1 while transfer is not progress in the dedicated direct memory access controller (E-DMAC).

(2) Check that the STBY114 bit in module standby status register 1 (MSTPSR1) is set to 1.

10. DMAC

(1) Set the MSTP111 bit in module standby control register 1 (MSTPCR1) to 1 while DMA transfer is not in progress.

(2) Check that the STBY111 bit in module standby status register 1 (MSTPSR1) is set to 1.

11. VIN0

(1) Clear the module enable (ME) bit in the main control register (MC) and the continuous capture (CC) and single capture (SC) bits in the frame capture register (FC) to 0 to stop the video input module.

(2) Check that the capture active (CA) bit in the module status register (MS) is cleared to 0.

(3) Set the MSTP108 bit in module standby control register 1 (MSTPCR1) to 1.

(4) Check that the STBY108 bit in module standby status register 1 (MSTPSR1) is set to 1.

12. VIN1

(1) Clear the module enable (ME) bit in the main control register (MC) and the continuous capture (CC) and single capture (SC) bits in the frame capture register (FC) to 0 to stop the video input module.

(2) Check that the capture active (CA) bit in the module status register (MS) is cleared to 0.

(3) Set the MSTP109 bit in module standby control register 1 (MSTPCR1) to 1.

(4) Check that the STBY109 bit in module standby status register 1 (MSTPSR1) is set to 1.

13. R-GPVG

(1) Set the MSTP107 bit in module standby control register 1 (MSTPCR1) to 1.
(2) Check that the STBY107 bit in module standby status register 1 (MSTPSR1) is set to 1.
(3) Set bit 7 in module reset register 1 (MRST1) to 1 (be sure to write 0 to the bits other than bit 7).
(4) Wait for at least 100 Pϕ cycles.
(5) Clear bit 7 in module reset register 1 (MRST1) to 0 (be sure to write 0 also to the bits other than bit 7).

14. 2DG

(1) Check that the current display list is completed if the 2DG is operating.
(2) Check that the trap flag (TRA) in the status register (SR) is set to 1.
(3) Set the software reset (SRES) bit to 1 and the rendering start (RS) bit to 0 in the system control register (SCLR).
(4) Set the MSTP106 bit in the module standby control register 1 (MSTPCR1) to 1.
(5) Check that the STBY106 bit in the module standby status register 1 (MSTPSR1) is set to 1.

15. DU

(1) Turn off the DU operation by clearing the display enable (DEN) and display reset (DRES) bits in the display unit system control register (DSYSR) to 0.
(2) Check the VBK bit in the display unit status register (DSSR) to confirm the next VBK (this is because the DU operation is turned off at the time the VBK flag is set). To stop display synchronization after module standby mode is canceled, set the display reset (DRES) bit to 1.
(3) Set the MSTP103 bit in module standby control register 1 (MSTPCR1) to 1.
(4) Check that the STBY103 bit in module standby status register 1 (MSTPSR1) is set to 1.

16. USB

When two ports serve as hosts (the order in which OHCI control and EHCI control are applied does not matter):

(1) OHCI control: Set the Host Controller Functional State field in HcControl register to 00.
(2) EHCI control: Set the Asynchronous Schedule Enable bit in USBCMD register to 0.
(3) EHCI control: Check that the Asynchronous Schedule Status bit in USBSTS register is cleared to 0.
(4) EHCI control: Set the Periodic Schedule Enable bit in USBCMD register to 0.
(5) EHCI control: Check that the Periodic Schedule Status bit in USBSTS register is cleared to 0.

(6) EHCI control: Set the Run/Stop bit in USBCMD register to 0.
(7) EHCI control: Check that the HCHalted bit in USBSTS register is set to 1.
(8) Common control: Set the PHYRST bit in port control 1 register to 1.
(9) Common control: Set the MSTP100 bit in module standby control register 1 (MSTPCR1) to 1.
(10) Common control: Check that the STBY100 bit in module standby status register 1 (MSTPSR1) is set to 1.

When one port serves as a function:

(1) Common control: Set the PORT1 bit in port control 0 register to 0.
(2) For both ports, follow the above procedure for two ports serving as hosts.

17. MMC

(1) Confirm that access to an MMC card is not in progress.
(2) Set the MSTP331 bit in module standby control register 3 (MSTPCR3) to 1.

18. MIMLB

Confirm the following register settings.

(1) All bits in the MOST control 1 register (MOST_CONTROL1) of the MIMLB are 0 (transfer stopped).
(2) The mlb_enable bit in the MOST control 2 register (MOST_CONTROL2) of the MIMLB is 0 (not connected with MediaLB).
(3) All bits in the FIFO interrupt enable register (FIFO_INT_ENABLE) of the MIMLB are 0 (interrupts disabled).
(4) All bits in SYSTEM interrupt enable register (SYSTEM_INT_ENABLE) of the MIMLB are 0 (interrupts disabled).
(5) All bits in SYSTEM _MEM interrupt enable register (SYSTEM _MEM_INT_ENABLE) of the MIMLB are 0 (interrupts disabled).
(6) The MDE bit in device control Cfg register (DCCR) of the OS62400 is 0 (disabled).
(7) All bits in system mask Cfg register (SMCR) of the OS62400 are 1 (masked).
(8) The CE bit is 0 (disabled) and the MASK7 to MASK0 bits are 1 (masked) in the channel n entry Cfg register (CECRn) of the OS62400.
(9) Set the MSTP330 bit in module standby control register 3 (MSTPCR3) to 1.

19. SDHI (0/1/2)

(1) Confirm that access to an SD card is not in progress.
(2) Set the MSTP323 to MSTP321 bits (corresponding to channels 0 to 2, respectively) in module standby control register 3 (MSTPCR3) to 1.

20. RQSPI

(1) Set the MSTP320 bit in module standby control register 3 (MSTPCR3) to 1.

21. SRC (0/1)

(1) Set the MSTP319 and MSTP318 bits in module standby control register 3 (MSTPCR3) to 1.

22. RSPI

(1) Check whether the data transfer has been completed. In other words, the transmit buffer must be empty, and the receive buffer must be read out to be empty.
(2) Disable all interrupt requests.
(3) Check that the SPE bit in control register is 0 (the RSPI module function is disabled).
(4) Set the MSTP317 bit in module standby control register 3 (MSTPCR3) to 1.

23. RCAN 1/0

(1) Complete the current CAN bus communication. Then check the completion of the communication as follows. During transmission, check that the transmit data stored in the transmit mailbox has been transmitted to the CAN bus by monitoring the interrupt request register (IRR). During reception, check that the receive data, which is provided from CAN bus and stored in the receive mailbox, has been popped by monitoring the interrupt request register (IRR).
(2) Set the MSTP316 and MSTP315 bits in module standby control register 3 (MSTPCR3) to 1.

24. FLCTL

(1) Check that the TRSTRT bit in the transfer control register is cleared to 0 (transfer is stopped) and the CE bit in the common control register is cleared to 0 (disabled (high level is output to the FCE pin)).
(2) Set the MSTP314 bit in module standby control register 3 (MSTPCR3) to 1.

25. ADC

(1) Set the MSTP313 bit in module standby control register 3 (MSTPCR3) to 1.

26. MTU

(1) Set the MSTP312 bit in module standby control register 3 (MSTPCR3) to 1.

27. IEBus

(1) Set the MSTP304 bit in module standby control register 3 (MSTPCR3) to 1.

28. RTC

(1) Set the MSTP303 bit in module standby control register 3 (MSTPCR3) to 1.

29. HIF

(1) Set the MSTP302 bit in module standby control register 3 (MSTPCR3) to 1.

30. STIF0/1

(1) Set the MSTP301 bit (module standby mode for STIF0) or MSTP300 bit (module standby for STIF1) in module standby control register 3 (MSTPCR3) to 1

31. LCDC

(1) Set the MSTP405 bit in module standby control register 4 (MSTPCR4) to 1.
(2) Check that the STBY405 bit in module standby status register 4 (MSTPSR4) is set to 1.

32. LMB

(1) Set the MSTP403 bit in module standby control register 4 (MSTPCR4) to 1.
(2) Check that the STBY403 bit in module standby status register 4 (MSTPSR4) is set to 1.

33. VEU3F0

(1) Set the MSTP502 bit in module standby control register 5 (MSTPCR5) to 1.

(b) Module Standby Mode Cancellation and Restart

After the transition to module standby mode, module standby mode can be cancelled by a power-on reset or the following procedure.

1. IIC3

(1) Clear the MSTP029 and MSTP030 bits (corresponding to IIC3_1 and IIC3_0, respectively) in module standby control register 0 (MSTPCR0) to 0.

2. SCIF

(1) Clear the MSTP026 to MSTP021 bits (corresponding to SCIF0 to SCIF5, respectively) in module standby control register 0 (MSTPCR0) to 0.

(2) Perform the same processing as initialization sequence after a power-on reset.

3. HSCIF

(1) Clear the MSTP019 bit in module standby control register 0 (MSTPCR0) to 0.

4. TIMER

(1) Clear the MSTP016 to MSTP014 bits (corresponding to TIMER0 to TIMER2, respectively) in module standby control register 0 (MSTPCR0) to 0.

(2) Perform the same processing as initialization sequence after a power-on reset.

5. SSI (0 to 3)

(1) Clear all the MSTP009 to MSTP012 bits in module standby control register 0 (MSTPCR0) to 0.

(2) Perform the same processing as initialization sequence after a power-on reset.

6. SSS

(1) Restart clock supply to the SSS module.

(2) Clear the MSTP008 bit in module standby control register 0 (MSTPCR0) to 0.

(3) Perform the same processing as the initialization sequence after a power-on reset.

Note: The SSS module switch registers SELOUT (output) and SELIN (input) cannot be reset (initialized) by the SSS software reset control register (SRCTL).

If an SSS module is selected by setting the SSS module switch registers SELOUT (output) and SELIN (input) when the SSS software reset control register SRCTL controls the SSS module reset, the SSS module pins enter the input (initial) state.

7. SPI

(1) Clear the MSTP007 bit in module standby control register 0 (MSTPCR0) to 0.

(2) Perform the same processing as initialization sequence after a power-on reset.

8. ADMAC

(1) Check that the STBY115 bit in module standby status register 1 (MSTPSR1) is set to 1.
(2) Clear the MSTP115 bit in module standby control register 1 (MSTPCR1) to 0.
(3) Check that the STBY115 bit in module standby status register 1 (MSTPSR1) is set to 0.

9. GETHER

(1) Check that the STBY114 bit in module standby status register 1 (MSTPSR1) is set to 1.
(2) Clear the MSTP114 bit in module standby control register 1 (MSTPCR1) to 0.
(3) Check that the STBY114 bit in module standby status register 1 (MSTPSR1) is set to 0.
(4) Perform the same processing as initialization sequence after a power-on reset.

10. DMAC

(1) Clear the MSTP111 bit in module standby control register 1 (MSTPCR1) to 0.
(2) Check that the STBY111 bit in module standby status register 1 (MSTPSR1) is cleared to 0.
(3) Perform the same processing as initialization sequence after a power-on reset.

11. VIN0

(1) Check that the STBY108 bit in module standby status register 1 (MSTPSR1) is set to 1.
(2) Clear the MSTP108 bit in module standby control register 1 (MSTPCR1) to 0.
(3) Check that the STBY108 bit in module standby status register 1 (MSTPSR1) is cleared to 0.
(4) Set the module enable (ME) bit in the main control register (MC) to 1 to activate the video input module.
(5) Set the continuous capture (CC) or single capture (SC) bit in the frame capture register (FC) to 1.

12. VIN1

(1) Check that the STBY109 bit in module standby status register 1 (MSTPSR1) is set to 1.
(2) Clear the MSTP109 bit in module standby control register 1 (MSTPCR1) to 0.
(3) Check that the STBY109 bit in module standby status register 1 (MSTPSR1) is cleared to 0.
(4) Set the module enable (ME) bit in the main control register (MC) to 1 to activate the video input module.
(5) Set the continuous capture (CC) or single capture (SC) bit in the frame capture register (FC) to 1.

13. R-GPVG

(1) Check that the STBY107 bit in module standby status register 1 (MSTPSR1) is set to 1.
(2) Clear the MSTP107 bit in module standby control register 1 (MSTPCR1) to 0.
(3) Execute the R-GPVG initialization sequence.

14. 2DG

(1) Check that the STBY106 bit in module standby status register 1 (MSTPSR1) is set to 1.
(2) Set bit 6 in module reset register 1 (MRST1) to 1 (be sure to write 0 to the bits other than bit 6).
(3) Clear the MSTP106 bit in module standby control register 1 (MSTPCR1) to 0.
(4) Check that the STBY106 bit in module standby status register 1 (MSTPSR1) is cleared to 0.
(5) Set bit 6 in module reset register 1 (MRST1) to 1 for at least four clkp cycles.
(6) Clear bit 6 in module reset register 1 (MRST1) to 0.
(7) Clear the software reset (SRES) and rendering start (RS) bits in the system control register (SCLR) to 0.
(8) Set the 2DG registers again.
(9) Clear the software reset (SRES) bit in the system control register (SCLR) to 0 and set the rendering start (RS) bit in SCLR to 1.

15. DU

(1) Check that the STBY103 bit in module standby status register 1 (MSTPSR1) is set to 1.
(2) Clear the MSTP103 bit in module standby control register 1 (MSTPCR1) to 0.
(3) Check that the STBY103 bit in module standby status register 1 (MSTPSR1) is cleared to 0.
(4) Turn on the display by setting the DU enable (DEN) bit and the DU reset (DRES) bit in the display unit system control register (DSYSR) to 1 and 0, respectively.

16. USB

The following procedure is common to both cases when both two ports serve as hosts and when one port serves as a function.

(1) Common control: Clear the MSTP100 bit in module standby control register 1 (MSTPCR1) to 0.
(2) Common control: Check that the STBY100 bit in module standby status register 1 (MSTPSR1) is cleared to 0.
(3) Perform the same processing as initialization sequence after a power-on reset.

17. MMC

(1) Clear the MSTP331 bit in module standby control register 3 (MSTPCR3) to 0.

(2) Perform the same processing as initialization sequence after a power-on reset.

18. MIMLB

(1) Clear the MSTP330 bit in module standby control register 3 (MSTPCR3) to 0.

(2) Perform the same processing as initialization sequence after a power-on reset.

19. SDHI (0/1/2)

(1) Clear the MSTP323 to MSTP321 bits (corresponding to channels 0 to 2, respectively) in module standby control register 3 (MSTPCR3) to 0.

(2) Perform the same processing as initialization sequence after a power-on reset.

20. RQSPI

(1) Clear the MSTP320 bit in module standby control register 3 (MSTPCR3) to 0.

21. SRC (0/1)

(1) Clear the MSTP319 and MSTP318 bits in module standby control register 3 (MSTPCR3) to 0.

22. RSPI

(1) Clear the MSTP317 bit in module standby control register 3 (MSTPCR3) to 0.

(2) Perform the same processing as initialization sequence after a power-on reset.

23. RCAN

(1) Clear the MSTP316 and MSTP315 bits in module standby control register 3 (MSTPCR3) to 0.

(2) Perform the same processing as initialization sequence after a power-on reset.

24. FLCTL

(1) Clear the MSTP314 bit in module standby control register 3 (MSTPCR3) to 0.

(2) Perform the same processing as initialization sequence after a power-on reset.

25. ADC

(1) Clear the MSTP313 bit in module standby control register 3 (MSTPCR3) to 0.

26. MTU

(1) Clear the MSTP312 bit in module standby control register 3 (MSTPCR3) to 0.

27. IEBus

(1) Clear the MSTP304 bit in module standby control register 3 (MSTPCR3) to 0.

28. RTC

(1) Clear the MSTP303 bit in module standby control register 3 (MSTPCR3) to 0.

29. HIF

(1) Clear the MSTP302 bit in module standby control register 3 (MSTPCR3) to 0.

30. STIF (0/1)

(1) Clear the MSTP301 bit (canceling module standby mode for STIF0) or MSTP300 bit (canceling module standby for STIF1) in module standby control register 3 (MSTPCR3) to 0.

31. LCDC

(1) Check that the STBY405 bit in module standby status register 4 (MSTPSR4) is set to 1.
(2) Clear the MSTP405 bit in module standby control register 4 (MSTPCR4) to 0.
(3) Check that the STBY405 bit in module standby status register 4 (MSTPSR4) is cleared to 0.

32. LMB

(1) Check that the STBY403 bit in module standby status register 4 (MSTPSR4) is set to 1.
(2) Clear the MSTP403 bit in module standby control register 4 (MSTPCR4) to 0.
(3) Check that the STBY403 bit in module standby status register 4 (MSTPSR4) is cleared to 0.

33. VEU3F0

(1) Clear the MSTP502 bit in module standby control register 5 (MSTPCR5) to 0.

9.5 Boot Mode

This LSI can be booted from the memory connected to the CS0 space, the NAND flash memory, the serial flash memory, the flash memory connected to the MMC controller, or the eSD device or booted by using the HIFRAM.

9.5.1 Boot Mode and Pin Function Setting

This LSI can determine the boot mode using external pins when $\overline{\text{PRESET}}$ is low. The external pin settings for selecting the boot mode are shown in table 9.7.

Table 9.7 Mode Pin (HIFMD (MD19), MD18, MD17, MD16, and MD14) Settings and Corresponding Boot Modes

HIFMD (MD19)	MD14	MD18	MD17	MD16	Boot Mode
0	0	0	0	0	Boots the LSI from the memory connected to the CS0 space. (CS0 boot)
		0	0	1	Reserved
		0	1	0	Boots the LSI from the NAND flash memory connected to the NAND flash memory controller. (NAND flash boot)
		0	1	1	Boots the LSI, through low-speed communication, from the flash memory connected to the Renesas serial peripheral interface. (Serial boot)
	1	0	0	0	Boots the LSI from the flash memory connected to the MMC controller. (MMC boot)
		0	0	1	Reserved
		0	1	0	Boots the LSI from the eSD device connected to the SDHI0. (eSD boot)
		0	1	1	Reserved
0	—	1	—	—	Reserved
1	—	—	—	—	Boots the LSI by using the HIFRAM. (HIF boot)

9.5.2 Operation

(1) CS0 Boot

In CS0 boot, this LSI is booted from the memory connected to the CS0 space. In this mode, this LSI starts program execution from address 0xA0000000 after $\overline{\text{PRESET}}$ goes high.

(2) NAND flash Boot, Serial Boot, MMC Boot, and eSD Boot

When the mode setting is for NAND flash boot, serial boot, MMC boot, or eSD boot, this LSI copies 16 Kbytes from the start address of the given device to the ILRAM after $\overline{\text{PRESET}}$ goes high, and then starts program execution from the address where the ILRAM starts.

(3) HIF Boot

When HIFMD (MD19) is driven high, this LSI is activated in HIF boot mode regardless of the MD18, MD17, MD16, and MD14 settings. In this mode, this LSI starts program execution from address 0xFF820000 after $\overline{\text{PRESET}}$ goes high. For details of the HIF boot mode, refer to section 21, Host Interface (HIF).

(4) Boot Mode after Return from Power Cut-Off State

When RRAMKP is 1 after the return from the power cut-off state, this LSI is booted by using the HIFRAM as the data-retention RAM regardless of the HIFMD (MD19), MD18, MD17, MD16, and MD14 settings. In this case, this LSI automatically executes the program that has been written to the HIFRAM before the power cut-off state is entered, so the external device does not need to write to the HIFRAM.

When RRAMKP is 0, the LSI is booted according to the HIFMD (MD19), MD18, MD17, MD16, and MD14 settings determined when $\overline{\text{PRESET}}$ goes low.

(5) Boot Modes other than CS0 Boot Mode and Reset Exception Vectors after Return from Power Cut-Off State

When a reset exception is generated in boot mode other than CS0 boot mode, the vector addresses for exception processing are as listed in table 9.8.

Table 9.8 Reset Exception Vectors and Exception Codes in NAND Flash Boot, Serial Boot, MMC Boot, eSD Boot, and HIF Boot

No	Exception	Activation Boot Mode at Generation of Exception	Type of Exception (Power-on or H-UDI Reset) before Generation of Exception	Destination Vector for Exception Transition		Exception Code
				Vector Base	Offset	
(1)	Power-on reset (except power-on reset generated on return from deep standby mode)	CS0 boot	—	Refer to table 2.25, Exceptions, in section 2, SH-4A.		The value is the same for all boot modes. Refer to table 2.25, Exceptions, in section 2, SH-4A.
		HIF boot	—	H'FF820000	—	
		eSD boot, serial boot, NAND flash boot, and MMC boot	—	H'E5200000	—	
		Data-retention RAM boot	—	This is followed by rebooting in the mode selected by the levels on the MD19, MD18, MD17, MD16 and MD14 pins.		
(2)	Power-on reset (on return from deep standby mode)	CS0 boot	—	H'A0000000	BARH/L	
		HIF boot	—	H'FF820000	—	
		eSD boot, serial boot, NAND flash boot, and MMC boot	—	H'E5200000	—	
		Data-retention RAM boot	—	H'FF820000	HIAR	
(3)	Manual reset	CS0 boot	(1), (4)	Refer to table 2.25, Exceptions, in section 2, SH-4A.		
			(2)	H'A0000000	BARH/L	
		HIF boot	—	H'FF820000	—	
		eSD boot, serial boot, NAND flash boot, and MMC boot	—	H'E5200000	—	
		Data-retention RAM boot	—	H'FF820000	HIAR	

No	Exception	Activation Boot Mode at Generation of Exception	Type of Exception (Power-on or H-UDI Reset) before Generation of Exception	Destination Vector for Exception Transition		Exception Code
				Vector Base	Offset	
(4)	H-UDI reset	CS0 boot	—	Refer to table 2.25, Exceptions, in section 2, SH-4A.		The value is the same for all boot modes. Refer to table 2.25, Exceptions, in section 2, SH-4A.
		HIF boot	—	H'FF820000	—	
		eSD boot, serial boot, NAND flash boot, and MMC boot	—	H'E5200000	—	
		Data-retention RAM boot	—	This is followed by rebooting in the mode selected by the levels on the MD19, MD18, MD17, MD16 and MD14 pins.		
(5)	Instruction TLB multiple-hit exception	CS0 boot	(1), (4)	Refer to table 2.25, Exceptions, in section 2, SH-4A.		
			(2)	H'A0000000	BARH/L	
		HIF boot	—	H'FF820000	—	
		eSD boot, serial boot, NAND flash boot, and MMC boot	—	H'E5200000	—	
		Data-retention RAM boot	—	H'FF820000	HIAR	
(6)	Data TLB multiple-hit exception	CS0 boot	(1), (4)	Refer to table 2.25, Exceptions, in section 2, SH-4A.		
			(2)	H'A0000000	BARH/L	
		HIF boot	—	H'FF820000	—	
		eSD boot, serial boot, NAND flash boot, and MMC boot	—	H'E5200000	—	
		Data-retention RAM boot	—	H'FF820000	HIAR	

9.6 Points for Caution when Manual Reset Exceptions are Generated

In booting up from a device in CS0 or on the HIF, when a manual reset is generated while a pin required for the given boot mode is in use with another function, the pin-function settings remain as they are. Accordingly, the LSI chip is incapable of booting up correctly. When booting up from a device in CS0 or on the HIF, change the settings so that required pins are not in use with other functions or ensure that a manual reset is not generated while the pin function settings in use have been changed.

Section 10 R-GPVG

The R-GPVG is a graphics processor for OpenVG.

Contact your local sales representatives for details.

Related functions:

- Module standby (see section 9, Operating Modes and Power-Down Modes)
- Interrupts (see section 7, INTC/INTC2)
- Display unit (see section 14, Display Unit (DU))

Section 11 2D Graphic Accelerator (2DG)

The 2DG is a module which draws 2D graphics.

11.1 Features

(1) Drawing functions

Four-vertex drawing
Polygon drawing
Line drawing
High-functional bold line drawing
Antialias line drawing
BITBLT type commands with raster operation/alpha blending

(2) Color specification

Source coordinates: 1, 8, or 16 bit(s)/pixel
Drawing control coordinates: 8 or 16 bits/pixel
Work coordinates: 1 bit/pixel

(3) Screen coordinates

X: 0 to 4095
Y: 0 to 4095

(4) Register settings

Current pointer [MOVE/RMOVE]
Local offset [LCOFS/RLCOFS]
Specific address-mapped register [WPR]

(5) Sequence control

Jump [JUMP]
Sub routine [GOSUB] (Nesting level: 1)

11.1.1 List of Commands and Rendering Attributes

Table 11.1 List of Commands and Rendering Attributes

									Rendering Attributes							
									b15	b14	b13	b12	b11	b10	b9	b8
	OP CODE													DTRANS	WORK	
Command	b31	b30	b29	b28	b27	b26	b25	b24	Reserved	Reserved	CLIP	RCLIP	STRANS	/LINKE	/LREL	SS
POLYGON4A	1	0	0	0	0	0	1	0			CLIP	RCLIP	STRANS		WORK	SS
POLYGON4B					0	0	0	1			CLIP	RCLIP	STRANS		WORK	SS
POLYGON4C					0	0	0	0			CLIP	RCLIP			WORK	
LINEA	1	0	1	1	0	0	1	0			CLIP	RCLIP	STRANS			SS(0)
LINEB					0	0	0	1			CLIP	RCLIP	STRANS			SS(0)
LINEC					0	0	0	0			CLIP	RCLIP		LINKE	LREL	
LINED					0	0	1	1			CLIP	RCLIP		LINKE	LREL	
RLINEA					0	1	1	0			CLIP	RCLIP	STRANS			SS(0)
RLINEB					0	1	0	1			CLIP	RCLIP	STRANS			SS(0)
RLINEC					0	1	0	0			CLIP	RCLIP		LINKE	LREL	
RLINED					0	1	1	1			CLIP	RCLIP		LINKE	LREL	
FTRAPC	1	1	0	1	0	0	0	0			CLIP	RCLIP		LINKE	LREL	
RFTRAPC					0	1	0	0			CLIP	RCLIP		LINKE	LREL	
CLRWC	1	1	1	0	0	0	0	0			CLIP	RCLIP				
LINEWC	1	1	1	1	0	0	0	0			CLIP	RCLIP				
RLINEWC					0	1	0	0			CLIP	RCLIP				
BitBLTA	1	0	1	0	0	0	1	0			CLIP	RCLIP	STRANS	DTRANS	WORK	SS
BitBLTB					0	0	0	1			CLIP	RCLIP	STRANS	DTRANS	WORK	SS
BitBLTC					0	0	0	0			CLIP	RCLIP		DTRANS	WORK	

									Rendering Attributes							
									b7	b6	b5	b4	b3	b2	b1	b0
	OP CODE									STYLE	BLKE	NET/EDG	EOS		AA	CLKW
Command	b31	b30	b29	b28	b27	b26	b25	b24	REL	/SRCDIRX	/SRCDIRY	/DSTDIRX	/DSTDIRY	COOF	/αE	/SαE
POLYGON4A	1	0	0	0	0	0	1	0	REL	STYLE	BLKE	NET	EOS	COOF	αE	SαE
POLYGON4B					0	0	0	1	REL	STYLE	BLKE	NET	EOS	COOF	αE	
POLYGON4C					0	0	0	0			BLKE	NET	EOS	COOF	αE	
LINEA	1	0	1	1	0	0	1	0	REL	STYLE(1)		NET	EOS	COOF	AA	
LINEB					0	0	0	1	REL	STYLE(1)		NET	EOS	COOF	AA	
LINEC					0	0	0	0				NET	EOS	COOF	AA	
LINED					0	0	1	1							AA (1)	CLKW
RLINEA					0	1	1	0	REL	STYLE(1)		NET	EOS	COOF	AA	
RLINEB					0	1	0	1	REL	STYLE(1)		NET	EOS	COOF	AA	
RLINEC					0	1	0	0				NET	EOS	COOF	AA	
RLINED					0	1	1	1							AA (1)	CLKW
FTRAPC	1	1	0	1	0	0	0	0			BLKE(1)	EDG	EOS			
RFTRAPC					0	1	0	0			BLKE(1)	EDG	EOS			
CLRWC	1	1	1	0	0	0	0	0			BLKE(1)					
LINEWC	1	1	1	1	0	0	0	0					EOS			
RLINEWC					0	1	0	0					EOS			
BitBLTA	1	0	1	0	0	0	1	0	REL	SRCDIRX	SRCDIRY	DSTDIRX	DSTDIRY	COOF	αE	SαE
BitBLTB					0	0	0	1	REL	SRCDIRX	SRCDIRY	DSTDIRX	DSTDIRY	COOF	αE	
BitBLTC					0	0	0	0				DSTDIRX	DSTDIRY	COOF	αE	

Command	OP CODE b31	b30	b29	b28	b27	b26	b25	b24	Draw Mode b15	b14	b13	b12	b11	b10	b9	B8
TRAP	0	0	0	0	0	0	0	0								
NOP/INT	0	0	0	0	1	0	0	0	INT							
Reserve	0	0	0	1	0	0	0	0	This is for internal verification only and should not be set. Even if this setting is made, the command error (CER) flag is not set.							
WPR	0	0	0	1	1	0	0	0						LINKE	LREL	
JUMP	0	0	1	0	1	0	0	0								
GOSUB	0	0	1	1	0	0	0	0								
RET	0	0	1	1	1	0	0	0								
LCOFS	0	1	0	0	0	0	0	0								
RLCOFS	0	1	0	0	0	1	0	0								
MOVE	0	1	0	0	1	0	0	0								
RMOVE	0	1	0	0	1	1	0	0								
SYNC	0	0	0	1	0	0	1	0							WCLR	WFLSH
Reserve	0	1	1	0	0	1	0	0	This is for internal verification only and should not be set. Even if this setting is made, the command error (CER) flag is not set.							
Reserve	0	1	0	1	0	0	0	0	This is for internal verification only and should not be set. Even if this setting is made, the command error (CER) flag is not set.							

Command	OP CODE b31	b30	b29	b28	b27	b26	b25	b24	Draw Mode b7	b6	b5	b4	b3	b2	b1	b0
TRAP	0	0	0	0	0	0	0	0								
NOP/INT	0	0	0	0	1	0	0	0	INT No							
Reserve	0	0	0	1	0	0	0	0	This is for internal verification only and should not be set. Even if this setting is made, the command error (CER) flag is not set.							
WPR	0	0	0	1	1	0	0	0					ByteM3	ByteM2	ByteM1	ByteM0
JUMP	0	0	1	0	1	0	0	0	REL							
GOSUB	0	0	1	1	0	0	0	0	REL							No
RET	0	0	1	1	1	0	0	0								No
LCOFS	0	1	0	0	0	0	0	0								
RLCOFS	0	1	0	0	0	1	0	0								
MOVE	0	1	0	0	1	0	0	0								
RMOVE	0	1	0	0	1	1	0	0								
SYNC	0	0	0	1	0	0	1	0				TCLR			DCLR	DFLSH
Reserve	0	1	1	0	0	1	0	0	This is for internal verification only and should not be set. Even if this setting is made, the command error (CER) flag is not set.							
Reserve	0	1	0	1	0	0	0	0	This is for internal verification only and should not be set. Even if this setting is made, the command error (CER) flag is not set.							

Note: Clear the shaded bits to 0.

REL: Valid only when SS = 0. Clear this bit to 0 when SS = 1.

COOF: Valid only in 16-bit/pixel mode (GBM = 1). Clear this bit to 0 in 8-bit/pixel mode (GBM = 0).

SαE: Valid only for the ARGB format (SPF = DPF = 1). Clear this bit to 0 for the RGB format (SPF = DPF = 0) and 8-bit/pixel mode (GBM = 0).
Clear this bit to 0 when αE = 0.

αE: Valid only in 16-bit/pixel mode (GBM = 1). Clear this bit to 0 in 8-bit/pixel mode (GBM = 0). In the POLYGON4A, POLYGON4B, or POLYGON4C command, valid only when BLKE = 1. Clear this bit to 0 when BLKE = 0. In the BITBLTA, BITBLTB, or BITBLTC command, valid only when the ROP code = H'CC. Clear this bit to 0 for other codes.

LREL: Valid only when LINKE = 1. The LREL bit should be cleared to 0 when LINKE = 0.

STYLE: Set this bit to 1 when BLKE = 1. In the LINEA, LINEB, RLINEA, or RLINEB command, set this bit to 1.

AA: Clear this bit to 0 when NET = 1. Valid only in 16-bit/pixel mode (GBM = 1). Clear this bit to 0 in 8-bit/pixel mode (GBM = 0).
In the LINED or RLINED command, set this bit to 1.

SS: In the LINEA, LINEB, RLINEA, RLINEB or AAFC command, clear this bit to 0. In the AAFA command, set this bit to 1.

BLKE: In the FTRAPC, RFTRAPC, or CLRWC command, set this bit to 1.

11.1.2 Block Diagram

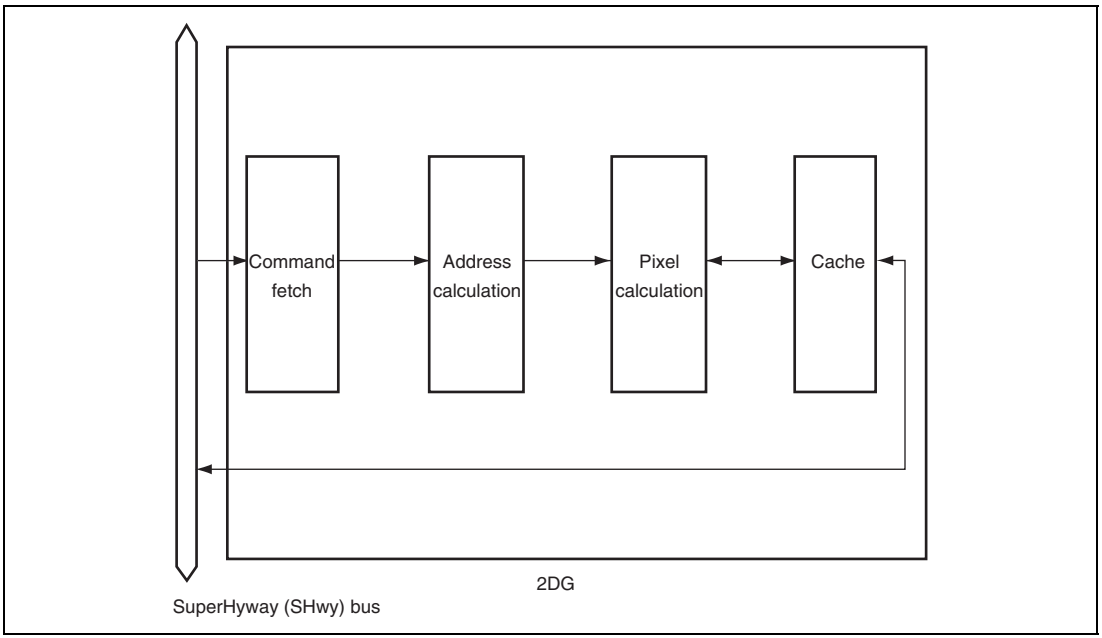


Figure 11.1 2DG Block Diagram

11.2 Register Description

Table 11.2 shows the 2DG register configuration.

The bit width of the 2DG registers is 32. Access the 2DG registers in units of 32 bits (longwords) and do not access registers other than ones in table 11.2. Otherwise, a correct operation cannot be guaranteed.

The CPU writing to registers, excluding the system control registers (SCLR), is prohibited during the time from the start of rendering to the TRAP command execution, except for the drawing halted period specified by the INT command. Hereafter, "reset" refers to both a hardware reset (a power-on reset and a manual reset) and software reset unless specified otherwise.

Each register values are retained in module standby mode, or sleep mode, but initialized by a reset or in deep standby mode.

Table 11.2 Register Configuration

(1) System control register

Register name	Abbreviation	R/W	Address	Access size	Setting by WPR Command	Byte M Control by WPR Command
System control registers	SCLR	R/W	H'FFE8 0000	32	×	—
Status register	SR	R	H'FFE8 0004	32	×	—
Status register clear register	SRCR	W	H'FFE8 0008	32	×	—
Interrupt enable register	IER	R/W	H'FFE8 000C	32	O	O
Interrupt command ID register	ICIDR	R	H'FFE8 0010	32	×	—

(2) Memory control registers

Register name	Abbreviation	R/W	Address	Access size	Setting by WPR Command	Byte M Control by WPR Command
Return address register 0	RTN0R	R/W	H'FFE8 0040	32	O	×
Return address register 1	RTN1R	R/W	H'FFE8 0044	32	O	×
Display list start address register	DLSAR	R/W	H'FFE8 0048	32	×	—
2-dimensional source area start address register	SSAR	R/W	H'FFE8 004C	32	O	×
Rendering start address register	RSAR	R/W	H'FFE8 0050	32	O	×
Work area start address register	WSAR	R/W	H'FFE8 0054	32	O	×
Source stride register	SSTRR	R/W	H'FFE8 0058	32	O	×
Destination stride register	DSTRR	R/W	H'FFE8 005C	32	O	×
Endian conversion control register	ENDCVR	R/W	H'FFE8 0060	32	×	—
Address extension register	ADREXTR	R/W	H'FFE8 006C	32	×	—

(3) Color control registers

Register name	Abbreviation	R/W	Address	Access size	Setting by WPR Command	Byte M Control by WPR Command
Source transparent color register	STCR	R/W	H'FFE8 0080	32	O	×
Destination transparent color register	DTCR	R/W	H'FFE8 0084	32	O	×
Alpha value register	ALPHR	R/W	H'FFE8 0088	32	O	×
Color offset register	COFSR	R/W	H'FFE8 008C	32	O	×

(4) rendering control register

Register name	Abbreviation	R/W	Address	Access size	Setting by WPR Command	Byte M Control by WPR Command
Rendering control register	RCLR	R/W	H'FFE8 00C0	32	O	O
Command status register	CSTR	R	H'FFE8 00C4	32	×	—
Current pointer register	CURR	R	H'FFE8 00C8	32	×	—
Local offset register	LCOR	R	H'FFE8 00CC	32	×	—
System clipping area MAX register	SCLMAR	R	H'FFE8 00D0	32	O	×
User clipping area MIN register	UCLMIR	R	H'FFE8 00D4	32	O	×
User clipping area MAX register	UCLMAR	R	H'FFE8 00D8	32	O	×
Relative user clipping area MIN register	RUCLMIR	R	H'FFE8 00DC	32	O	×
Relative user clipping area MAX register	RUCLMAR	R	H'FFE8 00E0	32	O	×
Rendering control 2 register	RCL2R	R/W	H'FFE8 00F0	32	O	O
Pattern offset register	POFSR	R/W	H'FFE8 00F8	32	O	×

11.2.1 System Control Register (SCLR)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SRES	HRES	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	1	0	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RS
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	0	—	—	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	SRES	1	R/W	Software Reset Resets the 2DG. 0: Command processing execution is enabled. 1: Reset state This bit is set to 1 when a hardware reset is performed. Clear this bit to 0 in initialization. When this bit is set to 1 by software, a reset is performed for drawing operations only. The 2DG registers are also initialized. While this bit is set to 1, this is the only register that can be written to. Note: Release a software rest according to the following procedure after performing the software reset during the drawing period (from rendering start to TRAP command execution). 1. Set SRES to 1. 2. Wait for one vsync period. 3. Set HRES to 1 (retain this state at least four cycles of CLKP). 4. Set HRES to 0. 5. Set SRES to 0.
30	HRES	0	R/W	Executes a reset (corresponding to a hardware reset) on the 2DG. Use this bit only to release a software reset. If it is used when the 2DG operates, a correct operation cannot be guaranteed.

Bit	Bit Name	Initial Value	R/W	Description
29 to 4	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2, 1	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
0	RS	0	R	Rendering Start Specifies the start of rendering. During the drawing period (from rendering start to TRAP command execution), writing 1 to this bit is prohibited. 0: Rendering is not started. 1: Rendering is started. This bit is cleared to 0 after rendering starts. Note: Setting both the SRES and RS bits to 1 simultaneously is prohibited.

11.2.2 Status Register (SR)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	VER[3:0]				—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	1	0	1	1	—	0	0	0	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	CER	INT	TRA
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	VER[3:0]	1011	R	Version Flag This flag is read as 1011.
27	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
26 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 4	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	CER	0	R	Command Error Flag Flag that indicates that an illegal command has been fetched. 0: Normal state. An illegal command has not been fetched since CER flag clearing by the SRES bit in SCLR or the CECL bit in SRCR. An illegal command is one in which the upper eight bits of the command code are undefined. The 2DG does not check the legality of the rendering attributes in the lower 16 bits. 1: Drawing operation halt state. Drawing operation remains halted because an illegal command was fetched after CER flag clearing by the SRES bit in SCLR or the CECL bit in SRCR. To resume drawing operation, after executing a software reset, make the bit setting for rendering start. The CER flag retains its state until cleared by a reset or by SRCR.

Bit	Bit Name	Initial Value	R/W	Description
1	INT	0	R	Interrupt Flag Flag that indicates that the NOP/INT command has been fetched (only when the rendering attribute INT bit is 1). 0: The NOP/INT command has not been fetched since INT flag clearing by the SRES bit in SCLR or the INCL bit in SRCR. 1: Drawing operation halt state. Drawing operation remains halted because the NOP/INT command was fetched after INT flag clearing by the SRES bit in SCLR or the INCL bit in SRCR (only when the rendering attribute INT bit is 1). Clearing the INT flag by the INCL bit in SRCR resumes drawing operation from the next command. The CPU writing to registers, excluding the system control registers (SCLR), is prohibited during the time from the start of drawing to the TRAP command execution. The INT flag retains its state until cleared by a reset or by SRCR. Note: Do not rewrite the display list when drawing operation is halted by the INT command.
0	TRA	0	R	Trap Flag Flag that indicates the end of command execution. 0: The TRAP command has not been fetched since TRA flag clearing by the SRES bit in SCLR or the TRCL bit in SRCR. 1: Command execution has ended, or the current command is not being executed. The TRA flag retains its state until cleared by a reset or by SRCR.

11.2.3 Status Register Clear Register (SRCR)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	CECL	INCL	TRCL
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	Undefined	W	Reserved The write value should always be 0.
3	—	0	W	Reserved The write value should always be 0.
2	CECL	0	W	Command Error Flag Selects whether the CER flag in SR is cleared or not. 0: The CER flag in SR is not cleared to 0. 1: The CER flag in SR is cleared to 0. When SR clearing is completed, all of the values in SRCR are cleared to 0 internally.
1	INCL	0	W	Interrupt Flag Clear Selects whether the INT flag in SR is cleared or not. 0: The INT flag in SR is not cleared to 0. 1: The INT flag in SR is cleared to 0. When SR clearing is completed, all of the values in SRCR are cleared to 0 internally.
0	TRCL	0	W	Trap Flag Clear Selects whether the TRA flag in SR is cleared or not. 0: The TRA flag in SR is not cleared to 0. 1: The TRA flag in SR is cleared to 0. When SR clearing is completed, all of the values in SRCR are cleared to 0 internally.

11.2.4 Interrupt Enable Register (IER)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	CEE	INE	TRE
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
3	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
2	CEE	0	R/W	Command Error Flag Enable Enables or disables interrupts initiated by the CER flag in SR. 0: Interrupts initiated by the CER flag in SR are disabled. 1: Interrupts initiated by the CER flag in SR are enabled.
1	INE	0	R/W	Interrupt Flag Enable Enables or disables interrupts initiated by the INT flag in SR. 0: Interrupts initiated by the INT flag in SR are disabled. 1: Interrupts initiated by the INT flag in SR are enabled.
0	TRE	0	R/W	Trap Flag Enable Enables or disables interrupts initiated by the TRA flag in SR. 0: Interrupts initiated by the TRA flag in SR are disabled. 1: Interrupts initiated by the TRA flag in SR are enabled.

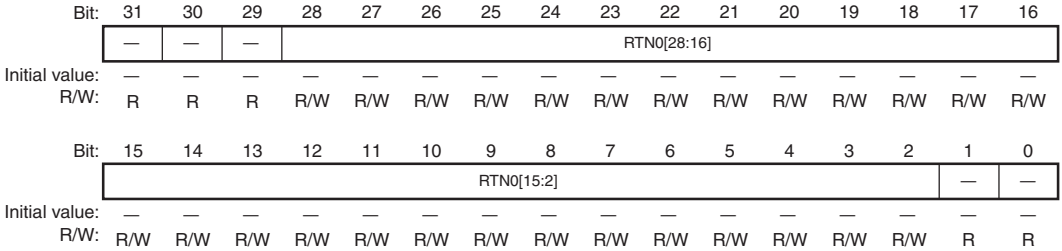
11.2.5 Interrupt Command ID Register (ICIDR)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ICID[7:0]							
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

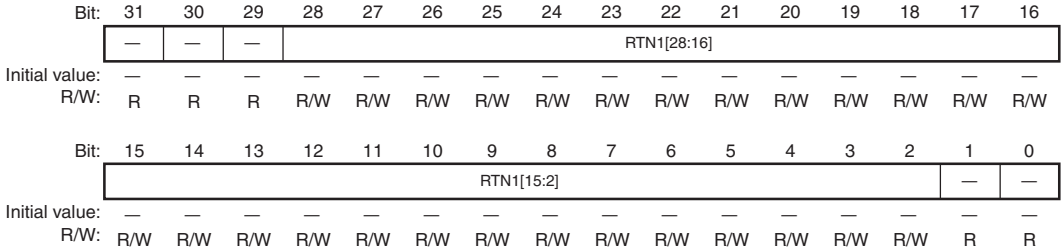
Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
7 to 0	ICID	Undefined	R	Interrupt command ID Stores the ID specified by the rendering attribute if the rendering attribute INT bit is set to 1 when the NOP/INT command is fetched.

11.2.6 Return Address Register 0 (RTN0R)



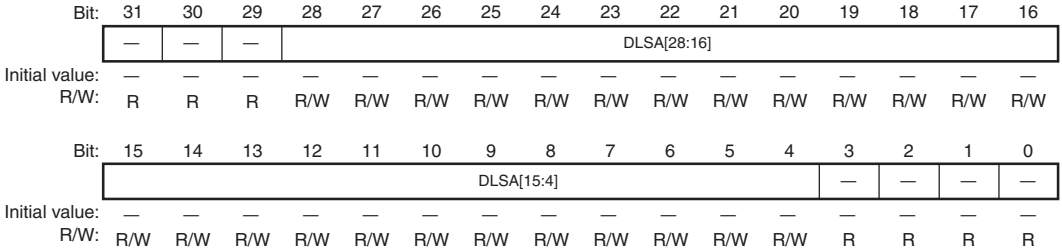
Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
28 to 2	RTN0 [28:2]	Undefined	R/W	Return Address 0 (A28 to A2) Stores the return address when the rendering attribute No bit is 0 in the GOSUB command. The address indicated by RTN0R is a longword address (bits A28 to A2).
1, 0	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.

11.2.7 Return Address Register 1 (RTN1R)



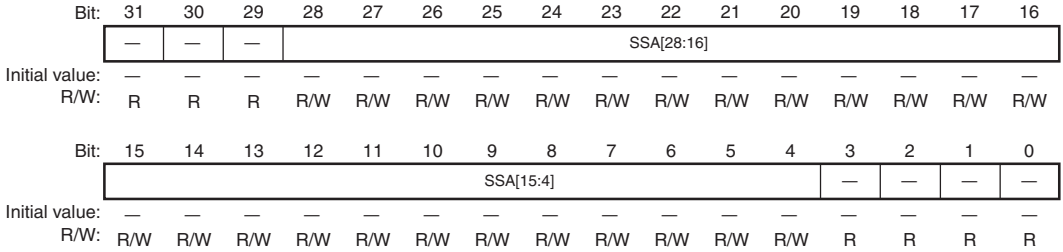
Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
28 to 2	RTN1 [28:2]	Undefined	R/W	Return Address 1 (A28 to A2) Stores the return address when the rendering attribute No bit is 1 in the GOSUB command. The address indicated by RTN1R is a longword address (bits A28 to A2).
1, 0	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.

11.2.8 Display List Start Address Register (DLSAR)



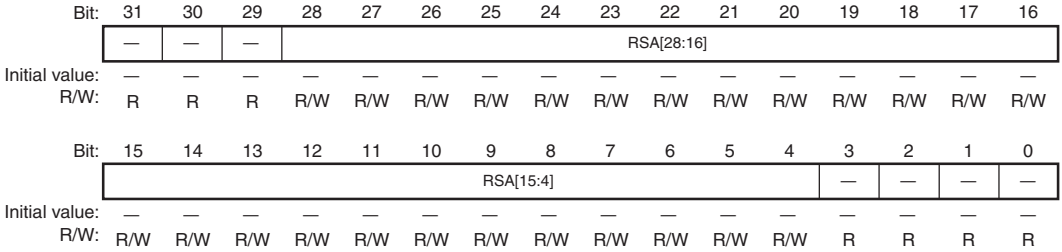
Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
28 to 4	DLSA[28:4]	Undefined	R/W	Display List Start Address (A28 to A4) Specifies the memory area to be used as the display list. The start physical address (bits A28 to A0) of the display list is set in 16-byte units. Even in 32-bit addressing mode, write the lower 29 bits of the specified 32-bit address to bits 28 to 0.
3 to 0	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.

11.2.9 2-Dimensional Source Area Start Address Register (SSAR)



Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
28 to 4	SSA[28:4]	Undefined	R/W	2-Dimensional Source Area Start Address (A28 to A4) Specifies the memory area to be used as the 2-dimensional source area. The physical address set in this register becomes the physical address for the origin of the 2-dimensional source coordinates. The start physical address (bits A28 to A0) of the 2-dimensional source area is set in 16-byte units. Even in 32-bit addressing mode, write the lower 29 bits of the specified 32-bit address to bits 28 to 0.
3 to 0	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.

11.2.10 Rendering Start Address Register (RSAR)



Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
28 to 4	RSA[28:4]	Undefined	R/W	Rendering Start Address (A28 to A4) Specifies the memory area to be used as the rendering area. The physical address set in this register becomes the physical address for the rendering coordinate origin. The start physical address (bits A28 to A0) of the rendering area is set in 16-byte units. Even in 32-bit addressing mode, write the lower 29 bits of the specified 32-bit address to bits 28 to 0. Set the rendering start address so that the rendering area does not overlap with the work area.
3 to 0	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.

11.2.11 Work Area Start Address Register (WSAR)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	WSA[28:16]												
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WSA[15:4]												—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	Undefined	R	Reserved The read value is undefined.
28 to 4	WSA[28:4]	Undefined	R/W	Work Area Start Address (A28 to A4) Specifies the memory area to be used as the work area. The physical address set in this register becomes the physical address for the work coordinate origin. The start physical address (bits A28 to A0) of the work area is set in 16-byte units. Even in 32-bit addressing mode, write the lower 29 bits of the specified 32-bit address to bits 28 to 0. Use only the work drawing commands for drawing in the work area. When writing to the work area by the CPU, avoid the drawing period (from rendering start to TRAP command execution (including the drawing halt period specified by the NOP/INT command)). Do not use a figure drawn by a work drawing command as the source figure.
3 to 0	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.

11.2.12 Source Stride Register (SSTRR)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	SSTR[12:3]										—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
12 to 3	SSSTR[12:3]	Undefined	R/W	Source Stride (b12 to b3) Specifies the stride of the 2-dimensional source area in pixel units. Set the value in the range of 8 ≤ SSTR ≤ 4096 (on a basis of 8 pixels).
2 to 0	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.

11.2.13 Destination Stride Register (DSTRR)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	DSTR[12:4]									—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
12 to 4	DSTR[12:4]	Undefined	R/W	Destination Stride (b12 to b4) Specifies the stride of the destination area in pixel units. Set the value in the range of 256 ≤ DSTR ≤ 4096 (on a basis of 64 pixels). Note: Set 0 to bits 5 and 4.
3 to 0	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.

11.2.14 Endian Conversion Control Register (ENDCVR)

For an endian conversion method, see section 11.3.3, Endian Conversion.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	—	—	—	0	0	—	—	0	0	0	—	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	LW SWAP	WSWAP	BYTE SWAP	BIT SWAP
Initial value:	0	0	0	0	0	0	0	—	0	0	0	—	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30 to 28	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
27, 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25, 24	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
23 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
19 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
3	LWSWAP	0	R/W	Longword Swap Swaps data in longword (32-bit) units. 0: Data is not swapped. 1: Data is swapped in longword (32-bit) units.
2	WSWAP	0	R/W	Word Swap Swaps data in word (16-bit) units. 0: Data is not swapped. 1: Data is swapped in word (16-bit) units.
1	BYTESWAP	0	R/W	Byte Swap Swaps data in byte (8-bit) units. 0: Data is not swapped. 1: Data is swapped in byte (8-bit) units.
0	BITSWAP	0	R/W	Bit Swap Swaps data in bit units. 0: Data is not swapped. 1: Data is swapped in one-bit units.

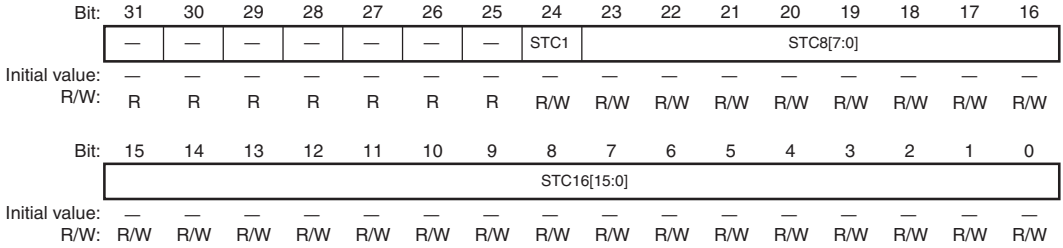
11.2.15 Address Extension Register (ADREXTR)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ADREXT[31:29]			—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	ADREXT [31:29]	All 0	R/W	Extended Address (A31 to A29) Specifies the upper three bits (A31 to A29) of the address to be output to the SuperHway.
28 to 0	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.

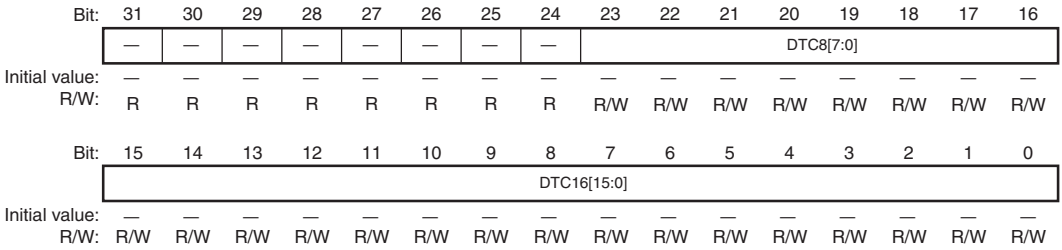
11.2.16 Source Transparent Color Register (STCR)



Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
24	STC1	Undefined	R/W	Source Transparent Color 1 Transparent color for 1-bit/pixel source
23 to 16	STC8[7:0]	Undefined	R/W	Source Transparent Color 8 Transparent color for 8-bit/pixel source
15 to 0	STC16[15:0]	Undefined	R/W	Source Transparent Color 16 Transparent color for 16-bit/pixel source

Note: For 16-bit/pixel source data, use the same format specified by the SPF bit in the rendering control register (RCLR). When SPF = 1 (ARGB = 1555), the A value is not compared.

11.2.17 Destination Transparent Color Register (DTCR)



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
23 to 16	DTC8[7:0]	Undefined	R/W	Destination Transparent Color 8 Transparent color for 8-bit/pixel destination
15 to 0	DTC16[15:0]	Undefined	R/W	Destination Transparent Color 16 Transparent color for 16-bit/pixel destination

Note: For 16-bit/pixel destination data, use the same format specified by the DPF bit in the rendering control register (RCLR).
When DPF = 1 (ARGB = 1555), the value A is not compared.

11.2.18 Alpha Value Register (ALPHR)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ALPH[7:2]						—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
7 to 2	ALPH[7:2]	Undefined	R/W	Alpha Value (b7 to b2) Specifies the alpha blending value when the rendering attribute α E bit is set to 1. For blending of the blue and red components, the upper five bits of the alpha value are valid. For blending of the green component, the upper six bits are valid when the destination pixel format is RGB and the upper five bits are valid when it is ARGB.
1, 0	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.

11.2.19 Color Offset Register (COFSR)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	COR[7:3]					—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	COG[7:2]						—	—	COB[7:3]					—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
23 to 19	COR[7:3]	Undefined	R/W	Color Offset R Color offset red component (b7 to b3). The offset components are treated as signed integers. Negative numbers are expressed as two's complement.
18 to 16	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
15 to 10	COG[7:2]	Undefined	R/W	Color Offset G Color offset green component (b7 to b2). The offset components are treated as signed integers. Negative numbers are expressed as two's complement. The upper five bits are valid in the ARGB format.
9, 8	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
7 to 3	COB[7:3]	Undefined	R/W	Color Offset B Color offset blue component (b7 to b3). The offset components are treated as signed integers. Negative numbers are expressed as two's complement.
2 to 0	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.

11.2.20 Rendering Control Register (RCLR)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	STP	DTP	—	—	SPF	DPF	—	GBM	SAU	AVALUE
Initial value:	0	0	0	0	0	—	0	0	—	—	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R/W	R/W	R	R/W	R/W	R/W

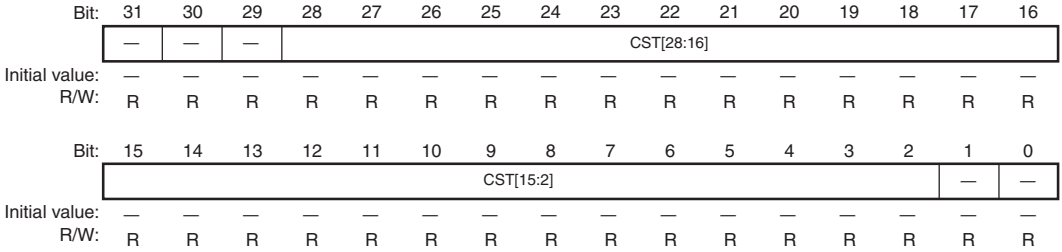
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LPCE	COM
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
25	STP	0	R/W	Source Transparent Color Polarity Selects whether source transparency occurs when the source data and the value set in the source transparent color register (DTCR) match or do not match. 0: Source transparency at a match 1: Source transparency at an unmatch
24	DTP	0	R/W	Destination Transparent Color Polarity Selects whether destination transparency occurs when the destination data and the value set in the destination transparent color register (DTCR) match or do not match. 0: Destination transparency at a match 1: Destination transparency at an unmatch
23, 22	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
21	SPF	0	R/W	Source Pixel Format Specifies the pixel format for the multi-valued source. This setting is valid only for a multi-valued 16-bit/pixel source. This bit should be cleared to 0 for an 8-bit-pixel destination. Set this bit to match the destination pixel format. 0: RGB = 565 format 1: ARGB = 1555 format
20	DPF	0	R/W	Destination Pixel Format Specifies the pixel format for the destination. This setting is valid only for a 16-bit/pixel destination. This bit should be cleared to 0 for an 8-bit-pixel destination. Set this bit to match the multi-valued source pixel format. 0: RGB = 565 format 1: ARGB = 1555 format
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
18	GBM	0	R/W	Graphic Bit Mode Specifies the graphic bit mode for the multi-valued source and destination. 0: 8-bit/pixel 1: 16-bit/pixel
17	SAU	0	R/W	Source Value A Use When the pixel format of the source and destination is the ARGB format, drawing is performed while referencing the source value A as the destination value A. 0: The destination value A is drawn as AVALUE. 1: The destination value A is drawn referencing the source value A. Note: If SAU is set to 1, command parameters Color0 and Color1 A values are referenced in the command which references a binary source and command parameter Color A value is referenced in the command which specifies monochrome. Ground (Destination) value A is redrawn in the LINED command regardless of the settings in the SAU and AVALUE bits.

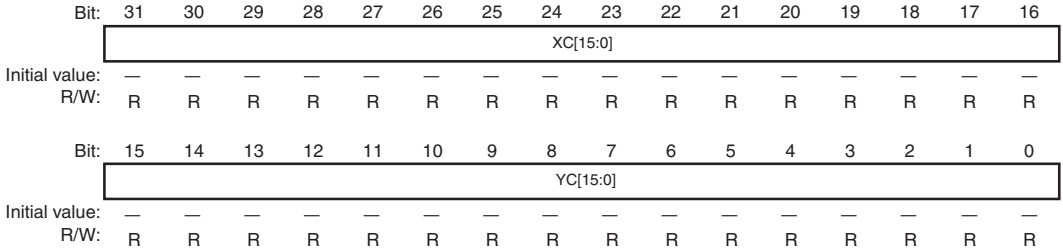
Bit	Bit Name	Initial Value	R/W	Description
16	AVALUE	0	R/W	Value A When the pixel format of the source and destination is the ARGB format, drawing is performed with the destination value A as 0 or 1. 0: The destination value A is drawn as 0. 1: The destination value A is drawn as 1.
15 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	—	1	R	Reserved This bit is always read as 0. The write value should always be 0.
1	LPCE	0	R/W	Line Pre-Clipping Enable This bit is valid for the RLINE and RLINEW type commands. When this bit is set to 1, pre-clipping is performed in line-segment units in the 2-dimensional clipping areas (system clipping, user clipping, and relative user clipping areas). If a line segment in the middle is pre-clipped, the pattern continuity is broken (the pattern starts from the final point of the line segment previously drawn). 0: Pre-clipping is not performed. 1: pre-clipping is performed in line-segment units in the 2-dimensional clipping areas.
0	CDM	0	R/W	Connection Drawing Mask Selects whether the linkage parts of bold lines are drawn or not. 0: The linkage parts of bold lines are drawn. 1: The linkage parts of bold lines are not drawn.

11.2.21 Command Status Register (CSTR)



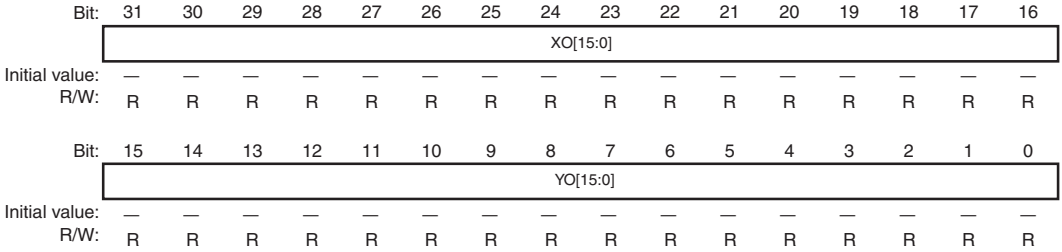
Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
28 to 2	CST[28:2]	Undefined	R	Command Status (A28 to A2) Stores the address of the fetched command word (op code word). The address indicated by CSTR is a longword address (bits A28 to A2).
1, 0	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.

11.2.22 Current Pointer Register (CURR)



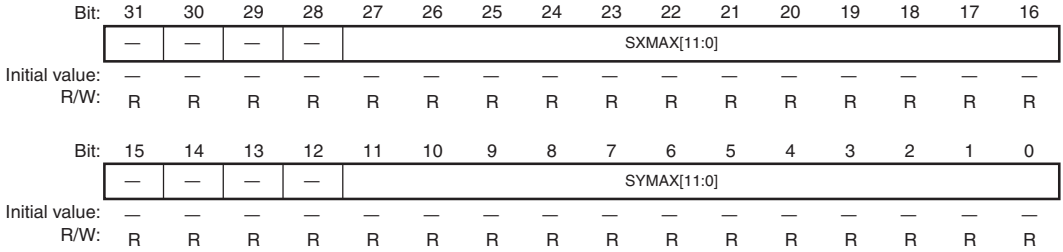
Bit	Bit Name	Initial Value	R/W	Description
31 to 16	XC[15:0]	Undefined	R	Current Pointer X The X coordinate of the current pointer
15 to 0	YC[15:0]	Undefined	R	Current Pointer Y The Y coordinate of the current pointer

11.2.23 Local Offset Register (LCOR)



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	XO[15:0]	Undefined	R	Local Offset X The X coordinate of the local offset
15 to 0	YO[15:0]	Undefined	R	Local Offset Y The Y coordinate of the local offset

11.2.24 System Clipping Area MAX Register (SCLMAR)



Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
27 to 16	SXMAX[11:0]	Undefined	R	System Clipping XMAX The XMAX of the system clipping coordinate.
15 to 12	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
11 to 0	SYMAX[11:0]	Undefined	R	System Clipping YMAX The YMAX of the system clipping coordinate.

Note: When setting this register by the WPR command, set the maximum values of the drawing range (Max. 4095. SXMAX < DSTRR).

11.2.25 User Clipping Area MIN Register (UCLMIR)

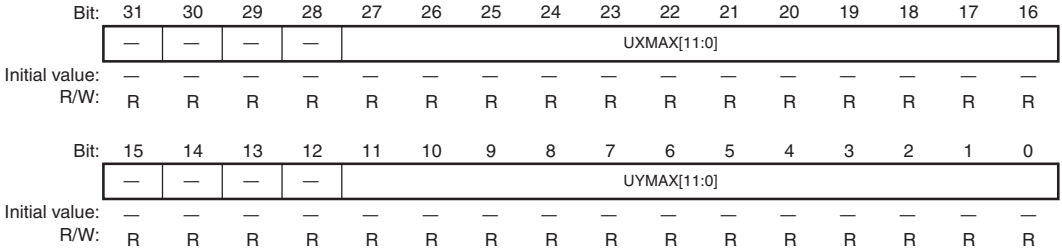
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	UXMIN[11:0]											
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	UYMIN[11:0]											
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
27 to 16	UXMIN[11:0]	Undefined	R	User Clipping XMIN The XMIN of the user clipping coordinates.
15 to 12	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
11 to 0	UYMIN[11:0]	Undefined	R	User Clipping YMIN The YMIN of the user clipping coordinates.

Note: When setting this register by the WPR command, set UXMAX and UYMAX in the following ranges: $0 \leq UXMIN < UXMAX \leq SXMAX \leq 4095$, $0 \leq UYMIN < UYMAX \leq SYMAX \leq 4095$.

11.2.26 User Clipping Area MAX Register (UCLMAR)



Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
27 to 16	UXMAX [11:0]	Undefined	R	User Clipping XMAX The XMAX of the user clipping coordinates.
15 to 12	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
11 to 0	UYMAX [11:0]	Undefined	R	User Clipping YMAX The YMAX of the user clipping coordinates.

Note: When setting this register by the WPR command, set UXMAX and UYMAX in the following ranges: $0 \leq UXMIN < UXMAX \leq SXMAX \leq 4095$, $0 \leq UYMIN < UYMAX \leq SYMAX \leq 4095$.

11.2.27 Relative User Clipping Area MIN Register (RUCLMIR)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	RUXMIN[11:0]											
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	RUYMIN[11:0]											
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
27 to 16	RUXMIN[11:0]	Undefined	R	Relative User Clipping XMIN The XMIN of the relative user clipping coordinates (offset values added to the local offset).
15 to 12	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
11 to 0	RUYMIN[11:0]	Undefined	R	Relative User Clipping YMIN The YMIN of the relative user clipping coordinates (offset values added to the local offset).

Note: When setting this register by the WPR command, set RUXMAX and RUYMAX in the following ranges: $0 \leq RUXMIN < RUXMAX \leq SXMAX \leq 4095$, $0 \leq RUYMIN < RUYMAX \leq SYMAX \leq 4095$.

For details on the setting ranges, see (5) Relative Clipping Specification (RCLIP), in section 11.3.4, Rendering Attributes.

11.2.28 Relative User Clipping Area MAX Register (RUCLMAR)

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
27 to 16	RUXMAX [11:0]	Undefined	R	Relative User Clipping XMAX The XMAX of the relative user clipping coordinates (offset values added to the local offset).
15 to 12	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
11 to 0	RUYMAX [11:0]	Undefined	R	Relative User Clipping YMAX The YMAX of the relative user clipping coordinates (offset values added to the local offset).

Note: When setting this register by the WPR command, set RUXMAX and RUYMAX in the following ranges: $0 \leq RUXMIN < RUXMAX \leq SXMAX \leq 4095$, $0 \leq RUYMIN < RUYMAX \leq SYMAX \leq 4095$.

For details on the setting ranges, see (5) Relative Clipping Specification (RCLIP), in section 11.3.4, Rendering Attributes.

11.2.29 Rendering Control 2 Register (RCL2R)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	DAE	PSTYLE	PXSIZE[1:0]		PYSIZE[1:0]	
Initial value:	—	—	—	—	0	0	—	—	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	1	0	0	—	0	0	0	0	0	0	0	0	1	0	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
27, 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25, 24	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
23, 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
21	DAE	0	R/W	Destination Alpha Enable This bit is used in combination with the alpha blend enable (αE) bit. In the ARGB = 1555 format, only the pixels whose destination (ground) value A is 1 are alpha blended. Pixels whose destination (ground) value A is 0 are not drawn. 0: Alpha blending is performed regardless of the destination (ground) value A. 1: Only the pixels whose destination (ground) value A is 1 are alpha blended. Notes: 1. Clear this bit to 0 for the RGB = 565 format or at 8-bit-pixel drawing. 2. Clear this bit to 0 for commands other than POLYGON4A, POLYGON4B, and POLYGON4C commands. 3. Clear this bit to 0 when the alpha blend enable bit (αE) is 0. 4. This bit is not decoded by a command so it must be set or cleared in each relevant command.
20	PSTYLE	0	R/W	Pattern Style Enable This bit is used in combination with the source style specification (STYLE). The source pattern is created repeatedly in the pattern size based on the destination coordinates. 0: Pattern style disabled 1: The source pattern is created based on the destination coordinates. Notes: 1. Set the source offset TXOFS and TYOFS to 0. 2. Clear this bit to 0 when the source style specification bit (STYLE) is 0. 3. Clear the source address specification bit (SS) to 0. 4. Clear this bit to 0 for commands other than the POLYGON4A and POLYGON4B commands. 5. This bit is not decoded by a command so it must be set or cleared in each relevant command.

Bit	Bit Name	Initial Value	R/W	Description
19, 18	PXSIZE	All 0	R/W	Pattern X Size These bits specify the pattern X size when the pattern style enable (PSTYLE) bit is 1. 00: Pattern X size = 8 pixels 01: Pattern X size = 16 pixels 10: Pattern X size = 32 pixels 11: Pattern X size = 64 pixels Note: Set the specified pattern X size (8, 16, 32, or 64) in the source size TDX.
17, 16	PYSIZE	All 0	R/W	Pattern Y Size These bits specify the pattern Y size when the pattern style enable (PSTYLE) bit is 1. 00: Pattern Y size = 8 pixels 01: Pattern Y size = 16 pixels 10: Pattern Y size = 32 pixels 11: Pattern Y size = 64 pixels Note: Set the specified pattern Y size (8, 16, 32, or 64) in the source size TDY.
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
13, 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
10 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.

Bit	Bit Name	Initial Value	R/W	Description
1	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
0	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.

11.2.30 Pattern Offset Register (POFSR)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	POFSX[11:0]											
Initial value:	—	—	—	—	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	POFSY[11:0]											
Initial value:	—	—	—	—	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
27 to 16	POFSX[11:0]	H'000	R/W	Pattern Offset X These bits specify the pattern offset value in the X direction as a 16-bit integer. Negative numbers are expressed as two's complement.
15 to 12	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
11 to 0	POFSY[11:0]	H'000	R/W	Pattern Offset Y These bits specify the pattern offset value in the Y direction as a 16-bit integer. Negative numbers are expressed as two's complement.

11.3 Operation

11.3.1 Basic Functions

(1) Bold Line Drawing

A bold line can be drawn by setting a value greater than 0 as line width W in a LINEA/B/C command or a RLINEA/B/C command. The bold line coordinates a, b, c, and d are obtained from the starting and final coordinate points and line width W, and the bold line drawn. W is set in the 6-bit integer part. When 0 is set in W, a line of line width 1 is drawn. The connection drawing mask bit (COM) in the rendering control register (RCLR) is used to select whether the linkage parts of bold lines are drawn or not. When the starting and final coordinate points of a line segment match in bold line drawing, nothing is drawn.

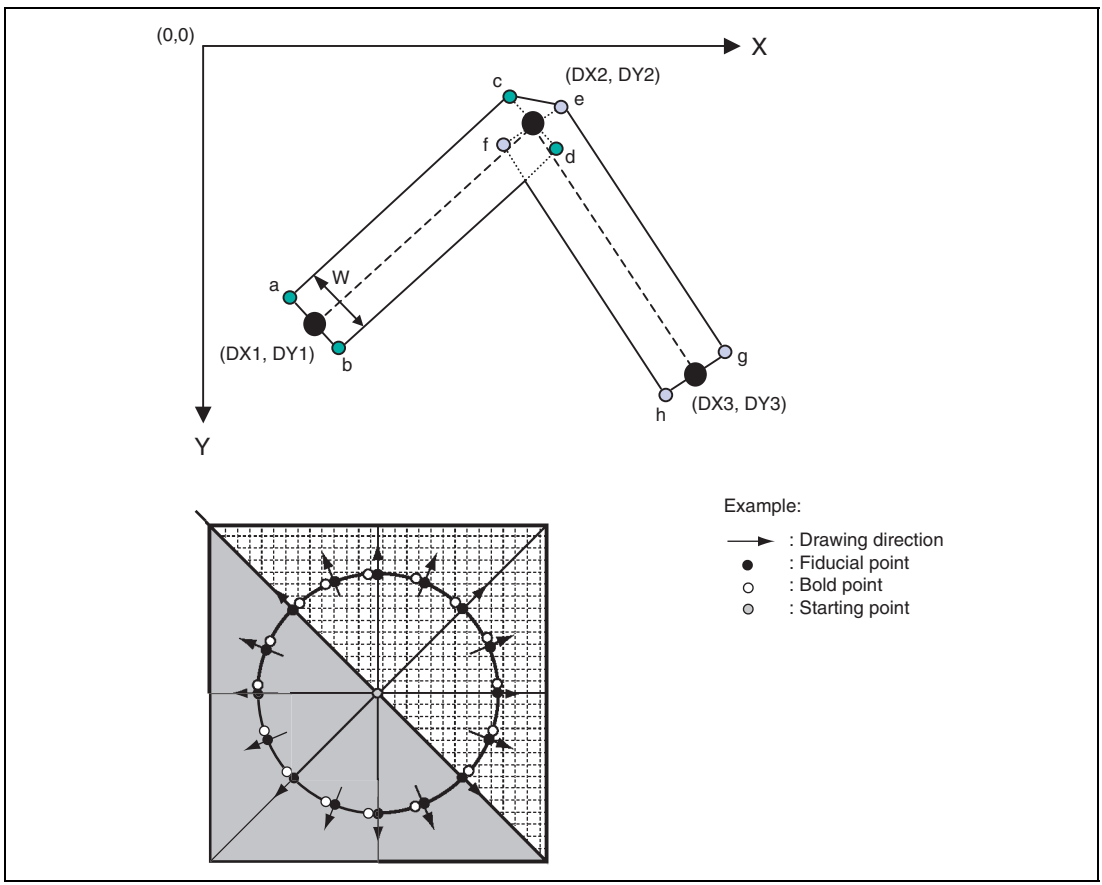


Figure 11.2 Bold Line Drawing

(2) Antialiasing

Antialiasing which reduces alias can be used in a LINEA/B/C/D or RLINEA/B/C/D command.

For LINEA/B/C/D commands and RLINEA/B/C/D commands, antialiasing is performed by setting the rendering attribute AA (antialias enable) bit to 1.

Notes: 1. For a dashed line in LINEA/B or RLINEA/B command, antialiasing is not performed for the gaps in the dashed line.
2. When the starting and final coordinate points of a line segment match in the LINEA, LINEB, LINEC, RLINEA, RLINEB, or RLINEC command, a single dot is drawn for a 1-bit-wide line (W = 0) without antialiasing and nothing is drawn for bold line drawing.
3. When the starting and final coordinate points of a line segment match in the LINED or RLINED command, nothing is drawn.
4. Antialiasing is not performed for horizontal, vertical, and 45-degree diagonal line segments in the LINEA, LINEB, LINEC, RLINEA, RLINEB, or RLINEC command.
5. Antialiasing is not performed for horizontal and vertical line segments in the LINED or RLINED command.

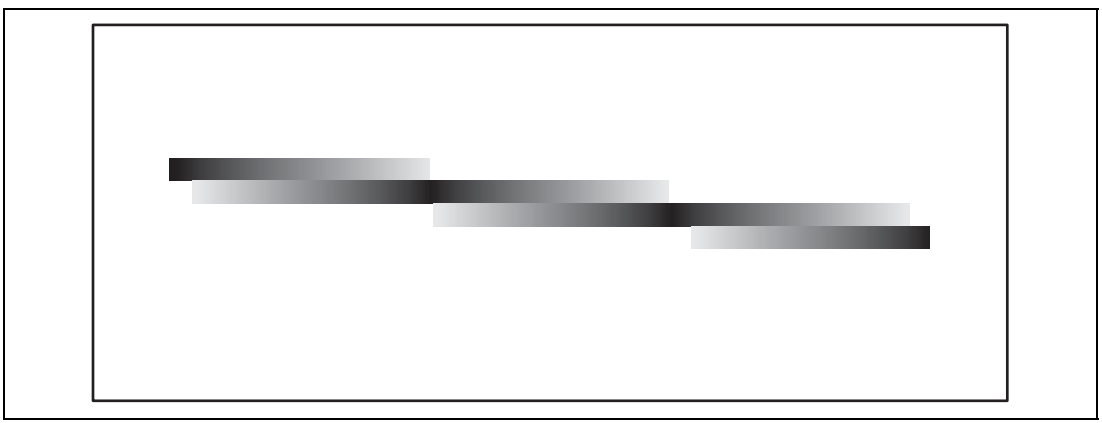


Figure 11.3 Example of Antialias Specification

(3) Coordinate Systems

The 2DG has four 2-dimensional coordinate systems (screen coordinates, rendering coordinates, 2-dimensional source coordinates and work coordinates), and one 1-dimensional coordinate system (1-dimensional source coordinates).

Screen coordinates are the display control coordinates. Screen coordinate X corresponds to the horizontal dimension of the display screen and Y to the vertical dimension. The origin is the top-left corner in the display screen. The screen coordinate positive directions are right for the X-axis and down for the Y-axis. Either 16 bits (16 bits/pixel) or 8 bits (8 bits/pixel) can be selected as the data width of one screen coordinate. The maximum values of the screen coordinates are X = 4095, Y = 4095.

Rendering coordinates are drawing control coordinates. Rendering coordinates are shifted horizontally and vertically with respect to screen coordinates by the offset amounts specified in drawing commands. According to the drawing commands, the 2DG performs drawing operations using these coordinates. Either 16 bits (16 bits/pixel) or 8 bits (8 bits/pixel) can be selected as the data width of one rendering coordinate.

2-dimensional source coordinates are drawing control coordinates. When a drawing command is executed with SS = 1, these are the source data (rectangle) coordinates specified by the drawing command. Either 16 bits (16 bits/pixel) or 8 bits (8 bits/pixel) can be selected as the data width of one 2-dimensional source coordinate.

1-dimensional source coordinates are drawing control coordinates. When a drawing command is executed with SS = 0, these are the source data (1-dimensional) coordinates specified by the drawing command. 1 bit (1 bit/pixel), 16 bits (16 bits/pixel), or 8 bits (8 bits/pixel) can be selected as the data width of one 1-dimensional source coordinate. For one 1-dimensional source, one physical address (top-left) and the horizontal width and vertical height of the 1-dimensional source are specified.

Work coordinates are drawing control coordinates that correspond one-to-one with the rendering coordinates. When a drawing command is executed, these are the work coordinates specified by the drawing command. The data width of one work coordinate is 1 bit.

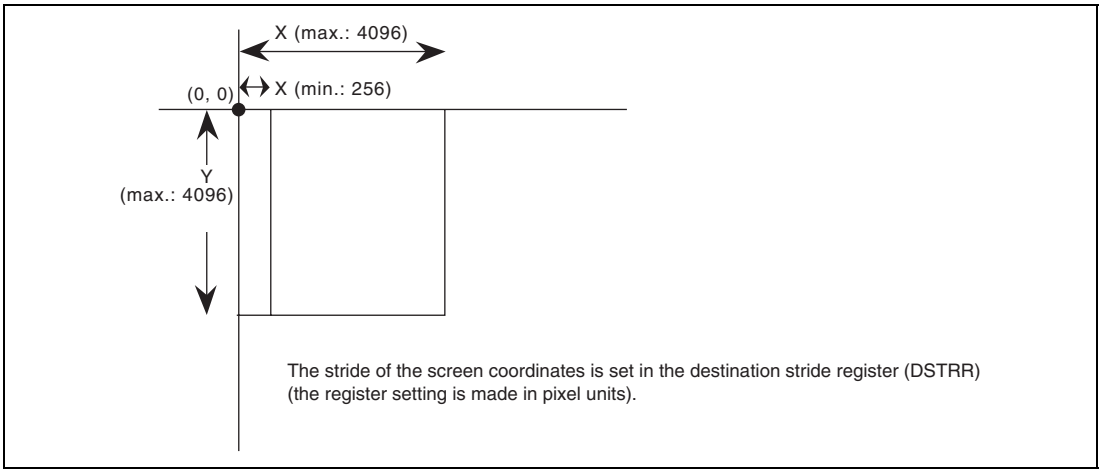


Figure 11.4 Screen Coordinates

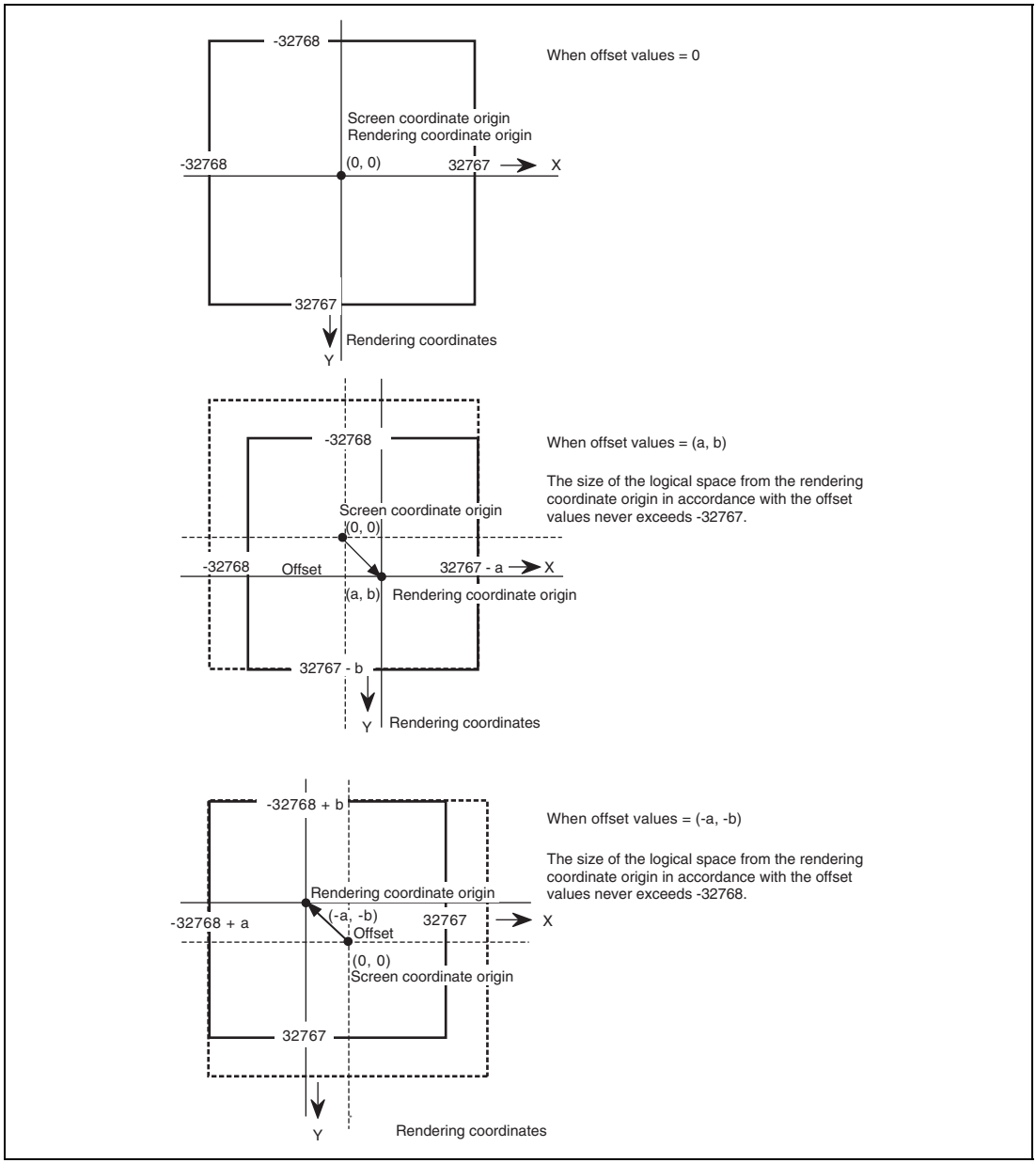


Figure 11.5 Rendering Coordinates

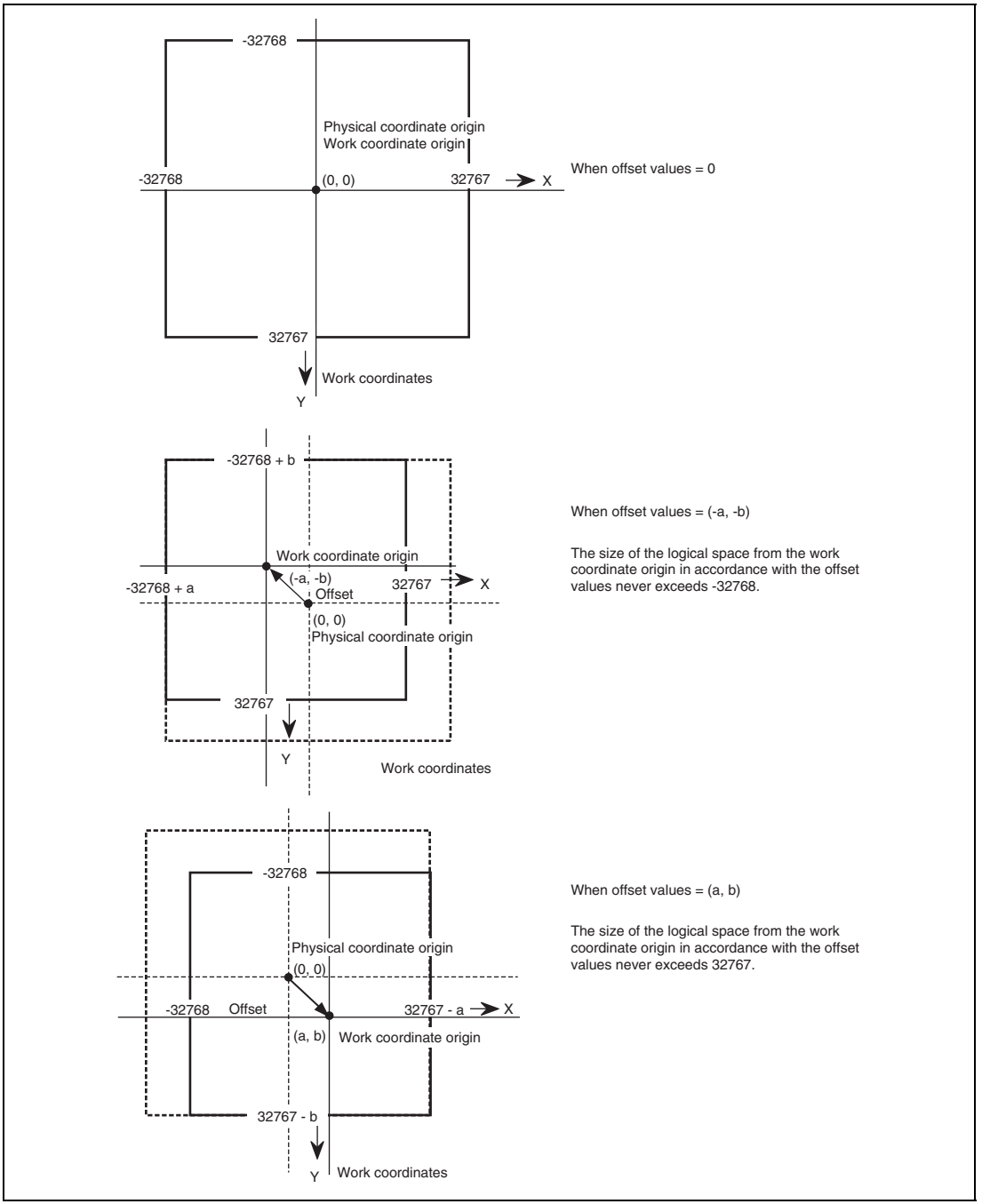


Figure 11.6 Work Coordinates

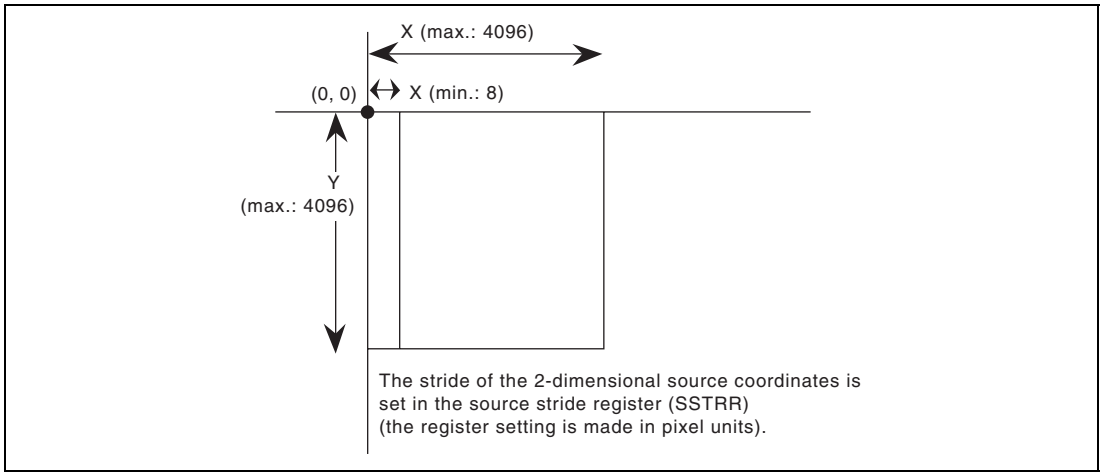


Figure 11.7 2-Dimensional Source Coordinates (SS = 1)

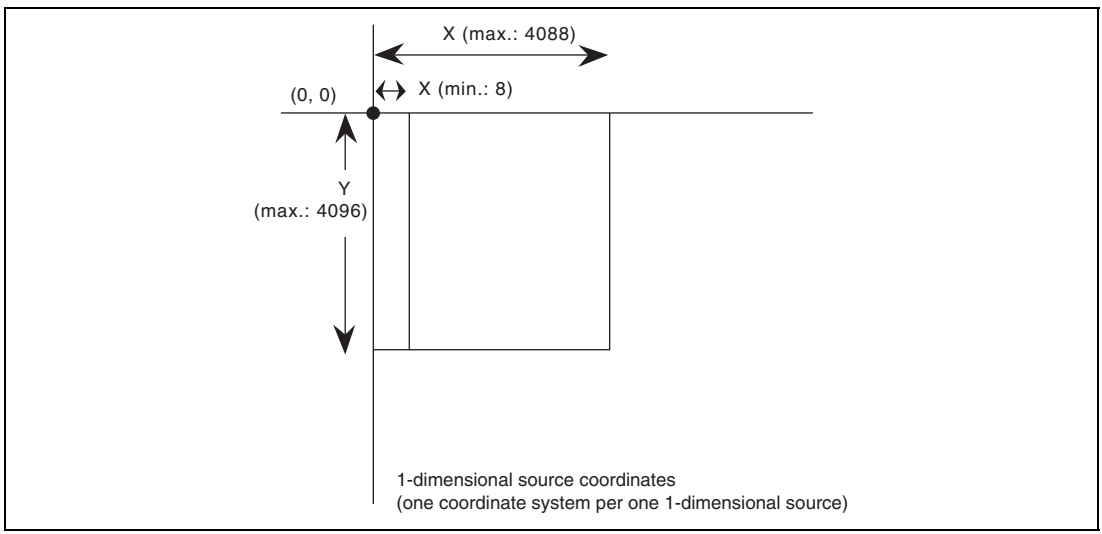
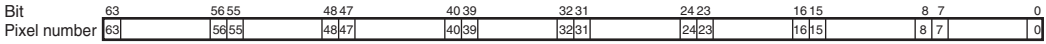


Figure 11.8 1-Dimensional Source Coordinates (SS = 0)

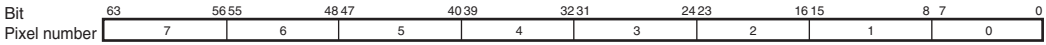
11.3.2 Data Formats

• 1-bit/pixel data



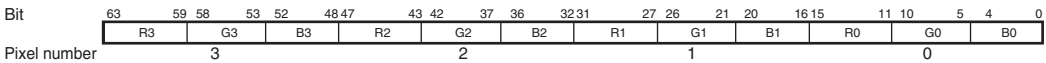
The pixel number is 0 at the left side of the screen, and increments as it shifts right.

• 8-bit/pixel data



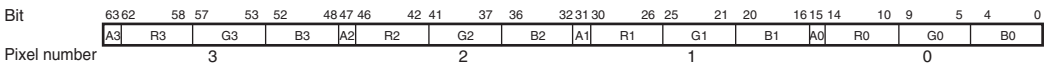
The pixel number is 0 at the left side of the screen, and increments as it shifts right.

• 16-bit/pixel data (RGB)



The pixel number is 0 at the left side of the screen, and increments as it shifts right.

• 16-bit/pixel data (ARGB)



The pixel number is 0 at the left side of the screen, and increments as it shifts right.

• 32-bit data (display list)

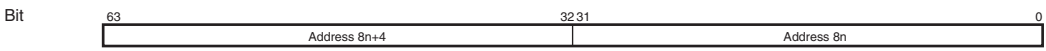


Figure 11.9 Data Formats

11.3.3 Endian Conversion

The endian conversion control register (ENDCVR) specifies an endian conversion method.

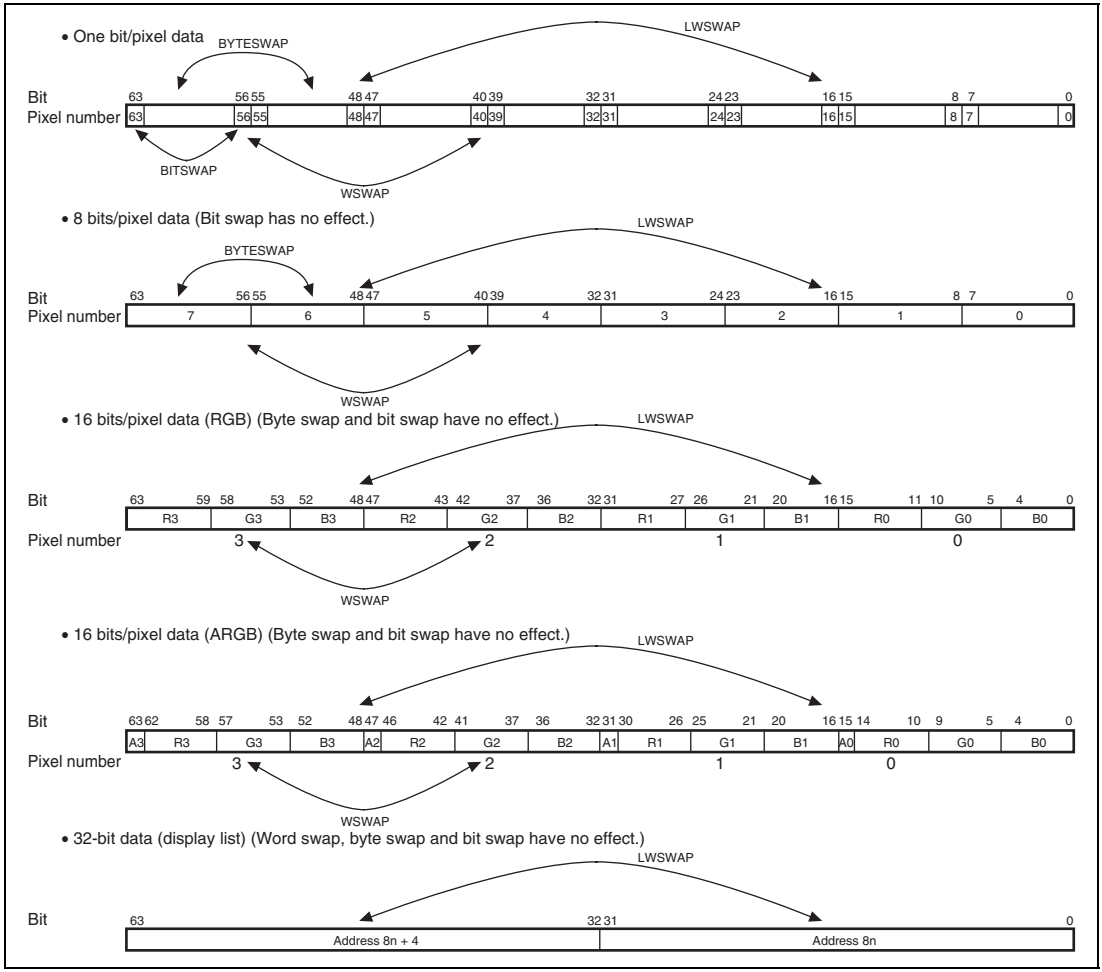


Figure 11.10 Endian Conversion

11.3.4 Rendering Attributes

(1) Source Transparency Specification (STRANS)

When referencing source data, the STRANS bit can be used to select transparency or non-transparency on an individual drawing command basis. If transparency is selected, the source color becomes transparent at register value = source color when the source transparent color polarity bit (STP) in the rendering control register (RCLR) is 0, and the source color becomes transparent at register value ≠ source color when the STP bit is 1, and the pixels are not drawn in either case.

The source transparency specification can be used with the POLYGON4A, POLYGON4B, LINEA, LINEB, RLINEA, RLINEB, BITBLTA, and BITBLTB commands. The STRANS bit should be cleared to 0 in other commands. When the source pixel format is ARGB, the A value is not compared. Note that when the STRANS bit is set to 1, the source data is always read in the BITBLTA or BITBLTB command, regardless of the ROP code.

(2) Destination Transparency Specification (DTRANS)

When referencing destination data, the DTRANS bit can be used to select transparency or non-transparency on an individual drawing command basis. If transparency is selected, the destination color becomes transparent at register value = destination color when the destination transparent color polarity bit (DTP) in the rendering control register (RCLR) is 0, and the destination color becomes transparent at register value ≠ destination color when the DTP bit is 1, and the pixels are not drawn in either case.

The destination transparency specification can be used with the BITBLTA, BITBLTB, and BITBLTC commands. The DTRANS bit should be cleared to 0 in other commands. When the destination pixel format is ARGB, the A value is not compared. Note that when the DTRANS bit is set to 1, the destination data is always read, regardless of the ROP code.

(3) Source Style Specification (STYLE)

The STYLE bit can be used to select, on an individual drawing command basis, whether to enlarge or reduce the source data or repeatedly reference it. If no style specification is made, the source data is enlarged or reduced in proportion to the size of the rendering area. When a style specification is made, the source data is referenced repeatedly in proportion to the size of the rendering area. This attribute is therefore used when drawing repeated patterns such as hatch patterns.

The source style specification can be used with the POLYGON4A, POLYGON4B, LINEA, LINEB, RLINEA, and RLINEB commands. The STYLE bit should be cleared to 0 in other

commands. The STYLE bit must be set to 1 when BLKE = 1 in the POLYGON4A, POLYGON4B, LINEA, LINEB, RLINEA, or RLINEB command.

In the LINEA, LINEB, RLINEA, and RLINEB commands, the source data is repeatedly referenced in only the X direction of the source data. The source data is enlarged or reduced in proportion to the line width in the Y direction of the source data.

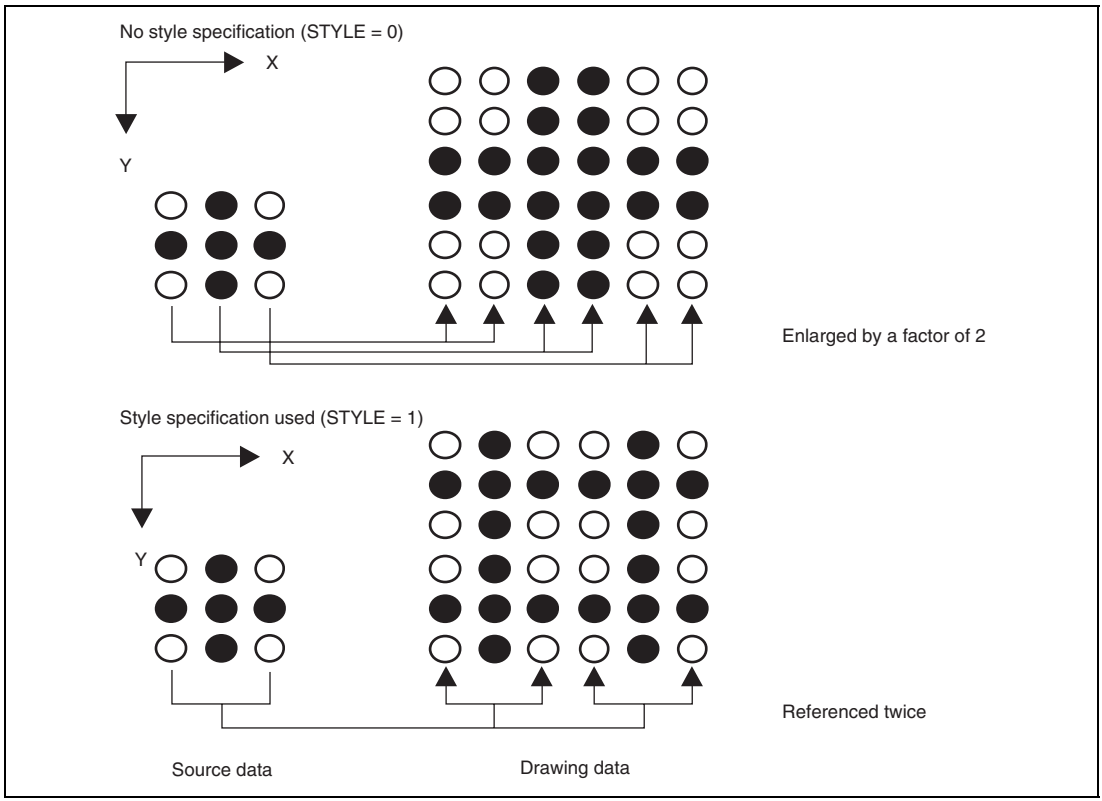


Figure 11.11 Example of Source Style Specification

(4) Clipping Specification (CLIP)

The 2DG can perform clipping area management. There are three kinds of clipping areas: system clipping area, user clipping area, and relative user clipping area.

The system clipping area has a fixed drawing range. The system clipping area is always valid, regardless of attribute specifications.

A user clipping area can be designated as desired within the system clipping area. Whether or not clipping is performed in that area can be selected on an individual command basis with the rendering attribute CLIP bit. The boundary is drawn. The local offset values specified by the LCOFS or RLCOFS command are not added.

When setting a user clipping area, the following ranges must be satisfied: XMIN < XMAX, YMIN < YMAX.

Clipping is set with screen coordinates. Since the clipping area is undefined after the power is turned on, set the clipping area by the WPR command at the top of the display list that is executed first. XMAX must be set to a value less than the value set in the destination stride register (DSTRR).

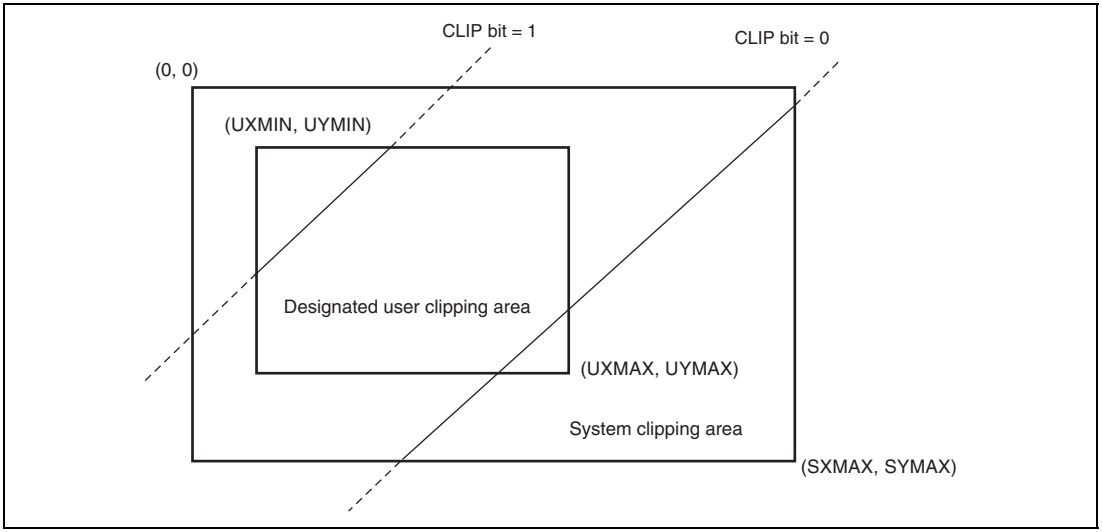


Figure 11.12 Example of Clipping Specification

(5) Relative Clipping Specification (RCLIP)

The 2DG can perform clipping area management. There are three kinds of clipping areas: system clipping area, user clipping area, and relative user clipping area.

The system clipping area has a fixed drawing range. The system clipping area is always valid, regardless of attribute specifications.

A relative user clipping area can be designated as desired within the system clipping area at a relative setting with respect to the local offset. Whether or not clipping is performed in that area can be selected on an individual command basis with the rendering attribute RCLIP bit. The boundary is drawn. The local offset values specified by the LCOFS or RLCOFS command are added.

When setting a relative user clipping area, the following ranges must be satisfied: XMIN < XMAX, YMIN < YMAX.

Clipping is set with screen coordinates. Since the clipping area is undefined after the power is turned on, set the clipping area by the WPR command at the top of the display list that is executed first. XMAX must be set to a value less than the value set in the destination stride register (DSTRR). If both the RCLIP and CLIP bits are set to 1 simultaneously, the region where the two clipping areas overlap is drawn.

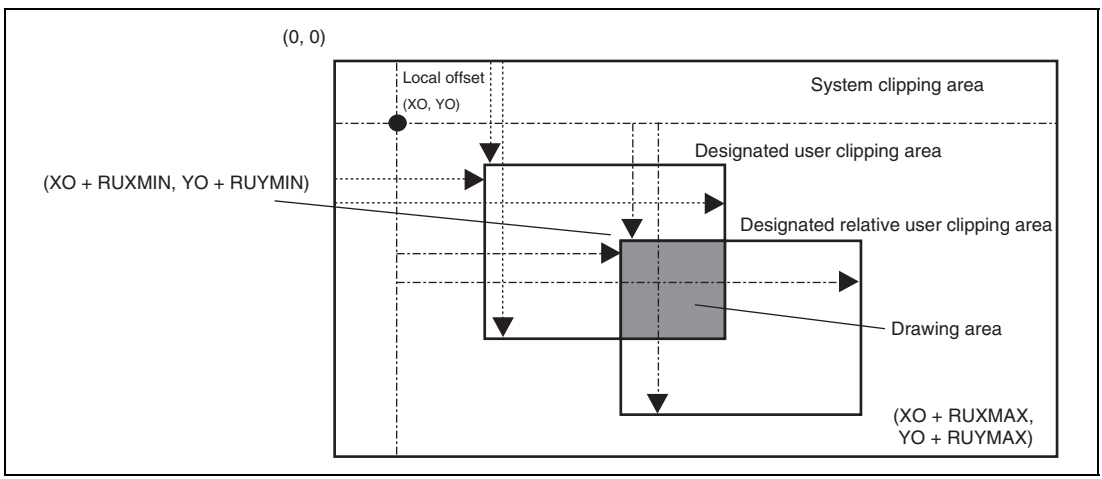


Figure 11.13 Example of Relative User Clipping Specification

When a relative user clipping area ((XO + RUXMIN, YO + RUYMIN) - (XO + RUXMAX, YO + RUYMAX)) intersects with the system clipping area, saturation processing is performed as follows:

XO + RUXMIN < 0 → XO + RUXMIN = 0

XO + RUXMAX > SXMAX → XO + RUXMAX = SXMAX

YO + RUYMIN < 0 → YO + RUYMIN = 0

YO + RUYMAX > SYMAX → YO + RUYMAX = SYMAX

Note: Set the local offset values and relative user clipping area without exceeding the following ranges:

-4096 ≤ XO + RUXMIN ≤ 4095

-4096 ≤ YO + RUYMIN ≤ 4095

0 ≤ XO + RUXMAX ≤ 8191

0 ≤ YO + RUYMAX ≤ 8191

When RCLIP = 1 and the relative user clipping area satisfies one of the following conditions, the relative user clipping area is disabled internally by the 2DG (same operation as RCLIP = 0).

4095 < XO + RUXMIN

4095 < YO + RUYMIN

XO + RUXMAX < 0

YO + RUYMAX < 0

(6) Net Drawing Specification (NET)

The NET bit can be used to select, on an individual drawing command basis, whether or not net drawing is to be performed. Net drawing is a function for drawing only pixels at coordinates for which the condition "rendering coordinates X + Y = EOS (0: even number, 1: odd number)" is true.

For example, if EOS = 0, drawing is only performed on the pixels at coordinates Y = 0, X = 0, 2, 4, 6, 8… and Y = 1, X = 1, 3, 5, 7, 9….

This function enables the drawn figure and ground to be mutually semi-composed.

The net drawing specification can be used with the POLYGON4 type, LINEA, LINEB, LINEC, RLINEA, RLINEB, and RLINEC commands. The NET bit should be cleared to 0 in other commands. The NET bit cannot be used together with the antialias enable bit (AA).

(7) Even/Odd Select Specification (EOS)

Even pixels are selected when EOS = 0, and odd pixels when EOS = 1.

The even/odd select specification is used together with the net drawing specification (NET). With the LINEWC and RLINEWC commands, drawing is performed at the work coordinates with 0 when EOS = 0, and with 1 when EOS = 1.

(8) Work Specification (WORK)

When drawing is performed at rendering coordinates with the POLYGON4 type or BITBLT type command, the WORK bit can be used to select, on an individual drawing command basis, whether or not binary work data is to be referenced.

When binary work data referencing is selected, drawing is performed if the work data for the pixel corresponding to the rendering coordinates is 1, but not if the work data is 0. The same shape as that drawn at work coordinates can thus be drawn at rendering coordinates.

Drawing at work coordinates can be performed either by means of the FTRAPC, RFTRAPC, LINEWC, RLINEWC, or CLRWC command or else by the CPU. The work specification can be used with the POLYGON4 type and BITBLT type commands. The WORK bit should be cleared to 0 in the other commands.

(9) Source Address Specification (SS)

The SS bit is used to select whether the source is to be referenced at a 2-dimensional source area address or at the address indicated by the Base Address parameter in the display list. The source address specification can be used with the POLYGON4A, POLYGON4B, BITBLTA, and BITBLTB commands. The command which is used to disable the SS bit should be set up at 0.

If the offset values are set, the source is referenced from (TXOFS, TYOFS).

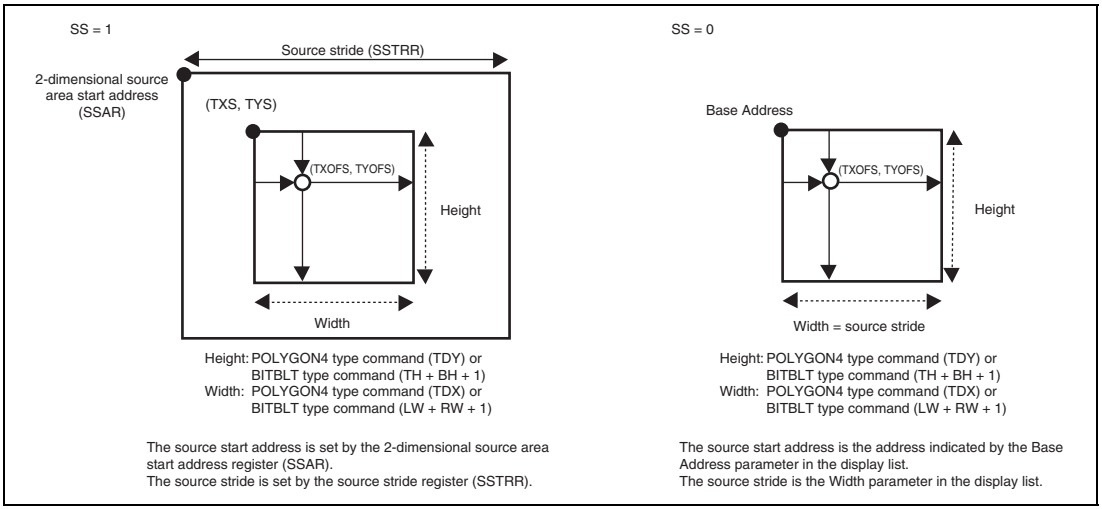


Figure 11.14 Example of Source Address Specification

Note: When SS = 1, settings must be made within the ranges of $0 \leq TXS \leq SSTRR - Width$ (TDX, LW + RW + 1), $0 \leq TYS \leq 4096 - Height$ (TDY, TH + BH + 1).

(10) Source Coordinate Relative Address Specification (REL)

Setting the REL bit to 1 in the POLYGON4A, POLYGON4B, BITBLTA, BITBLTB, LINEA, LINEB, RLINEA, RLINEB, JUMP, and GOSUB commands enables source referencing and branching to be performed at an address relative to (before or after) the command code.

Clear the SS bit to 0 in the POLYGON4A or BITBLTA command; correct operation is not guaranteed when the SS bit is set to 1.

The command code address is the origin of the relative address (longword address).

Note: With the POLYGON4A, POLYGON4B, BITBLTA, BITBLTB, LINEA, LINEB, RLINEA, and RLINEB commands, adding the address (longword: 32-bit units) where the command code is located to the source start relative address (longword: 32-bit units) must result in a quad word address (64-bit units).

(11) Edge Drawing (EDG)

With the FTRAP and RFTRAP commands, setting the EDG bit to 1 enables edge lines to be drawn after completion of trapezoid painting to the work area. Whether edge line drawing is performed with 0 or with 1 is specified by the EOS bit.

(12) Color Offset (COOF)

The color offset specification can be used with the POLYGON4 type, LINEA, LINEB, LINEC, RLINEA, RLINEB, RLINEC and BITBLT type commands. In 16-bit/pixel drawing, if the rendering attribute COOF bit is set to 1, the result of adding the value in COFSR to the value of the source data (color expanded data for a binary source and the specified color for the monochrome specification) is drawn. In 8-bit/pixel drawing, the COOF bit should be cleared to 0. When the source pixel format is ARGB, the A value is not used in operation.

(13) Source Direction X, Y (SRCDIRX, SRCDIRY)

The source direction X, Y specification can be used with the BITBLTA and BITBLTB commands. The directions in which to scan the source data are selected.

(TXS, TYS) or the Base Address specifies the top-left corner of the rectangle source, regardless of the source scan directions.

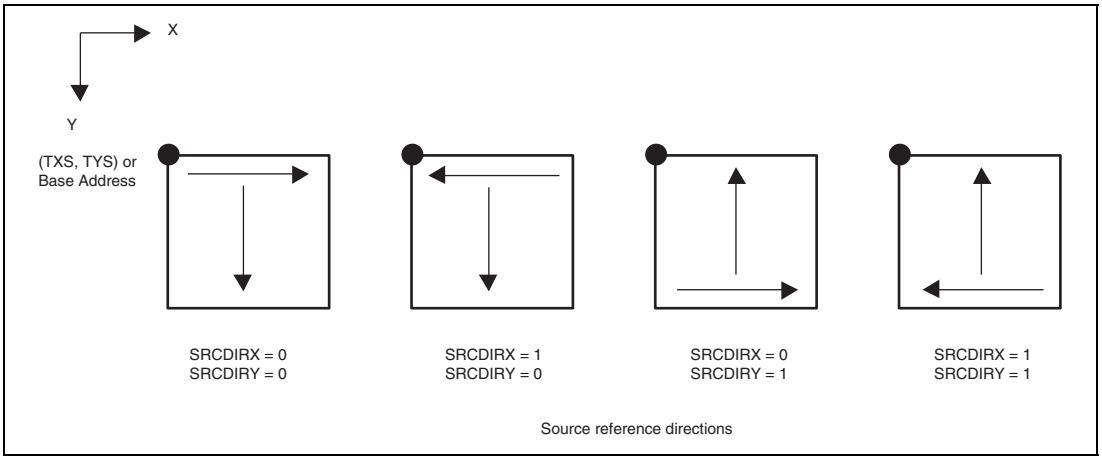


Figure 11.15 Example of Source Direction Specification

(14) Destination Direction X, Y (DSTDIRX, DSTDIRY)

The destination direction X, Y specification can be used with the BITBLTA, BITBLTB, and BITBLTC commands. The directions in which to draw the destination data are selected.

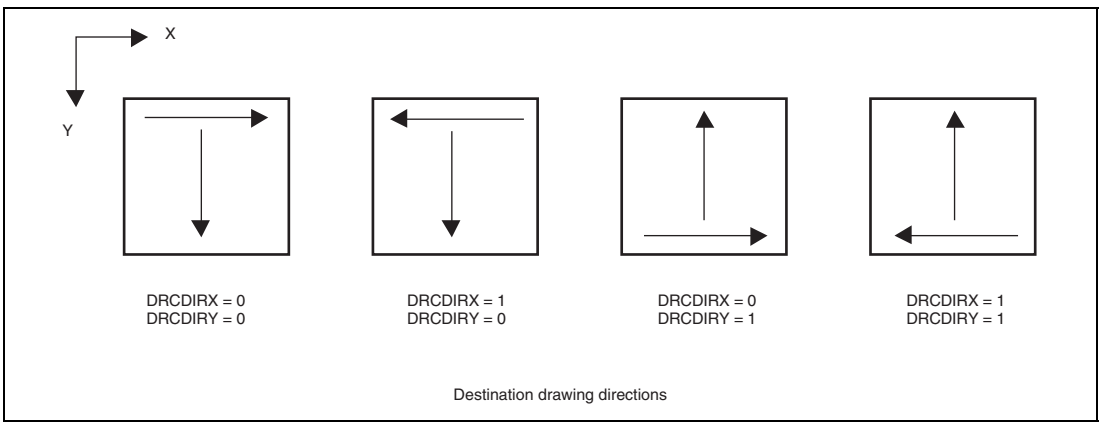


Figure 11.16 Example of Destination Direction Specification

(15) Antialias Enable (AA)

The antialias enable specification can be used with the LINE type and RLINE type commands to reduce alias.

The antialias enable specification is enabled only in 16-bit/pixel drawing. For 8-bit/pixel drawing, the AA bit should be cleared to 0. The AA bit should be set to 1 with the LINED and RLINED commands. The antialias enable specification cannot be used together with the net drawing specification (NET).

(16) Alpha Blend Enable (αE)

The alpha blend enable specification can be used with the POLYGON4 type and BITBLT type commands.

The source data (color expanded data for a binary source and the specified color for the monochrome specification) and ground data are alpha blended and drawn. The alpha value is set in the alpha value register (ALPHR). The alpha blend enable specification is enabled only in 16-bit/pixel drawing. For 8-bit/pixel drawing, the αE bit should be cleared to 0.

In the POLYGON4 type commands, the alpha blend enable specification is enabled only when BLKE = 1. The αE bit should be cleared to 0 when BLKE = 0. In the BITBLT type commands, the alpha blend enable specification is enabled only when the ROP code is H'CC (source copy). For other ROP codes, the αE bit should be cleared to 0.

The A value in the ARGB format is not alpha blended. The A value is drawn according to the source A value use (SAU) and A value (AVALUE) bits in the rendering control register (RCLR).

(17) Source Alpha Enable (SαE)

The source alpha enable specification can be used with the POLYGON4A and BITBLTA commands. The SαE bit is used together with the alpha blend enable bit (αE).

When αE = 0, the SαE bit should be cleared to 0. When the source pixel format bit (SPF) is 1 (ARGB format), only the pixels whose source A value is 1 are alpha blended. Pixels whose source A value is 0 are not alpha blended and the source data is drawn as it is.

The source alpha enable specification is enabled only when SPF = 1. The SαE bit should be cleared to 0 when SPF = 0.

(18) Block Enable (BLKE)

The block enable specification can be used with the POLYGON4 type commands.

When BLKE = 1, the input vertex coordinates (DXn, DYn) are internally transformed to circumscribed rectangle coordinates (DX'n, DY'n) and four-vertex drawing performed. When BLKE = 1, the fixed drawing direction is from the upper-left corner to the lower-right corner (up-and-down and right-and-left directions cannot be reversed).

The BLKE bit should be set to 1 with the CLRWC, FTRAPC, and RFTRAPC commands.

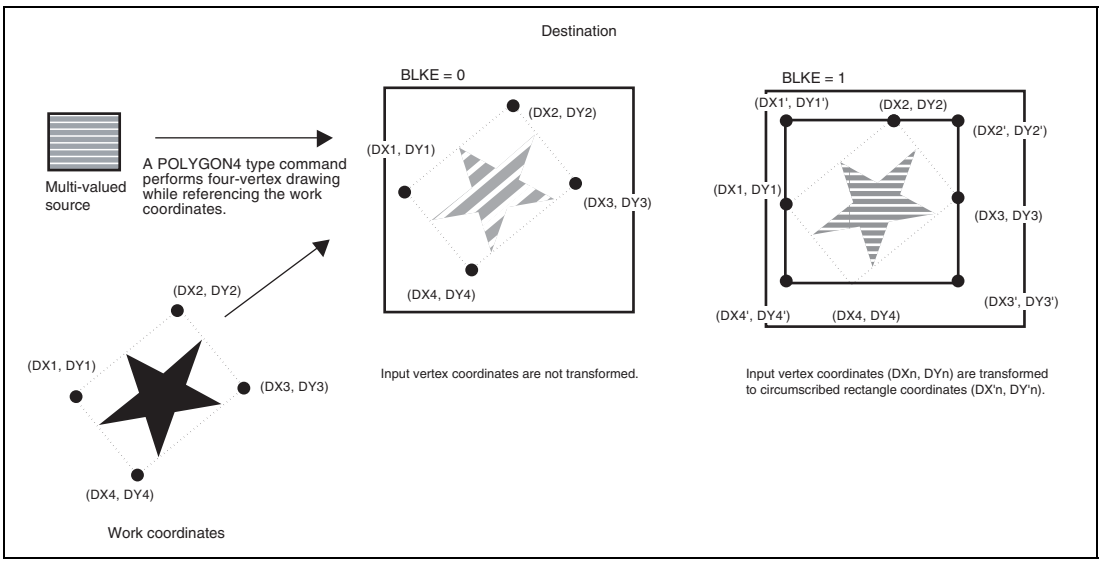


Figure 11.17 Example of Block Enable Specification

(19) Link Specification Enable (LINKE)

The link specification enable specification can be used with the LINEC, LINED, RLINEC, RLINED, FTRAPC, RFTRAPC, and WPR commands.

From the memory address specified by the LINK Address, the vertex coordinates are read with the LINEC, LINED, RLINEC, RLINED, FTRAPC, and RFTRAPC commands, and the register write data is read with the WPR command.

The longword address following the LINK Address is handled as the next command code. Therefore, do not specify the longword address following the address where the LINK Address is to be assigned as the link destination address specified by the LINK Address.

(20) Link Address Relative Specification (LREL)

The link address relative specification can be used with the LINEC, LINED, RLINEC, RLINED, FTRAPC, RFTRAPC, and WPR commands. The LREL bit is used together with the link specification enable bit (LINKE).

The LREL bit should be cleared to 0 when LINKE = 0. The link destination address is specified as a relative address. The command code address is the origin of the relative address.

The longword address following the LINK Address is handled as the next command code. Therefore, do not specify the longword address following the address where the LINK Address is to be assigned as the link destination address specified by the LINK Address.

(21) Clockwise (CLKW)

The clockwise specification can be used with the LINED and RLINED commands.

The CLKW bit is used to specify whether the order in giving the n vertices is clockwise or counterclockwise. The order is clockwise when CLKW = 1 and counterclockwise when CLKW = 0.

(22) Raster Operation (ROP)

The raster operation specification can be used with the BITBLT type commands. The ROP code is specified in the ROP field, which is a BITBLT command parameter.

Table 11.3 Raster Operation Codes

ROP Code	Operation
H'00	0
H'11	~(S | D)
H'22	~S & D
H'33	~S
H'44	S & ~D
H'55	~D
H'66	S ^ D
H'77	~(S & D)
H'88	S & D
H'99	~(S ^ D)
H'AA	D
H'BB	~S | D
H'CC	S
H'DD	S | ~D
H'EE	S | D
H'FF	1

Set the ROP code to H'CC when alpha blending is enabled (αE = 1). Neither alpha blending nor raster operation is performed for the A value in the ARGB format. The A value is drawn according to the source A value use (SAU) and A value (AVALUE) bits in the rendering control register (RCLR).

11.3.5 Structure of Cache Memories in 2DG

The 2DG incorporates four types of cache memories: command, source (texture), work, and destination cache memories. These cache memories are used to allow the data in the DDR memory to be temporarily held by the 2DG; the 2DG performs rendering with the data.

The four types of cache memories are described below.

- Command cache (64 bytes)

  Command cache is used to allow the display list in the DDR memory to be held by the 2DG. Command cache is cleared by the rendering start command.

  However, it is not cleared when rendering operation is resumed in the halt state having been caused by the NOP command (INT = 1).
- Source (texture) cache (8 kbytes)

  Source (texture) cache is used to allow the source (texture) data in the DDR memory to be held by the 2DG. Source (texture) cache is cleared by the TRAP command or NOP command (INT = 1).
- Work cache (1 kbyte)

  Work cache is used to allow the 2DG to perform rendering on or make work-references to the work coordinates in the DDR memory. Work cache is flushed by the TRAP command or NOP command (INT = 1).
- Destination cache (512 bytes)

  Destination cache is used to allow the 2DG to perform rendering on the rendering coordinates in the DDR memory. Destination cache is flushed by the TRAP command or NOP command (INT = 1).

If the above cache memories are not updated, that is, if the same address is referenced with the access size smaller than each cache size, or if the reference starts at the previous reference-start address and ends at the address smaller than each cache size, the previous cache contents are used even after the DDR memory data is updated. Accordingly, when there is an overlap between different cache areas, data coherency might not be maintained.

To avoid this, each cache needs to be updated intentionally. Specifically, the following should be performed.

Access the address such that exceeds each cache size. The contents of all the cache areas except command cache can be updated using the SYNC command.

11.4 Display List

11.4.1 4-Vertex Screen Drawing Commands

(1) POLYGON4A

(a) Function

Performs any four-vertex drawing in the destination area while referencing a multi-valued (8- or 16-bit/pixel) source.

(b) Command Format

SS = 1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1000_0010								Reserve (all 0)								Draw Mode															
0	0	0	0	TXS (0 ≤ TXS ≤ 4088)												0	0	0	0	TYS (0 ≤ TYS ≤ 4095)											
0	0	0	0	TDX (8 ≤ TDX ≤ 4095)												0	0	0	0	TDY (1 ≤ TDY ≤ 4095)											
0	0	0	0	TXOFS (0 ≤ TXOFS ≤ TDX – 1)												0	0	0	0	TYOFS (0 ≤ TYOFS ≤ TDY – 1)											
Sign	DX1 (-32768 ≤ DX1 ≤ 32767)															Sign	DY1 (-32768 ≤ DY1 ≤ 32767)														
Sign	DX2 (-32768 ≤ DX2 ≤ 32767)															Sign	DY2 (-32768 ≤ DY2 ≤ 32767)														
Sign	DX3 (-32768 ≤ DX3 ≤ 32767)															Sign	DY3 (-32768 ≤ DY3 ≤ 32767)														
Sign	DX4 (-32768 ≤ DX4 ≤ 32767)															Sign	DY4 (-32768 ≤ DY4 ≤ 32767)														

Note: 0 ≤ TXS ≤ SSTRR – TDX, 0 ≤ TYS ≤ 4096 – TDY (SSTRR: Source stride register setting)

SS = 0 and REL = 0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1000_0010								Reserve (all 0)								Draw Mode															
0	0	0	Base Address (quad word address)																										0	0	0
0	0	0	0	TDX (8 ≤ TDX ≤ 4088)									0	0	0	0	0	0	0	TDY (1 ≤ TDY ≤ 4095)											
0	0	0	0	TXOFS (0 ≤ TXOFS ≤ TDX – 1)												0	0	0	0	TYOFS (0 ≤ TYOFS ≤ TDY – 1)											
Sign	DX1 (-32768 ≤ DX1 ≤ 32767)															Sign	DY1 (-32768 ≤ DY1 ≤ 32767)														
Sign	DX2 (-32768 ≤ DX2 ≤ 32767)															Sign	DY2 (-32768 ≤ DY2 ≤ 32767)														
Sign	DX3 (-32768 ≤ DX3 ≤ 32767)															Sign	DY3 (-32768 ≤ DY3 ≤ 32767)														
Sign	DX4 (-32768 ≤ DX4 ≤ 32767)															Sign	DY4 (-32768 ≤ DY4 ≤ 32767)														

SS = 0 and REL = 1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1000_0010								Reserve (all 0)								Draw Mode															
Sign extended	Sign	Base Address (longword address)																											0	0	
0	0	0	0	TDX (8 ≤ TDX ≤ 4088)									0	0	0	0	0	0	0	TDY (1 ≤ TDY ≤ 4095)											
0	0	0	0	TXOFS (0 ≤ TXOFS ≤ TDX – 1)												0	0	0	0	TYOFS (0 ≤ TYOFS ≤ TDY – 1)											
Sign	DX1 (-32768 ≤ DX1 ≤ 32767)															Sign	DY1 (-32768 ≤ DY1 ≤ 32767)														
Sign	DX2 (-32768 ≤ DX2 ≤ 32767)															Sign	DY2 (-32768 ≤ DY2 ≤ 32767)														
Sign	DX3 (-32768 ≤ DX3 ≤ 32767)															Sign	DY3 (-32768 ≤ DY3 ≤ 32767)														
Sign	DX4 (-32768 ≤ DX4 ≤ 32767)															Sign	DY4 (-32768 ≤ DY4 ≤ 32767)														

Note: Adding the address (longword: 32-bit units) where the command code is located to the Base Address (longword: 32-bit units) must result in a quad word address (64-bit units).

(c) Code

B'10000010

(d) Rendering Attributes

Reference Data				Drawing Destination	
Multi-Valued Source	Binary Source	Binary Work	Specified Color	Rendering	Work
O		O (only WORK = 1)		O	

Draw Mode

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Fixed to 0	Fixed to 0	CLIP	RCLIP	STRANS	Fixed to 0	WORK	SS	REL	STYLE	BLKE	NET	EOS	COOF	αE	SαE

(e) Command Parameters

TXS, TYS: Source starting point. Write 0 to the unused bits.

Base Address: Source start absolute address (Quad word address. Write 0 to bits A31 to A29 and A2 to A0.)

Source start relative address (Longword address. Negative number expressed as two's complement. Bits A31 to A29 are used to extend the sign in bit A28. Write 0 to bits A1 and A0.)

Note: Even in 32-bit addressing mode, write the values in bits 28 to 3 of the specified 32-bit address to bits A28 to A3.

TDX, TDY: Source size. Write 0 to the unused bits.

DXn, DYn (n = 1 to 4): Rendering coordinates (absolute coordinates). Negative numbers expressed as two's complement.

TXOFS, TYOFS: Source offset. Write 0 to the unused bits.

(f) Description

Transfers multi-valued (8- or 16-bit/pixel) source data to any quadrilateral rendering coordinates. The source data is always scanned horizontally, but diagonal scanning may be used in the drawing, depending on the shape. In diagonally-scanned drawing, double-writing occurs to fill in gaps.

When SS = 0, set a multiple of 8 pixels as the TDX value. When SS = 1, set 8 or more pixels as the TDX value. If the TDX setting is less than 8 pixels, multi-valued source references will not be

performed normally. If TXOFS or TYOFS is set, the source at a location shifted by the offset amount is referenced. Make the TXOFS and TYOFS settings in pixel units.

1. When source style specification is selected as a rendering attribute (STYLE = 1), the source data is not enlarged or reduced, but is referenced repeatedly.
2. When work specification is selected as a rendering attribute (WORK = 1), only places where the work coordinate pixel is 1 are drawn at rendering coordinates while referencing work coordinates for the same coordinates as the rendering coordinates.
3. When SS = 1, the source data is referenced from the 2-dimensional source area. When SS = 0, the source data is referenced from the Base Address in the display list. When REL = 0, the source address can be specified as an absolute address. When REL = 1, the source address can be specified as a relative address with respect to the memory address at which the POLYGON4A command code is located.
4. In 16-bit/pixel drawing, if the rendering attribute COOF bit is set to 1, the result of adding the value in COFSR to the value of the multi-valued source data is drawn. The operation is performed by saturation processing. In 8-bit/pixel drawing, the COOF bit should be cleared to 0.

(g) Example

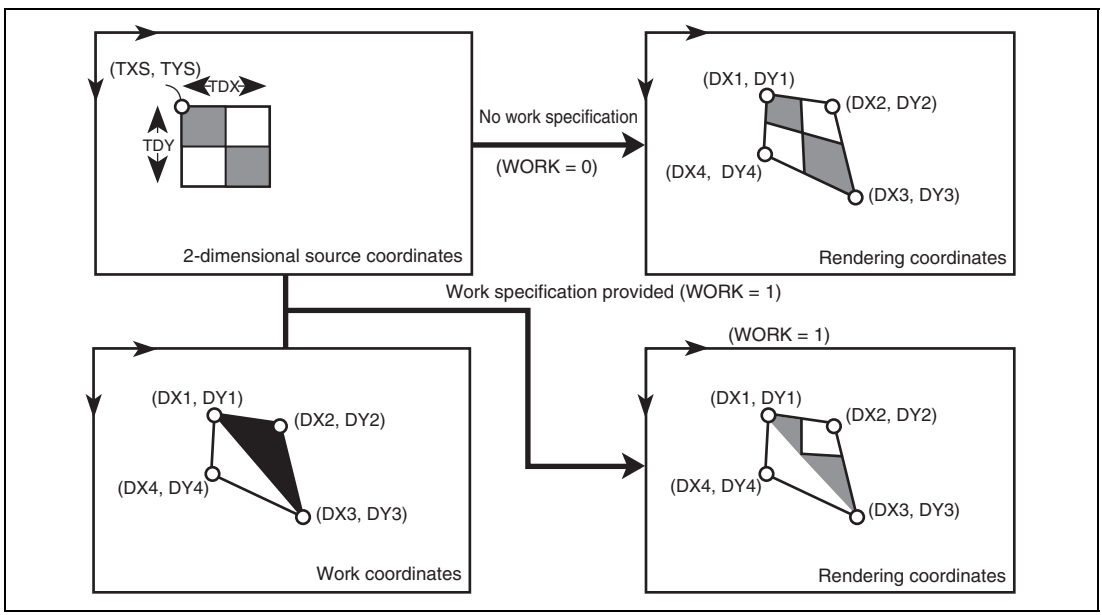


Figure 11.18 POLYGON4A Command Example

(2) POLYGON4B

(a) Function

Performs any four-vertex drawing in the destination area while referencing a binary (1-bit/pixel) source.

(b) Command Format

SS = 1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1000_0001								Reserve (all 0)								Draw Mode															
Color1																Color0															
0	0	0	0	TXS (0 ≤ TXS ≤ 4088)												0	0	0	0	TYS (0 ≤ TYS ≤ 4095)											
0	0	0	0	TDX (8 ≤ TDX ≤ 4088)									0	0	0	0	0	0	0	TDY (1 ≤ TDY ≤ 4095)											
0	0	0	0	TXOFS (0 ≤ TXOFS ≤ TDX – 1)												0	0	0	0	TYOFS (0 ≤ TYOFS ≤ TDY – 1)											
Sign	DX1 (-32768 ≤ DX1 ≤ 32767)															Sign	DY1 (-32768 ≤ DY1 ≤ 32767)														
Sign	DX2 (-32768 ≤ DX2 ≤ 32767)															Sign	DY2 (-32768 ≤ DY2 ≤ 32767)														
Sign	DX3 (-32768 ≤ DX3 ≤ 32767)															Sign	DY3 (-32768 ≤ DY3 ≤ 32767)														
Sign	DX4 (-32768 ≤ DX4 ≤ 32767)															Sign	DY4 (-32768 ≤ DY4 ≤ 32767)														

Note: 0 ≤ TXS ≤ SSTRR – TDX, 0 ≤ TYS ≤ 4096 – TDY (SSTRR: Source stride register setting)

SS = 0 and REL = 0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1000_0001								Reserve (all 0)								Draw Mode															
Color1																Color0															
0	0	0	Base Address (quad word address)																										0	0	0
0	0	0	0	TDX (8 ≤ TDX ≤ 4088)									0	0	0	0	0	0	0	TDY (1 ≤ TDY ≤ 4095)											
0	0	0	0	TXOFS (0 ≤ TXOFS ≤ TDX – 1)												0	0	0	0	TYOFS (0 ≤ TYOFS ≤ TDY – 1)											
Sign	DX1 (-32768 ≤ DX1 ≤ 32767)															Sign	DY1 (-32768 ≤ DY1 ≤ 32767)														
Sign	DX2 (-32768 ≤ DX2 ≤ 32767)															Sign	DY2 (-32768 ≤ DY2 ≤ 32767)														
Sign	DX3 (-32768 ≤ DX3 ≤ 32767)															Sign	DY3 (-32768 ≤ DY3 ≤ 32767)														
Sign	DX4 (-32768 ≤ DX4 ≤ 32767)															Sign	DY4 (-32768 ≤ DY4 ≤ 32767)														

SS = 0 and REL = 1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1000_0001								Reserve (all 0)								Draw Mode															
Color1																Color0															
Sign extended			Sign	Base Address (longword address)																										0	0
0	0	0	0	TDX (8 ≤ TDX ≤ 4088)								0	0	0	0	0	0	0	0	TDY (1 ≤ TDY ≤ 4095)											
0	0	0	0	TXOFS (0 ≤ TXOFS ≤ TDX – 1)												0	0	0	0	TYOFS (0 ≤ TYOFS ≤ TDY – 1)											
Sign	DX1 (-32768 ≤ DX1 ≤ 32767)															Sign	DY1 (-32768 ≤ DY1 ≤ 32767)														
Sign	DX2 (-32768 ≤ DX2 ≤ 32767)															Sign	DY2 (-32768 ≤ DY2 ≤ 32767)														
Sign	DX3 (-32768 ≤ DX3 ≤ 32767)															Sign	DY3 (-32768 ≤ DY3 ≤ 32767)														
Sign	DX4 (-32768 ≤ DX4 ≤ 32767)															Sign	DY4 (-32768 ≤ DY4 ≤ 32767)														

Note: Adding the address (longword: 32-bit units) where the command code is located to the Base Address (longword: 32-bit units) must result in a quad word address (64-bit units).

(c) Code

B'10000001

(d) Rendering Attributes

Reference Data			Drawing Destination	
Multi-Valued Source	**Binary Source**	**Binary Work**	**Specified Color**	**Rendering**	**Work**
	O	O (only WORK = 1)		O	

Draw Mode

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Fixed to 0	Fixed to 0	CLIP	RCLIP	STRANS	Fixed to 0	WORK	SS	REL	STYLE	BLKE	NET	EOS	COOF	αE	Fixed to 0

(e) Command Parameters

TXS, TYS:	Source starting point. Write 0 to the unused bits.
Base Address:	Source start absolute address (Quad word address. Write 0 to bits A31 to A29 and A2 to A0.) Source start relative address (Longword address. Negative number expressed as two's complement. Bits A31 to A29 are used to extend the sign in bit A28. Write 0 to bits A1 and A0.)

Note: Even in 32-bit addressing mode, write the values in bits 28 to 3 of the specified 32-bit address to bits A28 to A3.

TDX, TDY:	Source size. Write 0 to the unused bits.
DXn, DYn (n = 1 to 4):	Rendering coordinates (absolute coordinates). Negative numbers expressed as two's complement.
TXOFS, TYOFS:	Source offset. Write 0 to the unused bits.
Color0, Color1:	8- or 16-bit/pixel color specification. For 16-bit/pixel drawing, the color specification should match the destination pixel format. For 8-bit/pixel drawing, the same value should be set in the upper and lower bytes.

(f) Description

Draws binary (1-bit/pixel) source data in any quadrilateral rendering area, using the colors specified by parameters Color0 and Color1. For the color specifications (Color0 and Color1) in 8-bit/pixel drawing, set the same 8-bit data in the upper and lower bytes. The source data is always scanned horizontally, but diagonal scanning may be used in the drawing, depending on the shape. In diagonally-scanned drawing, double-writing occurs to fill in gaps. A multiple of 8 pixels must be set as the TDX value, regardless of the SS bit value. If TXOFS or TYOFS is set, the source at a location shifted by the offset amount is referenced. Make the TXOFS and TYOFS settings in pixel units.

1. When source style specification is selected as a rendering attribute (STYLE = 1), the source data is not enlarged or reduced, but is referenced repeatedly.
2. When work specification is selected as a rendering attribute (WORK = 1), only places where the work coordinate pixel is 1 are drawn at rendering coordinates while referencing work coordinates for the same coordinates as the rendering coordinates.
3. When REL = 0, the source address can be specified as an absolute address. When REL = 1, the source address can be specified as a relative address with respect to the memory address at which the POLYGON4B command code is located.

(g) Example

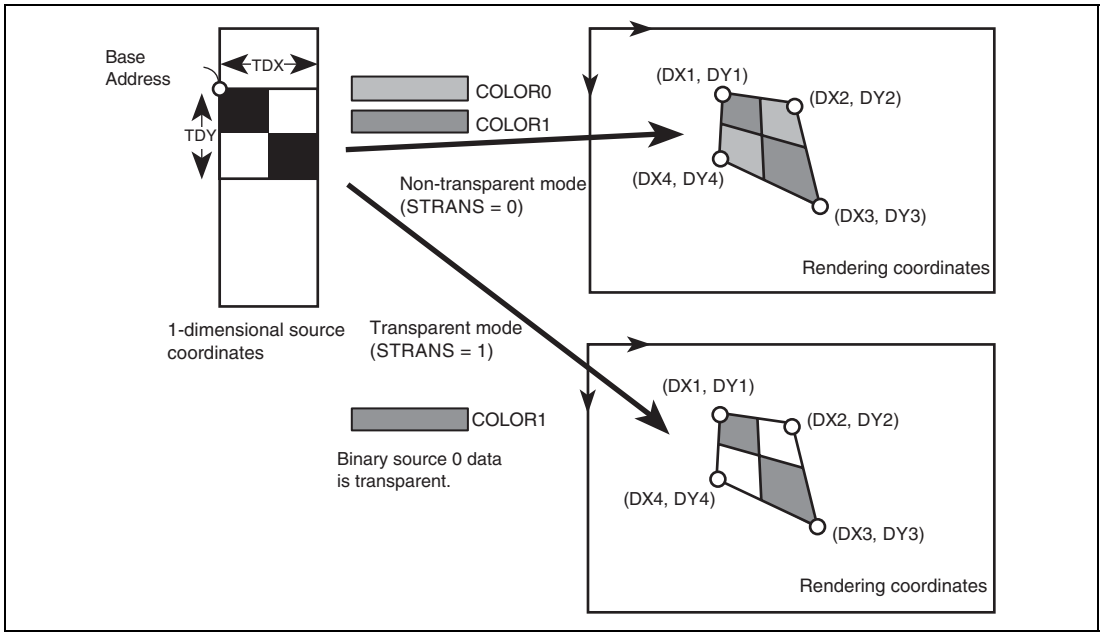


Figure 11.19 POLYGON4B Command Example

(3) POLYGON4C

(a) Function

Performs any four-vertex drawing at rendering coordinates with a monochrome specification.

(b) Command Format

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1000_0000								Reserve (all 0)								Draw Mode															
All 0																Color															
Sign	DX1 (-32768 ≤ DX1 ≤ 32767)															Sign	DY1 (-32768 ≤ DY1 ≤ 32767)														
Sign	DX2 (-32768 ≤ DX2 ≤ 32767)															Sign	DY2 (-32768 ≤ DY2 ≤ 32767)														
Sign	DX3 (-32768 ≤ DX3 ≤ 32767)															Sign	DY3 (-32768 ≤ DY3 ≤ 32767)														
Sign	DX4 (-32768 ≤ DX4 ≤ 32767)															Sign	DY4 (-32768 ≤ DY4 ≤ 32767)														

(c) Code

B'10000000

(d) Rendering Attributes

Reference Data			Drawing Destination	
Multi-Valued Source	Binary Source	Binary Work	Specified Color	Rendering	Work
		O (only WORK = 1)	O	O	

Draw Mode

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Fixed to 0	Fixed to 0	CLIP	RCLIP	Fixed to 0	Fixed to 0	WORK	Fixed to 0	Fixed to 0	Fixed to 0	BLKE	NET	EOS	COOF	αE	Fixed to 0

(e) Command Parameters

DXn, DYn (n = 1 to 4): Rendering coordinates (absolute coordinates). Negative numbers expressed as two's complement.

Color: 8- or 16-bit/pixel color specification. For 16-bit/pixel drawing, the color specification should match the destination pixel format.

For 8-bit/pixel drawing, the same value should be set in the upper and lower bytes.

(f) Description

Draws any quadrilateral in the rendering area in the single color specified by the Color parameter. When work specification is selected as a rendering attribute (WORK = 1), only places where the work coordinate pixel is 1 are drawn at rendering coordinates while referencing work coordinates for the same coordinates as the rendering coordinates.

(g) Example

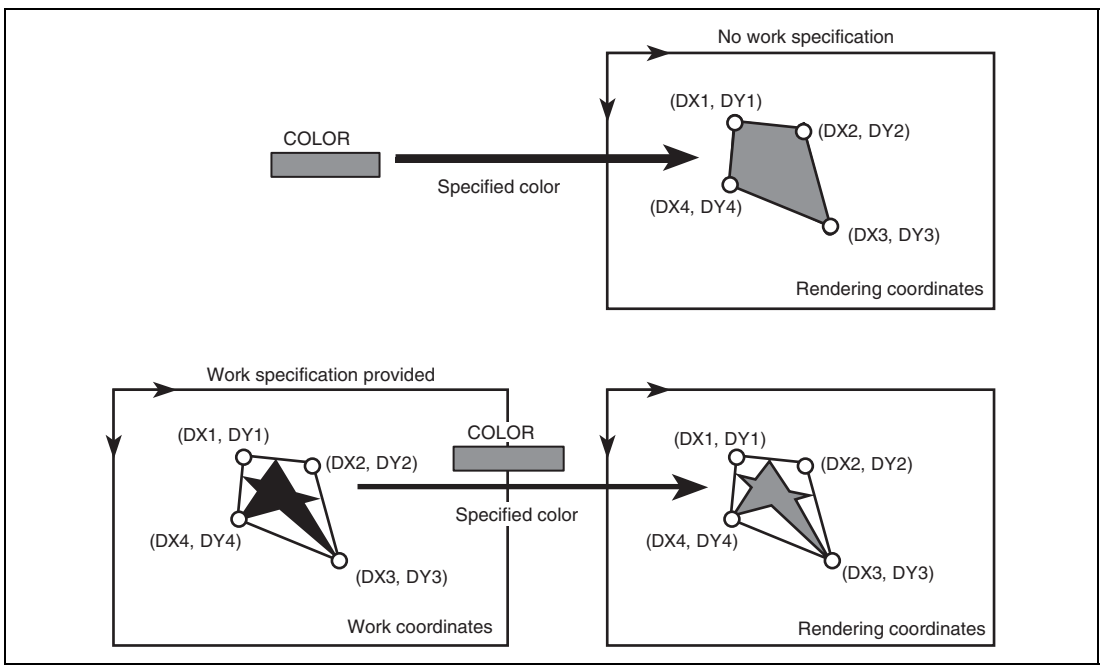


Figure 11.20 POLYGON4C Command Example

11.4.2 Line Drawing Commands

(1) LINEA

(a) Function

Draws a polygonal line with any width in the destination area while referencing a multi-valued (8- or 16-bit/pixel) source.

(b) Command Format

REL = 0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1011_0010								Reserve (all 0)								Draw Mode															
0	0	0	Base Address (quad word address)																										0	0	0
0	0	0	0	TDX (8 ≤ TDX ≤ 4088)												0	0	0	0	0	0	0	TDY (1 ≤ TDY ≤ 4095)								
0	0	0	0	TXOFS (0 ≤ TXOFS ≤ TDX – 1)												n (2 ≤ n ≤ 65535)															
Reserve (all 0)																0	0	0	0	0	0	0	0	0	0	W (0,2 ≤ W ≤ 63)					
Sign	DX1 (-32768 ≤ DX1 ≤ 32767)															Sign	DY1 (-32768 ≤ DY1 ≤ 32767)														
Sign	:															Sign	:														
Sign	:															Sign	:														
Sign	DXn (-32768 ≤ DXn ≤ 32767)															Sign	DYn (-32768 ≤ DYn ≤ 32767)														

Notes: 1. When W = 0, set TDY to 1.
2. When n = 0 or 1, correct operation is not guaranteed.

REL = 1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1011_0010								Reserve (all 0)								Draw Mode															
Sign extended			Sign	Base Address (longword address)																										0	0
0	0	0	0	TDX (8 ≤ TDX ≤ 4088)								0	0	0	0	0	0	0	0	TDY (1 ≤ TDY ≤ 4095)											
0	0	0	0	TXOFS (0 ≤ TXOFS ≤ TDX − 1)												n (2 ≤ n ≤ 65535)															
Reserve (all 0)																0	0	0	0	0	0	0	0	0	0	W (0,2 ≤ W ≤ 63)					
Sign	DX1 (-32768 ≤ DX1 ≤ 32767)															Sign	DY1 (-32768 ≤ DY1 ≤ 32767)														
Sign	:															Sign	:														
Sign	:															Sign	:														
Sign	DXn (-32768 ≤ DXn ≤ 32767)															Sign	DYn (-32768 ≤ DYn ≤ 32767)														

Notes: 1. Adding the address (longword: 32-bit units) where the command code is located to the Base Address (longword: 32-bit units) must result in a quad word address (64-bit units).
2. When W = 0, set TDY to 1.
3. When n = 0 or 1, correct operation is not guaranteed.

(c) Code

B'10110010

(d) Rendering Attributes

Reference Data			Drawing Destination	
Multi-Valued Source	**Binary Source**	**Binary Work**	**Specified Color**	**Rendering**	**Work**
O				O	

Draw Mode

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Fixed to 0	Fixed to 0	CLIP	RCLIP	STRANS	Fixed to 0	Fixed to 0	SS (0)	REL	STYLE (1)	Fixed to 0	NET	EOS	COOF	AA	Fixed to 0

Notes: 1. Clear the SS bit to 0.
2. Set the STYLE bit to 1.

(e) Command Parameters

Base Address:	Source start absolute address (Quad word address. Write 0 to bits A31 to A29 and A2 to A0.) Source start relative address (Longword address. Negative number expressed as two's complement. Bits A31 to A29 are used to extend the sign in bit A28. Write 0 to bits A1 and A0.)

Note: Even in 32-bit addressing mode, write the values in bits 28 to 3 of the specified 32-bit address to bits A28 to A3.

TDX, TDY:	Source size. Write 0 to the unused bits.
TXOFS:	Source offset. Write 0 to the unused bits.
n (n = 2 to 65,535):	Number of vertices
W:	Line width. Set a 6-bit integer. Write 0 to the unused bits. When 0 is set in W, a polygonal line of line width 1 is drawn. Setting 1 in W is prohibited.
DXn (n = 2 to 65,535):	Rendering coordinate (absolute coordinate). Negative number expressed as two's complement.
DYn (n = 2 to 65,535):	Rendering coordinate (absolute coordinate). Negative number expressed as two's complement.

(f) Description

Draws a polygonal line from vertex 1 (DX1, DY1), through vertex 2 (DX2, DY2), ..., vertex n – 1 (DXn – 1, DYn – 1), to vertex n (DXn, DYn). Set a multiple of 8 pixels as the TDX value. If TXOFS is set, the source at a location shifted by the offset amount is referenced. Make the TXOFS setting in pixel units. Pattern repetition selected by the STYLE bit is only performed in the X direction of the source data. The source data is enlarged or reduced in proportion to the line width in the Y direction.

When a value greater than 1 is set in W, a bold line can be drawn.

Notes: 1. 8-point drawing is used for a line width of 1. Both of the 8-point drawing and 4-point drawing are used for bold line drawing.
2. The final point of each line segment is drawn. When the starting and final coordinate points of a line segment match, a single dot is drawn for a line width of 1 and nothing is drawn for bold line drawing
3. When AA = 1, note the following:
For a dashed line, antialiasing is not performed for the gaps in the dashed line.
When the starting and final coordinate points of a line segment match, antialiasing is not performed.

(2) LINEB

(a) Function

Draws a polygonal line with any width in the destination area while referencing a binary (1-bit/pixel) source.

(b) Command Format

REL = 0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1011_0001								Reserve (all 0)								Draw Mode															
Color1																Color0															
0	0	0	Base Address (quad word address)																										0	0	0
0	0	0	0	TDX (8 ≤ TDX ≤ 4088)												0	0	0	0	0	0	0	TDY (1 ≤ TDY ≤ 4095)								
0	0	0	0	TXOFS (0 ≤ TXOFS ≤ TDX – 1)												n (2 ≤ n ≤ 65535)															
Reserve (all 0)																0	0	0	0	0	0	0	0	0	0	W (0,2 ≤ W ≤ 63)					
Sign	DX1 (-32768 ≤ DX1 ≤ 32767)															Sign	DY1 (-32768 ≤ DY1 ≤ 32767)														
Sign	:															Sign	:														
Sign	:															Sign	:														
Sign	DXn (-32768 ≤ DXn ≤ 32767)															Sign	DYn (-32768 ≤ DYn ≤ 32767)														

Notes: 1. When W = 0, set TDY to 1.
2. When n = 0 or 1, correct operation is not guaranteed.

REL = 1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1011_0001								Reserve (all 0)								Draw Mode															
Color1																Color0															
Sign extended				Sign				Base Address (longword address)																						0	0
0	0	0	0	TDX (8 ≤ TDX ≤ 4088)								0	0	0	0	0	0	0	TDY (1 ≤ TDY ≤ 4095)												
0	0	0	0	TXOFS (0 ≤ TXOFS ≤ TDX – 1)												n (2 ≤ n ≤ 65535)															
Reserve (all 0)																0	0	0	0	0	0	0	0	0	0	W (0,2 ≤ W ≤ 63)					
Sign	DX1 (-32768 ≤ DX1 ≤ 32767)															Sign	DY1 (-32768 ≤ DY1 ≤ 32767)														
Sign	:															Sign	:														
Sign	:															Sign	:														
Sign	DXn (-32768 ≤ DXn ≤ 32767)															Sign	DYn (-32768 ≤ DYn ≤ 32767)														

Notes: 1. Adding the address (longword: 32-bit units) where the command code is located to the Base Address (longword: 32-bit units) must result in a quad word address (64-bit units).
2. When W = 0, set TDY to 1.
3. When n = 0 or 1, correct operation is not guaranteed.

(c) Code

B'10110001

(d) Rendering Attributes

Reference Data			Drawing Destination	
Multi-Valued Source	Binary Source	Binary Work	Specified Color	Rendering	Work
	O			O	

Draw Mode

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Fixed to 0	Fixed to 0	CLIP	RCLIP	STRANS	Fixed to 0	Fixed to 0	SS (0)	REL	STYLE (1)	Fixed to 0	NET	EOS	COOF	AA	Fixed to 0

Notes: 1. Clear the SS bit to 0.
2. Set the STYLE bit to 1.

(e) Command Parameters

Color0, Color1:	8- or 16-bit/pixel color specification. For 16-bit/pixel drawing, the color specification should match the destination pixel format. For 8-bit/pixel drawing, the same value should be set in the upper and lower bytes.
Base Address:	Source start absolute address (Quad word address. Write 0 to bits A31 to A29 and A2 to A0. Source start relative address (Longword address. Negative number expressed as two's complement. Bits A31 to A29 are used to extend the sign in bit A28. Write 0 to bits A1 and A0.)

Note: Even in 32-bit addressing mode, write the values in bits 28 to 3 of the specified 32-bit address to bits A28 to A3.

TDX, TDY:	Source size. Write 0 to the unused bits.
TXOFS:	Source offset. Write 0 to the unused bits.
n (n = 2 to 65,535):	Number of vertices
W:	Line width. Set a 6-bit integer. Write 0 to the unused bits. When 0 is set in W, a polygonal line of line width 1 is drawn. Setting 1 in W is prohibited.
DXn (n = 2 to 65,535):	Rendering coordinate (absolute coordinate). Negative number expressed as two's complement.
DYn (n = 2 to 65,535):	Rendering coordinate (absolute coordinate). Negative number expressed as two's complement.

(f) Description

Draws a polygonal line from vertex 1 (DX1, DY1), through vertex 2 (DX2, DY2), ..., vertex n – 1 (DXn – 1, DYn – 1), to vertex n (DXn, DYn). Set a multiple of 8 pixels as the TDX value. If TXOFS is set, the source at a location shifted by the offset amount is referenced. Make the TXOFS setting in pixel units. Pattern repetition selected by the STYLE bit is only performed in the X direction of the source data. The source data is enlarged or reduced in proportion to the line width in the Y direction.

When a value greater than 1 is set in W, a bold line can be drawn.

Notes: 1. 8-point drawing is used for a line width of 1. Both of the 8-point drawing and 4-point drawing are used for bold line drawing.

2. The final point of each line segment is drawn. When the starting and final coordinate points of a line segment match, a single dot is drawn for a line width of 1 and nothing is drawn for bold line drawing.

3. When AA = 1, note the following:

   For a dashed line, antialiasing is not performed for the gaps in the dashed line.

   When the starting and final coordinate points of a line segment match, antialiasing is not performed.

(g) Example

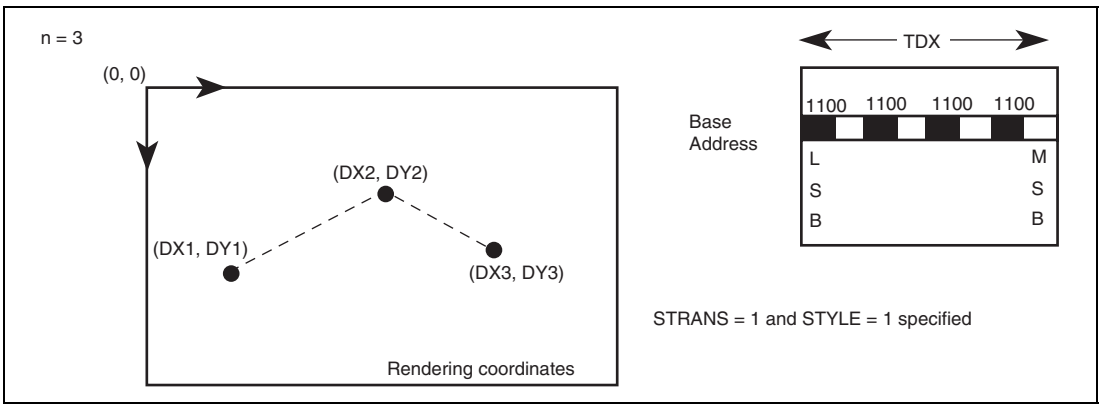


Figure 11.21 LINEB Command Example

(3) LINEC

(a) Function

Draws a polygonal line with any width in the destination area with a monochrome specification.

(b) Command Format

LINKE = 0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1011_0000								Reserve (all 0)								Draw Mode															
Color																n (2 ≤ n ≤ 65535)															
Reserve (all 0)																0	0	0	0	0	0	0	0	0	0	W (0,2 ≤ W ≤ 63)					
Sign	DX1 (-32768 ≤ DX1 ≤ 32767)															Sign	DY1 (-32768 ≤ DY1 ≤ 32767)														
Sign	:															Sign	:														
Sign	:															Sign	:														
Sign	DXn (-32768 ≤ DXn ≤ 32767)															Sign	DYn (-32768 ≤ DYn ≤ 32767)														

Note: When n = 0 or 1, correct operation is not guaranteed.

LINKE = 1 and LREL = 0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1011_0000								Reserve (all 0)								Draw Mode															
Color																n (2 ≤ n ≤ 65535)															
Reserve (all 0)																0	0	0	0	0	0	0	0	0	0	W (0,2 ≤ W ≤ 63)					
0	0	0	LINK Address (longword address)																											0	0

Notes: 1. When n = 0 or 1, correct operation is not guaranteed.

2. The longword address following the LINK Address is handled as the next command code. Therefore, do not specify the longword address following the address where the LINK Address is to be assigned as the link destination address specified by the LINK Address.

LINKE = 1 and LREL = 1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1011_0000								Reserve (all 0)								Draw Mode															
Color																n (2 ≤ n ≤ 65535)															
Reserve (all 0)																0	0	0	0	0	0	0	0	0	0	W (0,2 ≤ W ≤ 63)					
Sign extended								Sign	LINK Address (longword address)																					0	0

Notes: 1. When n = 0 or 1, correct operation is not guaranteed.

2. The longword address following the LINK Address is handled as the next command code. Therefore, do not specify the longword address following the address where the LINK Address is to be assigned as the link destination address specified by adding the address where the command code is located to the LINK Address.

(c) Code

B'10110000

(d) Rendering Attributes

Reference Data			Drawing Destination	
Multi-Valued Source	**Binary Source**	**Binary Work**	**Specified Color**	**Rendering**	**Work**
			O	O	

Draw Mode

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Fixed to 0	Fixed to 0	CLIP	RCLIP	Fixed to 0	LINKE	LREL	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	NET	EOS	COOF	AA	Fixed to 0

(e) Command Parameters

Color:	8- or 16-bit/pixel color specification. For 16-bit/pixel drawing, the color specification should match the destination pixel format. For 8-bit/pixel drawing, the same value should be set in the upper and lower bytes.
n (n = 2 to 65,535):	Number of vertices
W:	Line width. Set a 6-bit integer. Write 0 to the unused bits. When 0 is set in W, a polygonal line of line width 1 is drawn. Setting 1 in W is prohibited.
DXn (n = 2 to 65,535):	Rendering coordinate (absolute coordinate). Negative number expressed as two's complement.
DYn (n = 2 to 65,535):	Rendering coordinate (absolute coordinate). Negative number expressed as two's complement.
LINK Address:	LINK absolute address (Longword address. Write 0 to bits A31 to A29, A1, and A0.) LINK relative address (Longword address. Negative number expressed as two's complement. Bits A31 to A29 are used to extend the sign in bit A28. Write 0 to bits A1 and A0.)

Note: Even in 32-bit addressing mode, write the values in bits 28 to 3 of the specified 32-bit address to bits A28 to A3.

(f) Description

Draws a polygonal line from vertex 1 (DX1, DY1), through vertex 2 (DX2, DY2), ..., vertex n – 1 (DXn – 1, DYn – 1), to vertex n (DXn, DYn). When a value greater than 1 is set in W, a bold line can be drawn. When LINKE = 1, the vertex coordinates are read from the memory address specified by the LINK Address. The LINK Address can be specified through the LREL bit as an absolute address or a relative address with respect to the memory address at which the LINEC command code is located.

Notes: 1. 8-point drawing is used for a line width of 1. Both of the 8-point drawing and 4-point drawing are used for bold line drawing.

2. The final point of each line segment is drawn. When the starting and final coordinate points of a line segment match, a single dot is drawn for a line width of 1 and nothing is drawn for bold line drawing.

3. When AA = 1, note the following:

When the starting and final coordinate points of a line segment match, antialiasing is not performed.

(g) Example

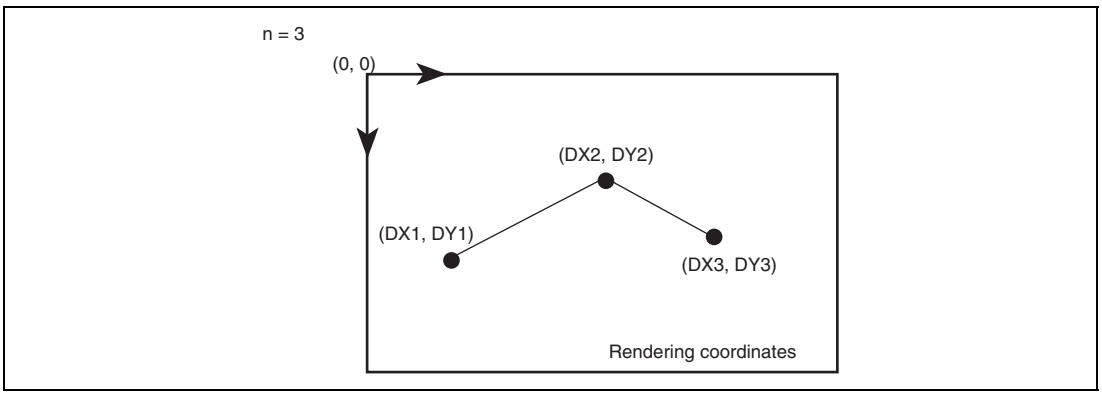


Figure 11.22 LINEC Command Example

(4) LINED

(a) Function

Performs antialiasing for the exterior frame of a polygon. This command can only be executed for a 16-bit/pixel destination.

(b) Command Format

LINKE = 0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1011_0011								Reserve (all 0)								Draw Mode															
Reserve (all 0)																n (2 ≤ n ≤ 65535)															
Sign	DX1 (-32768 ≤ DX1 ≤ 32767)															Sign	DY1 (-32768 ≤ DY1 ≤ 32767)														
Sign	:															Sign	:														
Sign	:															Sign	:														
Sign	DXn (-32768 ≤ DXn ≤ 32767)															Sign	DYn (-32768 ≤ DYn ≤ 32767)														

Note: When n = 0 or 1, correct operation is not guaranteed.

LINKE = 1 and LREL = 0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1011_0011								Reserve (all 0)								Draw Mode															
Reserve (all 0)																n (2 ≤ n ≤ 65535)															
0	0	0	LINK Address (longword address)																											0	0

Notes: 1. When n = 0 or 1, correct operation is not guaranteed.

2. The longword address following the LINK Address is handled as the next command code. Therefore, do not specify the longword address following the address where the LINK Address is to be assigned as the link destination address specified by the LINK Address.

LINKE = 1 and LREL = 1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1011_0011								Reserve (all 0)								Draw Mode															
Reserve (all 0)																n (2 ≤ n ≤ 65535)															
Sign extended							Sign	LINK Address (longword address)																						0	0

Notes: 1. When n = 0 or 1, correct operation is not guaranteed.

2. The longword address following the LINK Address is handled as the next command code. Therefore, do not specify the longword address following the address where the LINK Address is to be assigned as the link destination address specified by adding the address where the command code is located to the LINK Address.

(c) Code

B'10110011

(d) Rendering Attributes

Reference Data				Drawing Destination	
Multi-Valued Source	**Binary Source**	**Binary Work**	**Specified Color**	**Rendering**	**Work**
				O	

Draw Mode

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Fixed to 0	Fixed to 0	CLIP	RCLIP	Fixed to 0	LINKE	LREL	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	AA (1)	CLKW

Note: Set the AA bit to 1.

(e) Command Parameters

n (n = 2 to 65,535):	Number of vertices
DXn (n = 2 to 65,535):	Rendering coordinate (absolute coordinate). Negative number expressed as two's complement.
DYn (n = 2 to 65,535):	Rendering coordinate (absolute coordinate). Negative number expressed as two's complement.
LINK Address:	LINK absolute address (Longword address. Write 0 to bits A31 to A29, A1, and A0).
	LINK relative address (Longword address. Negative number expressed as two's complement. Bits A31 to A29 are used to extend the sign in bit A28. Write 0 to bits A1 and A0.)

Note: Even in 32-bit addressing mode, write the values in bits 28 to 3 of the specified 32-bit address to bits A28 to A3.

(f) Description

Performs antialiasing for the exterior frame of a polygon drawn using work reference.

The CLKW bit specifies whether the order to give the n vertices is clockwise or counterclockwise: CLKW = 1 selects clockwise and CLKW = 0 selects counterclockwise. When clockwise is specified, the left image with respect to the drawing direction is referenced by antialiasing. On the other hand, the right image is referenced when counterclockwise is selected. When LINKE = 1, the vertex coordinates are read from the memory address specified by the LINK Address. The LINK Address can be specified through the LREL bit as an absolute address or a relative address with respect to the memory address at which the LINED command code is located.

This command can only be executed for a 16-bit/pixel destination.

When a polygon used in work reference is drawn by the FTRAPC (RFTRAPC) command, perform drawing with both the EDG and EOS bits set to 1.

Notes: 1. 8-point drawing is used.

2. The final point of each line segment is not drawn. When antialiasing is performed for the exterior frame of a polygon drawn by a POLYGON4 type command, the paths may not match.
3. When the starting and final coordinate points of a line segment match, nothing is drawn.
4. Antialiasing is not performed for horizontal, vertical, and 45-degree diagonal line segments, which are pre-clipped inside the 2DG.
5. Clipping is performed on a pixel basis when either the referenced pixel or the pixel to be drawn is outside the clipping area, and antialiasing is not performed in such a case.

(g) Example

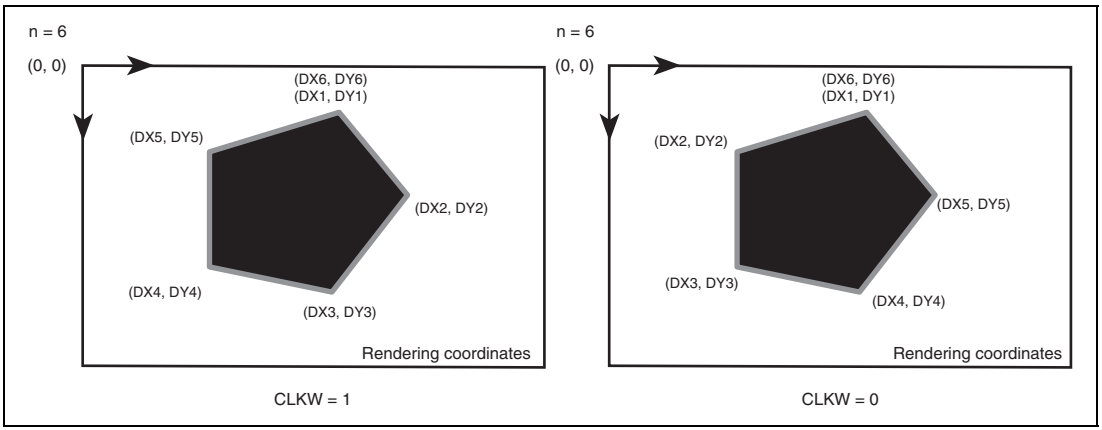


Figure 11.23 LINE Command Example

(5) RLINEA

(a) Function

Draws a polygonal line with any width in the destination area with a relative coordinate specification from the current pointer value while referencing a multi-valued (8- or 16-bit/pixel) source.

(b) Command Format

REL = 0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1011_0110								Reserve (all 0)								Draw Mode															
0	0	0	Base Address (quad word address)																										0	0	0
0	0	0	0	TDX (8 ≤ TDX ≤ 4088)								0	0	0	0	0	0	0	0	TDY (1 ≤ TDY ≤ 4095)											
0	0	0	0	TXOFS (0 ≤ TXOFS ≤ TDX – 1)												n (1 ≤ n ≤ 65535)															
Reserve (all 0)																0	0	0	0	0	0	0	0	0	0	W (0,2 ≤ W ≤ 63)					
Sign	DX2 (-128 ≤ DX2 ≤ 127)							Sign	DY2 (-128 ≤ DY2 ≤ 127)							Sign	DX1 (-128 ≤ DX1 ≤ 127)							Sign	DY1 (-128 ≤ DY1 ≤ 127)						
Sign	:							Sign	:							Sign	:							Sign	:						
Sign	:							Sign	:							Sign	:							Sign	:						
Sign	DXn (-128 ≤ DXn ≤ 127)							Sign	DYn (-128 ≤ DYn ≤ 127)							Sign	DXn-1 (-128 ≤ DXn-1 ≤ 127)							Sign	DYn-1 (-128 ≤ DYn-1 ≤ 127)						

Notes: 1. When W = 0, set TDY to 1.
2. When n = 0, correct operation is not guaranteed.
3. When n is an odd number, insert a dummy word of 0 at the end.

REL = 1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1011_0110								Reserve (all 0)								Draw Mode															
Sign extended			Sign	Base Address (longword address)																										0	0
0	0	0	0	TDX (8 ≤ TDX ≤ 4088)								0	0	0	0	0	0	0	0	TDY (1 ≤ TDY ≤ 4095)											
0	0	0	0	TXOFS (0 ≤ TXOFS ≤ TDX – 1)												n (1 ≤ n ≤ 65535)															
Reserve (all 0)																0	0	0	0	0	0	0	0	0	0	W (0,2 ≤ W ≤ 63)					
Sign	DX2 (-128 ≤ DX2 ≤ 127)			Sign	DY2 (-128 ≤ DY2 ≤ 127)			Sign	DX1 (-128 ≤ DX1 ≤ 127)							Sign	DY1 (-128 ≤ DY1 ≤ 127)														
Sign	:			Sign	:			Sign	:							Sign	:														
Sign	:			Sign	:			Sign	:							Sign	:														
Sign	DXn (-128 ≤ DXn ≤ 127)			Sign	DYn (-128 ≤ DYn ≤ 127)			Sign	DXn-1 (-128 ≤ DXn-1 ≤ 127)							Sign	DYn-1 (-128 ≤ DYn-1 ≤ 127)														

Notes: 1. Adding the address (longword: 32-bit units) where the command code is located to the Base Address (longword: 32-bit units) must result in a quad word address (64-bit units).
2. When W = 0, set TDY to 1.
3. When n = 0, correct operation is not guaranteed.
4. When n is an odd number, insert a dummy word of 0 at the end.

(c) Code

B'10110110

(d) Rendering Attributes

Reference Data			Drawing Destination	
Multi-Valued Source	**Binary Source**	**Binary Work**	**Specified Color**	**Rendering**	**Work**
O				O	

Draw Mode

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Fixed to 0	Fixed to 0	CLIP	RCLIP	STRANS	Fixed to 0	Fixed to 0	SS (0)	REL	STYLE (1)	Fixed to 0	NET	EOS	COOF	AA	Fixed to 0

Notes: 1. Clear the SS bit to 0.
2. Set the STYLE bit to 1.

(e) Command Parameters

Base Address:	Source start absolute address (Quad word address. Write 0 to bits A31 to A29 and A2 to A0.) Source start relative address (Longword address. Negative number expressed as two's complement. Bits A31 to A29 are used to extend the sign in bit A28. Write 0 to bits A1 and A0.)

Note: Even in 32-bit addressing mode, write the values in bits 28 to 3 of the specified 32-bit address to bits A28 to A3.

TDX, TDY:	Source size. Write 0 to the unused bits.
TXOFS:	Source offset. Write 0 to the unused bits.
n (n = 1 to 65,535):	Number of vertices
W:	Line width. Set a 6-bit integer. Write 0 to the unused bits. When 0 is set in W, a polygonal line of line width 1 is drawn. Setting 1 in W is prohibited.
DXn (n = 1 to 65,535):	Rendering coordinate (relative coordinate). Negative number expressed as two's complement.
DYn (n = 1 to 65,535):	Rendering coordinate (relative coordinate). Negative number expressed as two's complement.

(f) Description

Draws a polygonal line comprising line segments (XC, YC) – (XC + DX1, YC + DY1), (XC + DX1, YC + DY1) – (XC + DX1 + DX2, YC + DY1 + DY2), ..., (XC + ... + DXn – 1, YC + ... + DYn – 1) – (XC + ... + DXn – 1 + DXn, YC + ... + DYn – 1 + DYn) to the coordinates specified by the relative shift (DX, DY) from the current pointer values (XC, YC).

The final coordinate point is stored as the current pointer values (XC, YC). Set a multiple of 8 pixels as the TDX value. If TXOFS is set, the source at a location shifted by the offset amount is referenced. Make the TXOFS setting in pixel units.

Pattern repetition selected by the STYLE bit is only performed in the X direction of the source data. The source data is enlarged or reduced in proportion to the line width in the Y direction.

When a value greater than 0 is set in W, a bold line can be drawn.

Notes: 1. 8-point drawing is used for a line width of 1. Both of the 8-point drawing and 4-point drawing are used for bold line drawing.
2. The final point of each line segment is drawn. When the starting and final coordinate points of a line segment match, a single dot is drawn for a line width of 1 and nothing is drawn for bold line drawing.
3. When AA = 1, note the following:
For a dashed line, antialiasing is not performed for the gaps in the dashed line.
When the starting and final coordinate points of a line segment match, antialiasing is not performed.
4. The final coordinate point is stored as the current pointer values (XC, YC).

(6) RLINEB

(a) Function

Draws a polygonal line with any width in the destination area with a relative coordinate specification from the current pointer value while referencing a binary (1-bit/pixel) source.

(b) Command Format

REL = 0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1011_0101								Reserve (all 0)								Draw Mode															
Color1																Color0															
0	0	0	Base Address (quad word address)																									0	0	0	
0	0	0	0	TDX (8 ≤ TDX ≤ 4088)									0	0	0	0	0	0	0	TDY (1 ≤ TDY ≤ 4095)											
0	0	0	0	TXOFS (0 ≤ TXOFS ≤ TDX − 1)												n (1 ≤ n ≤ 65535)															
Reserve (all 0)																0	0	0	0	0	0	0	0	0	0	W (0,2 ≤ W ≤ 63)					
Sign	DX2 (-128 ≤ DX2 ≤ 127)							Sign	DY2 (-128 ≤ DY2 ≤ 127)							Sign	DX1 (-128 ≤ DX1 ≤ 127)							Sign	DY1 (-128 ≤ DY1 ≤ 127)						
Sign	:							Sign	:							Sign	:							Sign	:						
Sign	:							Sign	:							Sign	:							Sign	:						
Sign	DXn (-128 ≤ DXn ≤ 127)							Sign	DYn (-128 ≤ DYn ≤ 127)							Sign	DXn-1 (-128 ≤ DXn-1 ≤ 127)							Sign	DYn-1 (-128 ≤ DYn-1 ≤ 127)						

Notes: 1. When W = 0, set TDY to 1.
2. When n = 0, correct operation is not guaranteed.
3. When n is an odd number, insert a dummy word of 0 at the end.

REL = 1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1011_0101								Reserve (all 0)								Draw Mode															
Color1																Color0															
Sign extended			Sign	Base Address (longword address)																										0	0
0	0	0	0	TDX (8 ≤ TDX ≤ 4088)								0	0	0	0	0	0	0	0	TDY (1 ≤ TDY ≤ 4095)											
0	0	0	0	TXOFS (0 ≤ TXOFS ≤ TDX − 1)												n (1 ≤ n ≤ 65535)															
Reserve (all 0)																0	0	0	0	0	0	0	0	0	0	W (0,2 ≤ W ≤ 63)					
Sign	DX2 (-128 ≤ DX2 ≤ 127)							Sign	DY2 (-128 ≤ DY2 ≤ 127)							Sign	DX1 (-128 ≤ DX1 ≤ 127)							Sign	DY1 (-128 ≤ DY1 ≤ 127)						
Sign	:							Sign	:							Sign	:							Sign	:						
Sign	:							Sign	:							Sign	:							Sign	:						
Sign	DXn (-128 ≤ DXn ≤ 127)							Sign	DYn (-128 ≤ DYn ≤ 127)							Sign	DXn-1 (-128 ≤ DXn-1 ≤ 127)							Sign	DYn-1 (-128 ≤ DYn-1 ≤ 127)						

Notes: 1. Adding the address (longword: 32-bit units) where the command code is located to the Base Address (longword: 32-bit units) must result in a quad word address (64-bit units).
2. When W = 0, set TDY to 1.
3. When n = 0, correct operation is not guaranteed.
4. When n is an odd number, insert a dummy word of 0 at the end.

(c) Code

B'10110101

(d) Rendering Attributes

Reference Data			Drawing Destination	
Multi-Valued Source	**Binary Source**	**Binary Work**	**Specified Color**	**Rendering**	**Work**
	O			O	

Draw Mode

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Fixed to 0	Fixed to 0	CLIP	RCLIP	STRANS	Fixed to 0	Fixed to 0	SS (0)	REL	STYLE (1)	Fixed to 0	NET	EOS	COOF	AA	Fixed to 0

Notes: 1. Clear the SS bit to 0.
2. Set the STYLE bit to 1.

(e) Command Parameters

Color0, Color1:	8- or 16-bit/pixel color specification. For 16-bit/pixel drawing, the color specification should match the destination pixel format. For 8-bit/pixel drawing, the same value should be set in the upper and lower bytes.
Base Address:	Source start absolute address (Quad word address. Write 0 to bits A31 to A29 and A2 to A0.) Source start relative address (Longword address. Negative number expressed as two's complement. Bits A31 to A29 are used to extend the sign in bit A28. Write 0 to bits A1 and A0.)

Note: Even in 32-bit addressing mode, write the values in bits 28 to 3 of the specified 32-bit address to bits A28 to A3.

TDX, TDY:	Source size. Write 0 to the unused bits.
TXOFS:	Source offset. Write 0 to the unused bits.
n (n = 1 to 65,535):	Number of vertices
W:	Line width. Set a 6-bit integer. Write 0 to the unused bits. When 0 is set in W, a polygonal line of line width 1 is drawn. Setting 1 in W is prohibited.
DXn (n = 1 to 65,535):	Rendering coordinate (relative coordinate). Negative number expressed as two's complement.
DYn (n = 1 to 65,535):	Rendering coordinate (relative coordinate). Negative number expressed as two's complement.

(f) Description

Draws a polygonal line comprising line segments (XC, YC) – (XC + DX1, YC + DY1), (XC + DX1, YC + DY1) – (XC + DX1 + DX2, YC + DY1 + DY2), ..., (XC + ... + DXn – 1, YC + ... + DYn – 1) – (XC + ... + DXn – 1 + DXn, YC + ... + DYn – 1 + DYn) to the coordinates specified by the relative shift (DX, DY) from the current pointer values (XC, YC).

The final coordinate point is stored as the current pointer values (XC, YC).

Set a multiple of 8 pixels as the TDX value. If TXOFS is set, the source at a location shifted by the offset amount is referenced. Make the TXOFS setting in pixel units.

Pattern repetition selected by the STYLE bit is only performed in the X direction of the source data. The source data is enlarged or reduced in proportion to the line width in the Y direction.

When a value greater than 1 is set in W, a bold line can be drawn.

Notes: 1. 8-point drawing is used for a line width of 1. Both of the 8-point drawing and 4-point drawing are used for bold line drawing.
2. The final point of each line segment is drawn. When the starting and final coordinate points of a line segment match, a single dot is drawn for a line width of 1 and nothing is drawn for bold line drawing.
3. When AA = 1, note the following:
   For a dashed line, antialiasing is not performed for the gaps in the dashed line.
   When the starting and final coordinate points of a line segment match, antialiasing is not performed.
4. The final coordinate point is stored as the current pointer values (XC, YC).

(g) Example

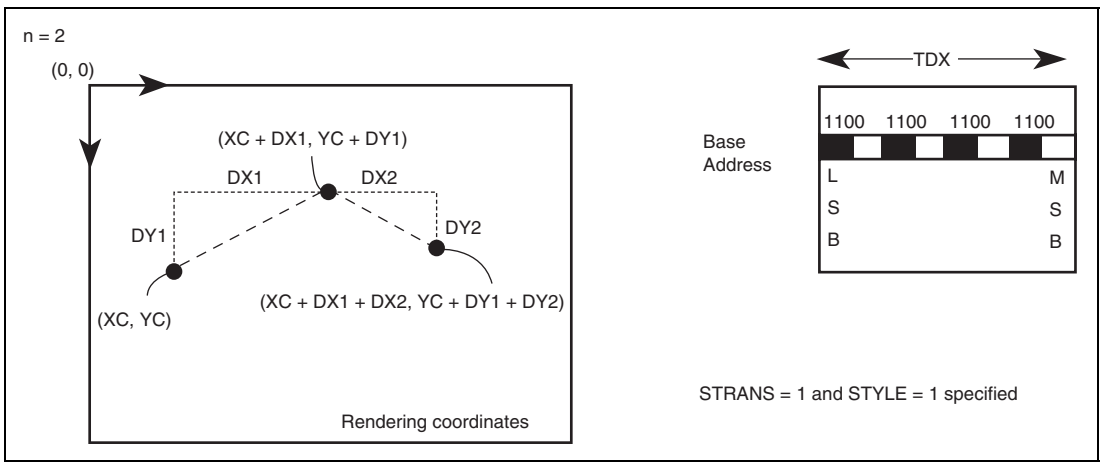


Figure 11.24 RLINEB Command Example

(7) RLINEC

(a) Function

Draws a polygonal line with any width in the destination area with a monochrome specification and with a relative coordinate specification from the current pointer value.

(b) Command Format

LINKE = 0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1011_0100								Reserve (all 0)								Draw Mode															
Color																n (1 ≤ n ≤ 65535)															
Reserve (all 0)																0	0	0	0	0	0	0	0	0	0	W (0,2 ≤ W ≤ 63)					
Sign	DX2 (-128 ≤ DX2 ≤ 127)							Sign	DY2 (-128 ≤ DY2 ≤ 127)							Sign	DX1 (-128 ≤ DX1 ≤ 127)							Sign	DY1 (-128 ≤ DY1 ≤ 127)						
Sign	:							Sign	:							Sign	:							Sign	:						
Sign	:							Sign	:							Sign	:							Sign	:						
Sign	DXn (-128 ≤ DXn ≤ 127)							Sign	DYn (-128 ≤ DYn ≤ 127)							Sign	DXn-1 (-128 ≤ DXn-1 ≤ 127)							Sign	DYn-1 (-128 ≤ DYn-1 ≤ 127)						

Notes: 1. When n = 0, correct operation is not guaranteed.
2. When n is an odd number, insert a dummy word of 0 at the end.

LINKE = 1 and LREL = 0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1011_0100								Reserve (all 0)								Draw Mode															
Color																n (1 ≤ n ≤ 65535)															
Reserve (all 0)																0	0	0	0	0	0	0	0	0	0	W (0,2 ≤ W ≤ 63)					
0	0	0	LINK Address (longword address)																											0	0

Notes: 1. When n = 0, correct operation is not guaranteed.
2. When n is an odd number, insert a dummy word of 0 at the end.
3. The longword address following the LINK Address is handled as the next command code. Therefore, do not specify the longword address following the address where the LINK Address is to be assigned as the link destination address specified by the LINK Address.

LINKE = 1 and LREL = 1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1011_0100								Reserve (all 0)								Draw Mode															
Color																n (1 ≤ n ≤ 65535)															
Reserve (all 0)																0	0	0	0	0	0	0	0	0	0	W (0,2 ≤ W ≤ 63)					
Sign extended	Sign	LINK Address (longword address)																												0	0

Notes: 1. When n = 0, correct operation is not guaranteed.

2. When n is an odd number, insert a dummy word of 0 at the end.

3. The longword address following the LINK Address is handled as the next command code. Therefore, do not specify the longword address following the address where the LINK Address is to be assigned as the link destination address specified by adding the address where the command code is located to the LINK Address.

(c) Code

B'10110100

(d) Rendering Attributes

Reference Data				Drawing Destination	
Multi-Valued Source	Binary Source	Binary Work	Specified Color	Rendering	Work
			O	O	

Draw Mode

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Fixed to 0	Fixed to 0	CLIP	RCLIP	Fixed to 0	LINKE	LREL	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	NET	EOS	COOF	AA	Fixed to 0

(e) Command Parameters

Color:	8- or 16-bit/pixel color specification. For 16-bit/pixel drawing, the color specification should match the destination pixel format. For 8-bit/pixel drawing, the same value should be set in the upper and lower bytes.
n (n = 1 to 65,535):	Number of vertices
W:	Line width. Set a 6-bit integer. Write 0 to the unused bits. When 0 is set in W, a polygonal line of line width 1 is drawn. Setting 1 in W is prohibited.
DXn (n = 1 to 65,535):	Rendering coordinate (relative coordinate). Negative number expressed as two's complement.
DYn (n = 1 to 65,535):	Rendering coordinate (relative coordinate). Negative number expressed as two's complement.
LINK Address:	LINK absolute address (Longword address. Write 0 to bits A31 to A29, A1, and A0.) LINK relative address (Longword address. Negative number expressed as two's complement. Bits A31 to A29 are used to extend the sign in bit A28. Write 0 to bits A1 and A0.)

Note: Even in 32-bit addressing mode, write the values in bits 28 to 3 of the specified 32-bit address to bits A28 to A3.

(f) Description

Draws a polygonal line comprising line segments (XC, YC) – (XC + DX1, YC + DY1), (XC + DX1, YC + DY1) – (XC + DX1 + DX2, YC + DY1 + DY2), ..., (XC + ... + DXn – 1, YC + ... + DYn – 1) – (XC + ... + DXn – 1 + DXn, YC + ... + DYn – 1 + DYn) to the coordinates specified by the relative shift (DX, DY) from the current pointer values (XC, YC). When a value greater than 1 is set in W, a bold line can be drawn. When LINKE = 1, the vertex coordinates are read from the memory address specified by the LINK Address. The LINK Address can be specified through the LREL bit as an absolute address or a relative address with respect to the memory address at which the RLINEC command code is located.

The final coordinate point is stored as the current pointer values (XC, YC).

Notes: 1. 8-point drawing is used for a line width of 1. Both of the 8-point drawing and 4-point drawing are used for bold line drawing.

2. The final point of each line segment is drawn. When the starting and final coordinate points of a line segment match, a single dot is drawn for a line width of 1 and nothing is drawn for bold line drawing.

3. When AA = 1, note the following:

   When the starting and final coordinate points of a line segment match, antialiasing is not performed.

4. The final coordinate point is stored as the current pointer values (XC, YC).

(g) Example

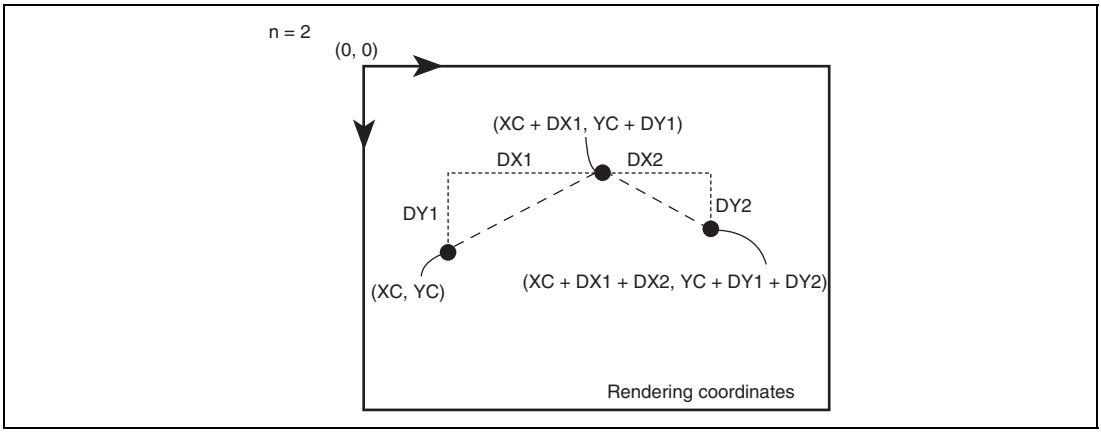


Figure 11.25 RLINEC Command Example

(8) RLINED

(a) Function

Performs antialiasing for the exterior frame of a polygon with a relative coordinate specification from the current pointer value. This command can only be executed for a 16-bit/pixel destination.

(b) Command Format

LINKE = 0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1011_0111								Reserve (all 0)								Draw Mode															
Reserve (all 0)																n (1 ≤ n ≤ 65535)															
Sign	DX2 (-128 ≤ DX2 ≤ 127)							Sign	DY2 (-128 ≤ DY2 ≤ 127)							Sign	DX1 (-128 ≤ DX1 ≤ 127)							Sign	DY1 (-128 ≤ DY1 ≤ 127)						
Sign	:							Sign	:							Sign	:							Sign	:						
Sign	:							Sign	:							Sign	:							Sign	:						
Sign	DXn (-128 ≤ DXn ≤ 127)							Sign	DYn (-128 ≤ DYn ≤ 127)							Sign	DXn-1 (-128 ≤ DXn-1 ≤ 127)							Sign	DYn-1 (-128 ≤ DYn-1 ≤ 127)						

Notes: 1. When n = 0, correct operation is not guaranteed.
2. When n is an odd number, insert a dummy word of 0 at the end.

LINKE = 1 and LREL = 0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1011_0111								Reserve (all 0)								Draw Mode															
Reserve (all 0)																n (1 ≤ n ≤ 65535)															
0	0	0	LINK Address (longword address)																											0	0

Notes: 1. When n = 0, correct operation is not guaranteed.
2. When n is an odd number, insert a dummy word of 0 at the end.
3. The longword address following the LINK Address is handled as the next command code. Therefore, do not specify the longword address following the address where the LINK Address is to be assigned as the link destination address specified by the LINK Address.

LINKE = 1 and LREL = 1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1011_0111								Reserve (all 0)								Draw Mode															
Reserve (all 0)																n (1 ≤ n ≤ 65535)															
Sign extended	Sign	LINK Address (longword address)																												0	0

Notes: 1. When n = 0, correct operation is not guaranteed.
2. When n is an odd number, insert a dummy word of 0 at the end.
3. The longword address following the LINK Address is handled as the next command code. Therefore, do not specify the longword address following the address where the LINK Address is to be assigned as the link destination address specified by adding the address where the command code is located to the LINK Address.

(c) Code

B'10110111

(d) Rendering Attributes

Reference Data				Drawing Destination	
Multi-Valued Source	Binary Source	Binary Work	Specified Color	Rendering	Work
				O	

Draw Mode

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Fixed to 0	Fixed to 0	CLIP	RCLIP	Fixed to 0	LINKE	LREL	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	AA (1)	CLKW

Note: Set the AA bit to 1.

(e) Command Parameters

n (n = 1 to 65,535):	Number of vertices
DXn (n = 1 to 65,535):	Rendering coordinate (relative coordinate). Negative number expressed as two's complement.
DYn (n = 1 to 65,535):	Rendering coordinate (relative coordinate). Negative number expressed as two's complement.
LINK Address:	LINK absolute address (Longword address. Write 0 to bits A31 to A29, A1, and A0.) LINK relative address (Longword address. Negative number expressed as two's complement. Bits A31 to A29 are used to extend the sign in bit A28. Write 0 to bits A1 and A0.)

Note: Even in 32-bit addressing mode, write the values in bits 28 to 3 of the specified 32-bit address to bits A28 to A3.

(f) Description

Performs antialiasing for the exterior frame of a polygon drawn using work reference, with a relative coordinate specification from the current pointer value.

The CLKW bit specifies whether the order to give the n vertices is clockwise or counterclockwise: CLKW = 1 selects clockwise and CLKW = 0 selects counterclockwise. When clockwise is specified, the left image with respect to the drawing direction is referenced by antialiasing. On the other hand, the right image is referenced when counterclockwise is selected. When LINKE = 1, the vertex coordinates are read from the memory address specified by the LINK Address. The LINK Address can be specified through the LREL bit as an absolute address or a relative address with respect to the memory address at which the RLINED command code is located.

This command can only be executed for a 16-bit/pixel destination.

When a polygon used in work reference is drawn by the FTRAPC (RFTRAPC) command, perform drawing with both the EDG and EOS bits set to 1.

The final coordinate point is stored as the current pointer values (XC, YC).

Notes: 1. 8-point drawing is used.
2. The final point of each line segment is not drawn. When antialiasing is performed for the exterior frame of a polygon drawn by a POLYGON4 type command, the paths may not match.
3. When the starting and final coordinate points of a line segment match, nothing is drawn.
4. Antialiasing is not performed for horizontal, vertical, and 45-degree diagonal line segments, which are pre-clipped inside the 2DG.
5. Clipping is performed on a pixel basis when either the referenced pixel or the pixel to be drawn is outside the clipping area, and antialiasing is not performed in such a case.
6. The final coordinate point is stored as the current pointer values (XC, YC).

(g) Example

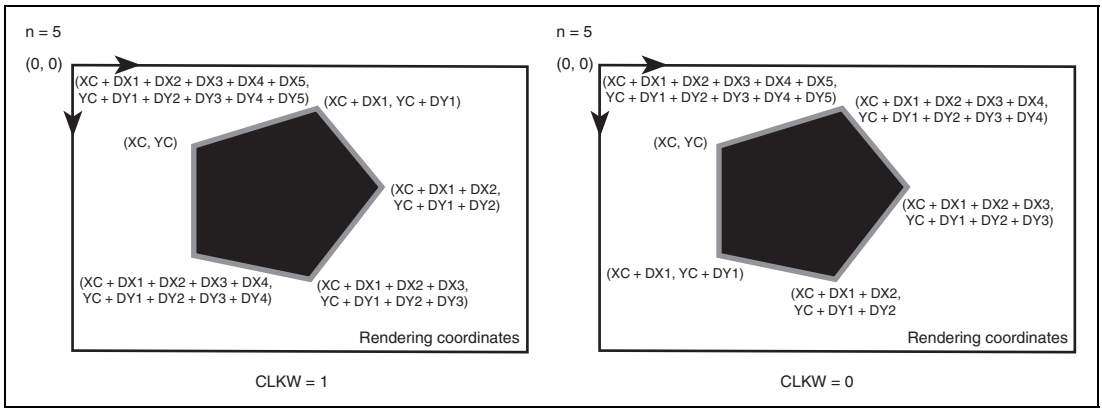


Figure 11.26 RLINED Command Example

11.4.3 Work Screen Drawing Commands

(1) FTRAPC

(a) Function

Draws a polygon at work coordinates.

(b) Command Format

LINKE = 0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1101_0000								Reserve (all 0)								Draw Mode															
Reserve (all 0)																n (2 ≤ n ≤ 65535)															
Sign	Xmin (-32768 ≤ Xmin ≤ 32767)															Sign	Ymin (-32768 ≤ Ymin ≤ 32767)														
Sign	Xmax (-32768 ≤ Xmax ≤ 32767)															Sign	Ymax (-32768 ≤ Ymax ≤ 32767)														
Sign	DX1 (-32768 ≤ DX1 ≤ 32767)															Sign	DY1 (-32768 ≤ DY1 ≤ 32767)														
Sign	:															Sign	:														
Sign	:															Sign	:														
Sign	DXn (-32768 ≤ DXn ≤ 32767)															Sign	DYn (-32768 ≤ DYn ≤ 32767)														

Note: When n = 0 or 1, correct operation is not guaranteed.

LINKE = 1 and LREL = 0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1101_0000								Reserve (all 0)								Draw Mode															
Reserve (all 0)																n (2 ≤ n ≤ 65535)															
Sign	Xmin (-32768 ≤ Xmin ≤ 32767)															Sign	Ymin (-32768 ≤ Ymin ≤ 32767)														
Sign	Xmax (-32768 ≤ Xmax ≤ 32767)															Sign	Ymax (-32768 ≤ Ymax ≤ 32767)														
0	0	0	LINK Address (longword address)																											0	0

Notes: 1. When n = 0 or 1, correct operation is not guaranteed.

2. The longword address following the LINK Address is handled as the next command code. Therefore, do not specify the longword address following the address where the LINK Address is to be assigned as the link destination address specified by the LINK Address.

LINKE = 1 and LREL = 1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1101_0000								Reserve (all 0)								Draw Mode															
Reserve (all 0)																n (2 ≤ n ≤ 65535)															
Sign	Xmin (-32768 ≤ Xmin ≤ 32767)															Sign	Ymin (-32768 ≤ Ymin ≤ 32767)														
Sign	Xmax (-32768 ≤ Xmax ≤ 32767)															Sign	Ymax (-32768 ≤ Ymax ≤ 32767)														
Sign extended	Sign	LINK Address (longword address)																												0	0

Notes: 1. When n = 0 or 1, correct operation is not guaranteed.

2. The longword address following the LINK Address is handled as the next command code. Therefore, do not specify the longword address following the address where the LINK Address is to be assigned as the link destination address specified by adding the address where the command code is located to the LINK Address.

(c) Code

B'11010000

(d) Rendering Attributes

Reference Data			Drawing Destination	
Multi-Valued Source	**Binary Source**	**Binary Work**	**Specified Color**	**Rendering**	**Work**
					O

Draw Mode

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Fixed to 0	Fixed to 0	CLIP	RCLIP	Fixed to 0	LINKE	LREL	Fixed to 0	Fixed to 0	Fixed to 0	BLKE (1)	EDG	EOS	Fixed to 0	Fixed to 0	Fixed to 0

Note: Set the BLKE bit to 1.

(e) Command Parameters

n (n = 2 to 65,535):	Number of vertices
Xmin:	Xmin value in the circumscribed quadrangle of the polygon. Work coordinate (absolute coordinate). Negative number expressed as two's complement.
Ymin:	Ymin value in the circumscribed quadrangle of the polygon. Work coordinate (absolute coordinate). Negative number expressed as two's complement.
Xmax:	Xmax value in the circumscribed quadrangle of the polygon. Work coordinate (absolute coordinate). Negative number expressed as two's complement.
Ymax:	Ymax value in the circumscribed quadrangle of the polygon. Work coordinate (absolute coordinate). Negative number expressed as two's complement.
DXn (n = 2 to 65,535):	Work coordinate (absolute coordinate). Negative number expressed as two's complement.
DYn (n = 2 to 65,535):	Work coordinate (absolute coordinate). Negative number expressed as two's complement.
LINK Address:	LINK absolute address (Longword address. Write 0 to bits A31 to A29, A1, and A0.) LINK relative address (Longword address. Negative number expressed as two's complement. Bits A31 to A29 are used to extend the sign in bit A28. Write 0 to bits A1 and A0.)

Note: Even in 32-bit addressing mode, write the values in bits 28 to 3 of the specified 32-bit address to bits A28 to A3.

(f) Description

Draws a polygon with n – 1 vertices at work coordinates. Paints n – 1 trapezoids at work coordinates using binary EOR, with X = Xmin as the left-hand side, and line segments (DX1, DY1) – (DX2, DY2), (DX2, DY2) – (DX3, DY3), ..., (DXn – 1, DYn – 1) – (DXn, DYn) as the right-hand sides, and with the top and bottom bases parallel to the X-axis. Bottom base drawing is not performed. Set (DXN, DYN) = (DX1, DY1) to give a closed figure. If the rendering attribute EDG bit is set to 1, an edge line is drawn after the paint operation. The line drawing data is selected with the EOS bit.

(g) Example

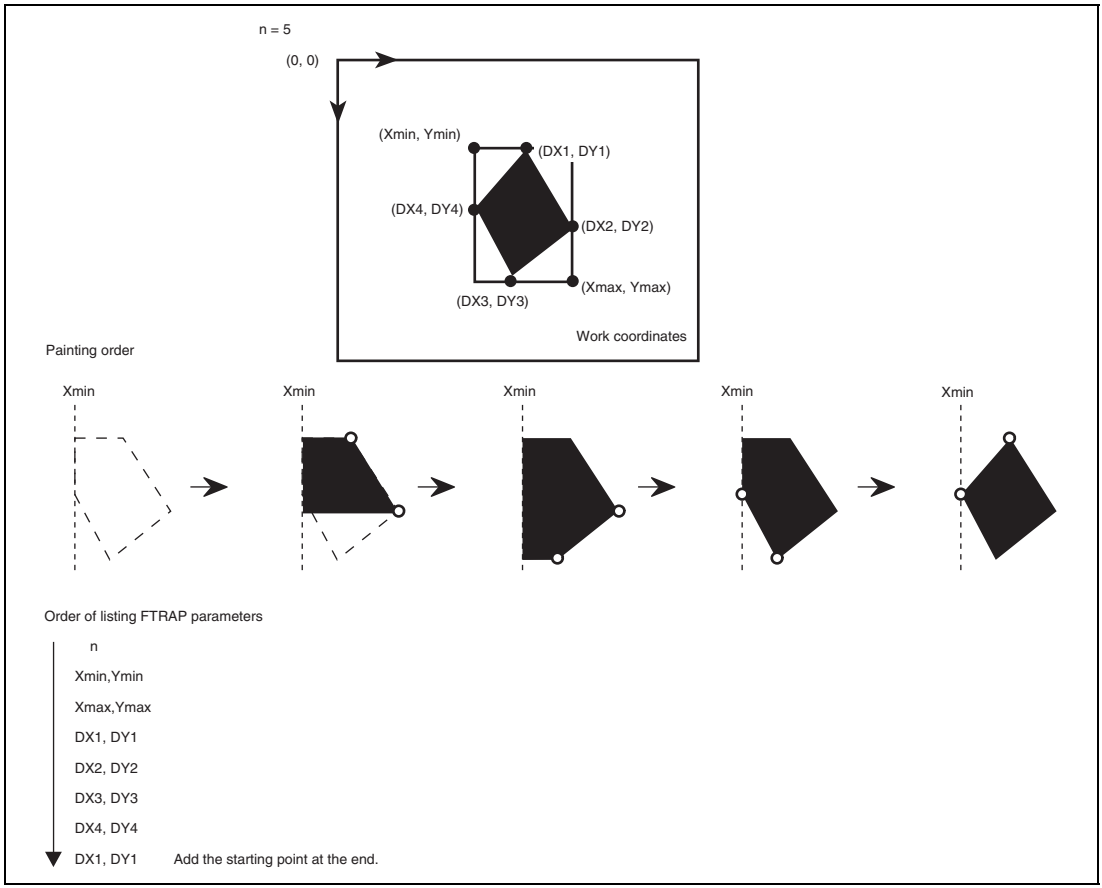


Figure 11.27 FTRAPC Command Example

(2) RFTRAPC

(a) Function

Draws a polygon at work coordinates with a relative coordinate specification from the current pointer value.

(b) Command Format

LINKE = 0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1101_0100								Reserve (all 0)								Draw Mode															
Reserve (all 0)																n (1 ≤ n ≤ 65535)															
Sign	Xmin (-32768 ≤ Xmin ≤ 32767)															Sign	Ymin (-32768 ≤ Ymin ≤ 32767)														
Sign	Xmax (-32768 ≤ Xmax ≤ 32767)															Sign	Ymax (-32768 ≤ Ymax ≤ 32767)														
Sign	DX2 (-128 ≤ DX2 ≤ 127)							Sign	DY2 (-128 ≤ DY2 ≤ 127)							Sign	DX1 (-128 ≤ DX1 ≤ 127)							Sign	DY1 (-128 ≤ DY1 ≤ 127)						
Sign	:							Sign	:							Sign	:							Sign	:						
Sign	:							Sign	:							Sign	:							Sign	:						
Sign	DXn (-128 ≤ DXn ≤ 127)							Sign	DYn (-128 ≤ DYn ≤ 127)							Sign	DXn-1 (-128 ≤ DXn-1 ≤ 127)							Sign	DYn-1 (-128 ≤ DYn-1 ≤ 127)						

Notes: 1. When n = 0, correct operation is not guaranteed.
2. When n is an odd number, insert a dummy word of 0 at the end.

LINKE = 1 and LREL = 0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1101_0100								Reserve (all 0)								Draw Mode															
Reserve (all 0)																n (1 ≤ n ≤ 65535)															
Sign	Xmin (-32768 ≤ Xmin ≤ 32767)															Sign	Ymin (-32768 ≤ Ymin ≤ 32767)														
Sign	Xmax (-32768 ≤ Xmax ≤ 32767)															Sign	Ymax (-32768 ≤ Ymax ≤ 32767)														
0	0	0	LINK Address (longword address)																											0	0

Notes: 1. When n = 0, correct operation is not guaranteed.
2. When n is an odd number, insert a dummy word of 0 at the end.
3. The longword address following the LINK Address is handled as the next command code. Therefore, do not specify the longword address following the address where the LINK Address is to be assigned as the link destination address specified by the LINK Address.

LINKE = 1 and LREL = 1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1101_0100								Reserve (all 0)								Draw Mode															
Reserve (all 0)																n (1 ≤ n ≤ 65535)															
Sign	Xmin (-32768 ≤ Xmin ≤ 32767)															Sign	Ymin (-32768 ≤ Ymin ≤ 32767)														
Sign	Xmax (-32768 ≤ Xmax ≤ 32767)															Sign	Ymax (-32768 ≤ Ymax ≤ 32767)														
Sign extended	Sign	LINK Address (longword address)																												0	0

Notes: 1. When n = 0, correct operation is not guaranteed.
2. When n is an odd number, insert a dummy word of 0 at the end.
3. The longword address following the LINK Address is handled as the next command code. Therefore, do not specify the longword address following the address where the LINK Address is to be assigned as the link destination address specified by the address where the command code is located plus the LINK Address.

(c) Code

B'11010100

(d) Rendering Attributes

Reference Data				Drawing Destination	
Multi-Valued Source	Binary Source	Binary Work	Specified Color	Rendering	Work
					O

Draw Mode

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Fixed to 0	Fixed to 0	CLIP	RCLIP	Fixed to 0	LINKE	LREL	Fixed to 0	Fixed to 0	Fixed to 0	BLKE (1)	EDG	EOS	Fixed to 0	Fixed to 0	Fixed to 0

Note: Set the BLKE bit to 1.

(e) Command Parameters

n (n = 1 to 65,535):	Number of vertices
Xmin:	Xmin value in the circumscribed quadrangle of the polygon. Work coordinate (absolute coordinate). Negative number expressed as two's complement.
Ymin:	Ymin value in the circumscribed quadrangle of the polygon. Work coordinate (absolute coordinate). Negative number expressed as two's complement.
Xmax:	Xmax value in the circumscribed quadrangle of the polygon. Work coordinate (absolute coordinate). Negative number expressed as two's complement.
Ymax:	Ymax value in the circumscribed quadrangle of the polygon. Work coordinate (absolute coordinate). Negative number expressed as two's complement.
DXn (n = 1 to 65,535):	Work coordinate (relative coordinate). Negative number expressed as two's complement.
DYn (n = 1 to 65,535):	Work coordinate (relative coordinate). Negative number expressed as two's complement.
LINK Address:	LINK absolute address (Longword address. Write 0 to bits A31 to A29, A1, and A0.) LINK relative address (Longword address. Negative number expressed as two's complement. Bits A31 to A29 are used to extend the sign in bit A28. Write 0 to bits A1 and A0.)

Note: Even in 32-bit addressing mode, write the values in bits 28 to 3 of the specified 32-bit address to bits A28 to A3.

(f) Description

Draws a polygon with n – 1 vertices at work coordinates. Paints n – 1 trapezoids at work coordinates using binary EOR, with X = Xmin as the left-hand side, and line segments specified by the relative shift (DX, DY) from the current pointer values (XC, YC) ((XC, YC) – (XC + DX1, YC + DY1), (XC + DX1, YC + DY1) – (XC + DX1 + DX2, YC + DY1 + DY2), ..., (XC + ... + DXn – 1, YC + ... + DYn – 1) – (XC + ... + DXn – 1 + DXn, YC + ...+ DYn – 1 + DYn)) as the right-hand sides, and with the top and bottom bases parallel to the X-axis. Bottom base drawing is not performed.

The final coordinate point is stored as the current pointer values (XC, YC). Set (DX1 + DX2 + ...+ DXn = 0, DY1 + DY2 + ... + DYn = 0) to give a closed figure. If the rendering attribute EDG bit

is set to 1, an edge line is drawn after the paint operation. The line drawing data is selected with the EOS bit.

Notes: The final coordinate point is stored as the current pointer values (XC, YC).

(g) Example

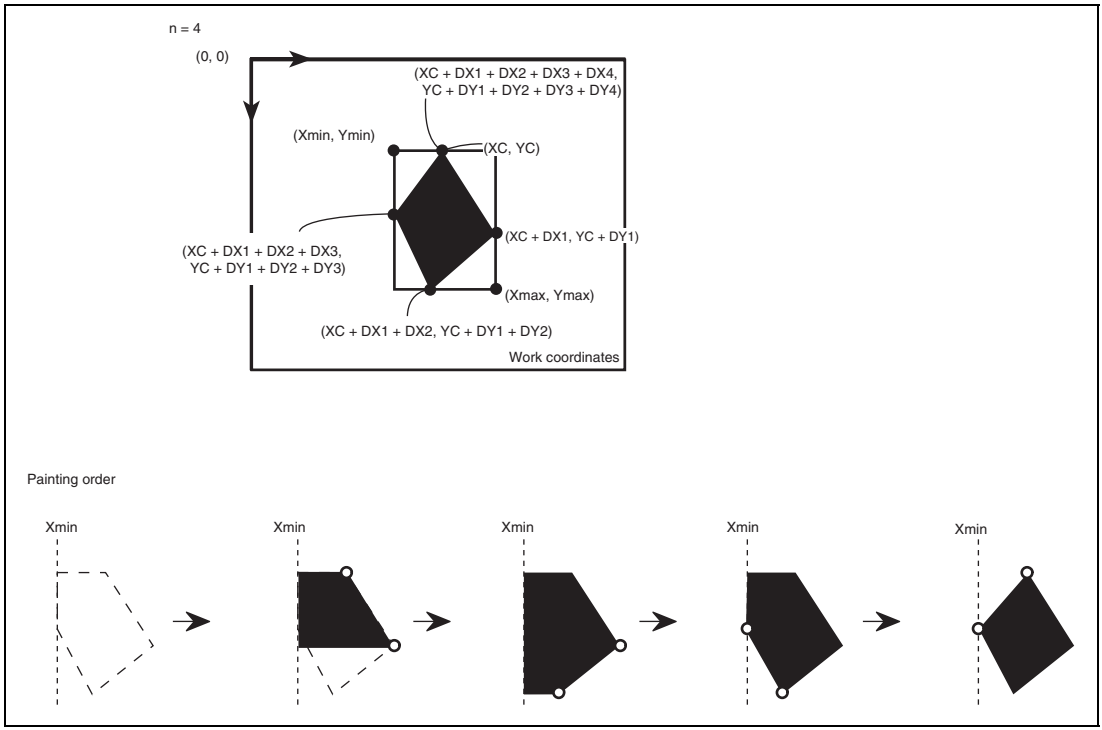


Figure 11.28 RFTRAPC Command Example

(3) CLRWC

(a) Function

Clear the work coordinates to 0.

(b) Command Format

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1110_0000								Reserve (all 0)								Draw Mode															
Sign	Xmin (-32768 ≤ Xmin ≤ 32767)															Sign	Ymin (-32768 ≤ Ymin ≤ 32767)														
Sign	Xmax (-32768 ≤ Xmax ≤ 32767)															Sign	Ymax (-32768 ≤ Ymax ≤ 32767)														

(c) Code

B'11100000

(d) Rendering Attributes

Reference Data			Drawing Destination	
Multi-Valued Source	Binary Source	Binary Work	Specified Color	Rendering	Work
					O

Draw Mode

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Fixed to 0	Fixed to 0	CLIP	RCLIP	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	BLKE (1)	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0

Note: Set the BLKE bit to 1.

(e) Command Parameters

Xmin, Xmax: Left and right X coordinate values. Work coordinates (absolute coordinates. Negative numbers expressed as two's complement.

Ymin, Ymax: Upper and lower Y coordinate values. Work coordinates (absolute coordinates. Negative numbers expressed as two's complement.

(f) Description

Zero-clears the area specified by upper-left coordinates (Xmin, Ymin) and lower-right coordinates (Xmax, Ymax) at work coordinates.

(g) Example

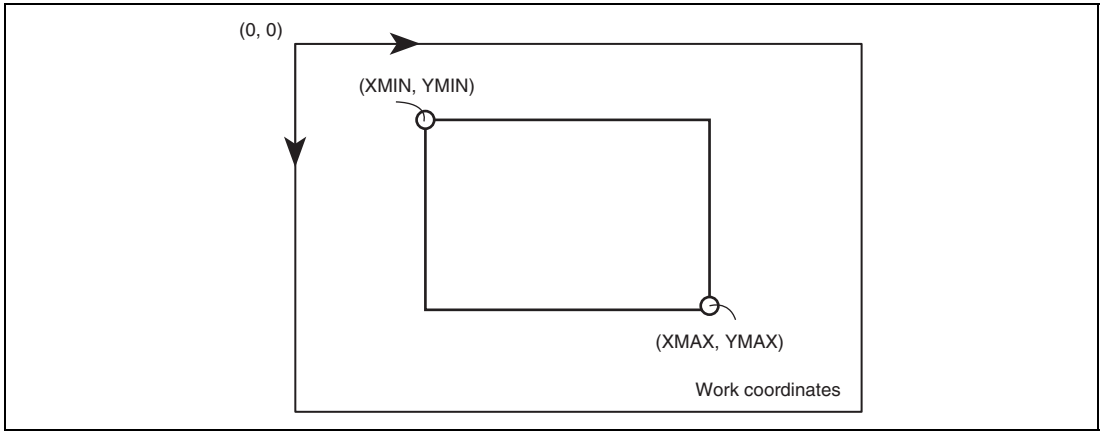


Figure 11.29 CLRWC Command Example

11.4.4 Work Line Drawing Commands

(1) LINEWC

(a) Function

Draws a polygon at work coordinates.

(b) Command Format

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
OP CODE = 1111_0000	Reserve (all 0)	Draw Mode	
Reserve (all 0)		n (2 ≤ n ≤ 65535)	
Sign	DX1 (-32768 ≤ DX1 ≤ 32767)	Sign	DY1 (-32768 ≤ DY1 ≤ 32767)
Sign	:	Sign	:
Sign	:	Sign	:
Sign	DXn (-32768 ≤ DXn ≤ 32767)	Sign	DYn (-32768 ≤ DYn ≤ 32767)

Note: When n = 0 or 1, correct operation is not guaranteed.

(c) Code

B'11110000

(d) Rendering Attributes

Reference Data			Drawing Destination		
Multi-Valued Source	**Binary Source**	**Binary Work**	**Specified Color**	**Rendering**	**Work**
			O (EOS of binary work)		O

Draw Mode

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Fixed to 0	Fixed to 0	CLIP	RCLIP	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	EOS	Fixed to 0	Fixed to 0	Fixed to 0

(e) Command Parameters

n (n = 2 to 65,535): Number of vertices

DXn (n = 2 to 65,535): Work coordinate (absolute coordinate). Negative number expressed as two's complement.

DYn (n = 2 to 65,535): Work coordinate (absolute coordinate). Negative number expressed as two's complement.

(f) Description

Performs binary drawing at work coordinates of a polygonal line from vertex 1 (DX1, DY1), through vertex 2 (DX2, DY2), ..., vertex n – 1 (DXn – 1, DYn – 1), to vertex n (DXn, DYn). 0 drawing or 1 drawing is selected with the drawing mode EOS bit. Drawing is performed at work coordinates with 0 when EOS = 0, and at work coordinates with 1 when EOS = 1 (Used for edge drawing at work coordinates for a polygonal painted figure).

Note: 8-point drawing is used. The final point of each line segment is drawn.

(g) Example

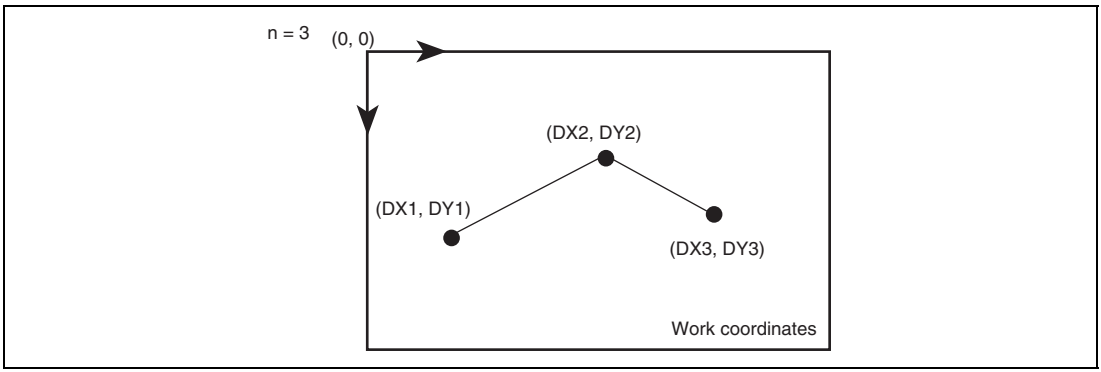


Figure 11.30 LINEWC Command Example

(2) RLINEWC

(a) Function

Draws a 1-bit-wide solid line at work coordinates with a relative coordinate specification from the current pointer value.

(b) Command Format

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
OP CODE = 1111_0100	Reserve (all 0)	Draw Mode	
Reserve (all 0)		n (1 ≤ n ≤ 65535)	
Sign DX2 (-128 ≤ DX2 ≤ 127)	Sign DY2 (-128 ≤ DY2 ≤ 127)	Sign DX1 (-128 ≤ DX1 ≤ 127)	Sign DY1 (-128 ≤ DY1 ≤ 127)
Sign :	Sign :	Sign :	Sign :
Sign :	Sign :	Sign :	Sign :
Sign DXn (-128 ≤ DXn ≤ 127)	Sign DYn (-128 ≤ DYn ≤ 127)	Sign DXn-1 (-128 ≤ DXn-1 ≤ 127)	Sign DYn-1 (-128 ≤ DYn-1 ≤ 127)

Notes: 1. When n = 0, correct operation is not guaranteed.
2. When n is an odd number, insert a dummy word of 0 at the end.

(c) Code

B'11110100

(d) Rendering Attributes

Reference Data				Drawing Destination	
Multi-Valued Source	Binary Source	Binary Work	Specified Color	Rendering	Work
			O (EOS of binary work)		O

Draw Mode

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Fixed to 0	Fixed to 0	CLIP	RCLIP	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	EOS	Fixed to 0	Fixed to 0	Fixed to 0

(e) Command Parameters

n (n = 1 to 65,535): Number of vertices

DXn (n = 1 to 65,535): Work coordinate (relative coordinate). Negative number expressed as two's complement.

DYn (n = 1 to 65,535): Work coordinate (relative coordinate). Negative number expressed as two's complement.

(f) Description

Performs binary drawing at work coordinates of a polygonal line comprising line segments (XC, YC) – (XC + DX1, YC + DY1), (XC + DX1, YC + DY1) – (XC + DX1 + DX2, YC + DY1 + DY2), ..., (XC + ... + DXn – 1, YC + ... + DYn – 1) – (XC + ... + DXn – 1 + DXn, YC + ... + DYn – 1 + DYn) to the coordinates specified by the relative shift (DX, DY) from the current pointer values (XC, YC). 0 drawing or 1 drawing is selected with the drawing mode EOS bit. Drawing is performed at work coordinates with 0 when EOS = 0, and at work coordinates with 1 when EOS = 1. (Used for edge drawing at work coordinates for a polygonal painted figure.)

The final coordinate point is stored as the current pointer values (XC, YC).

Notes: 1. 8-point drawing is used. The end of a line is drawn.
2. The final coordinate point is stored as the current pointer values (XC, YC).

(g) Example

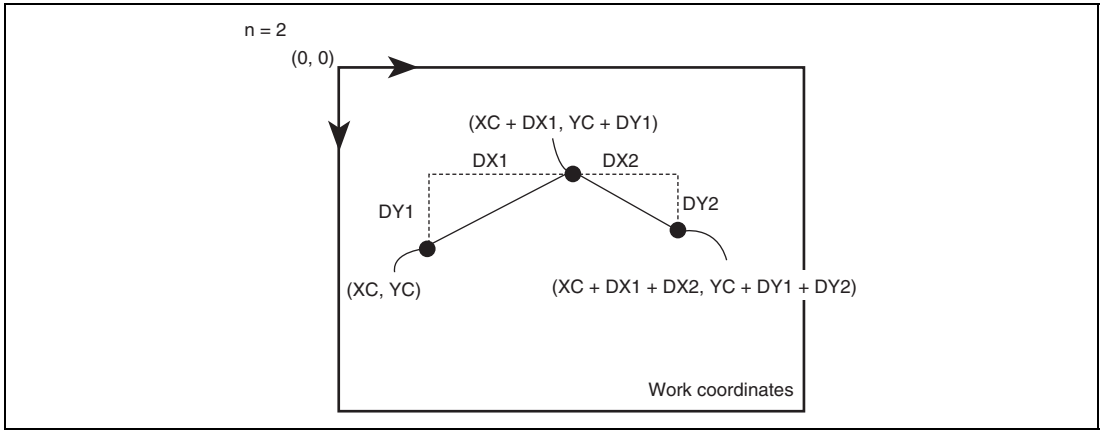


Figure 11.31 RLINEWC Command Example

11.4.5 Rectangle Drawing Commands

(1) BITBLTA

(a) Function

Transfers multi-valued (8- or 16-bit/pixel) rectangle source data to the destination area.

(b) Command Format

SS = 1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1010_0010								Reserve (all 0)								Draw Mode															
Reserve (all 0)																0	0	0	0	0	0	0	0	ROP							
0	0	0	0	TXS (0 ≤ TXS ≤ 4088)												0	0	0	0	TYS (0 ≤ TYS ≤ 4095)											
0	0	0	0	LW (0 ≤ LW ≤ 4094)												0	0	0	0	RW (0 ≤ RW ≤ 4094)											
0	0	0	0	TH (0 ≤ TH ≤ 4094)												0	0	0	0	BH (0 ≤ BH ≤ 4094)											
Sign	BXC (-32768 ≤ BXC ≤ 32767)															Sign	BYC (-32768 ≤ BYC ≤ 32767)														

Notes: 1. 0 ≤ TXS ≤ SSTRR – (LW + RW + 1), 0 ≤ TYS ≤ 4096 – (TH + BH + 1) (SSTRR: Source stride register setting)
2. 8 ≤ LW + RW + 1 ≤ 4095, 1 ≤ TH + BH + 1 ≤ 4095
3. –32768 ≤ BXC – LW ≤ 32767, –32768 ≤ BYC – TH ≤ 32767, –32768 ≤ BXC + RW ≤ 32767, –32768 ≤ BYC + BH ≤ 32767

SS = 0 and REL = 0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1010_0010								Reserve (all 0)								Draw Mode															
Reserve (all 0)																0	0	0	0	0	0	0	0	ROP							
0	0	0	Base Address (quad word address)																										0	0	0
0	0	0	0	LW (0 ≤ LW ≤ 4087)												0	0	0	0	RW (0 ≤ RW ≤ 4087)											
0	0	0	0	TH (0 ≤ TH ≤ 4094)												0	0	0	0	BH (0 ≤ BH ≤ 4094)											
Sign	BXC (-32768 ≤ BXC ≤ 32767)															Sign	BYC (-32768 ≤ BYC ≤ 32767)														

Notes: 1. 8 ≤ LW + RW + 1 ≤ 4088 (multiple of 8), 1 ≤ TH + BH + 1 ≤ 4095
2. –32768 ≤ BXC – LW ≤ 32767, –32768 ≤ BYC – TH ≤ 32767, –32768 ≤ BXC + RW ≤ 32767, –32768 ≤ BYC + BH ≤ 32767

SS = 0 and REL = 1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1010_0010								Reserve (all 0)								Draw Mode															
Reserve (all 0)																0	0	0	0	0	0	0	0	ROP							
Sign extended			Sign	Base Address (longword address)																										0	0
0	0	0	0	LW (0 ≤ LW ≤ 4087)												0	0	0	0	RW (0 ≤ RW ≤ 4087)										
0	0	0	0	TH (0 ≤ TH ≤ 4094)												0	0	0	0	BH (0 ≤ BH ≤ 4094)										
Sign	BXC (-32768 ≤ BXC ≤ 32767)															Sign	BYC (-32768 ≤ BYC ≤ 32767)														

Notes: 1. 8 ≤ LW + RW + 1 ≤ 4088 (multiple of 8), 1 ≤ TH + BH + 1 ≤ 4095

2. –32768 ≤ BXC – LW ≤ 32767, –32768 ≤ BYC – TH ≤ 32767, –32768 ≤ BXC + RW ≤ 32767, –32768 ≤ BYC + BH ≤ 32767

3. Adding the address (longword: 32-bit units) where the command code is located to the Base Address (longword: 32-bit units) must result in a quad word address (64-bit units).

(c) Code

B'10100010

(d) Rendering Attributes

Reference Data			Drawing Destination	
Multi-Valued Source	**Binary Source**	**Binary Work**	**Specified Color**	**Rendering**	**Work**
O		O (only WORK = 1)		O	

Draw Mode

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Fixed to 0	Fixed to 0	CLIP	RCLIP	STRANS	DTRANS	WORK	SS	REL	SRCDIRX	SRCDIRY	DSTDIRX	DSTDIRY	COOF	αE	SαE

(e) Command Parameters

TXS, TYS:	Source starting point. Write 0 to the unused bits.
Base Address:	Source start absolute address (Quad word address. Write 0 to bits A31 to A29 and A2 to A0.) Source start relative address (Longword address. Negative number expressed as two's complement. Bits A31 to A29 are used to extend the sign in bit A28. Write 0 to bits A1 and A0.)

Note: Even in 32-bit addressing mode, write the values in bits 28 to 3 of the specified 32-bit address to bits A28 to A3.

BXC, BYC:	Center X and Y coordinate values. Rendering coordinates (absolute coordinates). Negative numbers expressed as two's complement.
LW, RW:	Left and right widths. Relative value from (BXC, BYC). Rendering coordinates. Make the setting in pixel units. Write 0 to the unused bits.
TH, BH:	Top and bottom heights. Relative value from (BXC, BYC). Rendering coordinates. Make the setting in pixel units. Write 0 to the unused bits.
ROP:	Raster operation code

(f) Description

Transfers multi-valued (8- or 16-bit/pixel) rectangle source data to rendering coordinates.

When SS = 0, set the (LW + RW + 1) value to be a multiple of 8 pixels. When SS = 1, set the (LW + RW + 1) value to be 8 or more pixels.

1. When work specification is selected as a rendering attribute (WORK = 1), only places where the work coordinate pixel is 1 are drawn at rendering coordinates while referencing work coordinates for the same coordinates as the rendering coordinates.
2. When SS = 1, the source data is referenced from the 2-dimensional source area. When SS = 0, the source data is referenced from the Base Address in the display list. When REL = 0, the source address can be specified as an absolute address. When REL = 1, the source address can be specified as a relative address with respect to the memory address at which the BITBLTA command code is located.
3. In 16-bit/pixel drawing, if the rendering attribute COOF bit is set to 1, the result of adding the value in COFSR to the value of the multi-valued source data is drawn. The operation is performed by saturation processing. In 8-bit/pixel drawing, the COOF bit should be cleared to 0.

4. The direction to reference the source data can be selected by the SRCDIRX and SRCDIRY bits.
5. The drawing direction can be selected by the DSTDIRX and DSTDIRY bits.
6. When αE = 1, the source data and ground data are alpha blended before drawing. When setting αE = 1, also set the ROP code = H'CC (source copy). The A value in the ARGB format is not alpha blended. The A value is drawn according to the source A value use (SAU) and A value (AVALUE) bits in the rendering control register (RCLR). Alpha blending is valid only in 16-bit/pixel drawing.
7. 16 raster operations are possible. The A value in the ARGB format is not subject to raster operations. The A value is drawn according to the source A value use (SAU) and A value (AVALUE) bits in the rendering control register (RCLR).

Note: System clipping or (relative) user clipping is performed when drawing a rectangle.

(g) Example

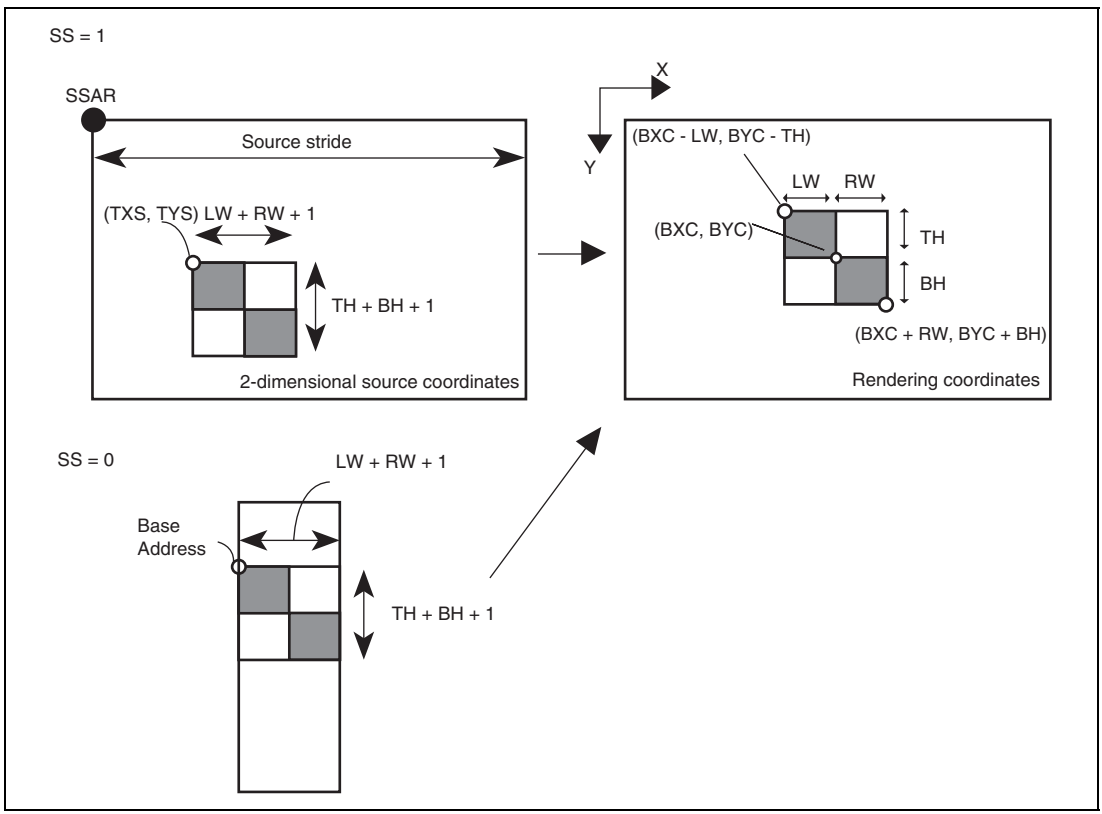


Figure 11.32 BITBLTA Command Example

(2) BITBLTB

(a) Function

Transfers binary (1-bit/pixel) rectangle source data that has been color expanded to the destination area.

(b) Command Format

SS = 1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1010_0001								Reserve (all 0)								Draw Mode															
Reserve (all 0)																0	0	0	0	0	0	0	0	ROP							
Color1																Color0															
0	0	0	0	TXS (0 ≤ TXS ≤ 4088)												0	0	0	0	TYS (0 ≤ TYS ≤ 4095)											
0	0	0	0	LW (0 ≤ LW ≤ 4087)												0	0	0	0	RW (0 ≤ RW ≤ 4087)											
0	0	0	0	TH (0 ≤ TH ≤ 4094)												0	0	0	0	BH (0 ≤ BH ≤ 4094)											
Sign	BXC (-32768 ≤ BXC ≤ 32767)															Sign	BYC (-32768 ≤ BYC ≤ 32767)														

Notes: 1. 0 ≤ TXS ≤ SSTRR – (LW + RW + 1), 0 ≤ TYS ≤ 4096 – (TH + BH + 1) (SSTRR: Source stride register setting)
2. 8 ≤ LW + RW + 1 ≤ 4088 (multiple of 8), 1 ≤ TH + BH + 1 ≤ 4095
3. –32768 ≤ BXC – LW ≤ 32767, –32768 ≤ BYC – TH ≤ 32767, –32768 ≤ BXC + RW ≤ 32767, –32768 ≤ BYC + BH ≤ 32767

SS = 0 and REL = 0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1010_0001								Reserve (all 0)								Draw Mode															
Reserve (all 0)																0	0	0	0	0	0	0	0	ROP							
Color1																Color0															
0	0	0	Base Address (quad word address)																										0	0	0
0	0	0	0	LW (0 ≤ LW ≤ 4087)												0	0	0	0	RW (0 ≤ RW ≤ 4087)											
0	0	0	0	TH (0 ≤ TH ≤ 4094)												0	0	0	0	BH (0 ≤ BH ≤ 4094)											
Sign	BXC (-32768 ≤ BXC ≤ 32767)															Sign	BYC (-32768 ≤ BYC ≤ 32767)														

Notes: 1. 8 ≤ LW + RW + 1 ≤ 4088 (multiple of 8), 1 ≤ TH + BH + 1 ≤ 4095
2. –32768 ≤ BXC – LW ≤ 32767, –32768 ≤ BYC – TH ≤ 32767, –32768 ≤ BXC + RW ≤ 32767, –32768 ≤ BYC + BH ≤ 32767

SS = 0 and REL = 1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1010_0001								Reserve (all 0)								Draw Mode															
Reserve (all 0)																0	0	0	0	0	0	0	0	ROP							
Color1																Color0															
Sign extended			Sign	Base Address (longword address)																										0	0
0	0	0	0	LW (0 ≤ LW ≤ 4087)												0	0	0	0	RW (0 ≤ RW ≤ 4087)											
0	0	0	0	TH (0 ≤ TH ≤ 4094)												0	0	0	0	BH (0 ≤ BH ≤ 4094)											
Sign	BXC (-32768 ≤ BXC ≤ 32767)															Sign	BYC (-32768 ≤ BYC ≤ 32767)														

Notes: 1. 8 ≤ LW + RW + 1 ≤ 4088 (multiple of 8), 1 ≤ TH + BH + 1 ≤ 4095

2. –32768 ≤ BXC – LW ≤ 32767, –32768 ≤ BYC – TH ≤ 32767, –32768 ≤ BXC + RW ≤ 32767, –32768 ≤ BYC + BH ≤ 32767

3. Adding the address (longword: 32-bit units) where the command code is located to the Base Address (longword: 32-bit units) must result in a quad word address (64-bit units).

(c) Code

B'10100001

(d) Rendering Attributes

Reference Data			Drawing Destination	
Multi-Valued Source	**Binary Source**	**Binary Work**	**Specified Color**	**Rendering**	**Work**
	O	O (only WORK = 1)		O	

Draw Mode

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Fixed to 0	Fixed to 0	CLIP	RCLIP	STRANS	DTRANS	WORK	SS	REL	SRCDIRX	SRCDIRY	DSTDIRX	DSTDIRY	COOF	αE	Fixed to 0

(e) Command Parameters

TXS, TYS:	Source starting point. Write 0 to the unused bits.
Base Address:	Source start absolute address (Quad word address. Write 0 to bits A31 to A29 and A2 to A0.) Source start relative address (Longword address. Negative number expressed as two's complement. Bits A31 to A29 are used to extend the sign in bit A28. Write 0 to bits A1 and A0.)

Note: Even in 32-bit addressing mode, write the values in bits 28 to 3 of the specified 32-bit address to bits A28 to A3.

BXC, BYC:	Center X and Y coordinate values. Rendering coordinates (absolute coordinates). Negative numbers expressed as two's complement.
LW, RW:	Left and right widths. Relative value from (BXC, BYC). Rendering coordinates. Make the setting in pixel units. Write 0 to the unused bits.
TH, BH:	Top and bottom heights. Relative value from (BXC, BYC). Rendering coordinates. Make the setting in pixel units. Write 0 to the unused bits.
ROP:	Raster operation code
Color0, Color1:	8- or 16-bit/pixel color specification. For 16-bit/pixel drawing, the color specification should match the destination pixel format. For 8-bit/pixel drawing, the same value should be set in the upper and lower bytes.

(f) Description

Transfers binary (1-bit/pixel) rectangle source data to rendering coordinates.

A multiple of 8 pixels must be set as the (LW + RW + 1) value, regardless of the SS bit value.

1. When work specification is selected as a rendering attribute (WORK = 1), only places where the work coordinate pixel is 1 are drawn at rendering coordinates while referencing work coordinates for the same coordinates as the rendering coordinates.
2. The binary source data is arranged in memory in linear fashion. When REL = 0, the source address can be specified as an absolute address. When REL = 1, the source address can be specified as a relative address with respect to the memory address at which the BITBLTB command code is located.
3. In 16-bit/pixel drawing, if the rendering attribute COOF bit is set to 1, the result of adding the value in COFSR to the value of the binary source data that has been color expanded is drawn.

The operation is performed by saturation processing. In 8-bit/pixel drawing, the COOF bit should be cleared to 0.

4. The direction to reference the source data can be selected by the SRCDIRX and SRCDIRY bits.
5. The drawing direction can be selected by the DSTDIRX and DSTDIRY bits.
6. When αE = 1, the data obtained by color expanding the binary source data and the ground data are alpha blended before drawing. When setting αE = 1, also set the ROP code = H'CC (source copy). The A value in the ARGB format is not alpha blended. The A value is drawn according to the source A value use (SAU) and A value (AVALUE) bits in the rendering control register (RCLR). Alpha blending is valid only in 16-bit/pixel drawing.
7. 16 raster operations are possible. The A value in the ARGB format is not subject to raster operations. The A value is drawn according to the source A value use (SAU) and A value (AVALUE) bits in the rendering control register (RCLR).

Note: System clipping or (relative) user clipping is performed when drawing a rectangle.

(g) Example

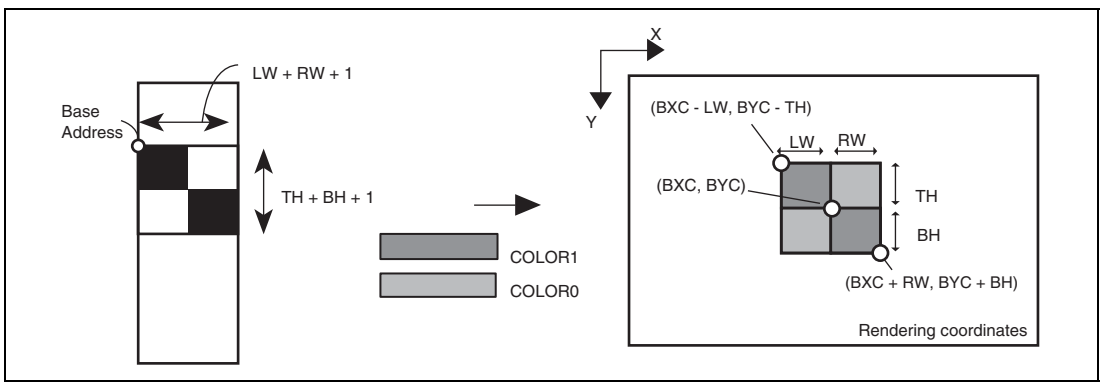


Figure 11.33 BITBLTB Command Example

(3) BITBLTC

(a) Function

Draws a rectangle with a monochrome specification to the destination area.

(b) Command Format

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 1010_0000								Reserve (all 0)								Draw Mode															
Reserve (all 0)																0	0	0	0	0	0	0	0	ROP							
Reserve (all 0)																Color															
0	0	0	0	LW (0 ≤ LW ≤ 4094)												0	0	0	0	RW (0 ≤ RW ≤ 4094)											
0	0	0	0	TH (0 ≤ TH ≤ 4094)												0	0	0	0	BH (0 ≤ BH ≤ 4094)											
Sign	BXC (-32768 ≤ BXC ≤ 32767)															Sign	BYC (-32768 ≤ BYC ≤ 32767)														

Notes: 1. 1 ≤ LW + RW + 1 ≤ 4095, 1 ≤ TH + BH + 1 ≤ 4095

2. –32768 ≤ BXC – LW ≤ 32767, –32768 ≤ BYC – TH ≤ 32767, –32768 ≤ BXC + RW ≤ 32767, –32768 ≤ BYC + BH ≤ 32767

(c) Code

B'10100000

(d) Rendering Attributes

Reference Data				Drawing Destination	
Multi-Valued Source	**Binary Source**	**Binary Work**	**Specified Color**	**Rendering**	**Work**
		O (only WORK = 1)	O	O	

Draw Mode

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Fixed to 0	Fixed to 0	CLIP	RCLIP	Fixed to 0	DTRANS	WORK	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	DSTDIRX	DSTDIRY	COOF	αE	Fixed to 0

(e) Command Parameters

BXC, BYC:	Center X and Y coordinate values. Rendering coordinates (absolute coordinates). Negative numbers expressed as two's complement.
LW, RW:	Left and right widths. Relative value from (BXC, BYC). Rendering coordinates. Make the setting in pixel units. Write 0 to the unused bits.
TH, BH:	Top and bottom heights. Relative value from (BXC, BYC). Rendering coordinates. Make the setting in pixel units. Write 0 to the unused bits.
Color:	8- or 16-bit/pixel color specification. For 16-bit/pixel drawing, the color specification should match the destination pixel format. For 8-bit/pixel drawing, the same value should be set in the upper and lower bytes.
ROP:	Raster operation code

(f) Description

Draws a rectangle in the destination area in the single color specified by the Color parameter.

1. When work specification is selected as a rendering attribute (WORK = 1), only places where the work coordinate pixel is 1 are drawn at rendering coordinates while referencing work coordinates for the same coordinates as the rendering coordinates.
2. In 16-bit/pixel drawing, if the rendering attribute COOF bit is set to 1, the result of adding the value in COFSR to the value of the specified color is drawn. The operation is performed by saturation processing. In 8-bit/pixel drawing, the COOF bit should be cleared to 0.
3. The drawing direction can be selected by the DSTDIRX and DSTDIRY bits.
4. When αE = 1, the specified color data and ground data are alpha blended before drawing. When setting αE = 1, also set the ROP code = H'CC (source copy). The A value in the ARGB format is not alpha blended. The A value is drawn according to the source A value use (SAU) and A value (AVALUE) bits in the rendering control register (RCLR). Alpha blending is valid only in 16-bit/pixel drawing.
5. 16 raster operations are possible. The A value in the ARGB format is not subject to raster operations. The A value is drawn according to the source A value use (SAU) and A value (AVALUE) bits in the rendering control register (RCLR).

Note: System clipping or (relative) user clipping is performed when drawing a rectangle.

(g) Example

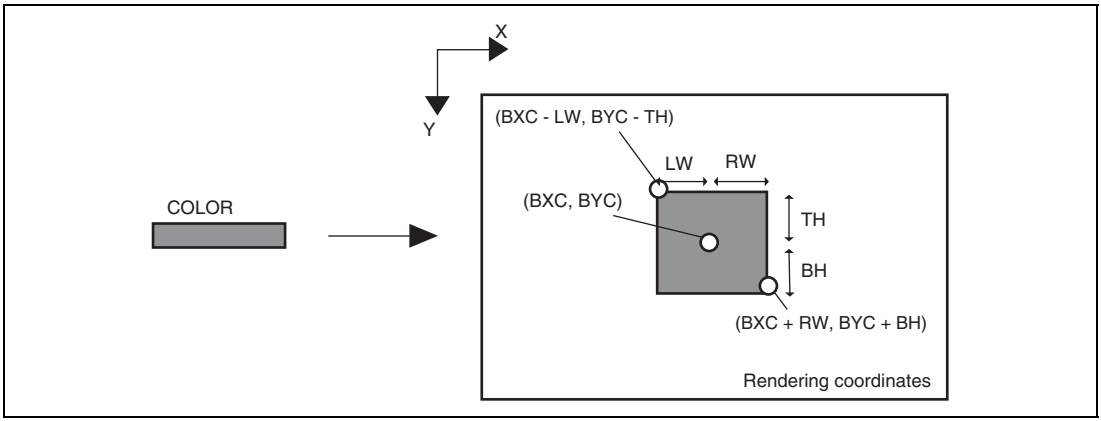


Figure 11.34 BITBLTC Command Example

11.4.6 Control Commands

(1) MOVE

(a) Function

Sets the current pointer.

(b) Command Format

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 0100_1000								Reserve (all 0)								Draw Mode															
XC (-32768 ≤ XC ≤ 32767)																YC (-32768 ≤ YC ≤ 32767)															

(c) Code

B'01001000

(d) Rendering Attributes

Draw Mode

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0

(e) Command Parameters

XC:	Rendering coordinate (absolute coordinate) or work coordinate (absolute coordinate). Negative number expressed as two's complement.
YC:	Rendering coordinate (absolute coordinate) or work coordinate (absolute coordinate). Negative number expressed as two's complement.

(f) Description

Sets the values obtained by adding the local offset values to XC and YC in the current pointers. XC and YC are set as absolute coordinates. The current pointers are used by relative drawing commands only. After issuing a MOVE command, use relative drawing commands in succession. If an absolute drawing command is used during this sequence, the current pointers will be used as registers for internal computation, and the current pointer values will be lost. A MOVE command must be therefore be issued before using relative drawing commands again.

(g) Example

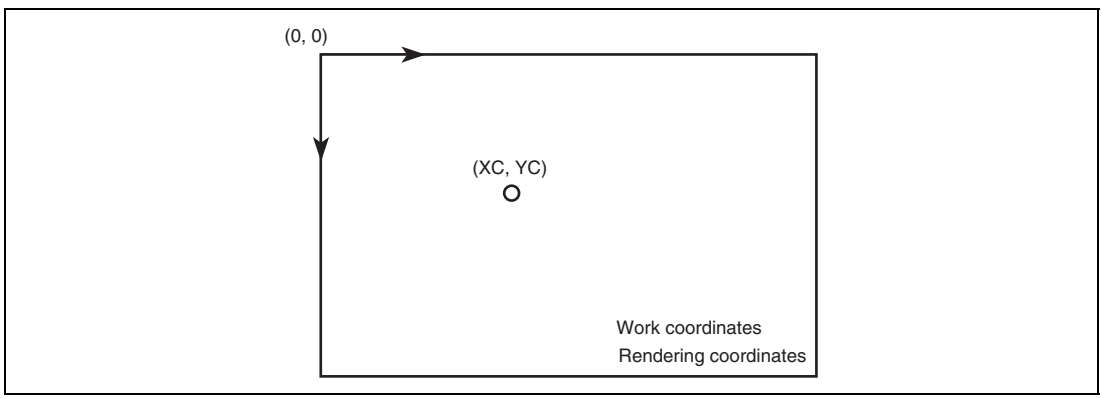


Figure 11.35 MOVE Command Example

(2) RMOVE

(a) Function

Adds XC and YC to the current pointers.

(b) Command Format

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 0100_1100								Reserve (all 0)								Draw Mode															
XC (-32768 ≤ XC ≤ 32767)																YC (-32768 ≤ YC ≤ 32767)															

(c) Code

B'01001100

(d) Rendering Attributes

Draw Mode

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0

(e) Command Parameters

XC: Rendering coordinate (relative coordinate) or work coordinate (relative coordinate). Negative number expressed as two's complement.

YC: Rendering coordinate (relative coordinate) or work coordinate (relative coordinate). Negative number expressed as two's complement.

(f) Description

Adds XC and YC to the current pointers.

(g) Example

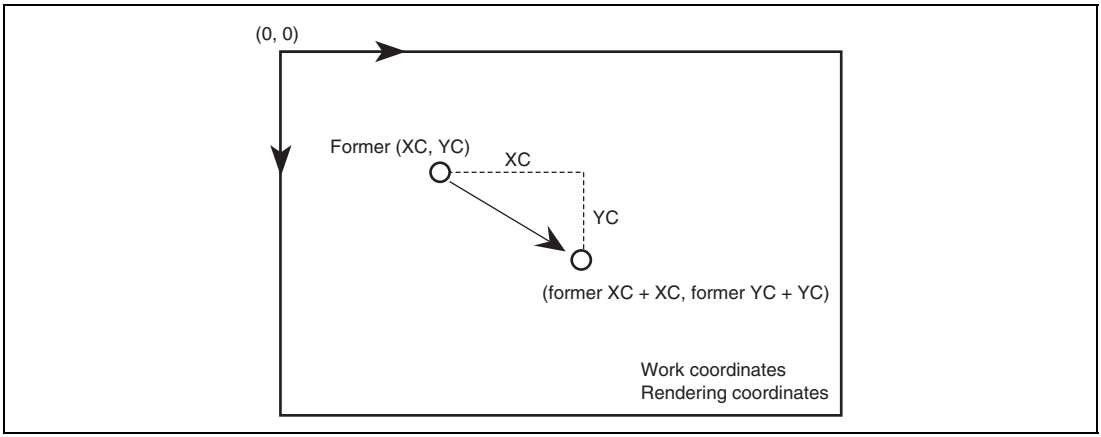


Figure 11.36 RMOVE Command Example

(3) LCOFS

(a) Function

Sets the offset values (local offset) of the destination area and work area.

(b) Command Format

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 0100_0000								Reserve (all 0)								Draw Mode															
XO (-32768 ≤ XO ≤ 32767)																YO (-32768 ≤ YO ≤ 32767)															

(c) Code

B'01000000

(d) Rendering Attributes

Draw Mode

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0

(e) Command Parameters

XO: Local offset value. Rendering coordinate (absolute coordinate) or work coordinate (absolute coordinate). Negative number expressed as two's complement.

YO: Local offset value. Rendering coordinate (absolute coordinate) or work coordinate (absolute coordinate). Negative number expressed as two's complement.

(f) Description

After the local offset values are set, these offset values are added in all subsequent coordinate specifications made in drawing commands.

These settings must be made at the start of the display list (the initial values are undefined).

To reflect the local offset values in the current pointers, issue a MOVE command after the LCOFS command.

(g) Example

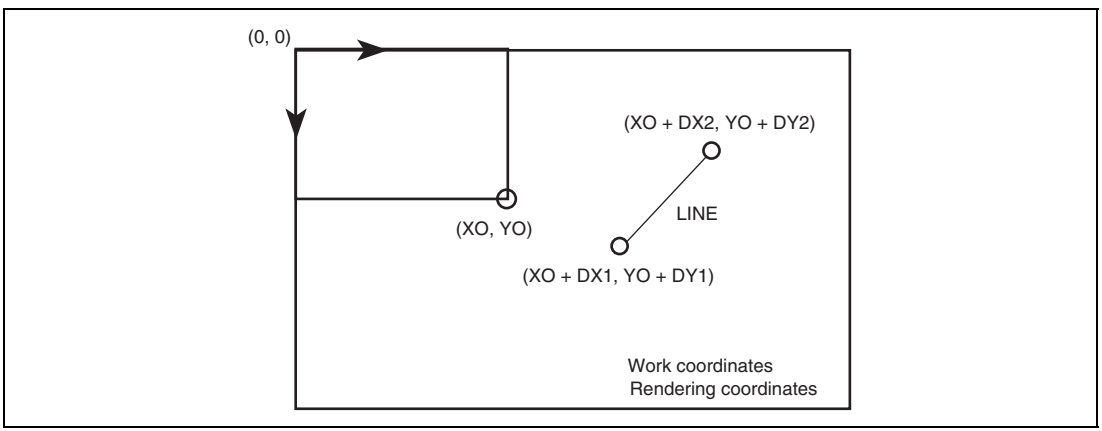


Figure 11.37 LCOFS Command Example

(4) RLCOFS

(a) Function

Adds XO and YO to the local offset.

(b) Command Format

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
OP CODE = 0100_0100	Reserve (all 0)	Draw Mode
XO (-32768 ≤ XO ≤ 32767)		YO (-32768 ≤ YO ≤ 32767)

(c) Code

B'01000100

(d) Rendering Attributes

Draw Mode

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0

(e) Command Parameters

XO: Local offset value. Rendering coordinate (relative coordinate) or work coordinate (relative coordinate). Negative number expressed as two's complement.

YO: Local offset value. Rendering coordinate (relative coordinate) or work coordinate (relative coordinate). Negative number expressed as two's complement.

(f) Description

Adding X0 and Y0 to the local offset makes the local offset values. After the local offset values are set, these offset values are added in all subsequent coordinate specifications made in drawing commands.

To reflect the local offset values in the current pointers, issue a MOVE command after setting the local offset with the LCOFS or RLCOFS command.

(g) Example

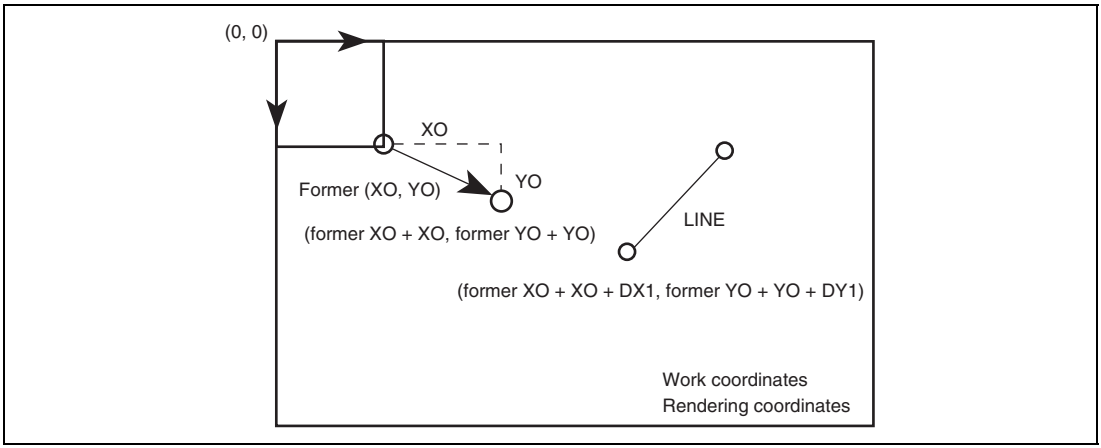


Figure 11.38 RLCOFS Command Example

(5) WPR

(a) Function

Sets a value in a specific address-mapped register.

(b) Command Format

LINKE = 0

31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
OP CODE = 0001_1000	Reserve (all 0)	Draw Mode
Reserve (all 0)	n – 1 (0 ≤ n – 1 ≤ 255)	0 0 0 0 W Reg No
Data 0		
:		
:		
Data n–1		

LINKE = 1 and LREL = 0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 0001_1000								Reserve (all 0)								Draw Mode															
Reserve (all 0)								n – 1 (0 ≤ n – 1 ≤ 255)								0	0	0	0	W Reg No											
0	0	0	LINK Address (longword address)																											0	0

Note: The longword address following the LINK Address is handled as the next command code. Therefore, do not specify the longword address following the address where the LINK Address is to be assigned as the link destination address specified by the LINK Address.

LINKE = 1 and LREL = 1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 0001_1000								Reserve (all 0)								Draw Mode															
Reserve (all 0)								n – 1 (0 ≤ n – 1 ≤ 255)								0	0	0	0	W Reg No											
Sign extended	Sign	LINK Address (longword address)																												0	0

Note: The longword address following the LINK Address is handled as the next command code. Therefore, do not specify the longword address following the address where the LINK Address is to be assigned as the link destination address specified by the address where the command code is located plus the LINK Address.

(c) Code

B'00011000

(d) Rendering Attributes

Draw Mode

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	LINKE	LREL	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	ByteM3	ByteM2	ByteM1	ByteM0

(e) Command Parameters

W reg No:	Register number
Data n (n = 1 to 256):	Write data
n − 1:	The number of write data
LINK Address:	LINK absolute address (Longword address. Write 0 to bits A31 to A29, A1, and A0.) LINK relative address (Longword address. Negative number expressed as two's complement. Bits A31 to A29 are used to extend the sign in bit A28. Write 0 to bits A1 and A0.)

Note: Even in 32-bit addressing mode, write the values in bits 28 to 3 of the specified 32-bit address to bits A28 to A3.

(f) Description

Writes data to the address-mapped registers. The register number is set in W reg No, and the write data in Data n.

Also ensure that there is no conflict with access by the CPU.

1. When the LINKE bit is set to 1, data is read from the memory address specified by the LINK Address and written to a register.
2. The LINK Address can be specified through the LREL bit as an absolute address or a relative address with respect to the memory address at which the WPR command code is located.
3. Setting the ByteM3 to ByteM0 bits to 1 allows writing to a register to be masked in byte units.

(6) JUMP

(a) Function

Changes the display list fetch destination.

(b) Command Format

REL = 0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 0010_1000								Reserve (all 0)								Draw Mode															
0	0	0	JUMP Address (longword address)																											0	0

REL = 1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 0010_1000								Reserve (all 0)								Draw Mode															
Sign extended Sign			JUMP Address (longword address)																											0	0

(c) Code

B'00101000

(d) Rendering Attributes

Draw Mode

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	REL	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0

(e) Command Parameter

JUMP Address: Jump destination absolute address (Longword address. Write 0 to bits A31 to A29, A1, and A0.)
Jump destination relative address (Longword address. Negative number expressed as two's complement. Bits A31 to A29 are used to extend the sign in bit A28. Write 0 to bits A1 and A0.)

Note: Even in 32-bit addressing mode, write the values in bits 28 to 3 of the specified 32-bit address to bits A28 to A3.

(f) Description

Changes the display list fetch destination to the specified address.

When REL = 0, the jump destination address can be specified as an absolute address. When REL = 1, the jump destination address can be specified as a relative address with respect to the memory address at which the command code is located.

(g) Example

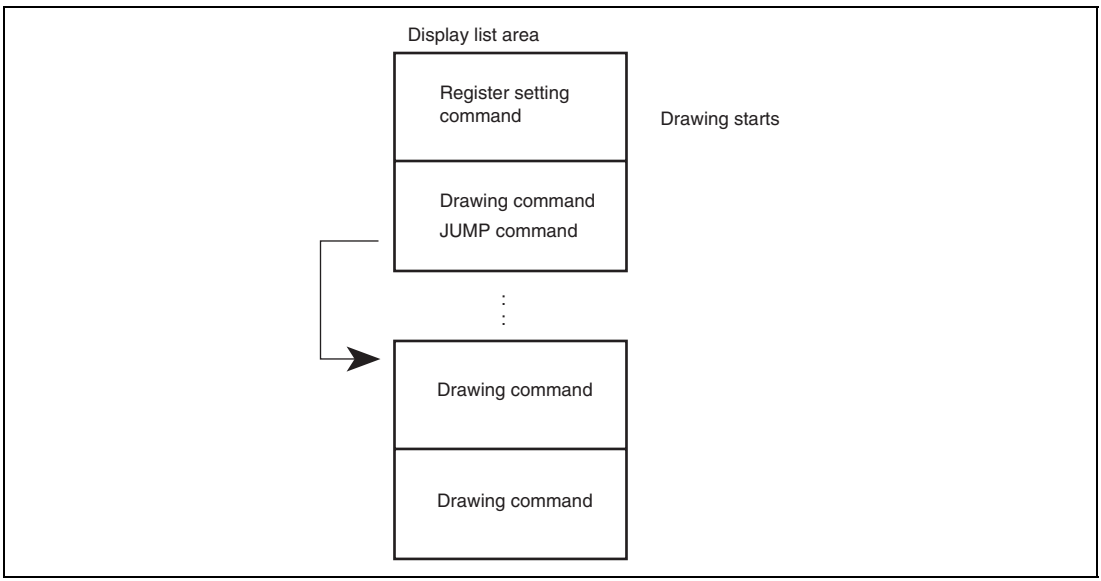


Figure 11.39 JUMP Command Example

(7) GOSUB

(a) Function

Makes a subroutine call for the display list.

(b) Command Format

REL = 0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 0011_0000								Reserve (all 0)								Draw Mode															
0	0	0	GOSUB Address (longword address)																											0	0

REL = 1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 0011_0000								Reserve (all 0)								Draw Mode															
Sign extended	Sign	GOSUB Address (longword address)																												0	0

(c) Code

B'00110000

(d) Rendering Attributes

Draw Mode

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	REL	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	No

(e) Command Parameter

GOSUB Address: Subroutine absolute address (Longword address. Write 0 to bits A31 to A29, A1, and A0.)
Subroutine relative address (Longword address. Negative number expressed as two's complement. Bits A31 to A29 are used to extend the sign in bit A28. Write 0 to bits A1 and A0.)

Note: Even in 32-bit addressing mode, write the values in bits 28 to 3 of the specified 32-bit address to bits A28 to A3.

(f) Description

Changes the display list fetch destination to the specified subroutine address. The fetch address is restored by an RET instruction. As only one level of nesting is permitted, it will not be possible to return if a subroutine call is issued within the subroutine.

When REL = 0, the subroutine address can be specified as an absolute address. When REL = 1, the jump destination address can be specified as a relative address with respect to the memory address at which the command code is located.

When the No bit is 0, the return address is set in the return address 0 register (RTN0R). When the No bit is 1, the return address is set in the return address 1 register (RTN1R).

(g) Example

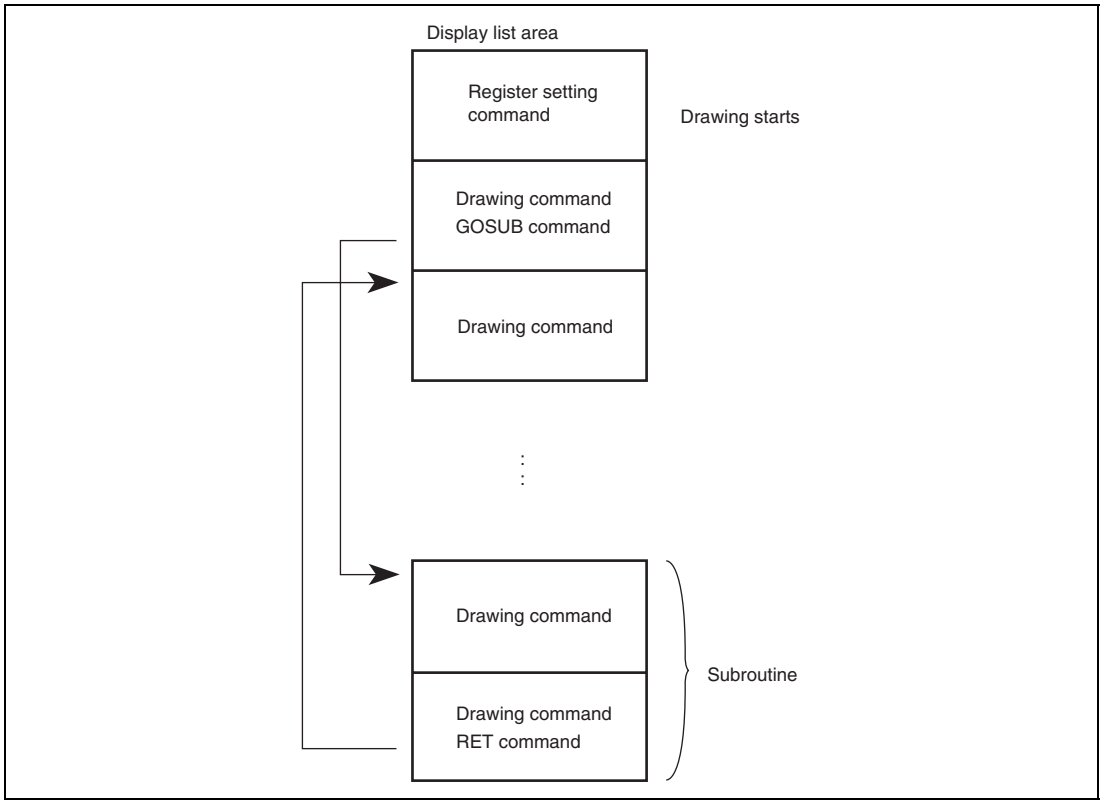


Figure 11.40 GOSUB Command Example

(8) RET

(a) Function

Returns from a subroutine call made by the GOSUB command.

(b) Command Format

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 0011_1000								Reserve (all 0)								Draw Mode															

(c) Code

B'00111000

(d) Rendering Attributes

Draw Mode

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	No

(e) Description

Restores the display list fetch destination to the address following the source of the subroutine call.

When the No bit is 0, the return address is set in the return address 0 register (RTN0R). When the No bit is 1, the return address is set in the return address 1 register (RTN1R).

(9) NOP/INT

(a) Function

Executes no operation.

(b) Command Format

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 0000_1000								Reserve (all 0)								Draw Mode															

(c) Code

B'00001000

(d) Rendering Attributes

Draw Mode

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
INT	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	INT No							

(e) Description

This command does not perform any operation. This command simply fetches the next instruction. However when the INT bit is set to 1 in this command, after this command has been fetched, the INT bit in the status register (SR) is set to 1, INT No is saved in the interrupt command ID register (ICIDR), and the drawing operation is halted. Clearing the INT bit in the status register (SR) restarts the drawing operation from the next command. To keep the coherence between the main memory and the cache when an interrupt occur, insert the SYNC command (rendering attribute: 0x0111) right before the NOP/INT command. When the INT bit is 0, inserting the SYNC command is not necessary.

(10) TRAP

(a) Function

Informs the end of the display list.

(b) Command Format

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 0000_0000								Reserve (all 0)								Draw Mode															

(c) Code

B'00000000

(d) Rendering Attributes

Draw Mode

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0

(e) Description

Halts the drawing operation and sets the TRA bit in the status register (SR) to 1. If the TRE bit in the interrupt enable register (IER) is set to 1, an interrupt is sent to the CPU. To keep the coherence between the main memory and the cache when an interrupt occur, insert the SYNC command (rendering attribute: 0x0111) immediately before the NOP/INT command.

This command must be placed at the end of the display list.

(f) Example

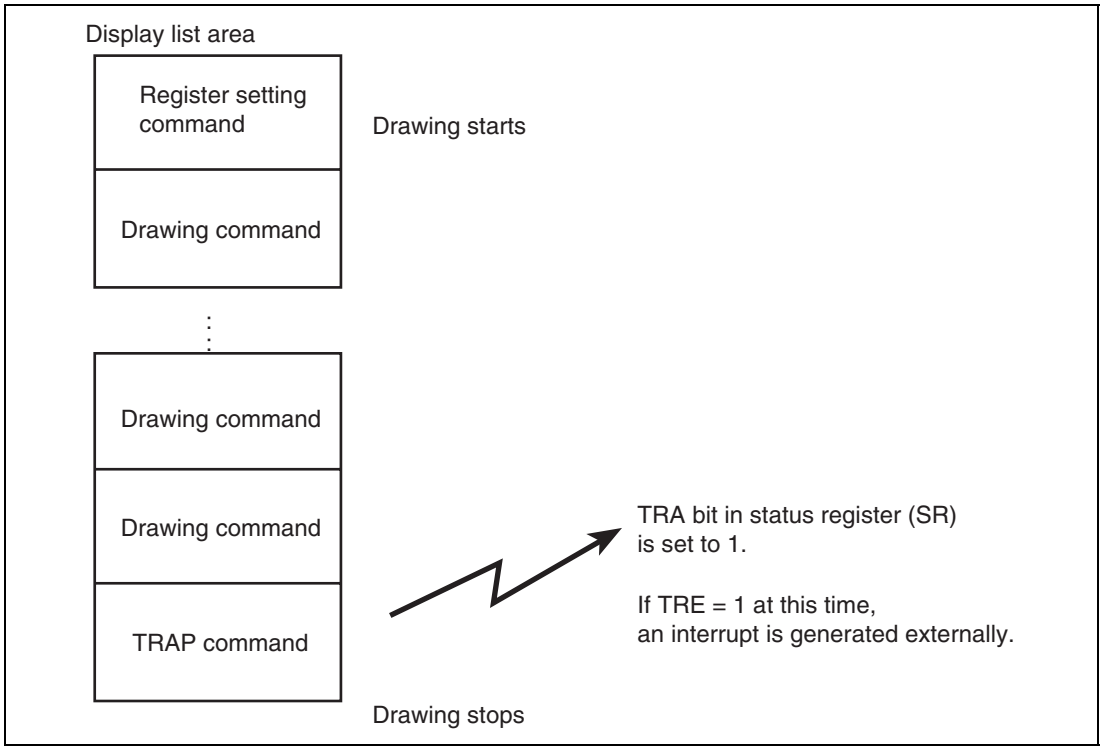


Figure 11.41 TRAP Command Example

(11) SYNC

(a) Function

Controls cache clearing/flushing.

(b) Command Format

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OP CODE = 0001_0010								Reserve (all 0)								Draw Mode															

(c) Code

B'00010010

(d) Rendering Attributes

Draw Mode

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0	WCLR	WFLSH	Fixed to 0	Fixed to 0	Fixed to 0	TCLR	Fixed to 0	Fixed to 0	DCLR	DFLSH

(e) Description

When all of the rendering attributes are 0, NOP operation is performed.

1. If TCLR is set to 1, the texture cache in the 3D command section is cleared.
2. If DCLR is set to 1, the destination cache is cleared.
3. If DFLSH is set to 1, the destination cache is flushed.
4. If WCLR is set to 1, the work cache is cleared.
5. If WFLSH is set to 1, the work cache is flushed.

Note: In the condition of the work cache and the destination cache,the bits for clearing and flushing cache should not be set to 1 at the same time.

Section 12 Video Input Module 0 (VIN0)

12.1 Overview

The Video Input Module 0 (hereinafter abbreviated as VIN0) is a video capture module that stores in external memory YCbCr-422 data through the ITU-R BT.601 or ITU-R BT.656 interface and RGB-666 data conforming to the ITU-R BT.709 color space definition through the ITU-R BT.601 interface.

This module can control the capture of interlaced video data from outside into a maximum of 2048 x 2048-pixel* capture area. It can also provide up to × 3 vertical and × 2 horizontal scaling for that data.

For captured video data, the VIN provides a color space conversion function from YCbCr-422 to RGB-565, a format conversion function from RGB-565 to ARGB-1555.

As the VIN internally generates a vertical sync signal and a field signal, it can capture progressive data in the ITU-R BT.1358 interface.

12.1.1 Input Interface

Four types of input interface can be selected for this channel.

Interface	Data Width	Data Type
ITU-R BT.601	8 bits	YCbCr-422 data (UYVY format)
ITU-R BT.656	8 bits	YCbCr-422 data (UYVY format)
ITU-R BT.1358	16 bits	YCbCr-422 data (8 bits (Y) + 8 bits (CbCr) format)
ITU-R BT.601	18 bits	RGB-666 data (ITU-R BT.709 color space format)

(1) Internal Sync Signal Generation

For video data capturing, the field signal can be internally generated even if the input sync signal stops.

(2) Capture Mode

The following four modes can be selected to capture the interlace images. In addition, single frame capture or continuous frame capture mode can be selected.

Triple-buffering control is provided in accordance with the captured field image and frame image to coordinate with the video capture mode of the display module.

- Odd-field capture mode
- Even-/odd-field capture mode
- Even-field capture mode
- Full interlace capture mode

(3) Vertical and Horizontal Scaling

The image can be scaled up and down up to three times in the vertical and two times in the horizontal directions.

(4) Size Clipping

The VIN has two clipping circuits, which independently handle images with up to 2048 × 2048 pixels. Any capture size within this limit can be specified before or after scaling. (When the scaling up function is specified, pre-clipping can be done within 768 × 2048 pixels.)

(5) YC (YCbCr-422)→RGB-565 Color Space Conversion

Image data stored in the YC (YCbCr-422) format can be converted to data of the RGB-565 color space and can be stored in external memory. Desired conversion coefficients can be specified through registers to adjust colors. The data in the YC (YCbCr-422) format can be stored in external memory without color space conversion.

(6) Format Conversion

In addition to YCbCr-422→RGB-565 color space conversion, the following data format conversion functions are available.

- RGB-565→ARGB-1555 data format conversion
- YCbCr-422→YCbCr-420 (NV12 format) data format conversion
- YC separation (data is separated into Y and UV components)

12.1.2 Block Diagram

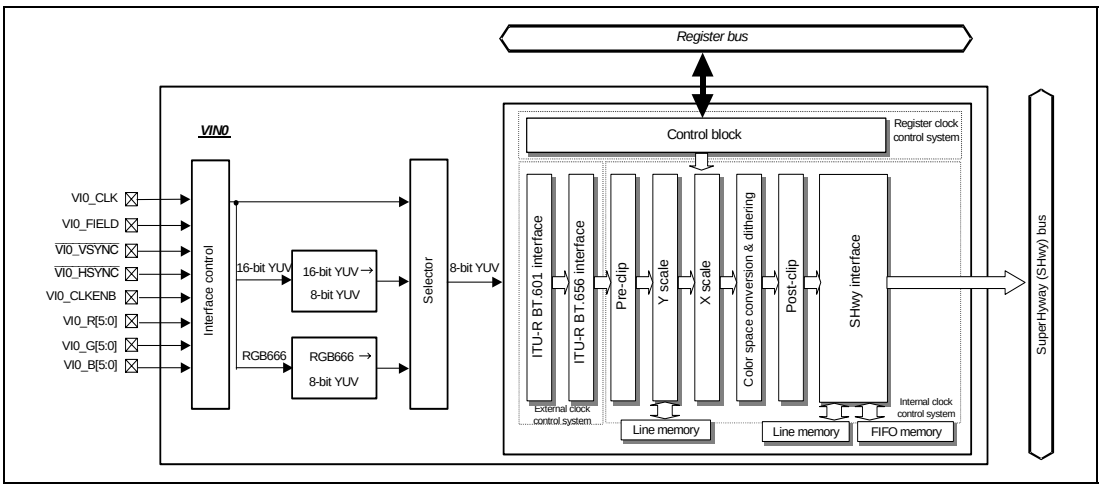


Figure 12.1 Block Diagram

12.2 External Pins

Table 12.1 Pin Configuration

Pin Name	Name	I/O	Function
VI0_CLK	VIN0 video clock	Input	External video clock in the ITU-R BT.601/BT.656 or BT.1358 interface
VI0_FIELD*	VIN0 field signal	Input	Field signal in the ITU-R BT.601 interface
$\overline{\text{VI0\_VSYNC}}$*	VIN0 vertical sync signal	Input	Vertical sync signal in the ITU-R BT.601 or BT.1358 interface.
$\overline{\text{VI0\_HSYNC}}$*	VIN0 horizontal sync signal	Input	Horizontal sync signal in the ITU-R BT.601 or BT.1358 interface
VI0_CLKENB*	VIN0 data enable	Input	Data enable signal in the ITU-R BT.601 or BT.1358 interface
VI0_R5 to VI0_R0* VI0_G5 to VI0_G0* VI0_B5 to VI0_B0*	VIN0 video data	Input	Data signals in the ITU-R BT.601/BT.656 or BT.1358 interface 16-/8-bit YCbCr-422 and RGB-666 are supported. For data pin connections, refer to table 12.2.

Note: * Fix the pins at a high or low level when they are not used.

Table 12.2 Data Pin Connections

Input Data Format	VI0_R[5:0]						VI0_G[5:0]						VI0_B[5:0]					
	5	4	3	2	1	0	5	4	3	2	1	0	5	4	3	2	1	0
ITU-R BT.601/BT.656 8-bit YCbCr-422	*	*	*	*	*	*	*	*	*	*	Video data [7:0]							
ITU-R BT.601/BT.1358 16-bit YCbCr-422	*	*	Y Video data [7:0]								CbCr Video data [7:0]							
ITU-R BT.601 RGB-666	R Video data [5:0]						G Video data [5:0]						B Video data [5:0]					

Note: * Fix the pins at a high or low level.

12.3 Register Configuration

Table 12.3 shows the VIN register configuration and table 12.4 shows the register state in each operating mode.

Table 12.3 VIN Registers

Name	Symbol	R/W	P4 Address	Area 7 Address	Access Size	Internal Update Mode Support*
Main control register	MC	R/W	H'FFC50000	H'1FC50000	32	Δ
Module status register	MS	R	H'FFC50004	H'1FC50004	32	—
Frame capture register	FC	R/W	H'FFC50008	H'1FC50008	32	—
Start line pre-clip register	SLPrC	R/W	H'FFC5000C	H'1FC5000C	32	Supported
End line pre-clip register	ELPrC	R/W	H'FFC50010	H'1FC50010	32	Supported
Start pixel pre-clip register	SPPrC	R/W	H'FFC50014	H'1FC50014	32	Supported
End pixel pre-clip register	EPPrC	R/W	H'FFC50018	H'1FC50018	32	Supported
Start line post-clip register	SLPoC	R/W	H'FFC5001C	H'1FC5001C	32	Supported
End line post-clip register	ELPoC	R/W	H'FFC50020	H'1FC50020	32	Supported
Start pixel post-clip register	SPPoC	R/W	H'FFC50024	H'1FC50024	32	Supported
End pixel post-clip register	EPPoC	R/W	H'FFC50028	H'1FC50028	32	Supported
Image stride register	IS	R/W	H'FFC5002C	H'1FC5002C	32	Supported
Memory base 1 register	MB1	R/W	H'FFC50030	H'1FC50030	32	Supported
Memory base 2 register	MB2	R/W	H'FFC50034	H'1FC50034	32	Supported
Memory base 3 register	MB3	R/W	H'FFC50038	H'1FC50038	32	Supported
Line count register	LC	R	H'FFC5003C	H'1FC5003C	32	—
Interrupt enable register	IE	R/W	H'FFC50040	H'1FC50040	32	—
Interrupt status register	INTS	R/W	H'FFC50044	H'1FC50044	32	—
Scanline interrupt register	SI	R/W	H'FFC50048	H'1FC50048	32	Supported
Memory transfer control register	MTC	R/W	H'FFC5004C	H'1FC5004C	32	Supported
Y scale register	YS	R/W	H'FFC50050	H'1FC50050	32	Supported
X scale register	XS	R/W	H'FFC50054	H'1FC50054	32	Supported
Data mode register	DMR	R/W	H'FFC50058	H'1FC50058	32	Supported
Data mode register 2	DMR2	R/W	H'FFC5005C	H'1FC5005C	32	—
UV address offset register	UVAOF	R/W	H'FFC50060	H'1FC50060	32	Supported

Name	Symbol	R/W	P4 Address	Area 7 Address	Access Size	Internal Update Mode Support*
CSC coefficient 1 register	CSCC1	R/W	H'FFC50064	H'1FC50064	32	Supported
CSC coefficient 2 register	CSCC2	R/W	H'FFC50068	H'1FC50068	32	Supported
CSC coefficient 3 register	CSCC3	R/W	H'FFC5006C	H'1FC5006C	32	Supported
Coefficient set 1A register	C1A	R/W	H'FFC50080	H'1FC50080	32	—
Coefficient set 1B register	C1B	R/W	H'FFC50084	H'1FC50084	32	—
Coefficient set 1C register	C1C	R/W	H'FFC50088	H'1FC50088	32	—
Coefficient set 2A register	C2A	R/W	H'FFC50090	H'1FC50090	32	—
Coefficient set 2B register	C2B	R/W	H'FFC50094	H'1FC50094	32	—
Coefficient set 2C register	C2C	R/W	H'FFC50098	H'1FC50098	32	—
Coefficient set 3A register	C3A	R/W	H'FFC500A0	H'1FC500A0	32	—
Coefficient set 3B register	C3B	R/W	H'FFC500A4	H'1FC500A4	32	—
Coefficient set 3C register	C3C	R/W	H'FFC500A8	H'1FC500A8	32	—
Coefficient set 4A register	C4A	R/W	H'FFC500B0	H'1FC500B0	32	—
Coefficient set 4B register	C4B	R/W	H'FFC500B4	H'1FC500B4	32	—
Coefficient set 4C register	C4C	R/W	H'FFC500B8	H'1FC500B8	32	—
Coefficient set 5A register	C5A	R/W	H'FFC500C0	H'1FC500C0	32	—
Coefficient set 5B register	C5B	R/W	H'FFC500C4	H'1FC500C4	32	—
Coefficient set 5C register	C5C	R/W	H'FFC500C8	H'1FC500C8	32	—
Coefficient set 6A register	C6A	R/W	H'FFC500D0	H'1FC500D0	32	—
Coefficient set 6B register	C6B	R/W	H'FFC500D4	H'1FC500D4	32	—
Coefficient set 6C register	C6C	R/W	H'FFC500D8	H'1FC500D8	32	—
Coefficient set 7A register	C7A	R/W	H'FFC500E0	H'1FC500E0	32	—
Coefficient set 7B register	C7B	R/W	H'FFC500E4	H'1FC500E4	32	—
Coefficient set 7C register	C7C	R/W	H'FFC500E8	H'1FC500E8	32	—
Coefficient set 8A register	C8A	R/W	H'FFC500F0	H'1FC500F0	32	—
Coefficient set 8B register	C8B	R/W	H'FFC500F4	H'1FC500F4	32	—
Coefficient set 8C register	C8C	R/W	H'FFC500F8	H'1FC500F8	32	—

Notes: Do not access any address not listed above.

* For the internal update mode, refer to the description of the VUP bit in the main control register (MC).

Table 12.4 Register State in Each Operating Mode

Name	Symbol	Power-on Reset	Manual Reset	Sleep	Module Standby	Deep Standby
Main control register	MC	H'00000000	H'00000000	Retained	Retained	H'00000000
Module status register	MS	H'00000018	H'00000018	Retained	Retained	H'00000018
Frame capture register	FC	H'00000000	H'00000000	Retained	Retained	H'00000000
Start line pre-clip register	SLPrC	H'00000000	H'00000000	Retained	Retained	H'00000000
End line pre-clip register	ELPrC	H'00000000	H'00000000	Retained	Retained	H'00000000
Start pixel pre-clip register	SPPrC	H'00000000	H'00000000	Retained	Retained	H'00000000
End pixel pre-clip register	EPPrC	H'00000000	H'00000000	Retained	Retained	H'00000000
Start line post-clip register	SLPoC	H'00000000	H'00000000	Retained	Retained	H'00000000
End line post-clip register	ELPoC	H'00000000	H'00000000	Retained	Retained	H'00000000
Start pixel post-clip register	SPPoC	H'00000000	H'00000000	Retained	Retained	H'00000000
End pixel post-clip register	EPPoC	H'00000000	H'00000000	Retained	Retained	H'00000000
Image stride register	IS	H'00000000	H'00000000	Retained	Retained	H'00000000
Memory base 1 register	MB1	H'00000000	H'00000000	Retained	Retained	H'00000000
Memory base 2 register	MB2	H'00000000	H'00000000	Retained	Retained	H'00000000
Memory base 3 register	MB3	H'00000000	H'00000000	Retained	Retained	H'00000000
Line count register	LC	H'00000000	H'00000000	Retained	Retained	H'00000000
Interrupt enable register	IE	H'00000000	H'00000000	Retained	Retained	H'00000000
Interrupt status register	INTS	H'00000000	H'00000000	Retained	Retained	H'00000000
Scanline interrupt register	SI	H'00000000	H'00000000	Retained	Retained	H'00000000
Memory transfer control register	MTC	H'0A080008	H'0A080008	Retained	Retained	H'0A080008
Y scale register	YS	H'00000000	H'00000000	Retained	Retained	H'00000000
X scale register	XS	H'00000000	H'00000000	Retained	Retained	H'00000000
Data mode register	DMR	H'00000000	H'00000000	Retained	Retained	H'00000000
Data mode register 2	DMR2	H'00000000	H'00000000	Retained	Retained	H'00000000
UV address offset register	UVAOF	H'00000000	H'00000000	Retained	Retained	H'00000000
CSC coefficient 1 register	CSCC1	H'01291080	H'01291080	Retained	Retained	H'01291080
CSC coefficient 2 register	CSCC2	H'019800D0	H'019800D0	Retained	Retained	H'019800D0
CSC coefficient 3 register	CSCC3	H'00640204	H'00640204	Retained	Retained	H'00640204
Coefficient set 1A register	C1A	H'00000000	H'00000000	Retained	Retained	H'00000000
Coefficient set 1B register	C1B	H'00000000	H'00000000	Retained	Retained	H'00000000

Name	Symbol	Power-on Reset	Manual Reset	Sleep	Module Standby	Deep Standby
Coefficient set 1C register	C1C	H'00000000	H'00000000	Retained	Retained	H'00000000
Coefficient set 2A register	C2A	H'00000000	H'00000000	Retained	Retained	H'00000000
Coefficient set 2B register	C2B	H'00000000	H'00000000	Retained	Retained	H'00000000
Coefficient set 2C register	C2C	H'00000000	H'00000000	Retained	Retained	H'00000000
Coefficient set 3A register	C3A	H'00000000	H'00000000	Retained	Retained	H'00000000
Coefficient set 3B register	C3B	H'00000000	H'00000000	Retained	Retained	H'00000000
Coefficient set 3C register	C3C	H'00000000	H'00000000	Retained	Retained	H'00000000
Coefficient set 4A register	C4A	H'00000000	H'00000000	Retained	Retained	H'00000000
Coefficient set 4B register	C4B	H'00000000	H'00000000	Retained	Retained	H'00000000
Coefficient set 4C register	C4C	H'00000000	H'00000000	Retained	Retained	H'00000000
Coefficient set 5A register	C5A	H'00000000	H'00000000	Retained	Retained	H'00000000
Coefficient set 5B register	C5B	H'00000000	H'00000000	Retained	Retained	H'00000000
Coefficient set 5C register	C5C	H'00000000	H'00000000	Retained	Retained	H'00000000
Coefficient set 6A register	C6A	H'00000000	H'00000000	Retained	Retained	H'00000000
Coefficient set 6B register	C6B	H'00000000	H'00000000	Retained	Retained	H'00000000
Coefficient set 6C register	C6C	H'00000000	H'00000000	Retained	Retained	H'00000000
Coefficient set 7A register	C7A	H'00000000	H'00000000	Retained	Retained	H'00000000
Coefficient set 7B register	C7B	H'00000000	H'00000000	Retained	Retained	H'00000000
Coefficient set 7C register	C7C	H'00000000	H'00000000	Retained	Retained	H'00000000
Coefficient set 8A register	C8A	H'00000000	H'00000000	Retained	Retained	H'00000000
Coefficient set 8B register	C8B	H'00000000	H'00000000	Retained	Retained	H'00000000
Coefficient set 8C register	C8C	H'00000000	H'00000000	Retained	Retained	H'00000000

[Legend for Register Description]

Initial value: Register value after a reset

—: Undefined value

R/W: Readable/writable. The written value can be read.

R: Read-only. The write value should always be 0.

Write 0 to the reserved bits.

12.3.1 Main Control Register (MC)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	CLP[1:0]		—	—	—	CFSL	—	—	FOC	—	YCAL	INF[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R	R/W	R	R	R/W	R	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	VUP	—	—	—	EN	EC	IM[1:0]		—	BPS	ME
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R	R	R	R/W	R/W	R/W	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
29, 28	CLP[1:0]	00	R/W	Pixel Data Clipping These bits specify the pixel data value to be clipped when the input format is ITU-R BT.601 and the input value is outside the range prescribed in the ITU-R BT.601 standard.

CLP	Luminance	Chrominance
00	No clipping	16 or a smaller value is clipped to 16. 240 or a greater value is clipped to 240.
01	16 or a smaller value is clipped to 16. 240 or a greater value is clipped to 240.	16 or a smaller value is clipped to 16. 240 or a greater value is clipped to 240.
10	No clipping	16 or a smaller value is clipped to 128. 240 or a greater value is clipped to 128.
11	0 or a smaller value is clipped to 1.	

Note: These bits support the internal update mode.

27 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
24	CFSL	0	R/W	RGB Color Format Select For RGB-666 input data, capture control is done after conversion to 8-bit YCbCr-422 data. As RGB-666 data supports both the ITU-R BT.601 and BT.709 color space settings, these bits select the color space for the input RGB data. 0: BT.601 color space is selected for RGB-666 input data. 1: BT.709 color space is selected for RGB-666 input data. Note: This bit supports the internal update mode.
23, 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21	FOC	0	R/W	Field Order Control This bit controls the field order for full interlace capturing. 0: Top field = Odd field (field 1) 1: Top field = Even field (field 2) Note: This bit supports the internal update mode.
20	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
19	YCAL	0	R/W	YCbCr-422 Input Data Alignment This bit controls data alignment for 16-bit YCbCr-422 input. 0: 16-bit video data is input without change. 1: The upper and lower bytes of 16-bit input video data are swapped.
18 to 16	INF[2:0]	000	R/W	Input Interface Format These bits specify the image format input to the VIN. 000: ITU-R BT.656 8-bit YCbCr-422 data is input. 001: ITU-R BT.601 8-bit YCbCr-422 data is input. 101: ITU-R BT.1358 16-bit YCbCr-422 data is input. 111: ITU-R BT.601 18-bit RGB-666 data is input. Others: Setting prohibited
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
10	VUP	0	R/W	VIN Register Update Control This bit specifies the internal register update timing after register writing. See the list of registers for applicable registers. 0: The register contents are updated immediately after register writing. 1: The register contents are updated after a valid field is detected in the ITU-R BT.601 data or the field (F) bit changes in the ITU-R BT.656 data.
9 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6	EN	0	R/W	Endian Type This bit specifies the endian type for data to be output to external memory. 0: Image data is packed and allocated in little endian. 1: Image data is packed and allocated in big endian. Note: When selecting big endian, be sure to set the BPSM bit in DMR to 1.
5	EC	0	R/W	Error Correction Control This bit specifies whether error correction with the parity bit is performed on the ITU-R BT.656 input. 0: Error correction is not performed on the ITU-R BT.656 input. 1: Error correction with the parity bit is performed on the ITU-R BT.656 input. Error correction must not be performed in the following cases: • When data is captured in the ITU-R BT.601 interface • When input data does not meet the standard of the ITU-R BT.656 parity bit

Bit	Bit Name	Initial Value	R/W	Description
4, 3	IM[1:0]	00	R/W	Interlace Mode These bits specify the capture mode. Do not modify this setting during capture operation. 00: Odd-field (field 1) capture mode Handles only odd fields as frames and stores them in external memory. 01: Odd-/even-field capture mode Handles odd and even fields as separate frames and stores them in external memory. This mode is available only in continuous frame capture mode. 10: Even-field (field 2) capture mode Handles only even fields as frames and stores them in external memory. 11: Full interlace mode Handles combinations of odd and even fields as single frames and stores them in external memory.
2	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
1	BPS	0	R/W	YCbCr-422→RGB-565 Conversion Bypass Mode This bit controls YCbCr-422→RGB-565 conversion. 0: YCbCr-422→RGB-565 conversion is performed. 1: YCbCr-422→RGB-565 conversion is not performed. Note: Desired RGB-565 conversion coefficients can be specified through CSCC1 to CSCC3.
0	ME	0	R/W	Module Enable This is the enable bit for the VIN. Set this bit before setting the other registers. 0: The module operation is stopped. 1: The module operation is enabled.

12.3.2 Module Status Register (MS)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	FBS[1:0]		FS	AV	CA
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4, 3	FBS[1:0]	11	R	Frame Buffer Status These bits show the frame buffer status. 00: The latest valid frame buffer has the base address defined by the memory base 1 register. 01: The latest valid frame buffer has the base address defined by the memory base 2 register. 10: The latest valid frame buffer has the base address defined by the memory base 3 register. 11: There is no valid frame buffer.
2	FS	0	R	Field Status This bit shows the type of the current capture field. 0: The current field is field 1 (odd field). 1: The current field is field 2 (even field).

Bit	Bit Name	Initial Value	R/W	Description
1	AV	0	R	Active Video Status This bit shows whether the current field is in the active video area defined by the pre-clipping register. 0: The current field is not in the active video area. 1: The current field is in the active video area. Note: This bit will be 0 if no input data is captured.
0	CA	0	R	Video Capture Active Status This bit shows the current video capture operation status. This bit is updated by the captured field signal. 0: Video capture is not operating. 1: Video capture is operating. Note: In field capture mode, this bit is set to 1 even for the field that does not capture data.

12.3.3 Frame Capture Register (FC)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CC	SC
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	CC	0	R/W	Continuous Frame Capture Mode This bit specifies the continuous frame capture mode. In this mode, the first capture frame is written into the memory address specified by the memory base 1 (MB1) register. After that, the capture operation is repeated in the order of MB2, MB3, MB1, MB2, and such. Writing 0 into this bit during continuous capture operation will immediately terminate the capture operation if the current frame is completed or it has not been captured. 0: The continuous frame capture mode is not set. 1: The continuous frame capture mode is set.
0	SC	0	R/W	Single Frame Capture Mode This bit specifies the single frame capture mode. In this mode, the capture frame is written into the memory address specified by the memory base 1 (MB1) register. Immediately after this bit is set to 1, the frame buffer status (FBS) bits in MS are initialized and the SC bit is also cleared to 0. 0: The single frame capture mode is not set. 1: The single frame capture mode is set. Note: Do not set this bit to 1 when the interlace mode (IM) bits are set to 01 (odd-/even-field capture mode).

Note: Do not specify the single frame capture mode and continuous frame capture mode at the same time.

12.3.4 Start Line Pre-Clip Register (SLPrC)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SLPrC[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SLPrC[10:0]	H'000	R/W	Start Line Pre-Clip These bits specify the (pre-clipping start line − 1) value in line units. This value is used before scaling. Specify a value in the range from 0 to 2046 so that the number of lines after pre-clipping will be 2 or more. (The value of 0 indicates the first valid line.)

12.3.5 End Line Pre-Clip Register (ELPrC)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ELPrC[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	ELPrC[10:0]	H'000	R/W	End Line Pre-Clip These bits specify the (pre-clipping end line − 1) value in line units. This value is used before scaling. Specify a value in the range from 1 to 2047 so that the number of lines after pre-clipping will be 2 or more.

12.3.6 Start Pixel Pre-Clip Register (SPPrC)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SPPrC[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SPPrC [10:0]	H'000	R/W	Start Pixel Pre-Clip These bits specify the (pre-clipping start pixel – 1) value in pixel units. This value is used before scaling. Specify a value in the range from 0 to 2042 so that the number of pixels after pre-clipping will be 5 or more. Notes: 1. The number of pixels to be clipped should be a multiple of 2 to support clipping processing in the YCbCr-422 format. Accordingly, the LSB in this register is always ignored and if an odd value is specified, clipping is done with an even value obtained by (set value – 1). 2. The capacity of the internal buffer is limited, so if the horizontal scaling up function is in use, the value here should be such that ELPrC – SLPrC is no greater than 768 pixels.

12.3.7 End Pixel Pre-Clip Register (EPPrC)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	EPPrC[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	EPPrC [10:0]	H'000	R/W	End Pixel Pre-Clip These bits specify the (pre-clipping end pixel – 1) value in pixel units. This value is used before scaling. Specify a value in the range from 5 to 2047 so that the number of pixels after pre-clipping will be 5 or more. Notes: 1. The number of pixels to be clipped should be a multiple of 2 to support the clipping processing in the YCbCr-422 format. Accordingly, the LSB in this register is always ignored and if an even value is specified, clipping is done with an odd value obtained by (set value + 1). 2. The capacity of the internal buffer is limited, so if the horizontal scaling up function is in use, the value here should be such that ELPrC – SLPrC is no greater than 768 pixels.

12.3.8 Start Line Post-Clip Register (SLPoC)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SLPoC[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SLPoC [10:0]	H'000	R/W	Start Line Post-Clip These bits specify the post-clipping start line value in line units. This value is used after scaling. Specify a value in the range from 0 to 2046 so that the number of lines after post-clipping will be 2 or more. (The value of 0 indicates the start line after scaling.)

12.3.9 End Line Post-Clip Register (ELPoC)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ELPoC[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	ELPoC [10:0]	H'000	R/W	End Line Post-Clip These bits specify the (post-clipping end line − 1) value in line units. This value is used after scaling. Specify a value in the range from 1 to 2047 so that the number of lines after post-clipping will be 2 or more.

12.3.10 Start Pixel Post-Clip Register (SPPoC)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SPPoC[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SPPoC [10:0]	H'000	R/W	Start Pixel Post-Clip These bits specify the (post-clipping start pixel − 1) value in pixel units. This value is used after scaling. Specify a value in the range from 0 to 2042 so that the number of pixels after post-clipping will be 5 or more. Note: The LSB in this register is valid to support clipping processing in the YCbCr-444 format.

12.3.11 End Pixel Post-Clip Register (EPPoC)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	EPPoC[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	EPPoC [10:0]	H'000	R/W	End Pixel Post-Clip These bits specify the (post-clipping end pixel − 1) value in pixel units. This value is used after scaling. Specify a value in the range from 5 to 2047 so that the number of pixels after post-clipping will be 5 or more. Note: The LSB in this register is valid to support clipping processing in the YCbCr-444 format.

Note: Even if the settings produce a post-clipping size in pixels that is an odd number, the output to memory will be in even pixel units since the format is YCbCr-422. Take note that if the clipping values specify an odd size (i.e. if EPPoc – SPPoc is odd), one will be added to round the value up to an even number.

EPPoC − SPPoC = odd value (= even size)

If an odd size is specified (the calculation result is an even value), the even size obtained by adding 1 is used for clipping.

Example: When SPPoC is set to 0 and EPPoC is set to 62:

The actual processing is done for 64 pixels.
62 - 0 = 62 (63 pixels)
63 pixels + 1 = 64 pixels

12.3.12 Image Stride Register (IS)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	IS[8:0]									—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 4	IS[8:0]	H'000	R/W	Image Stride These bits specify the width of the transfer destination memory within the range from 16 to 4096 pixels in units of 16 pixels. Specify a value equal to or greater than the post-clip width (EPPoC − SPPoC). When the data format is 16 bits/pixel, this value is used to generate a word address. When the data format is 8 bits/pixel in YC separation mode, this value is used to generate a byte address.
3 to 0	—	All 0	R	Reserved bits that indicate the lower-order four bits of the image stride (a multiple of 16 pixels). These bits are always read as 0. The write value should always be 0.

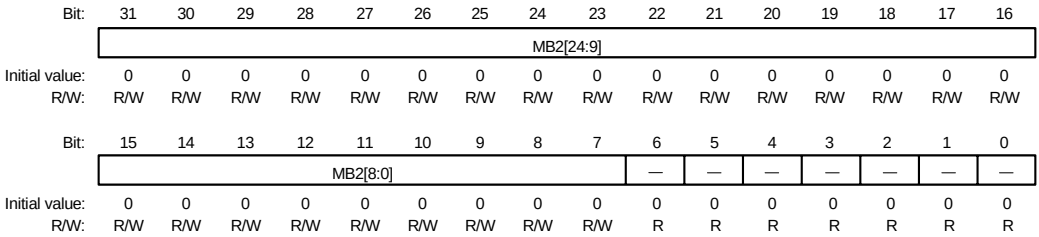
12.3.13 Memory Base 1 Register (MB1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MB1[24:9]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MB1[8:0]									—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	MB1[24:0]	H'00000000	R/W	Memory Base Address 1 These bits specify the transfer start address in frame buffer 1. Specify a value for physical address bits [31:7] in units of 128 bytes. If the module is in continuous frame capture mode, this value is used as the MB1 address in the following capture sequence: MB1 → MB2 → MB3 → MB1 → MB2 → MB3. In single frame capture mode, this value is used as the capture address. Specify a memory address taking into account the image size so that the image data does not exceed an area boundary on the address map.
6 to 0	—	All 0	R	Reserved bits that indicate the lower-order seven bits in of memory base address 1 (a multiple of 128 bytes). These bits are always read as 0. The write value should always be 0.

12.3.14 Memory Base 2 Register (MB2)



Bit	Bit Name	Initial Value	R/W	Description
31 to 7	MB2[24:0]	H'00000000	R/W	Memory Base Address 2 These bits specify the transfer start address in frame buffer 2. Specify a value for physical address bits [31:7] in units of 128 bytes. If the module is in continuous frame capture mode, this value is used as the MB2 address in the following capture sequence: MB1 → MB2 → MB3 → MB1 → MB2 → MB3. Specify a memory address taking into account the image size so that the image data does not exceed an area boundary on the address map.
6 to 0	—	All 0	R	Reserved bits that indicate the lower-order seven bits in of memory base address 2 (a multiple of 128 bytes). These bits are always read as 0. The write value should always be 0.

12.3.15 Memory Base 3 Register (MB3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MB3[24:9]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MB3[8:0]									—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	MB3[24:0]	H'00000000	R/W	Memory Base Address 3 These bits specify the transfer start address in frame buffer 3. Specify a value for physical address bits [31:7] in units of 128 bytes. If the module is in continuous frame capture mode, this value is used as the MB3 address in the following capture sequence: MB1 → MB2 → MB3 → MB1 → MB2 → MB3. Specify a memory address taking into account the image size so that the image data does not exceed an area boundary on the address map.
6 to 0	—	All 0	R	Reserved bits that indicate the lower-order seven bits in of memory base address 3 (a multiple of 128 bytes). These bits are always read as 0. The write value should always be 0.

12.3.16 Line Count Register (LC)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	LC[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	LC[10:0]	H'000	R	Line Count These bits show the line position in the current capture field.

12.3.17 Interrupt Enable Register (IE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FIE2	—	—	—	—	—	—	—	—	—	—	—	—	—	VFE	VRE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	FIE	CEE	SIE	EFE	FOE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	FIE2	0	R/W	Field Interrupt Enable 2 This bit enables or disables INTC output for field interrupts. Interrupt signals by this enable bit is asserted irrespective of whether capture is taking place. 0: Field interrupts are disabled. 1: Field interrupts are enabled.
30 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17	VFE	0	R/W	VSYNC Falling Edge Detect Interrupt Enable This bit enables or disables VSYNC falling edge detect interrupts. Interrupt signals by this enable bit is asserted irrespective of whether capture is taking place. 0: VSYNC falling edge detect interrupts are disabled. 1: VSYNC falling edge detect interrupts are enabled.
16	VRE	0	R/W	VSYNC Rising Edge Detect Interrupt Enable This bit enables or disables VSYNC rising edge detect interrupts. Interrupt signals by this enable bit is asserted irrespective of whether capture is taking place. 0: VSYNC rising edge detect interrupts are disabled. 1: VSYNC rising edge detect interrupts are enabled.

Bit	Bit Name	Initial Value	R/W	Description
15 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	FIE	0	R/W	Field Interrupt Enable This bit enables or disables field-switching interrupts. This interrupt enable setting is valid when the CA bit in MS is 1. 0: Field-switching interrupts are disabled. 1: Field-switching interrupts are enabled.
3	CEE	0	R/W	Correction Error Interrupt Enable This bit enables or disables interrupts due to error correction in the timing reference code (SAV/EAV) described in the ITU-R BT.656 specification. This interrupt enable setting is valid when the CA bit in MS is 1. 0: ITU-R BT.656 timing reference code error interrupts are disabled. 1: ITU-R BT.656 timing reference code error interrupts are enabled.
2	SIE	0	R/W	Scanline Interrupt Enable This bit enables or disables scanline interrupts. This interrupt enable setting is valid when the CA bit in MS is 1. 0: Scanline interrupts are disabled. 1: Scanline interrupts are enabled.
1	EFE	0	R/W	End of Frame Interrupt Enable This bit enables or disables end of frame interrupts. This interrupt enable setting is valid when the CA bit in MS is 1. 0: End of frame interrupts are disabled. 1: End of frame interrupts are enabled.
0	FOE	0	R/W	FIFO Overflow Interrupt Enable This bit enables or disables FIFO overflow interrupts. This interrupt enable setting is valid when the CA bit in MS is 1. 0: FIFO overflow interrupts are disabled. 1: FIFO overflow interrupts are enabled.

12.3.18 Interrupt Status Register (INTS)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FIS2	—	—	—	—	—	—	—	—	—	—	—	—	—	VFS	VRS
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	FIS	CES	SIS	EFS	FOS
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	FIS2	0	R/W	Field Interrupt Status 2 This bit shows that the field has changed. This bit is set to 1 when a valid field is detected in the ITU-R BT.601 interface or the F bit defined in ITU-R BT.656 changes. After being set to 1, this bit is cleared to 0 by writing 1. Note: This bit is set to 1 irrespective of whether or not capture is taking place. Be sure to clear this bit to 0 before using it.
30 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17	VFS	0	R/W	VSYNC Falling Edge Detect Interrupt Status This bit shows that a VSYNC falling edge has been detected in the ITU-R BT.601 input. After being set to 1, this bit is cleared to 0 by writing 1. Note: This bit is set to 1 irrespective of whether or not capture is taking place. Be sure to clear this bit to 0 before using it.
16	VRS	0	R/W	VSYNC Rising Edge Detect Interrupt Status This bit shows that a VSYNC rising edge has been detected in the ITU-R BT.601 input. After being set to 1, this bit is cleared to 0 by writing 1. Note: This bit is set to 1 irrespective of whether or not capture is taking place. Be sure to clear this bit to 0 before using it.
15 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
4	FIS	0	R/W	Field Interrupt Status This bit shows that a field has been captured in the active capture operation. This bit is set to 1 when a valid field is detected in the ITU-R BT.601 interface or the F bit defined in ITU-R BT.656 changes. After being set to 1, this bit is cleared to 0 by writing 1.
3	CES	0	R/W	Correction Error Interrupt Status This bit shows that the timing reference code in the active capture operation has an error involving at least two bits. After being set to 1, this bit is cleared to 0 by writing 1. This bit is set to 1 if the EC bit in MC is enabled and the timing reference code has an error involving at least two bits. If a 1-bit error occurs when the EC bit is enabled, this bit is not set to 1.
2	SIS	0	R/W	Scanline Interrupt Status This bit shows that the number of lines specified by SI has been reached in the active capture operation. After being set to 1, this bit is cleared to 0 by writing 1. This bit is set to 1 at the next line start timing after the value in the LC register matches the SI register setting. This timing is shown in figure 12.2, Scanline Interrupt Status Generation Timing.
1	EFS	0	R/W	End of Frame Interrupt Status This bit shows that the last frame has been reached in the active capture operation. This bit is set to 1 at the end of field 2 (even field). After being set to 1, this bit is cleared to 0 by writing 1.
0	FOS	0	R/W	FIFO Overflow Interrupt Status This bit shows that the FIFO has overflowed in the active capture operation. After being set to 1, this bit is cleared to 0 by writing 1. If the FIFO overflows, the FIFO data is overwritten by the pixel data captured after the overflow, and the resultant data is sent to the frame buffer.

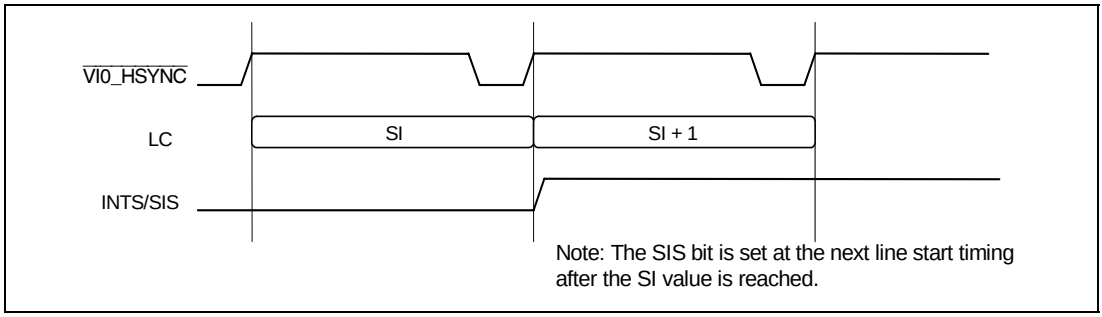


Figure 12.2 Scanline Interrupt Status Generation Timing

12.3.19 Scanline Interrupt Register (SI)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SI[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SI[10:0]	H'000	R/W	Scanline Interrupt Setting These bits specify a value to be compared with the LC register value in each field while the SIE bit in the IE register is set to 1. When this value matches the LC register value, an interrupt signal is asserted.

12.3.20 Memory Transfer Control Register (MTC)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	PRIH[3:0]				—	—	—	—	PRIL[3:0]			
Initial value:	0	0	0	0	1	0	1	0	0	0	0	0	1	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 24	PRIH[3:0]	H'A	R/W	Priority High Level Value Setting These bits specify a priority value for the second and subsequent words in memory transfer transactions. The initial value is 10.
23 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
19 to 16	PRIL[3:0]	H'8	R/W	Priority High Level Value Setting These bits specify a priority value for the first word in memory transfer transactions. The initial value is 8. This bit should always be 9 during the operation.
15 to 0	—	H'0008	R	Reserved These bits are always read as H'0008. The write value should always be H'0008.

Note: Only set the register to H'0A090008 during the operation. If any combination of values not listed above is specified, correct operation is not guaranteed.

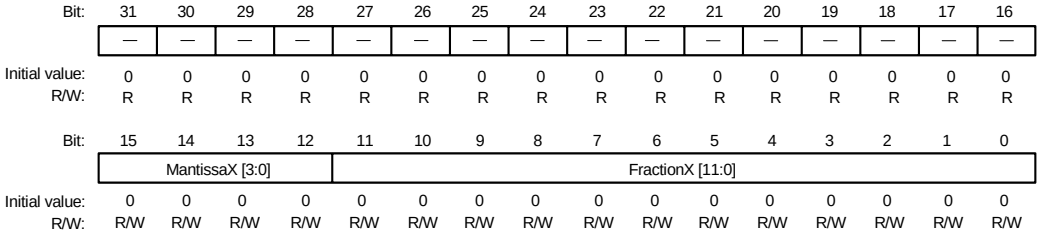
12.3.21 Y Scale Register (YS)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MantissaY[3:0]				FractionY[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 12	MantissaY [3:0]	H'0	R/W	The scaling ratio in the Y direction should be specified through MantissaY and FractionY. This register specifies the value of (number of capture lines per field) / (number of lines output to memory per field) and it is calculated from MantissaY and FractionY by the following equation. Y scaling = 4096 / (4096 x MantissaY + FractionY) For example, to obtain a Y scaling ratio of 1/2, specify MantissaY = H'2 and FractionY = H'000. Note that the maximum scaling-up ratio is × 3. When both MantissaY and FractionY are set to 0, the scaling function is disabled. To specify a scaling ratio of × 1, disabling of the scaling function is recommended.
11 to 0	FractionY [11:0]	H'000	R/W	

12.3.22 X Scale Register (XS)



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 12	MantissaX [3:0]	H'0	R/W	The scaling ratio in the X direction should be specified through MantissaX and FractionX. This register specifies the value of (number of input pixels per line) / (number of pixels output to memory per line) and it is calculated from MantissaX and FractionX by the following equation. X scaling = 4096 / (4096 x MantissaX + FractionX) As the multiphase filter specified by the coefficient set CmA (m = 1 to 8) registers are used for scaling, be sure to specify the coefficient set registers for scaling. Note that the maximum scaling-up ratio is ×2. When both MantissaX and FractionY are set to 0, the scaling function is disabled, To specify a scaling ratio of ×1, disabling of the scaling function is recommended. The following shows typical examples of this register setting and scaling ratio for 720 input pixels.
11 to 0	FractionX [11:0]	H'000	R/W	

Table 12.5 Examples of X Scaling Setting

Output Pixels	Scaling Ratio	Setting
680	0.943	H'10F8
800	1.111	H'0E68
854	1.185	H'0D80

12.3.23 Data Mode Register (DMR)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	YMODE	—	—	—	—	—	—	QWSM	BPSM	—	ABIT	DTMD[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R	R	R	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	YMODE	0	R/W	Y Data Transfer Mode This bit selects the Y and UV (CbCr) transfer mode when the output data conversion mode (DTMR) is set to 10 (YC separation). 0: Both Y and UV are transferred to memory. 1: Only Y is transferred to memory.
11 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	QWSM	0	R/W	Output Data Quadword Swap Mode 0: Quadwords (64 bits) are not swapped in output data. 1: Quadwords (64 bits) are swapped in output data.
4	BPSM	0	R/W	Output Data Byte Swap Mode 0: Bytes are not swapped in output data. 1: Bytes are swapped in output data. Note: When YCbCr-422 data is output in big endian, data is transferred in the YUYV format in most cases. To transfer data in the UYVY format, set this bit to 1.
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2	ABIT	0	R/W	Alfa Bit This bit specifies the alpha value for data in ARGB-1555 output mode. 0: The alpha value is set to 0. 1: The alpha value is set to 1.
1, 0	DTMD[1:0]	00	R/W	Data Conversion Mode These bits specify the conversion mode for the data output to external memory. 00: RGB-565 or YCbCr-422 data is output. 01: ARGB-1555 data is output. 10: Y and C are separated before output. 11: YCbCr-420 data is output (NV12 format) Note: When selecting the output data format, set the ABIT, YMODE, and BPS bits to appropriate values according to table 12.6, Data Conversion Settings.

Table 12.6 Data Conversion Settings

DMR		MC		
DTMD[1:0]	**YMODE**	**BPS**	**Output Data Format**	**Remarks**
00	0	0	RGB-565 transfer	
		1	YCbCr-422 transfer	
01	0	0	ARGB-1555 transfer	
10	0	1	YC separation transfer	Scaling up in the Y direction is prohibited.
	1	1	Y transfer (monochrome)	Scaling up in the Y direction is prohibited.
11	0	1	YCbCr-420 transfer	Scaling up in the Y direction is prohibited. Up to 1024 pixels can be captured in the X direction.

Note: If any combination of values not listed above is specified, correct operation is not guaranteed.

12.3.24 Data Mode Register 2 (DMR2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FPS	VPS	HPS	CES	—	—	—	—	—	—	—	—	—	—	FTEV	FTEH
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W

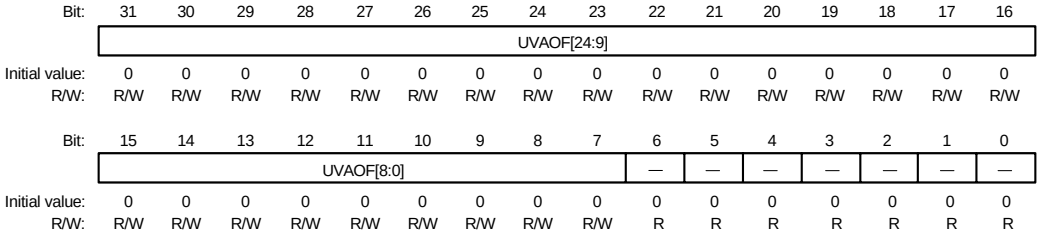
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VLV[3:0]				HLV[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	FPS	0	R/W	Field Signal Polarity Select This bit specifies the polarity of the input field signal in the ITU-R BT.601 interface. 0: 0/1 = Odd/Even 1: 0/1 = Even/Odd
30	VPS	0	R/W	Vsync Signal Polarity Select This bit specifies the polarity of the input vertical sync signal in the ITU-R BT.601 interface. 0: Active low 1: Active high
29	HPS	0	R/W	Hsync Signal Polarity Select This bit specifies the polarity of the input horizontal sync signal in the ITU-R BT.601 interface. 0: Active low 1: Active high
28	CES	0	R/W	Clock Enable Signal Polarity Select This bit specifies the polarity of the input clock enable signal in the ITU-R BT.601. 0: Active high 1: Active low
27 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
17	FTEV	0	R/W	VSYNC Field Toggle Mode Enable The VSYNC field toggle mode changes the capture field signal level according to the count of input VSYNC signal assertion. As the VIN controls capture operation only when the input field signal level changes, select the VSYNC field toggle mode when capturing progressive images. 0: The field toggle function according to the VSYNC count is disabled. 1: The field toggle function according to the VSYNC count is enabled. The period before a toggle should be specified in the VLV bits. Note: Do not set both FTEH and FTEV at the same time.
16	FTEH	0	R/W	HSYNC Field Toggle Counter Enable 0: The field toggle function according to the capture active line is disabled. 1: The field toggle function according to the capture active line is enabled. The period before a toggle should be specified in the HLV bits. Note: Do not set both FTEH and FTEV at the same time.

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	VLV[3:0]	H'0	R/W	VSYNC Field Toggle Mode Transition Period These bits specify the count of vertical sync signal input before the VSYNC field toggle mode is entered. After a transition to the VSYNC field toggle mode, the capture field signal is toggled every time VSYNC is input. When a change in the input field signal is detected, the toggle mode is canceled. H'0: The field signal is toggled at every VSYNC input. H'1: Toggle mode is entered after VSYNC is input once. H'2: Toggle mode is entered after VSYNC is input two times. H'3: Toggle mode is entered after VSYNC is input three times. : H'E: Toggle mode is entered after VSYNC is input 14 times H'F: Toggle mode is entered after VSYNC is input 15 times. Note: If the field signal changes while the transition period is counted, the counter is initialized.
11 to 0	HLV[11:0]	H'000	R/W	HSYNC Field Toggle Count Value The HSYNC field toggle counter counts the capture active lines. If the external field signal does not change before the prespecified counter value is reached, the capture field signal is toggled. H'000: The field signal is toggled for every valid line. H'001: The field signal is toggled for every single valid line. H'002: The field signal is toggled for every two valid lines. : H'FFF: The field signal is toggled for every 4095 valid lines. Note: For the period before a toggle, specify a value greater than one VSYNC period.

12.3.25 UV Address Offset Register (UVAOF)



Bit	Bit Name	Initial Value	R/W	Description
31 to 7	UVAOF [24:0]	H'0000 000	R/W	UV Data Address Offset This register specifies the offset address of UV data when YCbCr-422 and YCbCr-420 data are separated into Y and UV. Specify a value for physical address bits [31:7] in units of 128 bytes. Note: Specify an address greater than the Y transfer size. If the Y data area is smaller than the transfer size, the UV area is overwritten by Y data.
6 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

12.3.26 Color Space Change Coefficient 1 Register (CSCC1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	YMUL[9:0]									
Initial value:	0	0	0	0	0	0	0	1	0	0	1	0	1	0	0	1
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	YSUB[7:0]								CSUB[7:0]							
Initial value:	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25 to 16	YMUL[9:0]	H'129	R/W	Y Data Multiplication Coefficient These bits specify the multiplication coefficient for Y data in YC (YCbCr-422)→RGB-565 color space conversion. (Initial value: 1.164; common for R, G, and B) Specify an unsigned 10-bit integer obtained by multiplying the desired coefficient value by 256.
15 to 8	YSUB[7:0]	H'10	R/W	Y Data Subtraction Coefficient These bits specify the subtraction coefficient for Y data in YC (YCbCr-422)→RGB-565 color space conversion. (Initial value: H'16; common for R, G, and B) Specify an unsigned 8-bit integer.
7 to 0	CSUB[7:0]	H'80	R/W	CbCr Data Subtraction Coefficient These bits specify the subtraction coefficient for Cb and Cr data in YC (YCbCr-422)→RGB-565 color space conversion. (Initial value: H'128: common for R, G, and B) Specify an unsigned 8-bit integer.

12.3.27 Color Space Change Coefficient 2 Register (CSCC2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	RCRMUL[9:0]									
Initial value:	0	0	0	0	0	0	0	1	1	0	0	1	1	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	GCRMUL[9:0]									
Initial value:	0	0	0	0	0	0	0	0	1	1	0	1	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25 to 16	RCRMUL [9:0]	H'198	R/W	Cr Multiplication Coefficient for R Data Calculation These bits specify the Cr multiplication coefficient for the R data calculation equation in YC (YCbCr-422)→RGB-565 color space conversion. (Initial value: 1.596) Specify an unsigned 10-bit integer obtained by multiplying the desired coefficient value by 256.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	GCRMUL [9:0]	H'0D0	R/W	Cr Multiplication Coefficient for G Data Calculation These bits specify the Cr multiplication coefficient for the G data calculation equation in YC (YCbCr-422)→RGB-565 color space conversion. (Initial value: 0.813) Specify an unsigned 10-bit integer obtained by multiplying the desired coefficient value by 256.

12.3.28 Color Space Change Coefficient 3 Register (CSCC3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	GCBMUL[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	BCBMUL[9:0]									
Initial value:	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25 to 16	GCBMUL [9:0]	H'064	R/W	Cb Multiplication Coefficient for G Data Calculation These bits specify the Cb multiplication coefficient for the G data calculation equation in YC (YCbCr-422)→RGB-565 color space conversion. (Initial value: 0.392) Specify an unsigned 10-bit integer obtained by multiplying the desired coefficient value by 256.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	BCBMUL [9:0]	H'204	R/W	Cb Multiplication Coefficient for B Data Calculation These bits specify the Cb multiplication coefficient for the B data calculation equation in YC (YCbCr-422)→RGB-565 color space conversion. (Initial value: 2.017) Specify an unsigned 10-bit integer obtained by multiplying the desired coefficient value by 256.

The values specified in CSCC1 to CSCC3 are used as the coefficients in the following YC→RGB color space conversion equations.

$$R' = YMUL \times (Y - YSUB) + RCRMUL \times (Cr - CSUB)$$

$$G' = YMUL \times (Y - YSUB) - GCRMUL \times (Cr - CSUB) - GCBMUL \times (Cb - CSUB)$$

$$B' = YMUL\ (Y - YSUB) + BCBMUL \times (Cb - CSUB)$$

Note: The initial values are the coefficients specified in the ITU-R BT.601 standard. For details of each coefficient setting, refer to section 12.4.7, Color Space Conversion.

12.3.29 Coefficient Set Registers (CmA, CmB, CmC) (m = 1 to 8)

The coefficient set registers specify the coefficients used for calculation of X scaling specified in XS. The X scaling function processes input pixels with 8-step resolution and generates scaled pixels by using nine calculation coefficients (nine taps) around the determined pixel.

The following is an overview of nine tap coefficients. The MSB of each coefficient is a sign bit.

Table 12.7 Bit Count for Tap Coefficients

Register Name (m = 1 to 8)	CmA			CmB			CmC		
Nine tap coefficients	L1	L2	L3	L4	M	R4	R3	R2	R1
Bit width	10	10	10	10	10	10	10	10	10

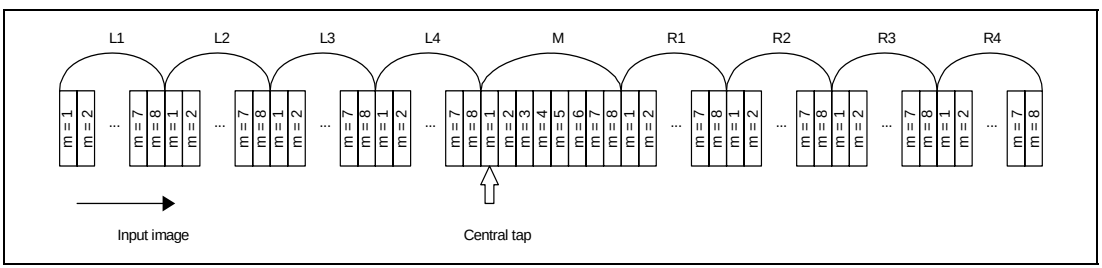


Figure 12.3 Bit Count for Tap Coefficients

(1) Coefficient Set CmA Register (CmA) (m = 1 to 8)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	L1[9:0]										L2[9:6]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	L2[5:0]						L3[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
29 to 20	L1[9:0]	H'000	R/W	L1 coefficient
19 to 10	L2[9:0]	H'000	R/W	L2 coefficient
9 to 0	L3[9:0]	H'000	R/W	L3 coefficient

(2) Coefficient Set CmB Register (CmB) (m = 1 to 8)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	R1[9:0]										R2[9:6]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R2[5:0]						R3[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
29 to 20	R1[9:0]	H'000	R/W	R1 coefficient
19 to 10	R2[9:0]	H'000	R/W	R2 coefficient
9 to 0	R3[9:0]	H'000	R/W	R3 coefficient

(3) Coefficient Set CmC Register (CmC) (m = 1 to 8)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	R4[9:0]										L4[9:6]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	L4[5:0]						M[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
29 to 20	R4[9:0]	H'000	R/W	R4 coefficient
19 to 10	L4[9:0]	H'000	R/W	L4 coefficient
9 to 0	M[9:0]	H'000	R/W	M coefficient

12.4 Operation

12.4.1 Input Interface

The VIN captures video data in the ITU-R BT.601, ITU-R BT.656, or ITU-R BT.1358 interface and stores it in external memory.

The following shows the interface and data format supported in each channel.

Table 12.8 Video Channels and Supported Interfaces

INF[2:0] in MC	ITU-R BT.656	ITU-R BT.601		ITU-R BT.1358
	8-Bit YCbCr-422	8-Bit YCbCr-422	18-Bit RBB-666	16-Bit YCbCr-422
000	Supported	—	—	—
001	—	Supported	—	—
101	—	—	—	Supported
111	—	—	Supported	—
Others	—	—	—	—

Note: When capturing progressive images (ITU-R BT.1358), be sure to enable the internal field signal generation function.

This module uses external sync signals to control capturing. As frames are controlled according to the field signal, correct control cannot be achieved if the field signal level in the ITU-R BT.601 interface or SAV/EAV (F bit) in the ITU-R BT.656 interface does not change. Be sure to input a correct field signal and timing reference code.

In the capture processing, the input interface and data format are converted to the ITU-R BT.656 8-bit YCbCr-422 format internally. Accordingly, note that video capture data is stored in memory with 8-bit precision.

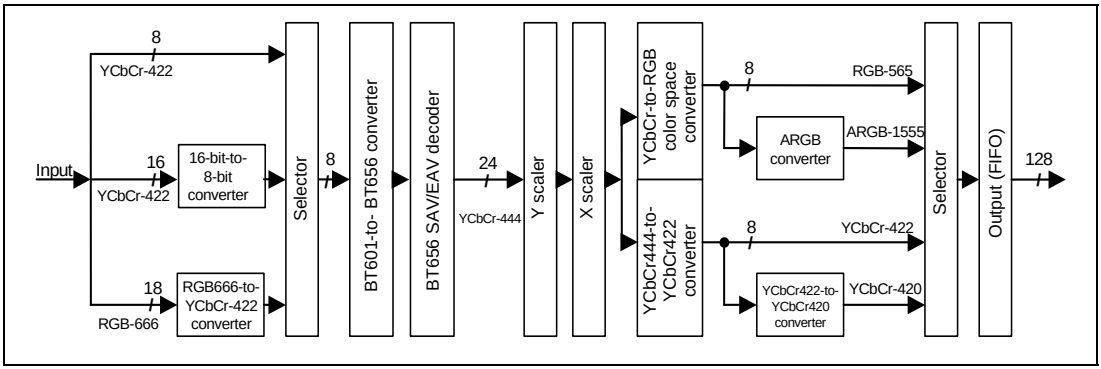


Figure 12.4 Data Format Conversion Flow in Video Input Module

Note: As the VIN converts data into the 8-bit YCbCr-422 format before capture control, the scaling-up function is not available because of the traffic limitation in some cases depending on the sampling frequency of the input interface. Refer to "Limits on input video clock" in table 12.11.

(1) ITU-R BT.601 or ITU-R BT.656 YCbCr-422 8-Bit Data Format

The 8-bit YCbCr-422 data in the ITU-R BT.601 or ITU-R BT.656 interface has the multiplexed YCbCr = 4:2:2 format. The data is constructed in the UYVY format (Cb0Y0Cr0Y1 format).

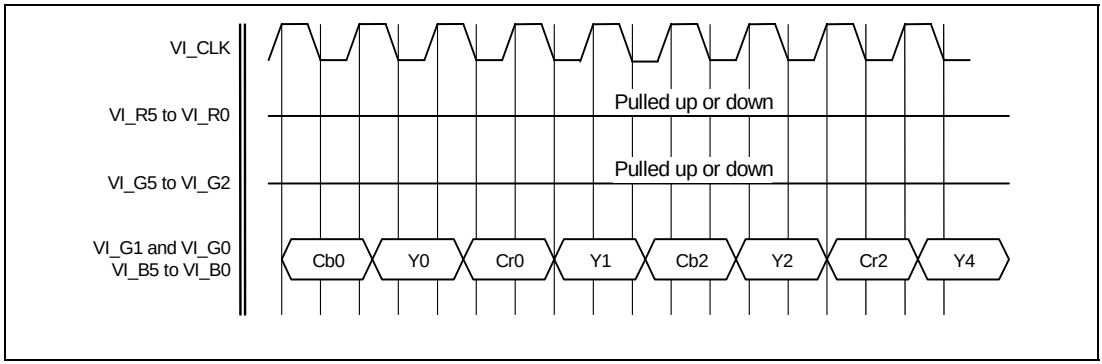


Figure 12.5 ITU-R BT.601 or ITU-R BT.656 YCbCr-422 8-Bit Data Format

(2) ITU-R BT.601 YCbCr-422 16-Bit Interface

The 16-bit YCbCr-422 data in the ITU-R BT.601 interface has the YCbCr = 4:2:2 format consisting of 8 bits (Y) + 8 bits (CbCr).

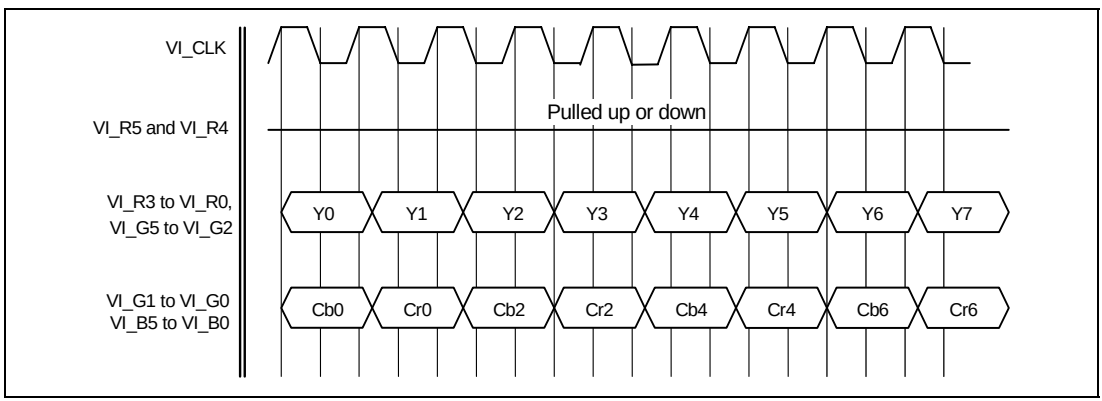


Figure 12.6 ITU-R BT.601 YCbCr-422 16-Bit Data Format

The 16-bit data is converted (multiplexed) into the 8-bit data format according to the internal sampling clock before capture control. In this format conversion, the word swap function is available by setting the YCAL bit in the main control (MC) register.

(3) ITU-R BT.601 RGB-666 18-Bit Interface

The 18-bit RGB-666 data in the ITU-R BT.601 interface has the RGB = 6:6:6 format whose color space conforms to that specified in the ITU-R BT.601 or ITU-R BT.709 standard. Specify the input color space through the CFSL bit in the main control (MC) register.

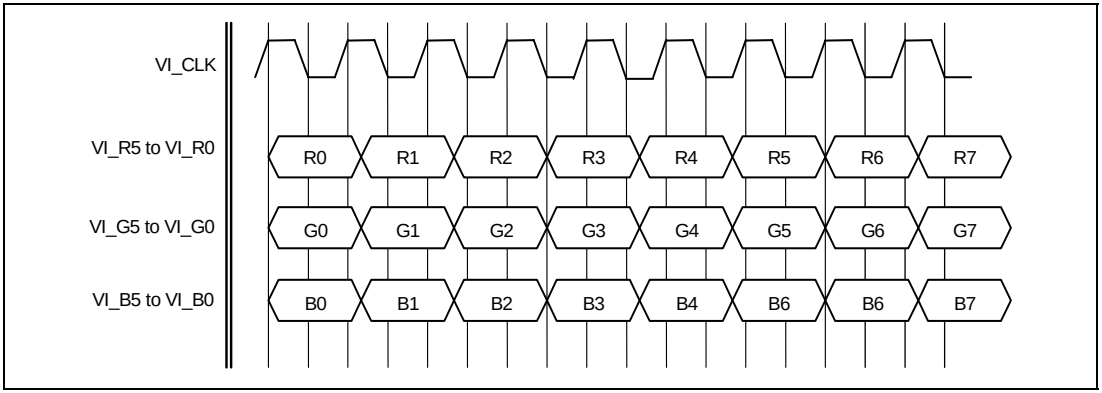


Figure 12.7 ITU-R BT.601 RGB-666 18-Bit Data Format

12.4.2 Error Correction

At the time of capturing with ITU-R BT.656, the VIN allows errors to be corrected with the timing reference code (SAV/EAV). The timing reference code (SAV/EAV) of ITU-R BT.656 has four protection bits, which can be used to correct only one-bit errors in the interface.

If the VIN cannot correct errors while the CEE bit in the interrupt enable register (IE) has been set to 1, an interrupt signal is generated as soon as the CES bit in the interrupt status (INTS) register is set. Note that no interrupt signal is generated if errors can be corrected.

12.4.3 Capture Mode

With the VIN, either of the single frame capture mode or continuous frame capture mode can be selected.

Specifying the capture field in IM bits of the main control (MC) register and then setting the SC bit in the frame capture (FC) register to 1 provides single frame capture mode. In this mode, the current frame is captured when the SC bit write timing (current scanline position) is smaller than the value of the start line pre-clip (SLPrC) register, or the next frame is captured in other cases. The capture data is transferred to the memory address that is set in the memory base 1 (MB1) register.

Specifying the capture field in IM bits of MC and then setting the CC bit of FC to 1 provides continuous frame capture mode, in which capture data is sequentially transferred to the addresses that are set in MB1 to MB3. In this case, the latest captured frame ID is shown in the FBS bits in the module status (MS) register.

When the IM bits in MC are set to the full interlace mode, data in the capture start field is stored in every other line in the memory as the top field data and then data in the next field is stored between the written lines so that the top and bottom field lines alternate with each other for interlace composition. The capture start field (top field) can be changed through the FOC bit in MC.

The following is a schematic diagram of capturing in full interlace mode when the odd field is selected as the top field, the memory width is set to H'200, and MB1 is set to H'0000.

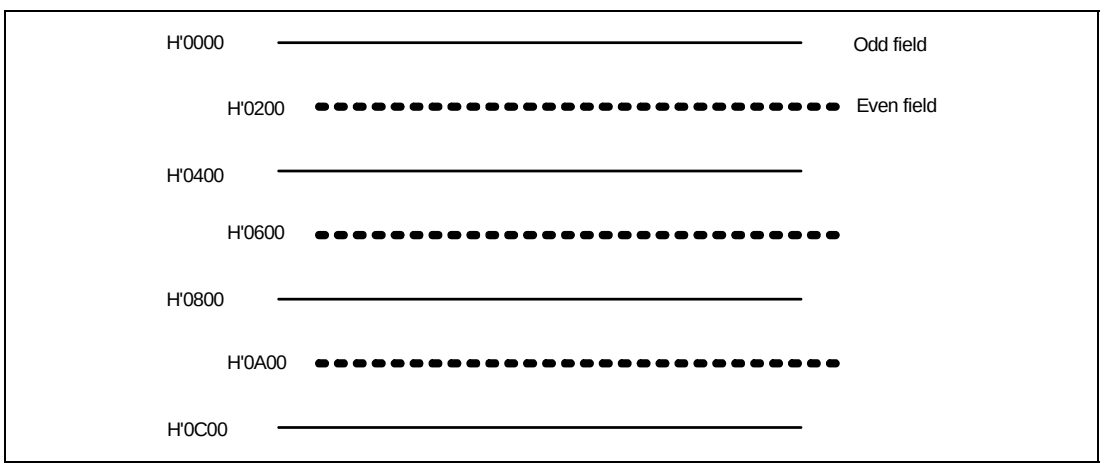


Figure 12.8 Example of Capturing Full Interlace

The VIN starts capture operation after detecting a frame signal switching in the ITU-R BT.601 or 656 interface. When capturing progressive data, in which the field signal does not change, use the internal field signal generation function to toggle the internal field signal.

12.4.4 Size Clipping

Image data that has been captured is pre-clipped according to the settings of the following registers: start line pre-clip (SLPrC), end line pre-clip (ELPrC), start pixel pre-clip (SPPrC), and end pixel pre-clip (EPPrC).

After the horizontal and vertical scaling, post-clipping takes place according to the settings of the following registers: start line post-clip (SLPoC), end line post-clip (ELPoC), start pixel post-clip (SPPoC), and end pixel post-clip (EPPoC). The following shows an example of size clipping.

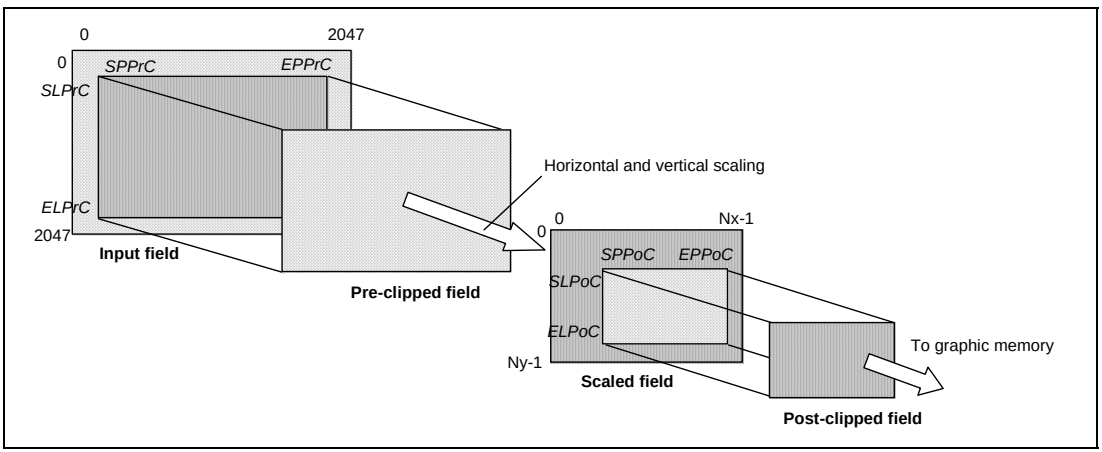


Figure 12.9 Example of Clipping

For all the post-clipped lines, the lengths of individual lines written into the memory are defined by the image stride (IS) register. The setting can be larger than the post-clipped frame width, but cannot be smaller than the width. The IS register must be filled with a value larger than the horizontal post-clip width. The input field in the above figure shows an effective image area from the video decoder; the VIN does not allow anything exceeding the image area to be captured.

Note: Each of the following registers specifies a distance from the starting point in the effective image area: start line pre-clip (SLPrC), end line pre-clip (ELPrC), start pixel pre-clip (SPPrC), and end pixel pre-clip (EPPrC). Specifically, in ITU-R BT.601, the distance is from the starting point of VI_CLKENB; in ITU-R BT.656, the distance is from the SAV (start of active video) signal.

12.4.5 Vertical Scaling

The vertical scaling function in the VIN creates lines by scaling up or down in the vertical direction through interpolation between two points in the neighborhood of captured lines. With the combinations of mantissaY and fractionY in the Y scaling (YS) register, the vertical scaling creates new lines by selecting line positions from captured lines.

The Y scaling function is disabled if mantissaY and fractionY are both cleared to 0 in YS.

Scaling down is implemented by creating fewer lines than captured lines. Examples of scaling up and down are shown in the following figure.

Note: When vertical scaling up is specified in full interlace capture mode, the scaling-up processing is applied separately to each field before full interlace composition on memory. Accordingly, the top and bottom field lines are inverted in some cases depending on the scaling ratio. Be sure to evaluate the scaled image quality before practical application.

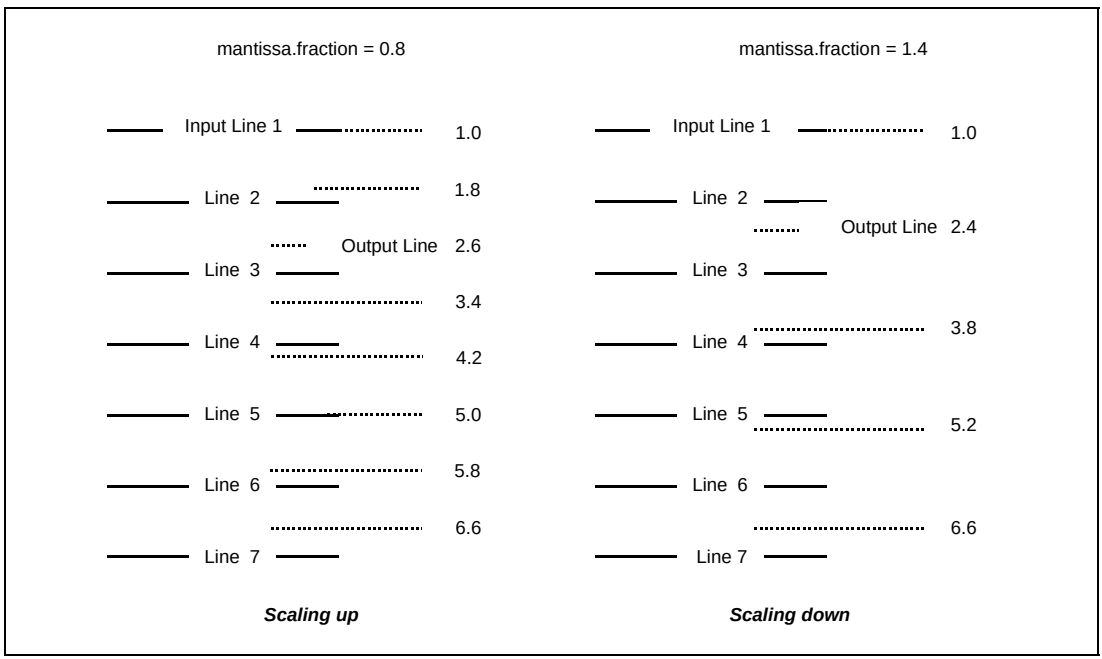


Figure 12.10 Examples of Scaling Up and Down in Vertical Direction

The number of lines created by the vertical scaling block is expressed by the following equation:

$$Ny = \begin{cases} \dfrac{4096 \times (ELPrC - SLPrC)}{4096 \times MantissaY + FractionY} - 1, \\ \qquad \text{when } \{4096 \times (ELPrC - SLPrC)\} \% (4096 \times MantissaY + FractionY) = 0 \\ Int\left(\dfrac{4096 \times (ELPrC - SLPrC)}{4096 \times MantissaY + FractionY} \right), \text{ otherwise} \end{cases}$$

where:
ELPrC is the value in the end line pre-clip (ELPrC) register.
SLPrC is the value in the start line pre-clip (SLPrC) register.
MantissaY and FractionY are the values in YS.

12.4.6 Horizontal Scaling

Using a 9-tap multiphase filter, the VIN creates pixels by scaling down in the horizontal direction. The horizontal scaling-up doubles the number of input pixels captured and provides the function of expansion of up to two-fold by the scaling-down in the 9-tap multiphase filter. The horizontal scaling, which is set through combinations of mantissaX and fractionX in the X scale (XS) register, determines the position of new pixels with a poly-phase filter. The selected coefficient, which is one of the eight coefficients, is determined by the position of output pixels.

In the example shown below, mantissaX and fractionX are set at 1.2. In this case, coefficient set C2 is selected.

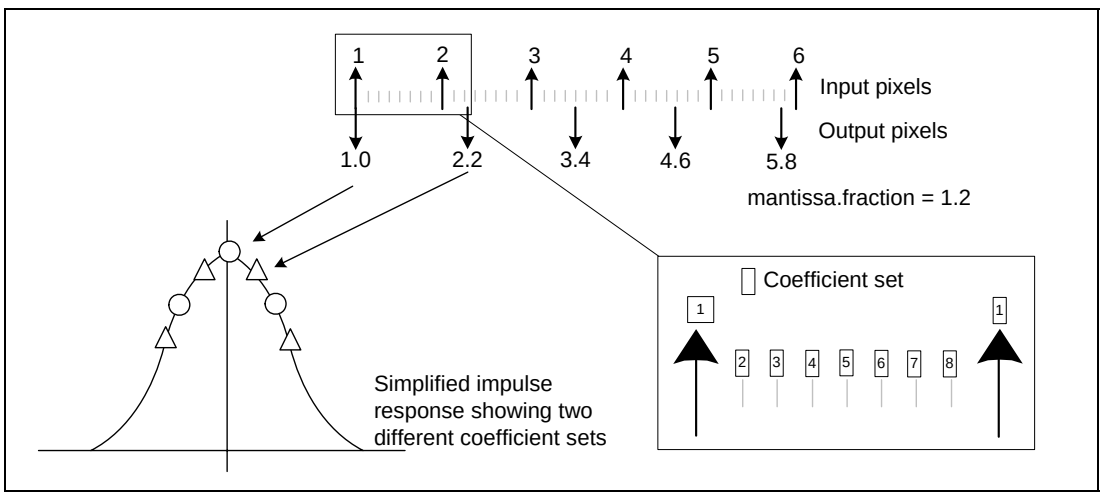


Figure 12.11 Pixel Position and Coefficient Set

The following figure shows an example in which different coefficient sets are used for different output pixel positions. Nine coefficient sets are assigned to each. A total of 72 coefficients are used in the horizontal scaling. Each of the coefficients in these sets has a width of 10 bits, where the MSB serves as the sign bit. By setting the same coefficients into the entire eight coefficient set registers, the horizontal scaling allows the use of a multiphase filter just like a single-phase filter.

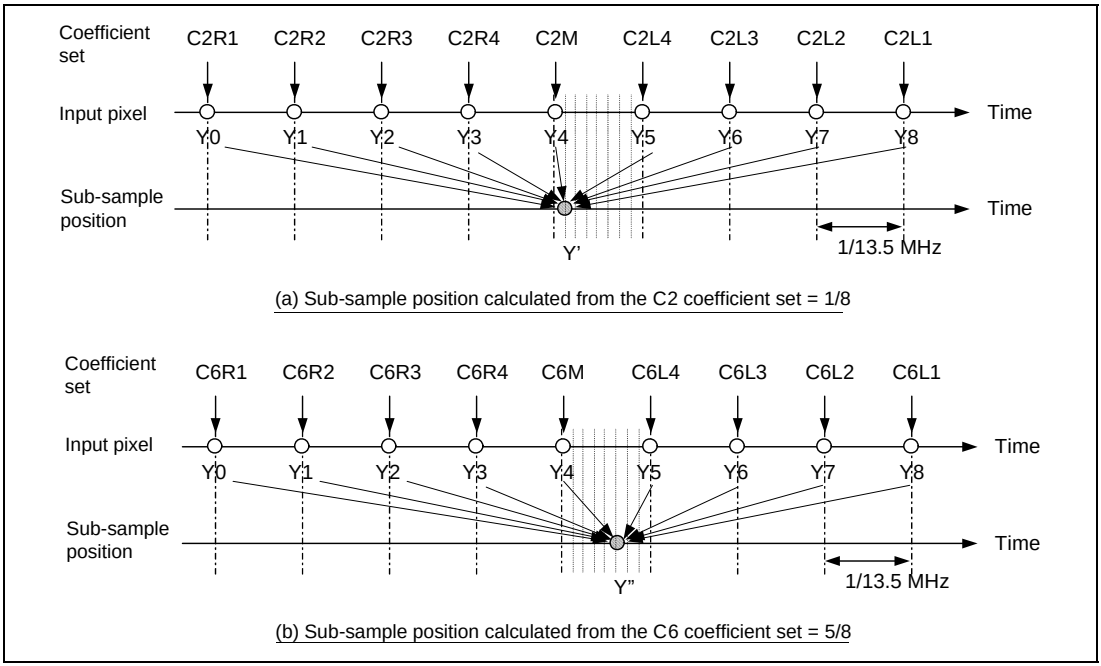


Figure 12.12 Example of Coefficients in Selected Coefficient Set

This scaling mechanism requires different coefficient values dependent on the scaling ratio. Here is an example of coefficient sets to be selected. The coefficients, CmM, CmLi, and CmRi (m = 1, 2, 3, - - - , 8; i = 1, 2, 3, and 4), are determined by the following equations:

$$C_nM = \beta \cdot h(-(n-1))$$
$$C_nR_i = \beta \cdot h(-(n-1)-8(5-i))$$
$$C_nL_i = \beta \cdot h(-(n-1)+8(5-i))$$
$$h(t) = \frac{\sin\left(\frac{\pi t}{T}\right)}{\frac{\pi t}{T}} \cdot \frac{\cos\left(\frac{\alpha \pi t}{T}\right)}{1-\frac{4\alpha^2 t^2}{T^2}}$$
$$T = 8 \times MantissaX + FractionX[11:9]$$

Mantissa X and Fraction X are the values in XS.

The parameter β is a standardized one. In the equation, h(t) follows the cosine characteristic and is executed with a 9-tap filter if the value of α is in the $0 < \alpha \leq 1$ range.

Obtaining dependable scale images requires you to use and meet the following equation in which all the coefficient sets are standardized.

This must be executed with the inherent β value that has been selected from the equation above.

$$C_n M + \sum_{i=1}^{4} C_n R_i + \sum_{i=1}^{4} C_n L_i = 512$$

The number of pixels created by the horizontal scaling block is calculated by the following equation.

$$Nx = Int\left(\frac{4096 \times (EPPrC - SPPrC)}{4096 \times MantissaX + FractionX}\right) + 1$$

where:
EPPrC is the value in the end pixel pre-clip (EPPrC) register.
SPPrC is the value in the start pixel pre-clip (SPPrC) register.
Mantissa X and Fraction X are the values in XS.

12.4.7 Color Space Conversion

The color space of the YCbCr data after scaling can be converted to the RGB color space. Color space conversion from 8-bit internal YCbCr data to R'G'B'-888 data uses the 3×3 matrix specified in the CSC coefficient 1 to 3 registers (CSCC1 to CSCC3).

Each matrix coefficient can be set to a desired value to flexibly meet the input data range. Captured YCbCr data can also be stored in memory without RGB conversion through the BPS bit setting in the main control register (MC).

The CSC coefficient 1 to 3 register (CSCC1 to CSCC3) settings for color space conversion correspond to input YCbCr data as shown below. The R'G'B'-888 data calculated by the following equations is in the internal data format, and it is converted to RGB-565 before being stored in memory.

$$R' = YMUL\ (Y - YSUB) + RCRMUL\ (Cr - CSUB)$$

$$G' = YMUL\ (Y - YSUB) - GCRMUL\ (Cr - CSUB) - GCBMUL\ (Cb - CSUB)$$

$$B' = YMUL\ (Y - YSUB) + BCBMUL\ (Cb - CSUB)$$

In the above equations,

YMUL: 8-bit data range (255) / Y signal data level (level 219 ($16 \leq Y \leq 235$))

YSUB: Y data normalization offset (16 when $16 \leq Y \leq 235$)

CSUB: Cb and Cr data normalization offset (128 when the center value is 128)

RCRMUL: Cr coefficient for red

GCRMUL: Cr coefficient for green

GCBMUL: Cb coefficient for green

BCBMUL: Cb coefficient for blue

The following shows examples of CSC coefficient 1 to 3 register (CSCC1 to CSCC3) settings. For details of each coefficient, refer to the description of the corresponding register.

Table 12.9 Examples of CSC Coefficient 1 to 3 Register Settings

Color Space Conversion Example	CSCC1			CSCC2		CSCC3	
	YMUL	YSUB	CSUB	RCRMUL	GCRMUL	GCBMUL	BCBMUL
ITU-R BT.601 (initial value) 16 ≤ Y ≤ 235, 16 ≤ Cb, Cr ≤ 240	1.164	16	128	1.596	0.813	0.392	2.017
Luminance expansion 1 ≤ Y ≤ 254, 16 ≤ Cb, Cr ≤ 240	1.008	1	128	1.596	0.392	0.813	2.017

Specify an integer in each addition coefficient bit field. For each multiplication coefficient, specify a value obtained by multiplying the desired coefficient value by 256.

Example: When the desired multiplication coefficient is 1.164

$1.164 \times 256 = 297$ (set value: b'0100101001)

Note: Specify appropriate values in the CSC coefficient 1 to 3 registers (CSCC1 to CSCC3) to keep the RGB image data within the range 0 ≤ R', G', B' ≤ 255. The RGB value is set to 0 when it is below 0, and set to 255 when it exceeds 255.

12.4.8 Dithering

By using a method of error diffusion that adds the lower bits of each RGB data, dithering is performed when the internal R'G'B'-888 format after color space conversion is converted to the RGB-565 format. This function is always executed during RGB conversion.

$$R_n[7:3] <= (R'_n[7:0] + R'_{n-1}[2:0]) >> 3$$
$$G_n[7:2] <= (G'_n[7:0] + G'_{n-1}[1:0]) >> 2$$
$$B_n[7:3] <= (B'_n[7:0] + B'_{n-1}[2:0]) >> 3$$

Where:
(R_n, G_n, B_n) = Output RGB-565 pixel
(R'_n, G'_n, B'_n) = Input R'G'B'-888 pixel
(R'_{n-1}, G'_{n-1}, B'_{n-1}) = Pseudo random error (LSB of the cumulative error)

12.4.9 Internal Field Signal Generation

As the video input module controls capture of data in interlaced mode, correct capture control is not achieved if the external field signal level does not change.

Through the internal field signal generation function, the VIN can control the capture field signal even when the input field signal does not change, such as during progressive data capturing. The following settings can be made for the internal field generation function through the FTEV and FTEH bits in the data mode register 2 (DTMR2).

- VSYNC field toggle mode (FTEV = 1 in DTMR2)
  When this setting is made, the VSYNC field toggle mode is entered for capture field signal control if the input field signal does not change for the VSYNC cycles specified by the VLV bits in DTMR2. The toggle mode is canceled when a change in the external field signal level is detected (the capture operation is controlled according to the input field signal).
- HSYNC field toggle counter (FTEH = 1 in DTMR2)
  This counter counts the capture active lines. If the external field signal does not change until the count reaches the HLV setting in DTMR2, the capture field signal is controlled.

Note: Do not set both the FTEV and FTEH bits in DTMR2 at the same time.
Immediately after cancellation of the toggle mode, capture control is skipped for one VSYNC cycle in some cases depending on the input field signal status.

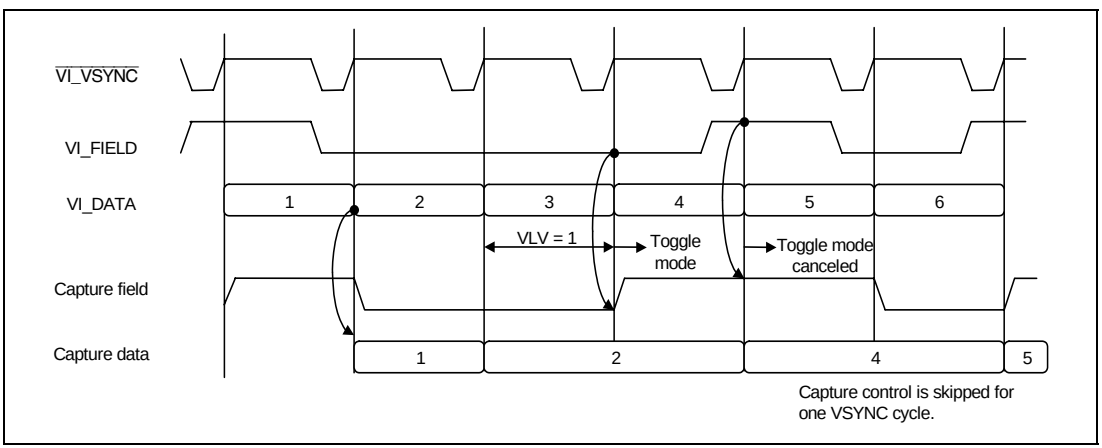


Figure 12.13 VSYNC Field Toggle Mode Overview and Notes

12.4.10 YUV Data Separation into Y and UV Data

The YUV data can be separated into Y and UV (CbCr) data, each of which can be transferred to separate address areas.

Set the DTMD[1:0] bits in the data mode register (DMR) to 10 to enable YUV separation transfer.

Y data is transferred to the address specified in the memory base register, and UV data is transferred to the memory base address + value specified in the UV address offset register (UVAOF).

Set the YMODE bit in DMR to 1 to transfer only the separated Y data. (In this mode, the UV data is not transferred to memory.)

Note: Do not specify scaling up in the Y direction in this mode.

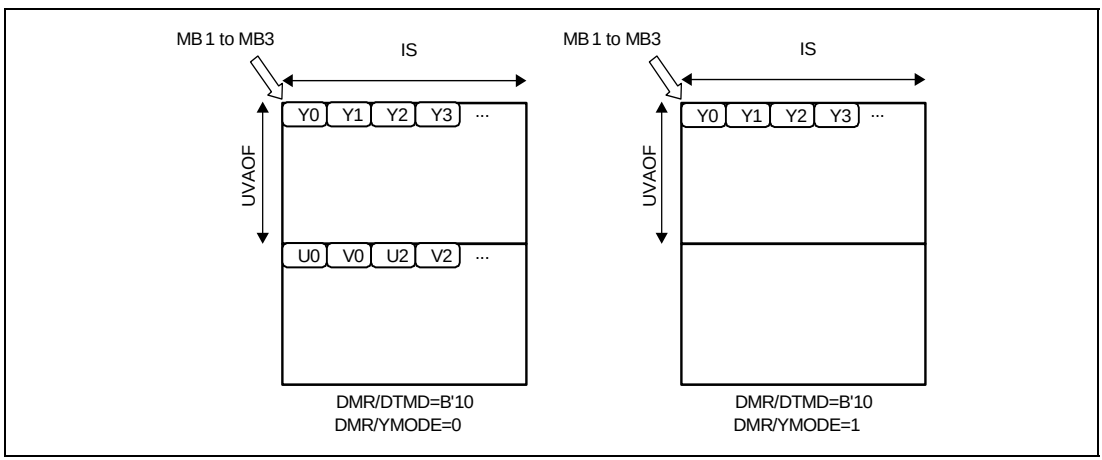


Figure 12.14 Y and UV Separation in Big Endian

12.4.11 YCbCr-422→YCbCr-420 Conversion

Set the DTMD[1:0] bits in the data mode register (DMR) to 11 to convert the YCbCr-422 format into the YCbCr-420 format in the NV12 format.

In this mode, 2-line average data is generated from the UV (CbCr) data in the capture field and 2-line data is transferred to memory at one time.

Y data is transferred at every line in the same manner as in the YUV separation mode.

UV (CbCr) data is transferred to the memory base address + value specified in the UV address offset register (UVAOF) as in the YUV separation mode.

Notes: 1. Do not specify scaling up in the Y direction in this mode.
2. The capture size in the X direction is up to 1024 pixels in this mode.

12.4.12 Output Data Format

The VIN can output image data in the following formats. The figures in this section assume that data is stored in unified memory in little endian.

(1) YC: YCbCr-422

The YUV image data in the YC (YCbCr) = 4:2:2 format is shown below. YC data can be switched between the UYVY format and YUYV format through the BPSM bit in DMR.

When big endian is selected through the EN bit in MC, the YUYV format is selected when BPSM = 0 in DMR and the UYVY format is selected when BPSM = 1.

- BPSM = 0 in DMR: UYVY format

YCbCr-422 data (UYVY format)

	63	56	55	48
D63 to D48 Image data 3 and 4	Y3		V2	

	47	40	39	32
D47 to D32 Image data 3 and 4	Y2		U2	

	31	24	23	16
D31 to D16 Image data 1 and 2	Y1		V0	

	15	8	7	0
D15 to D0 Image data 1 and 2	Y0		U0	

- BPSM = 1 in DMR: YUYV format

YCbCr-422 data (YUYV format)

	63	56	55	48
D63 to D48 Image data 3 and 4	V2		Y3	

	47	40	39	32
D47 to D32 Image data 3 and 4	U2		Y2	

	31	24	23	16
D31 to D16 Image data 1 and 2	V0		Y1	

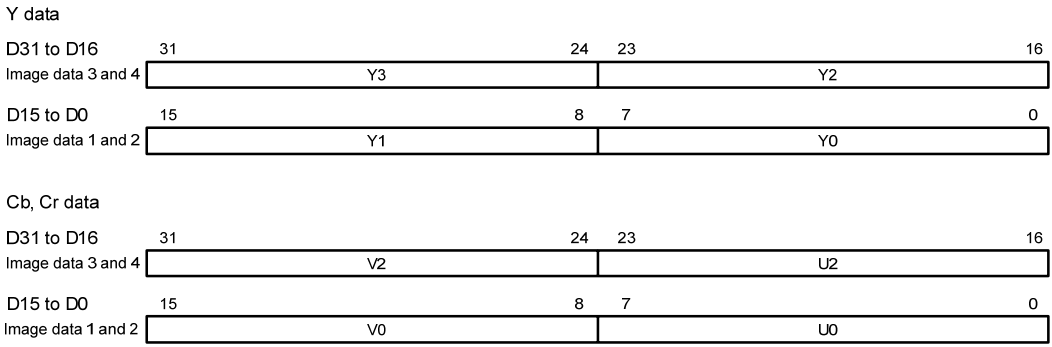
	15	8	7	0
D15 to D0 Image data 1 and 2	U0		Y0	

(2) YC Separation: YCbCr-422 and YCbCr-420

The YUV image data in the YC (YCbCr) = 4:2:2 or YC (YCbCr) = 4:2:0 format is shown below. Only the NV12 format is supported for UV data.

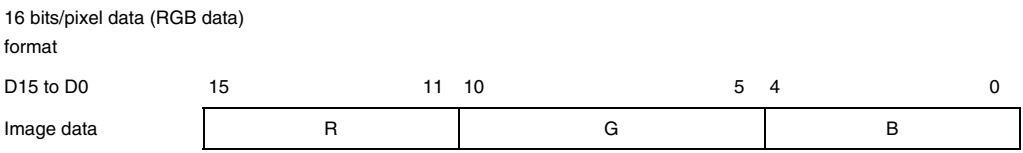
Set the DTMD[1:0] bits in the data mode register (DMR) to 10 to specify the 4:2:2 format, and set to 11 to specify the 4:2:0 format.

UV data is separately transferred to the memory base address + value specified in the UV address offset register (UVAOF). UV data is not transferred when only Y data transfer is selected.



(3) 16 Bits/Pixel: RGB-565

The RGB levels are expressed through 5 bits for R, 6 bits for G, and 5 bits for B.

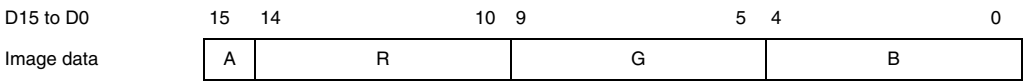


(4) 16 Bits/Pixel: ARGB-1555

The ARGB levels are expressed through 1 bit for A, 5 bits for R, 5 bits for G, and 5 bits for B. For data conversion to ARGB-1555, the lowest bit of the G data in RGB-565 data is truncated, and the A value specified through the register is added.

Set the DTMD[1:0] bits in the data mode register (DMR) to 01 to specify conversion to ARGB-1555, and specify the A value in the ABIT bit in DMR.

16 bits/pixel data (ARGB data) format



12.4.13 Endian Conversion

The VIN stores captured data in memory in little endian with the initial settings. Set the EN bit in the main control register (MC) to 1 to convert data into big endian before storing in memory.

Endian conversion in word units is controlled through the EN bit in MC, and swapping in byte units is controlled through the BPSM bit in the data mode register (DMR). For conversion to big endian, specify the BPSM bit as shown in the following table according to the data format specified through the DTMD bits in DMR and BPS bit in MC.

Table 12.10 Data Format and Conversion to Big Endian

Data Format	DTMD in DMR	BPS in MC	BPSM in DMR	Endian Conversion Unit
RGB-565	B'00	0	0	Word units
YCbCr-422	B'00	1	1	Byte units
ARGB-1555	B'01	0	0	Word units
YC	B'10 or B'11	1	0	Byte units

The following figure shows endian conversion in byte and word units.

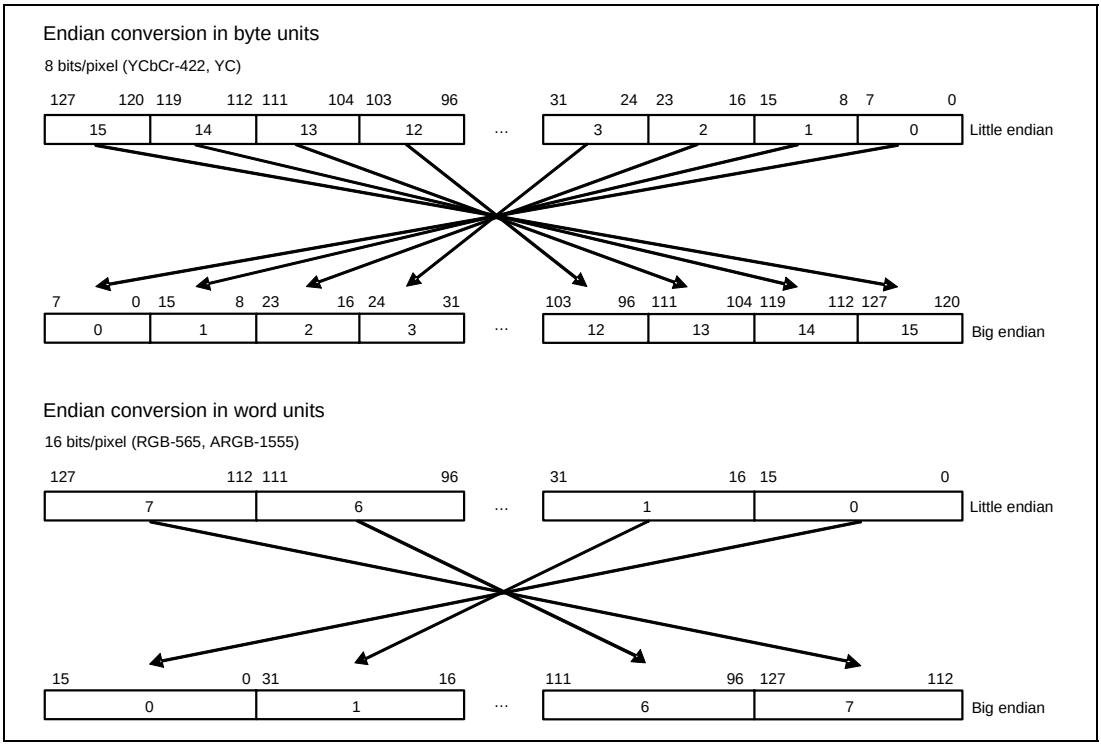


Figure 12.15 Data Alignment Conversion from Little Endian to Big Endian

In addition to the above endian conversion, quadword-unit (64-bit-unit) swap control can be specified through the QWSM bit in the data mode register (DMR). By means of this swap control, image data can be processed and transferred to a 64-bit module in a different endian.

The following shows quadword-unit swap conversion of 16 bits/pixel data.

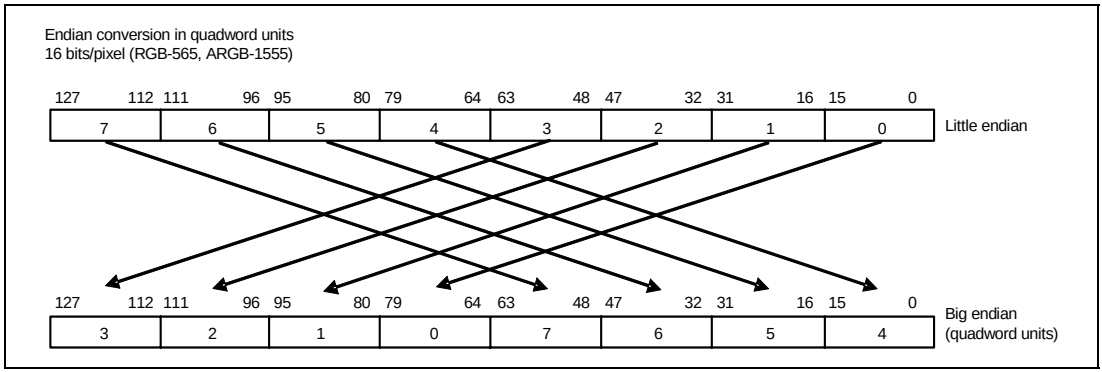


Figure 12.16 Data Swap Conversion in Quadword Units

12.5 Usage Notes

12.5.1 Module Standby Mode

The LSI supports module standby mode in which clock supply to the video input module is stopped. The video input module should not be accessed during module standby mode.

12.5.2 Transition to Module Standby Mode

1. Clear the module enable (ME) bit in the main control register (MC) and the continuous frame capture (CC) bit and the single frame capture (SC) bit in the frame capture register (FC) to 0 to stop the video input module.
2. Confirm that the capture active (CA) bit in the module status register (MS) is cleared to 0.
3. Stop the clock supply.

12.5.3 Cancellation of Module Standby Mode and Restarting of Video Input Module

1. Start the clock supply.
2. Set the module enable (ME) bit in the main control register (MC) to 1 to start the video input module.
3. Set the continuous frame capture (CC) bit or the single frame capture (SC) bit in the frame capture register (FC) to 1.

12.5.4 Limitations on Usage

The VIN does not operate correctly in some cases depending on the usage. The following shows cases that require attention.

Table 12.11 Limitations on Usage

Item	Description
Limits on input video clock	The upper limit on frequency of the input video clock varies with the clock mode, capture interface and scaling ratio. The upper limits on frequency of the video clock for this module on its own are shown below. Usage of other modules may make reaching the upper limits impossible. Fully evaluate performance before deciding on the settings for the input video clock, particularly if you are using magnification. Correct capture operations cannot be guaranteed if these limits are not observed.

1. Upper limits on video-clock frequency for this module on its own

- 177-MHz Clock Mode

Capture Mode	Horizontal Scaling Ratio XS	Vertical Scaling Ratio YS		
		0.0 ≤ YS ≤ 1.0	1.0 < YS ≤ 2.0	2.0 < YS ≤ 3.0
ITU-R BT.601 8-bit YCbCr-422, ITU-R BT.656 8-bit YCbCr-422	0.0 ≤ XS ≤ 1.0	80 MHz	80 MHz	80 MHz
	1.0 < XS ≤ 2.0	45 MHz	45 MHz	40 MHz
ITU-R BT.601 18-bit RGB-666, ITU-R BT.1358 16-bit YCbCr-422	0.0 ≤ XS ≤ 1.0	40 MHz	40 MHz	40 MHz
	1.0 < XS ≤ 2.0	20 MHz	20 MHz	15 MHz

- 200-MHz Clock Mode

Capture Mode	Horizontal Scaling Ratio XS	Vertical Scaling Ratio YS		
		0.0 ≤ YS ≤ 1.0	1.0 ≤ YS ≤ 2.0	2.0 < YS ≤ 3.0
ITU-R BT.601 8-bit YCbCr-422, ITU-R BT.656 8-bit YCbCr-422	0.0 ≤ XS ≤ 1.0	80 MHz	80 MHz	80 MHz
	1.0 < XS ≤ 2.0	52 MHz	52 MHz	45 MHz
ITU-R BT.601 18-bit RGB-666, ITU-R BT.1358 16-bit YCbCr-422	0.0 ≤ XS ≤ 1.0	45 MHz	45 MHz	45 MHz
	1.0 < XS ≤ 2.0	22 MHz	22 MHz	20 MHz

Item	Description
ITU-R BT.601 18-bit RGB-666 capturing	ITU-R BT.601 18-bit RGB-666 input data is first converted to YUV data and then capture processing is applied. Note that the quality is degraded in comparison with the input image.
Limitation on register update	If a register is updated during capture, data captured immediately after the register update cannot be guaranteed. To update a register which supports the internal update mode as shown in table 12.3 VIN Registers, specify an internal update for VIN registers by using the MC/VUP bit. To update other registers, stop capture operation and then update them.
Video display operation	With video display using this module, a capture frame and display frame are updated asynchronously. This disables smooth video display faithful to capture frames.
Field capture mode image quality	Images of odd-numbered, odd/even-numbered and even-numbered fields, captured by MC/IM interlace mode bit settings, contain every other line from the input interlace images. Therefore, note that the horizontal resolution for video display is in units of fields.
Full interlace mode image quality	In full interlace composition mode, horizontal stripe noise (such as combing noise) is generated in composite images as fields based on different timelines are combined in memory due to the interlacing method.
Limitation on vertical scaling	For vertical scaling and full interlace composition, the capture lines are inverted in some cases depending on the scaling ratio because the scaling processing is applied before interlace composition in memory. Be sure to evaluate the image quality before practical application.
Interrupt event timing	Interrupt event asserted by this module indicates the time when an interrupt event occurs in VIN, not the time when the transfer of capture data to memory is completed.
YCbCr-420 data format conversion	YCbCr-420 data format conversion is impossible when vertical scaling up is specified.
	In full interlace capture mode, YCbCr-420 data format conversion is done in field units before interlace composition in memory due to the capture method for this module. Accordingly, the UV data in memory is not UV data generated between input interlace images.
Pixel post-clip setting	After pixel post-clip setting is performed for the clipping data, this data is transferred in YCbCr-422 data format to memory. Therefore, if the EPPoC-SPPoC clipping size is an odd number, it is normalized to an even size which is the sum of that odd number and 1.
Coefficient settings for color space conversion	Specify appropriate values in the CSC coefficient 1 to 3 registers (CSCC1 to CSCC3) to keep the RGB image data within the range $0 \leq R', G', B' \leq 255$. After calculation for color space conversion, minus pixel data is normalized to 0 and pixel data more than or equal to 255 is normalized to 255.

<table>
<tr><th>Item</th><th>Description</th></tr>
<tr><td>Scaling up</td><td>When horizontal and vertical scaling up is specified, the amount of memory transfer becomes greater with an increase in traffic. Note the amount of traffic on the entire system when using the scaling-up function because the overall transfer efficiency of the system might degrade due to the increased use of the internal buses.</td></tr>
<tr><td>Upper limit of pixels captured in one line</td><td>The pixels to be captured in one line are limited as follows due to the scaling setting and output data type.
<table>
<tr><th rowspan="2">Condition</th><th colspan="2">Upper Limit of Pixels Captured in One Line</th></tr>
<tr><th>Pre-Clip</th><th>Post-Clip</th></tr>
<tr><td>X-direction scaling up and Y-direction scaling specified together</td><td>768 pixels</td><td>1536 pixels</td></tr>
<tr><td>Y-direction scaling up (no X-direction scaling)</td><td>1536 pixels</td><td>1536 pixels</td></tr>
<tr><td>Output data format is YCbCr-420</td><td>2048 pixels</td><td>1024 pixels</td></tr>
<tr><td>Others</td><td>2048 pixels</td><td>2048 pixels</td></tr>
</table>
There is no limitation on the number of lines.</td></tr>
<tr><td>Limitation on YC separation function</td><td>The offset register (UVAOF) that controls the address for storage of UV data should be set so that the regions for storage of Y data and of UV data do not overlap. If the regions where Y and UV data are stored overlap, Y data will be overwritten.
Vertical scaling up, YC separation transfer, and transfer settings are prohibited.</td></tr>
</table>

12.6 Supplementary Information

12.6.1 Example of Coefficient Set Register Settings for Horizontal Scaling

Table 12.12 shows the example of coefficient set register settings for horizontal (X) scaling (Recommended values).

Table 12.12 Coefficient Set Register Settings

- α = 0.6 and X Scale (XS) = H'00001600

m	CmA	CmB	CmC
1	H'00000BDD	H'00000BDD	H'06519578
2	H'3FF007DA	H'00000BE3	H'03C24973
3	H'3FF003D9	H'00000BE9	H'01B30D5F
4	H'3FFFF7DF	H'001003F1	H'0003C542
5	H'000FDFEC	H'001003F7	H'3EC4711D
6	H'000FC400	H'002FFFFD	H'3DF504F1
7	H'001FA81A	H'002FFC00	H'3D9578C3
8	H'002F8C3C	H'00100000	H'3DB5C492

- α = 0.6 and X Scale (XS) = H'00002000

m	CmA	CmB	CmC
1	H'000FA400	H'000FA400	H'09625902
2 to 8	H'00000000	H'00000000	H'00000000

Section 13 Video Input Module 1 (VIN1)

The Video Input Module (hereinafter abbreviated as Video Input) is a video capture module supporting the ITU-R BT.656 interface. The Video Input performs capture control according to the even or odd field from the video decoder. The captured 4:2:2YCbCr image data is stored in memory.

The fields under capture control are scaled up or down in both horizontal and vertical directions with the maximum input size of 720 × 480 pixels. The maximum output size is 1024 × 960 pixels.

13.1 Features

- ITU-R BT.656 interface
- Size clipping before/after scaling
- Scaling up (up to two-fold*) or down in the horizontal (x) direction with a 9-tap multi-phase filer

Note: * However, the horizontal output is up to 1024 pixels.

- Scaling up (up to two-fold) or down in the vertical (y) direction using point-to-point interpolation

The entire block diagram of VIN1 is shown in Figure 13.1.

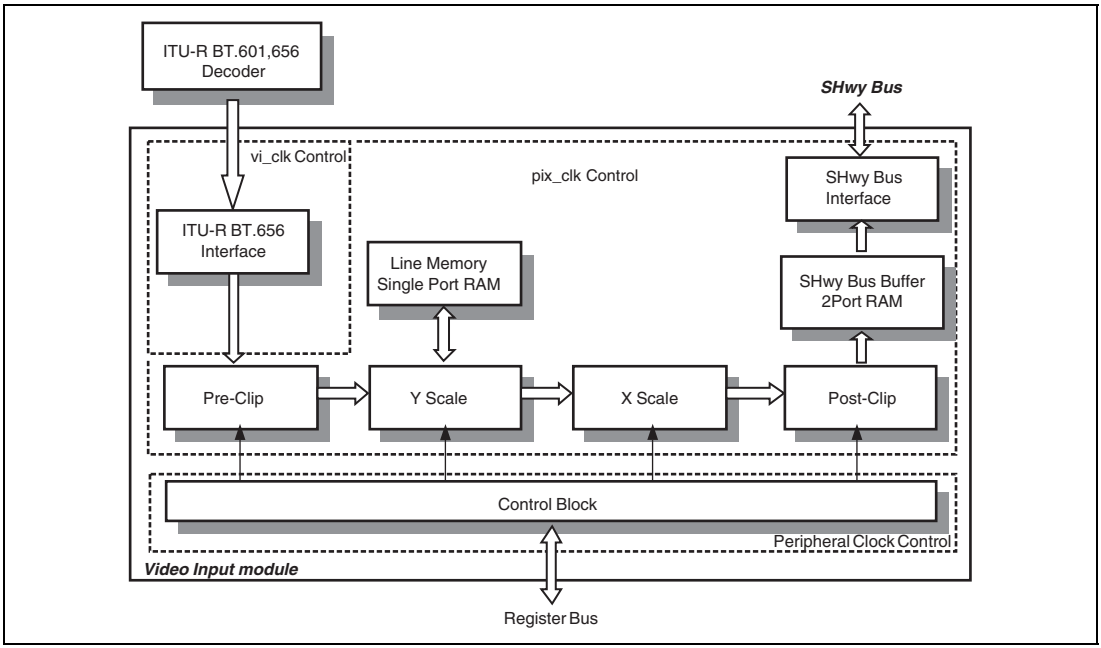


Figure 13.1 Block Diagram

13.2 Input/Output Pins

Table 13.1 Input/Output Pins

Terminal Name	Name	I/O	Remarks
VI1_CLK_x *[1].	Video clock input	Input	ITU-R BT.656 interface External asynchronous signal 27-MHz clock (typ.)
VI1_7_x to VI1_0_x *[2].	Video data input	Input	Video data

Legend: x: A or B (Mirror pin A or mirror pin B)

Notes: 1. Do not leave this pin floating when it is not in use. Fix it at the high level or low level.
2. Fix this pin at the low level when it is not in use.

13.3 Functional Overview

The functions of this module are listed below.

- ITU-R BT.656 interfaces
- Error correction
- Vertical scaling
- Horizontal scaling
- Size clipping
- Capture mode
- Endian conversion and output address

13.3.1 ITU-R BT.656 Interfaces

This module can capture video streams supporting the ITU-R BT.656 specifications.

The VI1_CLK allows an input of up to 29.4 MHz*. At the time of capturing with ITU-R BT.656, this module decides the frame switching, according to the F bit of SAV/EAV. For the capturing of video streams, a valid timing reference code must be entered because the process of field capturing will not end unless the F bit is switched.

Note that the capture operation is not guaranteed if the data input does not meet the specifications.

Note: * The frequency of VI1_CLK is limited to values no greater than the frequency of the system clock (clks1) multiplied by 1/3.3.

For example, when the system clock (clks1) is running at 88.9 MHz, the frequency of VI1_CLK should be no greater than 26.9 MHz.

13.3.2 Error Correction

At the time of capturing with ITU-R BT.656, this module allows errors to be corrected with the timing reference code (SAV/EAV). The timing reference code (SAV/EAV) of ITU-R BT.656 has four protection bits, which can be used to correct only one-bit errors in the interface.

If this module cannot correct errors because the CEE bit in the interrupt enable register (IE) has been set to 1, an interrupt signal is generated as soon as the CES bit in the interrupt status (INTS) register is set. Note that no interrupt signal is generated if errors can be corrected.

13.3.3 Vertical Scaling

This module creates lines by scaling up/down in the vertical direction through interpolation between two points in the neighborhood of captured lines. With the combinations of mantissaY and fractionY in the Y scaling (YS) register, the vertical scaling creates new lines by selecting line positions from captured lines.

The Y scaling function is disabled if mantissaY and fractionY are both cleared to 0 in the YS register.

Scaling down is implemented by creating fewer lines than captured lines. Examples of scaling up and down are shown in figure 13.2.

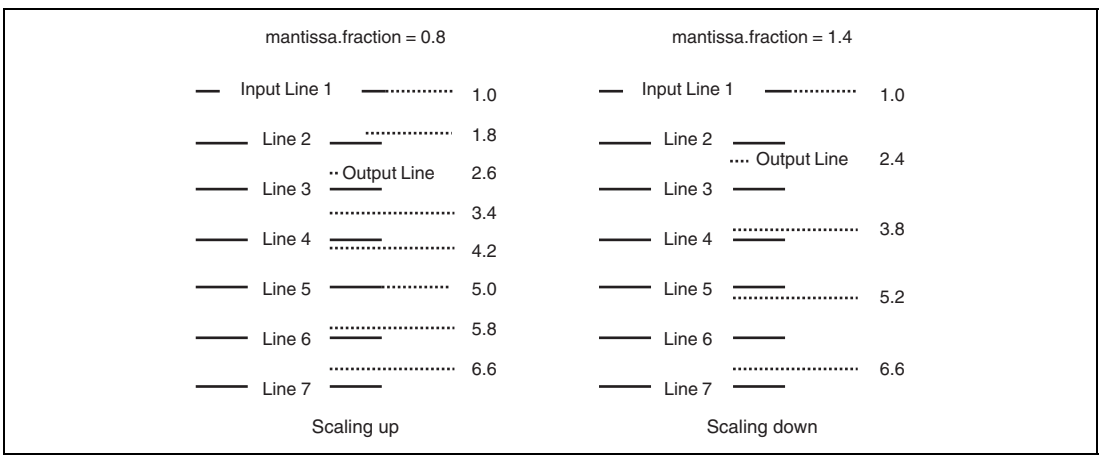


Figure 13.2 Examples of Scaling Up and Down in Vertical Direction

The number of lines created by this vertical scaling block is expressed by the following equation:

$$Ny = \begin{cases} \dfrac{4096 \times (ELPrC - SLPrC)}{4096 \times MantissaY + FractionY} - 1, \\ \qquad \text{when } \{4096 \times (ELPrC - SLPrC)\} \% (4096 \times MantissaY + FractionY) = 0 \\ Int\left(\dfrac{4096 \times (ELPrC - SLPrC)}{4096 \times MantissaY + FractionY} \right), \text{(otherwise)} \end{cases}$$

where:

ELPrC is the value in the end line pre-clip (ELPrC) register.

SLPrC is the value in the start line pre-clip (SLPrC) register.

MantissaY and FractionY are the values in the YS register.

13.3.4 Horizontal Scaling

Using a 9-tap multiphase filter, this module creates pixels by scaling down in the horizontal direction. The horizontal up-scaling doubles the number of input pixels captured and provides the function of expansion of up to two-fold by the down-scaling in the 9-tap multiphase filter. The horizontal scaling, which is set trough combinations of mantissaX and fractionX in the X scale (XS) register, determines the position of new pixels with a poly-phase filter. The selected coefficient, which is one of the eight coefficients, is determined by the position of output pixels.

In the example shown in figure 13.3, mantissaX and fractionX are set at 1.2. In this case, C2 from the coefficient set is selected.

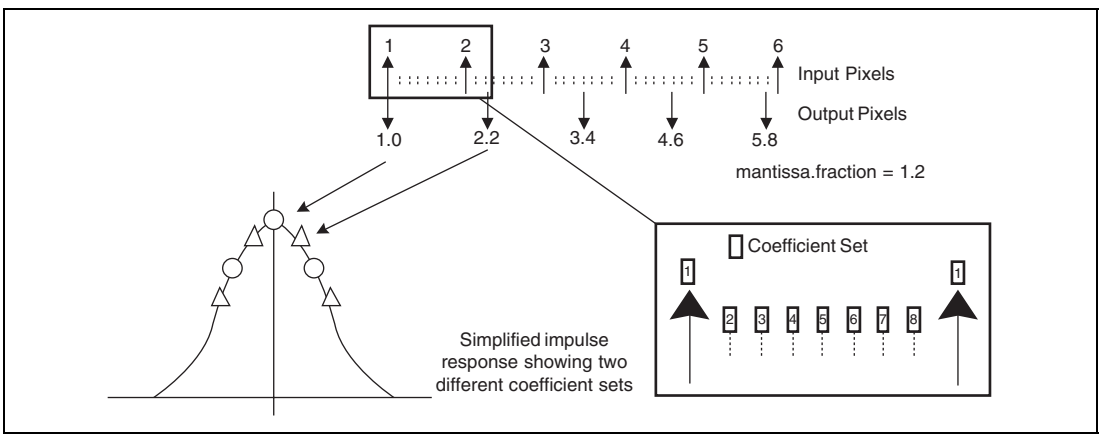


Figure 13.3 Pix Position and Coefficient Set

Figure 13.4 shows an example in which different coefficient sets are used for different output pixel positions. Nine coefficient sets are assigned to each. A total of 72 coefficients are used in the horizontal scaling. Each of the coefficients in these sets has a width of 10 bits, where the MSB serves as the sign bit. By setting the same coefficients into all of the eight coefficient set registers, the horizontal scaling allows the use of a multiphase filter just like a single-phase filter.

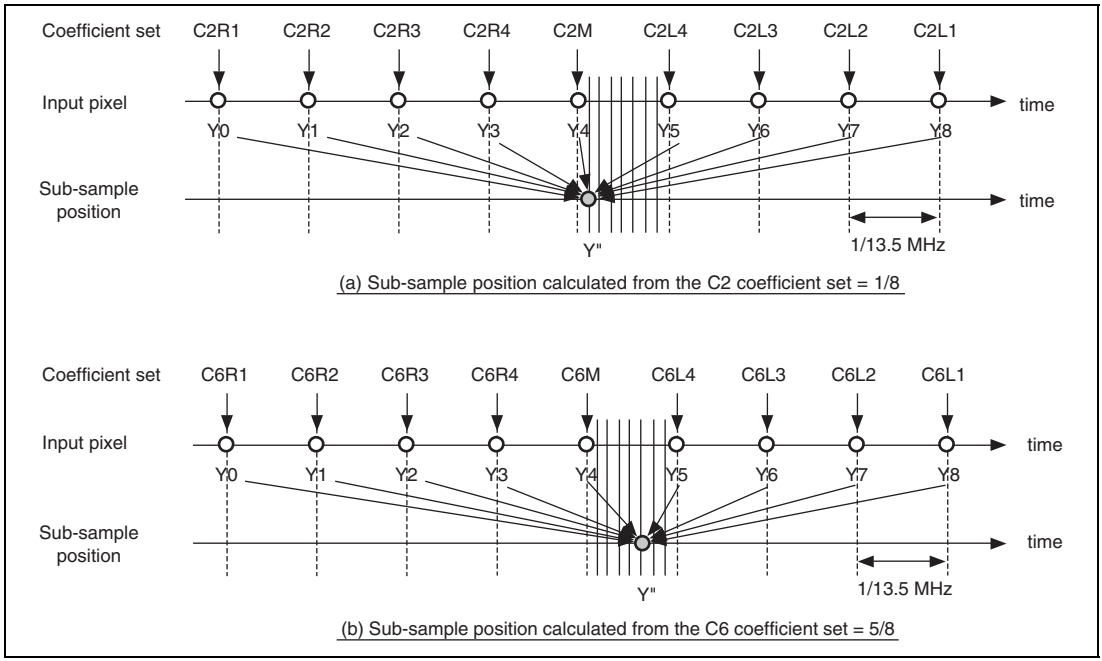


Figure 13.4 Example of Coefficients in Selected Coefficient Set

This scaling mechanism requires different coefficient values dependent on the scaling ratio. Here is an example of coefficient sets to be selected. The coefficients, CnM, CnLi, and CnRi (n = 1, 2, 3, - - - , 8; i = 1, 2, 3, and 4), are determined by the following equation:

$$C_nM = \beta \cdot h(-(n-1))$$

$$C_nR_i = \beta \cdot h(-(n-1) - 8(5-i))$$

$$C_nL_i = \beta \cdot h(-(n-1) + 8(5-i))$$

$$h(t) = \frac{\sin\left(\frac{\pi t}{T}\right)}{\frac{\pi t}{T}} \times \frac{\cos\left(\frac{\alpha \pi t}{T}\right)}{\frac{1-4\alpha^2 t^2}{T^2}}$$

$$T = 8 \times \text{MantissaX} + \text{FractionX}\ [11:9]$$

Mantissa X and Fraction X are the values in the XS register.

The parameter β is a standardized one. In the equation, h(t) follows the cosine characteristic and is executed with a 9-tap filter if the value of α is in the $0 < \alpha \leq 1$ range.

Obtaining dependable scale images requires you to use and meet the following equation in which all the coefficient sets are standardized.

This must be executed with the inherent β value that has been selected from the equation above.

$$C_nM + \sum_{i=1}^{4} C_nR_i + \sum_{i=1}^{4} C_nL_i = 512$$

The number of pixels created by the horizontal scaling block is calculated by the following equation.

$$Nx = Int\left(\frac{4096 \times (EPPrC - SPPrC)}{4096 \times MantissaX + FractionX}\right) + 1$$

Where:

EPPrC is the value in the end pixel pre-clip (EPPrC) register.

SPPrC is the value in the start pixel pre-clip (SPPrC) register.

MantissaX and FractionX are the values in the XS register.

13.3.5 Size Clipping

Image data that has been captured is pre-clipped according to the settings of the following registers: start line pre-clip (SLPrC), end line pre-clip (ELPrC), start pixel pre-clip (SPPrC), and end pixel pre-clip (EPPrC).

After the horizontal and vertical scaling, post-clipping takes place according to the settings of the following registers: start line post-clip (SLPoC), end line post-clip (ELPoC), start pixel post-clip (SPPoC), and end pixel post-clip (EPPoC). Figure 13.5 shows an example of size clipping.

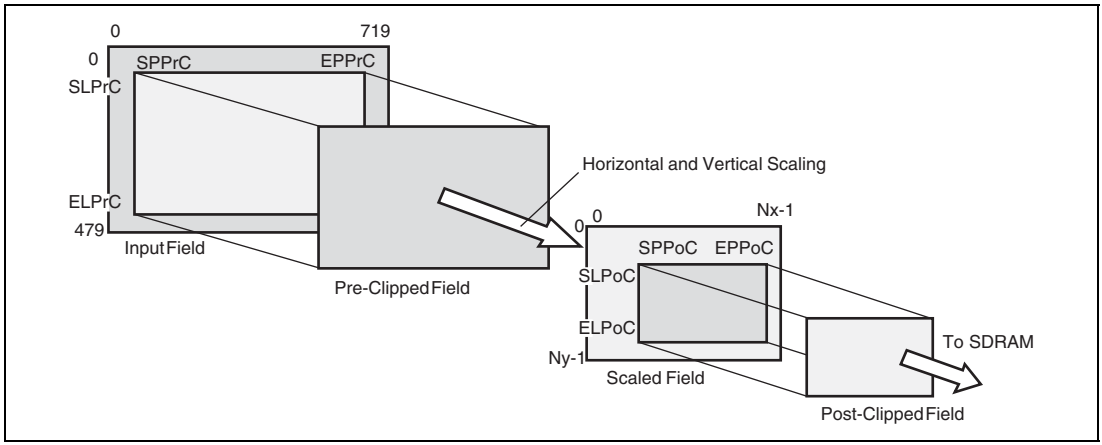


Figure 13.5 Example of Clippings

For all the post-clipped lines, the lengths of individual lines written into the memory are defined by the image stride (IS) register. The setting can be larger than the post-clipped frame width, but cannot be smaller than the width. The IS register must be filled with a value larger than the horizontal post-clip width. The Input Field in figure 13.5 shows an effective image area from the video decoder; this module does not allow anything exceeding the image area to be captured.

Note: Each of the settings for the following registers is given the value at the starting point in the effective image area: start line pre-clip (SLPrC), end line pre-clip (ELPrC), start pixel pre-clip (SPPrC), and end pixel pre-clip (EPPrC). Specifically, ITU-R BT.656 starts with the SAV (start of active video) signal.

13.3.6 Capture Mode

With this module, either of single capture mode or continuous capture mode can be selected.

Setting the SC bit in the frame capture (FC) register to 1 provides single capture mode, in which capture data is transferred to the memory address that is set in the memory base 1 (MB1) register.

Setting the CC bit in the FC register to 1 provides continuous capture mode, in which capture data is sequentially transferred to the address that is set in the memory base 1 to 3 registers. In this case, the latest frame ID captured is shown the FBS bit in the module status (MS) register.

This module starts capture operation after detecting a frame switching in the ITU-R BT.656 specification. The loading start field in full interlace mode is fixed at field 1 (odd field).

13.3.7 Endian Conversion and Output Address

With this module, switching between big endian and little endian is possible by setting the EN bit in the MC register. With the EN bit in the MC register set to 1, capture data is allocated in big endian form and output to memory.

For the transfer start address that is set by the memory base register, this module generates linear addresses and outputs capture data to memory.

The default is big endian.

13.4 Register Descriptions

Table 13.2 shows the VIN1 register configuration.

Table 13.2 VIN1 Register Configuration

Register Name	Abbreviation	R/W	P4 Area Address *1	Area 7 Address *1	Access Size	Vsync Update Mode Support *2
Main Control	MC	R/W	H'FFC5 1000	H'1FC5 1000	32	Disabled
Module Status	MS	R	H'FFC5 1004	H'1FC5 1004	32	Disabled
Frame Capture	FC	R/W	H'FFC5 1008	H'1FC5 1008	32	Disabled
Start Line Pre-Clip	SLPrC	R/W	H'FFC5 100C	H'1FC5 100C	32	Enabled
End Line Pre-Clip	ELPrC	R/W	H'FFC5 1010	H'1FC5 1010	32	Enabled
Start Pixel Pre-Clip	SPPrC	R/W	H'FFC5 1014	H'1FC5 1014	32	Enabled
End Pixel Pre-Clip	EPPrC	R/W	H'FFC5 1018	H'1FC5 1018	32	Enabled
Start Line Post-Clip	SLPoC	R/W	H'FFC5 101C	H'1FC5 101C	32	Enabled
End Line Post-Clip	ELPoC	R/W	H'FFC5 1020	H'1FC5 1020	32	Enabled
Start Pixel Post-Clip	SPPoC	R/W	H'FFC5 1024	H'1FC5 1024	32	Enabled
End Pixel Post-Clip	EPPoC	R/W	H'FFC5 1028	H'1FC5 1028	32	Enabled
Image Stride	IS	R/W	H'FFC5 102C	H'1FC5 102C	32	Enabled
Memory Base 1	MB1	R/W	H'FFC5 1030	H'1FC5 1030	32	Enabled
Memory Base 2	MB2	R/W	H'FFC5 1034	H'1FC5 1034	32	Enabled
Memory Base 3	MB3	R/W	H'FFC5 1038	H'1FC5 1038	32	Enabled
Line Count	LC	R	H'FFC5 103C	H'1FC5 103C	32	Disabled
Interrupt Enable	IE	R/W	H'FFC5 1040	H'1FC5 1040	32	Disabled
Interrupt Status	INTS	R/W	H'FFC5 1044	H'1FC5 1044	32	Disabled
Scanline Interrupt	SI	R/W	H'FFC5 1048	H'1FC5 1048	32	Enabled
Y Scale	YS	R/W	H'FFC5 1050	H'1FC5 1050	32	Enabled
X Scale	XS	R/W	H'FFC5 1054	H'1FC5 1054	32	Enabled
Coeff Set 1A	C1A	R/W	H'FFC5 1080	H'1FC5 1080	32	Disabled
Coeff Set 1B	C1B	R/W	H'FFC5 1084	H'1FC5 1084	32	Disabled

Register Name	Abbreviation	R/W	P4 Area Address *1	Area 7 Address *1	Access Size	Vsync Update Mode Support *2
Coeff Set 1C	C1C	R/W	H'FFC5 1088	H'1FC5 1088	32	Disabled
Coeff Set 2A	C2A	R/W	H'FFC5 1090	H'1FC5 1090	32	Disabled
Coeff Set 2B	C2B	R/W	H'FFC5 1094	H'1FC5 1094	32	Disabled
Coeff Set 2C	C2C	R/W	H'FFC5 1098	H'1FC5 1098	32	Disabled
Coeff Set 3A	C3A	R/W	H'FFC5 10A0	H'1FC5 10A0	32	Disabled
Coeff Set 3B	C3B	R/W	H'FFC5 10A4	H'1FC5 10A4	32	Disabled
Coeff Set 3C	C3C	R/W	H'FFC5 10A8	H'1FC5 10A8	32	Disabled
Coeff Set 4A	C4A	R/W	H'FFC5 10B0	H'1FC5 10B0	32	Disabled
Coeff Set 4B	C4B	R/W	H'FFC5 10B4	H'1FC5 10B4	32	Disabled
Coeff Set 4C	C4C	R/W	H'FFC5 10B8	H'1FC5 10B8	32	Disabled
Coeff Set 5A	C5A	R/W	H'FFC5 10C0	H'1FC5 10C0	32	Disabled
Coeff Set 5B	C5B	R/W	H'FFC5 10C4	H'1FC5 10C4	32	Disabled
Coeff Set 5C	C5C	R/W	H'FFC5 10C8	H'1FC5 10C8	32	Disabled
Coeff Set 6A	C6A	R/W	H'FFC5 10D0	H'1FC5 10D0	32	Disabled
Coeff Set 6B	C6B	R/W	H'FFC5 10D4	H'1FC5 10D4	32	Disabled
Coeff Set 6C	C6C	R/W	H'FFC5 10D8	H'1FC5 10D8	32	Disabled
Coeff Set 7A	C7A	R/W	H'FFC5 10E0	H'1FC5 10E0	32	Disabled
Coeff Set 7B	C7B	R/W	H'FFC5 10E4	H'1FC5 10E4	32	Disabled
Coeff Set 7C	C7C	R/W	H'FFC5 10E8	H'1FC5 10E8	32	Disabled
Coeff Set 8A	C8A	R/W	H'FFC5 10F0	H'1FC5 10F0	32	Disabled
Coeff Set 8B	C8B	R/W	H'FFC5 10F4	H'1FC5 10F4	32	Disabled
Coeff Set 8C	C8C	R/W	H'FFC5 10F8	H'1FC5 10F8	32	Disabled
Data Mode	DMR	R/W	H'FFC5 2100	H'1FC5 2100	32	Disabled

Notes: 1. These P4 addresses are for the P4 area in the virtual address space. These area 7 addresses are accessed from area 7 in the physical address space by means of the TLB. Do not access to any addresses other than listed above.

2. For details on the vsync update mode, refer to the description under the VUP bit in the MC register.

Table 13.3 Register State in Each Operating Mode

Abbreviation	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
MC	Initialized	Initialized	Retained	Retained	Retained	Initialized
MS	Initialized	Initialized	Retained	Retained	Retained	Initialized
FC	Initialized	Initialized	Retained	Retained	Retained	Initialized
SLPrC	Initialized	Initialized	Retained	Retained	Retained	Initialized
ELPrC	Initialized	Initialized	Retained	Retained	Retained	Initialized
SPPrC	Initialized	Initialized	Retained	Retained	Retained	Initialized
EPPrC	Initialized	Initialized	Retained	Retained	Retained	Initialized
SLPoC	Initialized	Initialized	Retained	Retained	Retained	Initialized
ELPoC	Initialized	Initialized	Retained	Retained	Retained	Initialized
SPPoC	Initialized	Initialized	Retained	Retained	Retained	Initialized
EPPoC	Initialized	Initialized	Retained	Retained	Retained	Initialized
IS	Initialized	Initialized	Retained	Retained	Retained	Initialized
MB1	Initialized	Initialized	Retained	Retained	Retained	Initialized
MB2	Initialized	Initialized	Retained	Retained	Retained	Initialized
MB3	Initialized	Initialized	Retained	Retained	Retained	Initialized
LC	Initialized	Initialized	Retained	Retained	Retained	Initialized
IE	Initialized	Initialized	Retained	Retained	Retained	Initialized
INTS	Initialized	Initialized	Retained	Retained	Retained	Initialized
SI	Initialized	Initialized	Retained	Retained	Retained	Initialized
YS	Initialized	Initialized	Retained	Retained	Retained	Initialized
XS	Initialized	Initialized	Retained	Retained	Retained	Initialized
C1A	Initialized	Initialized	Retained	Retained	Retained	Initialized
C1B	Initialized	Initialized	Retained	Retained	Retained	Initialized
C1C	Initialized	Initialized	Retained	Retained	Retained	Initialized
C2A	Initialized	Initialized	Retained	Retained	Retained	Initialized
C2B	Initialized	Initialized	Retained	Retained	Retained	Initialized
C2C	Initialized	Initialized	Retained	Retained	Retained	Initialized
C3A	Initialized	Initialized	Retained	Retained	Retained	Initialized
C3B	Initialized	Initialized	Retained	Retained	Retained	Initialized
C3C	Initialized	Initialized	Retained	Retained	Retained	Initialized

Abbreviation	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
C4A	Initialized	Initialized	Retained	Retained	Retained	Initialized
C4B	Initialized	Initialized	Retained	Retained	Retained	Initialized
C4C	Initialized	Initialized	Retained	Retained	Retained	Initialized
C5A	Initialized	Initialized	Retained	Retained	Retained	Initialized
C5B	Initialized	Initialized	Retained	Retained	Retained	Initialized
C5C	Initialized	Initialized	Retained	Retained	Retained	Initialized
C6A	Initialized	Initialized	Retained	Retained	Retained	Initialized
C6B	Initialized	Initialized	Retained	Retained	Retained	Initialized
C6C	Initialized	Initialized	Retained	Retained	Retained	Initialized
C7A	Initialized	Initialized	Retained	Retained	Retained	Initialized
C7B	Initialized	Initialized	Retained	Retained	Retained	Initialized
C7C	Initialized	Initialized	Retained	Retained	Retained	Initialized
C8A	Initialized	Initialized	Retained	Retained	Retained	Initialized
C8B	Initialized	Initialized	Retained	Retained	Retained	Initialized
C8C	Initialized	Initialized	Retained	Retained	Retained	Initialized
DMR	Initialized	Initialized	Retained	Retained	Retained	Initialized

13.4.1 Main Control (MC) Register

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LCKEN	PRICNT[2:0]			—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	VUP	—	—	—	EN	EC	IM[1:0]		—	—	ME
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W:	R	R	R	R	R	R/W	R	R	R	R/W	R/W	R/W	R/W	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	LCKEN	0	R/W	LCK Enable Enables or disables the lck bit of the SHwy bus for the VIN1 module. 0: lck bit disabled 1: lck bit enabled This bit should always be set to 0 when the LSI is operating.
30 to 28	PRICNT[2:0]	000	R/W	PRI Control Sets the pri [2:0] bits of the SHwy bus for the VIN1 module. These bits should be always set to 1 when the LSI is operating. Since the pri [3] bit is fixed to 1, the value of the pri [3:0] bits becomes 9 when that of the pri [2:0] bits is 1.
27 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
10	VUP	0	R/W	Vsync Update This bit sets the internal update timing based on the register setting. Refer to the list of registers for applicable registers. 0: Internal updating will occur immediately after register writing. 1: The register value will be updated after a valid field is detected in ITU-R BT.601 or the Field (F) bit in ITU-R BT.656 is changed.
9 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6	EN	0	R/W	Endian Type This bit sets the Endian type for data to be output to memory. 0: 4:2:2YCbCr data is packed and allocated in the big endian format. 1: 4:2:2YCbC data is packed and allocated in the little endian format.
5	EC	0	R/W	Error Correction This bit specifies whether error correction with the parity bit is performed on the ITU-R BT.656 input. 0: Error correction is not performed on the ITU-R BT.656 input. 1: Error correction with the parity bit is performed on the ITU-R BT.656 input. Error correction must not be performed in the following cases: • When input data does not meet the standard of the ITU-R BT.656 parity bit

Bit	Bit Name	Initial Value	R/W	Description
4, 3	IM[1:0]	00	R/W	Interlace Mode These bits set the capture mode for this module. This module supports four different frame capture modes listed below. Combinations with the settings for single and continuous capture modes in the frame capture (FC) register are shown below. 00: Odd field (field 1) capture mode Handles only odd field. Both continuous capture modes and single capture modes can be set. 01: Odd-/even-field capture mode Handles odd or even field as single separate frames. Continuous capture modes can be set, but single capture modes cannot. 10: Even-field (field 2) capture mode Handles only even field. Both continuous capture modes and single capture modes can be set. 11: Full interlace mode Handles combinations of odd or even field as single frames. Both continuous capture modes and single capture modes can be set. Capturing in full interlace mode takes as follows: First, the lines in odd fields are written into the memory every other line. After all the odd fields are covered, the lines in even fields are written into the memory between the written lines in the odd field so that they will be alternated with each other. Figure 13.6 shows an example in which the memory base 1 (MB1) register is set at H'0000 in the H'200. interval. During capture, the interlace mode (IM) bit should not be changed.

Bit	Bit Name	Initial Value	R/W	Description
2	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
1	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
0	ME	0	R/W	Module Enable This is the enable bit for this module. After setting this bit, set all the registers. 0: The module will not operate. 1: The module will operate.

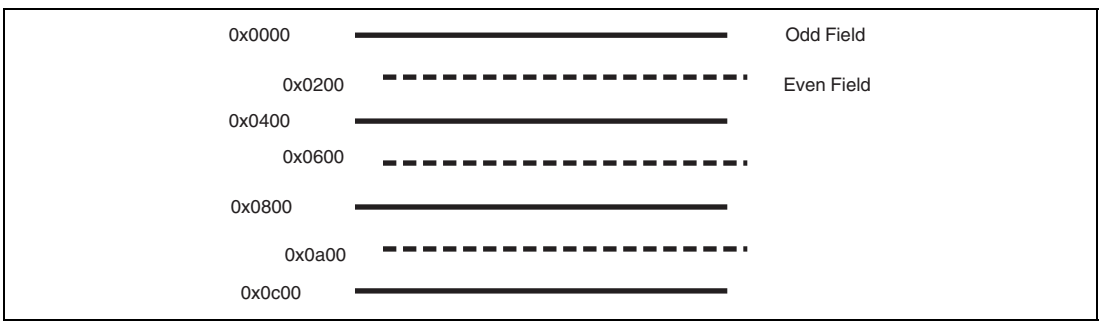


Figure 13.6 Example of Capturing Full Interlace

13.4.2 Module Status (MS) Register

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	FBS[1:0]		FS	AV	CA
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4, 3	FBS[1:0]	00	R	Frame Buffer State These bits indicate the state of frame buffer. 00: This indicates that the latest valid frame buffer has the base address defined by the Memory Base 1 (MB1) register. 01: This indicates that the latest valid frame buffer has the base address defined by the Memory Base 2 (MB2) register. 10: This indicates that the latest valid frame buffer has the base address defined by the Memory Base 3 (MB3) register. 11: This indicates that there is no valid frame buffer.
2	FS	0	R	Field State This bit shows the type of the current capture field. 0: This indicates that the current field is Field 1 (odd field). 1: This indicates that the current field is Field 2 (even field).

Bit	Bit Name	Initial Value	R/W	Description
1	AV	0	R	Active Video This bit shows whether or not the current pointer at the time of capture is in the active video area. This bit will be 0 if no input data is captured. The active video area is defined by the pre-clipping register. 0: The current field is not in the active video area. 1: The current field is in the active video area.
0	CA	0	R	Capture Active This bit shows whether or not video capture is currently taking place. When the interlace mode (IM) bits are set to 00 (field 1 capturing), however, the current video input field corresponds to field 2 and this status bit will be 1 even if no field is captured. This bit will be updated at the timing when the effective field in ITU-R BT.601 changes or when the F bit defined by ITU-R BT.656 changes. 0: This indicates that video capture is not operating. 1: This indicates that video capture is operating.

13.4.3 Frame Capture (FC) Register

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CC	SC
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	CC	0	R/W	Continuous Capture This bit provides a setting as to whether or not capture mode is to be continuous-frame. In continuous frame capture mode, the first capture frame is written into the memory address specified by the Memory Base 1 (MB1) register before the second capture frame is written into the memory address specified by the Memory Base 2 (MB2) register. The capture operation is repeated in the order of MB3, MB1, MB2, MB3, MB1, and such. 0: The continuous frame capture mode is not set. 1: The continuous frame capture mode is set. In continuous frame capture mode, writing 0 into this bit will immediately terminate the capture operation if the current frame is over or it has not been captured.

Bit	Bit Name	Initial Value	R/W	Description
0	SC	0	R/W	Single Capture This bit provides a setting as to whether or not capture mode is to be single-frame. In single frame capture mode, the current frame is captured if the current value of the frame line counter is smaller than the value of the Start Line Pre-Clip (SLPrC) register; otherwise, the next frame is captured. Frames captured in single capture mode are written into the memory address specified by the Memory Base 1 (MB1) register. 0: This means that single frame capture mode will not be set. 1: This means that single frame capture mode will be set. Don't set the value of 1 if the Frame Capture (FC) is set as continuous capture mode or if Interlace Mode (IM) is set as the "01" mode (odd-/even-field capture). In single capture mode, this bit is set to 1 before the Frame Buffer Status (FBS) bit in the Module Status register is immediately initialized and this bit is also set to 0.

Note: Do not set the SC and CC bits to 1 at the same time.

13.4.4 Start Line Pre-Clip (SLPrC) Register

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SLPrC[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	SLPrC[9:0]	H'000	R/W	Start Line Pre-Clip These bits set the start line which is to become valid at the time of capture. Set a value in the register in the range from 0 to 478 so that the number of lines that is pre-clipped by the SLPrC and ELPrC settings will be 2 or more. This value will be used before scaling. The value of 0 here indicates the first valid line captured when there is a change from 1 to 0 in the V bit in ITU-R BT.656.

13.4.5 End Line Pre-Clip (ELPrC) Register

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	ELPrC[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	ELPrC[9:0]	H'000	R/W	End Line Pre-Clip These bits set the end line which is to become valid at the time of capture. Set a value in the register in the range from 1 to 479 so that the number of lines that is pre-clipped by the SLPrC and ELPrC settings will be 2 or more. This value will be used before scaling.

13.4.6 Start Pixel Pre-Clip (SPPrC) Register

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SPPrC[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	SPPrC[9:0]	H'000	R/W	Start Pixel Pre-Clip These bits set the start pixel that is valid at the time of capture and present on the current line. Set a value in the register in the range from 0 to 715 so that the number of pixels that is pre-clipped by the SPPrC and EPPrC settings will be 5 or more. This value will be used before scaling. For the ITU-R BT.656 input (4:2:2 YCbCr) format, this module enables only the pre-clipping with a multiple of 2, so that the LSB in the register will be ignored. If an odd value is set in the address, the number will be decremented by 1 and actually handled as an even address.

13.4.7 End Pixel Pre-Clip (EPPrC) Register

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	EPPrC[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	EPPrC[9:0]	H'000	R/W	End Pixel Pre-Clip These bits set the end pixel that is valid at the time of capture and present on the current line. Set a value in the register in the range from 4 to 719 so that the number of pixels that is pre-clipped by the SPPrC and EPPrC settings will be 5 or more. This value will be used before scaling. For the ITU-R BT.656 input (4:2:2 YCbCr) format, this module enables only the pre-clipping with a multiple of 2; if an even value is set in the address, the number will be incremented by 1 and actually handled as an odd address. Therefore, the LSB in the register will be ignored.

13.4.8 Start Line Post-Clip (SLPoC) Register

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SLPoC[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	SLPoC	H'000	R/W	Start Line Post-Clip These bits set the start line in an image that is to be written into the memory. Set a value in the register in the range from 0 to 958 so that the number of lines that is post-clipped by the SLPoC and ELPoC settings will be 2 or more. This value will be used after scaling. A value of 0 indicates that it is the start line in the image after scaling.

13.4.9 End Line Post-Clip (ELPoC) Register

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	ELPoC[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	ELPoC	H'000	R/W	End Line Post-Clip These bits set the end line in an image that is to be written into the memory. Set a value in the register in the range from 1 to 959 so that the number of lines that is post-clipped by the SLPoC and ELPoC settings will be 2 or more. This value will be used after scaling.

13.4.10 Start Pixel Post-Clip (SPPoC) Register

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SPPoC[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	SPPoC	H'000	R/W	Start Pixel Post-Clip These bits set the start pixel of each line in an image that is to be written into the memory. Set a value in the register in the range from 0 to 1019 so that the number of pixels that is post-clipped by the SPPoC and EPPoC settings will be 5 or more. This value will be used after scaling. The post-clipping takes place after the conversion from 4:2:2 YCbCr to 4:4:4 YCbCr; therefore, the post-clipping value need not be limited to a multiple of 2.

13.4.11 End Pixel Post-Clip (EPPoC) Register

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	EPPoC[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	EPPoC	H'000	R/W	End Pixel Post-Clip These bits set the end pixel of each line in an image that is to be written into the memory. Set a value in the register in the range from 4 to 1023 so that the number of pixels that is post-clipped by the SPPoC and EPPoC settings will be 5 or more. This value will be used after scaling

13.4.12 Image Stride (IS) Register

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	IS[8:0]									—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 4	IS	H'000	R/W	Image Stride These bits set the width of the memory at the destination. For this register, set the width of the memory at the destination in the range from 16 to 4096 pixels at an increment of 16 pixels if the memory controller does not handle linear tile conversion. If the memory controller handles linear tile conversion, set the width of the memory at the destination by using one of the following: 512, 1024, 2048, and 4096 pixels. The setting must be a value equal to or greater than the width of the post-clip (EPPoC-SPPoC).
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

13.4.13 Memory Base 1 (MB1) Register

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	MBA1[2:0]			MB1[18:9]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MB1[8:0]									—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28 to 26	MBA1 [2:0]	000	R/W	Memory Base Area 1 These bits set the area of memory showing where to start the transfer of the frame that has been captured. 010: Area 2 011: Area 3 100: Area 4 101: Area 5 Setting other than above is prohibited.
25 to 7	MB1 [18:0]	H'00000	R/W	Memory Base 1 These bits set the memory address showing where to start the transfer of the frame that has been captured. The value must be set at an increment of 128B. If the module is in continuous capture mode, this register is used in the following capture sequence: MB 1- MB 2- MB 3- MB 1- MB 2- MB 3. In single capture mode, the settings in this register are used as a start memory address.
6 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

13.4.14 Memory Base 2 (MB2) Register

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	MBA2[2:0]			MB2[18:9]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MB2[8:0]									—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28 to 26	MBA2[2:0]	000	R/W	Memory Base Area 2 These bits set the area of memory showing where to start the transfer of the second frame that has been captured in continuous capture mode. 010: Area 2 011: Area 3 100: Area 4 101: Area 5 Setting other than above is prohibited.
25 to 7	MB2[18:0]	H'00000	R/W	Memory Base 2 These bits set the memory address showing where to start the transfer of the second frame that has been captured in continuous capture mode. The value must be set at an increment of 128B. If the module is in continuous capture mode, this register is used in the following capture sequence: MB 1- MB 2- MB 3- MB 1- MB 2- MB 3.
6 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

13.4.15 Memory Base 3 (MB3) Register

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	MBA3[2:0]			MB3[16:7]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MB3[6:0]							—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28 to 26	MBA3[2:0]	000	R/W	Memory Base Area 3 These bits set the area of memory showing where to start the transfer of the third frame that has been captured in continuous capture mode. 010: Area 2 011: Area 3 100: Area 4 101: Area 5 Setting other than above is prohibited.
25 to 9	MB3[16:0]	H'00000	R/W	Memory Base 3 These bits set the memory address showing where to start the transfer of the third frame that has been captured in continuous capture mode. The value must be set at an increment of 128B. If the module is in continuous capture mode, this register is used in the following capture sequence: MB 1- MB 2- MB 3- MB 1- MB 2- MB 3.
8 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

13.4.16 Line Count (LC) Register

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	LC[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	LC[9:0]	H'000	R	Line Count These bits show the line position in the current capture field.

13.4.17 Interrupt Status (INTS) Register

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FIS2	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	FIS	CES	SIS	EFS	FOS
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	FIS2	0	R/W	Field Interrupt Status 2 This bit shows that the field has changed. This bit is set to 1 at the change point of the F bit detected by the valid field in ITU-R BT.601 and defined by ITU-R BT.656. This bit is set to 1 irrespective of whether or not capture is taking place. Writing 1 to this bit clears the setting.
30 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	FIS	0	R/W	Field Interrupt Status This bit shows that a field has been captured. This bit is set to 1 at the change point of the F bit defined by ITU-R BT.656. Writing 1 to this bit clears the setting.

Bit	Bit Name	Initial Value	R/W	Description
3	CES	0	R/W	Timing Reference Code Error Status This bit shows that the current timing reference code has encountered an error involving at least two bits. This bit is set to 1 if the EC bit in the Main Control (MC) register is enabled and if it has encountered an error involving at least two bits. If a 1-bit error occurs when the EC bit is enabled, this bit is not set to 1. Writing 1 to this bit clears the setting.
2	SIS	0	R/W	Scanline Interrupt Status This bit shows that the number of lines specified by the SI register has been reached. The lines for which this bit is set to 1 are determined by the Scanline Interrupt (SI) register. This bit is set to 1 if the value in the LC register matches the setting for the SI register. Writing 1 to this bit clears the setting.
1	EFS	0	R/W	End of Frame Interrupt Status This bit shows that the last frame has been reached. This bit is set to 1 at the end of field 2 (even field). Writing 1 to this bit clears the setting.
0	FOS	0	R/W	FIFO Overflow Interrupt Status This bit shows that the FIFO has caused an overflow. The FIFO is used by the module which transfers pixel data to the pixel bus. An FIFO overflow would cause the pixel data to be lost. Writing 1 to this bit clears the setting.

13.4.18 Interrupt Enable (IE) Register

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FIE2	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	FIE	CEE	SIE	EFE	FOE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	FIE2	0	R/W	Field Interrupt Enable 2 This bit provides a setting as to whether the INTC output for field interrupts is to be enabled or disabled. Interrupt signals by this enable bit will be asserted irrespective of whether or not capture is taking place. 0: Field interrupts are disabled. 1: Field interrupts are enabled.
30 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	FIE	0	R/W	Field Interrupt Enable This bit provides a setting as to whether the INTC output for field-switching interrupts is to be enabled or disabled. This interrupt enable will be valid if the CA bit in the Module Status (MS) register is 1. 0: Field-switching interrupts are disabled. 1: Field-switching interrupts are enabled.

Bit	Bit Name	Initial Value	R/W	Description
3	CEE	0	R/W	Timing Reference Code Error Enable This bit provides a setting as to whether the INTC output for error correction interrupt is to be enabled or disabled in the timing reference code (SAV/EAV) described in the ITU-R BT.656 specification. This interrupt enable will be valid if the CA bit in the Module Status (MS) register is 1. 0: ITU-R BT.656 timing reference code error interrupts are disabled. 1: ITU-R BT.656 timing reference code error interrupts are enabled.
2	SIE	0	R/W	Scanline Interrupt Enable This bit provides a setting as to whether the INTC output for scan line interrupts is to be enabled or disabled. This interrupt enable will be valid if the CA bit in the Module Status (MS) register is 1. 0: Scan line interrupts are disabled. 1: Scan line interrupts are enabled.
1	EFE	0	R/W	End of Frame Interrupt Enable This bit provides a setting as to whether the INTC output for end of frame interrupts is to be enabled or disabled. This interrupt enable will be valid if the CA bit in the Module Status (MS) register is 1. 0: The last frame interrupt is disabled. 1: The last frame interrupt is enabled.
0	FOE	0	R/W	FIFO Overflow Interrupt Enable This bit provides a setting as to whether the INTC output for FIFO overflow interrupts is to be enabled or disabled. This interrupt enable will be valid if the CA bit in the Module Status (MS) register is 1. 0: FIFO overflow interrupts are disabled. 1: FIFO overflow interrupts are enabled.

13.4.19 Scanline Interrupt (SI) Register

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SI[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	SI[9:0]	H'000	R/W	Scanline Interrupt These bits set a value which is to be compared with the value of the LC register in each field when the SIE bit in the IE register is set to 1. If there is a match with the value of the LC register, an interrupt signal is asserted.

13.4.20 Y Scale (YS) Register

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Mantissa Y[3:0]				Fraction Y[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 12	Mantissa Y[3:0]	H'0	R/W	Mantissa Y The value of scaling up and down in Y direction is specified by mantissa Y and fraction Y. The value of this register shows the ratio of the number of capture lines per field to the number of lines written into the memory per field and it can be calculated by the following equation: Y scaling = 4096 / (4096 × Mantissa Y + Fraction Y) Note, however, that the maximum of the scaling up is 2. For example, to halve the Y scaling, set as follows: Mantissa Y = H'2 and Fraction Y = H'000. Scaling-based lines are created by interpolating two points in the vicinity of a capture line. If both mantissa Y and fraction Y are 0, then the scaling will be invalidated.
11 to 0	Fraction Y[11:0]	H'000	R/W	Fraction Y

13.4.21 X Scale (XS) Register

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Mantissa X[3:0]				Fraction X[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 12	Mantissa X[3:0]	H'0	R/W	Mantissa X The value of scaling up and down in the X direction is specified through mantissa X and fraction X. The value of the register shows the ratio of the number of input pixels per line to the number of memory output pixels per line and it is calculated by the following equation: X scaling = 4096 / (4096 × Mantissa X + Fraction X) Note, however, that the maximum of the scaling up is 2. Setting any value higher than 2 will be ignored. The multiphase filter is used to create pixels in the Coefficient Set CnA (n = 1 to 8) register. If both mantissa X and fraction X are 0, then the scaling will be invalidated.
11 to 0	Fraction X[11:0]	H'000	R/W	Fraction X

Table 13.4 shows a typical example of setting 720 input pixels along with their ratios of scaling up and down.

Table 13.4 Example of Setting the X Scaling

Output Pixels	Ratio of Scaling Up or Down	Set Value
680	0.943	H'10F8
800	1.111	H'0E68
854	1.185	H'0D80

13.4.22 Coefficient Set (CnA, CnC, CnB) Registers (n = 1 to 8)

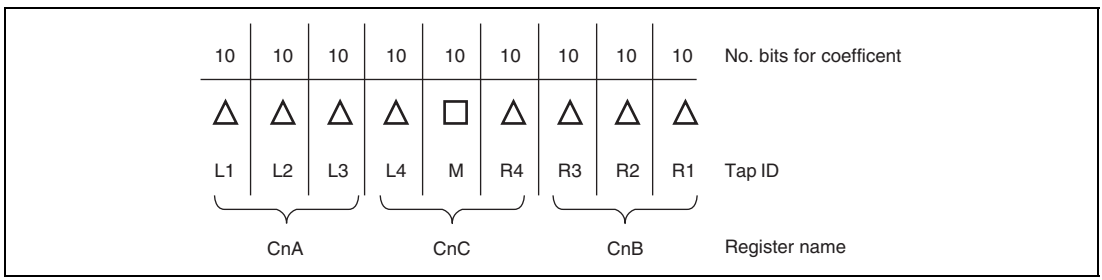


Figure 13.7 Tap Coefficient Bit Size

Figure 13.7 shows the number of bits used for each of the coefficients in nine taps. Each of the eight coefficients consists of three different 32-bit registers as referenced below. The MSB of each coefficient is a sign bit.

- Coefficient Set CnA Register (n = 1 to 8)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	L1[9:0]										L2[9:6]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	L2[5:0]						L3[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note that the value of n refers to the number of a coefficient set.

- Coefficient Set CnB Register (n = 1 to 8)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	R1[9:0]										R2[9:6]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R2[5:0]						R3[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note that the value of n refers to the number of a coefficient set.

- Coefficient Set CnC Register (n = 1 to 8)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	R4[9:0]										L4[9:6]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	L4[5:0]						M[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note that the value of n refers to the number of a coefficient set.

13.4.23 Data Mode (DMR) Register

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	BPSM	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	BPSM	0	R/W	Output Data Byte Swap Mode 0: Bytes are not swapped in output data. (Initial value) 1: Bytes are swapped in output data. Note: When YCbCr-422 data is output in big endian, data is transferred in the YUYV format in most cases. To transfer data in the UYVY format, set this bit to 1.
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

13.5 Sample Programs

This section describes sample programs for video input module setup.

- Initial setting of video input registers
- Function for specifying X-direction filter coefficients

13.5.1 Example of Initial Setting of Video Input 1 Registers

List 1 shows an example of the initial setting routine for video input registers. In this example, NTSC (image size: 720 × 480) video images are scaled down to 360 × 240 and NTSC fields 1 and 2 are captured into the respective frame memory areas.

setVideoReg() is a function for setting up a video input register. The first argument of this function is the address of the target video input register, and the second argument is the value to be stored in the address.

List 1: Example of Initial Setting Routine for Video Input Registers

```
1.    setVideoReg (vi_mc, 0x000B);
2.    setVideoReg (vi_slprc, 0x0000);       // SLPrC = 0
3.    setVideoReg (vi_elprc, 0x00EF);       // ELPrC = 239
4.    setVideoReg (vi_spprc, 0x0000);       // SPPrC = 0
5.    setVideoReg (vi_epprc, 0x02CF);       // EPPoC = 719
6.    setVideoReg (vi_slpoc, 0x0000);       // SLPoC = 0
7.    setVideoReg (vi_elpoc, 0x00EF);       // ELPoC = 239
8.    setVideoReg (vi_sppoc, 0x0000);       // SPPoC = 0
9.    setVideoReg (vi_eppoc, 0x0167);       // EPPoC = 359
10.   setVideoReg (vi_xs, 0x2000);
11.   setVideoReg (vi_ys, 0x0000);
12.   setVideoReg (vi_mb1, 0x00300000);
13.   setVideoReg (vi_mb2, 0x00380000);
14.   setVideoReg (vi_mb3, 0x00400000);
15.   setVideoReg (vi_is, 0x0200);
16.   setVideoReg (vi_ie, 0x0000);
17.   setVideoReg (vi_ints, 0x0000);
18.   setVideoReg (vi_si, 0x0001);
19.   //======== video_in coeff registers ========
20.   setVideoReg (vi_c1a, 0x000fa400);
21.   setVideoReg (vi_c1b, 0x000fa400);
22.   setVideoReg (vi_c1c, 0x09625902);
23.   setVideoReg (vi_c2a, 0x00000000);
24.   setVideoReg (vi_c2b, 0x00000000);
25.   setVideoReg (vi_c2c, 0x00000000);
```

```
26.    setVideoReg (vi_c3a, 0x00000000);
27.    setVideoReg (vi_c3b, 0x00000000);
28.    setVideoReg (vi_c3c, 0x00000000);
29.    setVideoReg (vi_c4a, 0x00000000);
30.    setVideoReg (vi_c4b, 0x00000000);
31.    setVideoReg (vi_c4c, 0x00000000);
32.    setVideoReg (vi_c5a, 0x00000000);
33.    setVideoReg (vi_c5b, 0x00000000);
34.    setVideoReg (vi_c5c, 0x00000000);
35.    setVideoReg (vi_c6a, 0x00000000);
36.    setVideoReg (vi_c6b, 0x00000000);
37.    setVideoReg (vi_c6c, 0x00000000);
38.    setVideoReg (vi_c7a, 0x00000000);
39.    setVideoReg (vi_c7b, 0x00000000);
40.    setVideoReg (vi_c7c, 0x00000000);
41.    setVideoReg (vi_c8a, 0x00000000);
42.    setVideoReg (vi_c8b, 0x00000000);
43.    setVideoReg (vi_c8c, 0x00000000);
44.    //======== Continuous capture starts ========
45.    setVideoReg (vi_fc, 0x0002);
```

Note: If liner tile conversion is not being performed by the memory controller, 15. setVideoReg (vi_is, 0x0168); can be used instead of 15. setVideoReg (vi_is, 0x0200);.

13.5.2 Function for Specifying X-Direction Filter Coefficients

This video input module uses a nine-tap filter to scale down images in the X direction. In this architecture, the filter coefficients required to obtain good image quality depend on the position of the pixels to be generated and the image scaling ratio.

In general, to obtain good quality after scaling down an image, the frequency band should be limited according to the scaling ratio so that ringing does not appear in the scaled image. A general low-pass filter requires a large number of taps to obtain a good-quality image, but in this video input module the appropriate filter characteristics need to be determined to achieve good quality through a limited number of taps. The raised cosine function is a well established way to determine the required characteristics. List 2 shows a program that obtains filter characteristics suitable for the video input module by using this function.

Function gen_tap shown in list 2 uses arguments creg[], xs, and alpha100. Filter coefficients creg[] and xs can be obtained by assigning values in arguments xs and alpha100. The details of these arguments are shown in the following table.

Argument	Type	Description
creg[ ]	Unsigned long	Values to be set in CnA, CnB, and CnC registers (n = 1 to 8). The first index for the array specifies the position of the pixel to be generated. The second index specifies one of coefficient set registers A to C (0 for CnA, 1 for CnB, and 2 for CnC). For example, creg[5][2] specifies the C6C register.
xs	Unsigned long	Value to be set in the X scale register (XS) in the video input module. Specify a positive value within the range from H'1000 to H'FFFF.
alpha100	Int	The < value (damping factor) multiplied by 100 to be used in the raised cosine function. Specify a positive value within the range of $0 \leq$ alpha100 ≤ 100.

As the < value (damping ratio) of the raised cosine function comes closer to 0, the characteristics of the low-pass filter become more appropriate. However, due to the limited number of filter taps (nine taps) in the video input module, appropriate characteristics may not be obtained and a false contour may appear in some cases. As the α value comes closer to 1, the frequency characteristic curve toward the cutoff value of the low-pass filter becomes more gradual and contour blurring will appear.

Accordingly, the < value is set to 0.6 (alpha100 = 60) in this example.

List 2: Function for Specifying X-Direction Filter Coefficients

```
1.     #include <stdio.h>
2.     #include <math.h>
3.
4.     #define MAXBSIZ 255
5.     #define TAPSIDE 4
6.     #define TAPNUM (TAPSIDE*2+1)
7.     #define TAPDIV 8
8.
9.     #define CLIP 512
10.    #define SCLE 512
11.
12.    // control option
13.    #define COEFFCLIP 1
14.
15.    // #define DISPLAY
16.    // doublecoeff[TAPNUM*TAPDIV];
17.    // intc[TAPDIV][TAPNUM];
18.
19.    char
20.    gen_tap( unsigned long creg[][3], unsigned long xs, int alpha100 )
21.    {
22.    charoverflag;
23.
24.    inti, iofst j, k;
25.
26.    doublex, T, tT;
27.    doublecoeff[TAPNUM*TAPDIV];
28.    intc[TAPDIV][TAPNUM]
29.    doublecoeff_diff;
30.    doublesum;
31.    intsumd;
32.    doublealpha;
33.    // Raised Cosine Charactaristics Mode1
```

```
34.    // 0.0 <= alpha <= 1.0
35.    // 0   <= alpha100 <= 100
36.    if ((XS & 0xf000) == 0)  //Double xs when scaling up.
37.          XS * = 2.0;
38.    alpha = (double)(alpha100)/(double)100;
39.    //
40.    // calc coeff (double order)
41.    //
42.    T = (double)((unsigned long)(xs>>9));
43.    for ( j = 0; j < TAPDIV; j++) {
44.    for ( I = -TAPSIDE; I <= TAPSIDE; I++) {
45.    k = -j + I * TAPDIV;
46.    iofst = k + (TAPSIDE+1)*TAPDIV-1;
47.    if ( alpha100 == 0 ) {
48.    if ( k == 0 ) coeff[iofst] = 1.0;
49.    else {
50.    x = (double)M_PI*(double)k/T;
51.    coeff[iofst] = sin( x ) / x;
52.    }
53.    }
54.    else {
55.    if ( k == 0 ) coeff[iofst] = 1.0;
56.    else {
57.    x = (double)M_PI* (double)k/T;
58.    sum = (double)4.0 * (double)(alpha100*alpha100) * (double)(k*k);
59.    tT = (double)1.0 - sum/T/T/(double)10000;
60.    /*                                                    */
61.    /* if tT == 0 -> lim f(tT) = sin(PI/2alpha)/PI/2alpha * PI/4 */
62.    /* tT -> 0                                */
63.    if ( sum == T*T* (double)10000 ) {
64.    coeff[iofst] = sin((double)M_PI/(2.0*alpha))/((double)M_PI/
       (2.0*alpha)) * (double)M_PI/4.0;
65.    }
66.    else {
67.    coeff[iofst] = (sin(x)/x) * (cos(alpha*x)/tT);
```

```
68.     }
69.     }
70.     }
71.     }
72.     }
73.
74.     //
75.     // trans : double -> integer
76.     //
77.     overflag = 0;
78.     for ( j = 0; j < TAPDIV; j++) {
79.     sum = 0.0;
80.     for ( i = -TAPSIDE; i <= TAPSIDE; i++) {
81.     k = -j + i * 8;
82.     k += (TAPSIDE+1) *TAPDIV-1;
83.     sum += coeff[k];
84.     }
85.
86.     coeff_diff = 0.0;
87.     sumd = 0;
88.     for ( i = -TAPSIDE; i < ((j<=(TAPDIV/2))?0:1); i++) {
89.     k = -j + i * 8;
90.     k += (TAPSIDE+1)*TAPDIV-1;
91.     iofst = i+TAPSIDE;
92.     c[j][iofst] = (int)( (double)SCLE*(coeff[k]/sum) + coeff_diff );
93.
94.     // for jitter
95.     coeff_diff = (double)SCLE*(coeff[k]/sum)) - (double)c[j][iofst];
96.     sumd += c[j][iofst];
97.     if( c[j][iofst] >= CLIP || c[j][iofst] < -CLIP ) {
98.     overflag = 1;
99.     }
100.    }
101.    coeff_diff = 0.0;
102.    for ( i = TAPSIDE; i >= ((j<=(TAPDIV/2))?0:1); i-- ) {
```

```
103.   k = -j + i * 8;
104.   k += (TAPSIDE+1)*TAPDIV-1;
105.   iofst = i+TAPSIDE;
106.   c[j][iofst] = (int)( (double)SCLE*(coeff[k]/sum) + coeff_diff );
107.
108.   // for jitter
109.   if ( i != ((j<=(TAPDIV/2))?0:1) ) {
110.   coeff_diff = ((double)SCLE*(coeff[k]/sum)) - (double)c[j][iofst];
111.   sumd += c[j][iofst];
112.   }
113.   else {
114.   c[j][iofst] = SCLE - sumd;
115.   }
116.   if( c[j][iofst] >= CLIP || c[j][iofst] < -CLIP ) {
117.   overflag = 1;
118.   }
119.   }
120.
121.   //
122.   // when coeff[center] == CLIP && coeff[else] == 0 then coeff
      [center]--
123.   //
124.   if ( COEFFCLIP && overflag == 1) {
125.   for ( i = -TAPSIDE; i <= TAPSIDE; i++) {
126.   iofst = i+TAPSIDE;
127.   if( c[j][iofst] >= CLIP ) {
128.   c[j][iofst] = CLIP-1;
129.   }
130.   else if( c[j][iofst] < -CLIP ) {
131.   c[j][iofst] = -CLIP;
132.   }
133.   }
134.   overflag = 0;
135.   }
136.   }
```

```
137.
138.   for ( j = 0; j < TAPDIV; j++) {
139.   creg[j][0] = ((unsigned long)c[j][8]&0x3ff)<<20;
140.   creg[j][0] |= ((unsigned long)c[j][7]&0x3ff)<<10;
141.   creg[j][0] |= ((unsigned long)c[j][6]&0x3ff);
142.
143.   creg[j][1] = ((unsigned long)c[j][0]&0x3ff)<<20;
144.   creg[j][1] |= ((unsigned long)c[j][1]&0x3ff)<<10;
145.   creg[j][1] |= ((unsigned long)c[j][2]&0x3ff);
146.
147.   creg[j][2] = ((unsigned long)c[j][3]&0x3ff)<<20;
148.   creg[j][2] |= ((unsigned long)c[j][5]&0x3ff)<<10;
149.   creg[j][2] |= ((unsigned long)c[j][4]&0x3ff);
150.   }
151.
152.   return( overflag );
153.
154.   }
```

13.6 Usage Notes

13.6.1 Cautions Required for Scaling Up

When using the module's function of scaling up, displaying only video images is recommended.

The module transfers capture data to memory according to the access control of the memory controller.

If, however, the setting for scaling up and access to a different module are mixed, the realtime property available from the memory controller may not be met. For further details, refer to the overview.

Section 14 Display Unit (DU)

14.1 Overview

14.1.1 Features

(1) Display Channel

One channel is provided for this module.

(2) Plane

The display surfaces normally called the foreground, background, and cursor, are called planes in this document. The maximum display resolution of planes is WXGA (1280 × 768)*. Parameters for each plane can be set independently through the settings of an internal register. The internal register settings can also be used to set the display priority order.

Note: * In cases of high-resolution display, the unified memory traffic volume may be considerable depending on the number of combined planes and display size, and constraints may arise owing to the traffic volume.

- Display size
- Display position
- Display data format (8 bits/pixel, 16 bits/pixel, 32 bits/pixel, ARGB, or YC)
- Plane superpositioning
- Scrolling
- Wrapping-around
- Blinking
- Buffering control

The internal register settings can be used to select four different control modes.

- Auto rendering mode (double buffering)
- Manual display change mode (double buffering)
- Auto display change mode (double buffering)
- Video capture mode (triple buffering)

(3) Type and Number of Planes

There are eight planes (display planes) available, each using both image data and an alpha value.

Up to eight display planes can be superposed on each other. Correspondences between planes and display channels are not fixed but can be selected as desired through register settings.

(4) Synchronization Method

Internal register settings can be used to select any of three synchronization modes for the display output timing.

- Master mode (internal sync mode)
- TV sync mode (external sync mode)
- Sync method switching mode

(5) CRT Scan Mode (CRT Scan Method)

Internal register settings can be used to select from among three scan modes.

- Non-interlaced mode
- Interlaced sync mode
- Interlaced sync & video mode

(6) YC→RGB Color Space Conversion Functions

Image data stored in YC format can be converted into the RGB color space and displayed in a window. The conversion coefficients can be set in a register.
However, data for two or more pixels cannot be converted into the RGB color space at the same time.
Data in YUV422 and YUV420 format are convertible into the RGB color space.

(7) Color Palette

Four internal color palette planes are provided, capable of simultaneously displaying 256 colors out of a possible 260,000 colors. Eight-bit blending ratios are provided for every 256 colors.

(8) Display Capture

The RGB888 or RGB666 data for output is convertible into RGB565 or ARGB1555 data for separate storage in external memory.

(9) Register Access Control

The module has internal control registers; these are writable/readable by the HPB protocol over the HPB. The unit of access is fixed to 32 bits.

(10) Display Output Compare Function

Whether the current display output data is as expected can be checked. This function calculates a CRC value for a specified plane area and compares it with the expected CRC value that has been calculated in advance.

14.1.2 Block Diagram

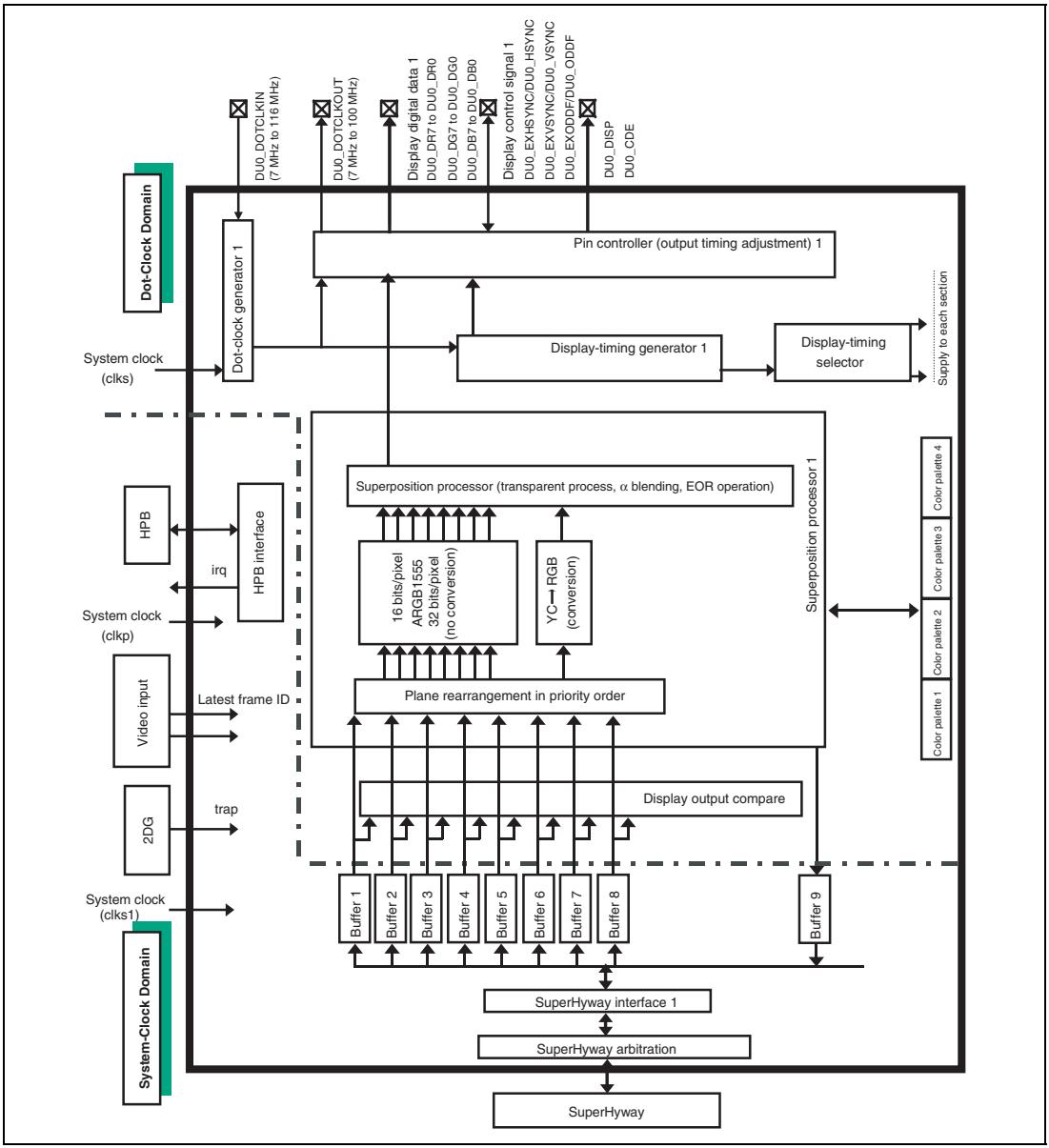


Figure 14.1 Block Diagram of the Display Unit (DU)

14.1.3 External Pins

Table 14.1 Pin Functions (DU0)

Name	Pin Name	I/O	Function	Signal Name Used in This Document
DU0 input dot clock	DU0_DOTCLKIN	Input	Input dot clock	DCLKIN
DU0 output dot clock	DU0_DOTCLKOUT	Output	Output dot clock	DCLKOUT
DU0 horizontal synchronous output/DU0 external horizontal synchronous input	DU0_HSYNC/ DU0_EXHSYNC	I/O	Composite synchronous output signal (Initial value)	CSYNC
			Horizontal synchronous output/External horizontal synchronous input	HSYNC or EXHSYNC*
DU0 vertical synchronous output/ DU0 external vertical synchronous input	DU0_VSYNC/ DU0_EXVSYNC	I/O	Vertical synchronous output/ External vertical synchronous input (Initial value)	VSYNC or EXVSYNC*
			Composite synchronous output signal	CSYNC
DU0 odd/even field	DU0_ODDF/ DU0_EXODDF/	I/O	Odd/even field (Initial value)	ODDF or EXODDF*
			CLAMP output signal	CLAMP
DU0 display interval	DU0_DISP	Output	Display interval (Initial value)	DISP
			Composite synchronous output signal	CSYNC
			DE output signal	DE
DU0 color detection	DU0_CDE	Output	Color detection	CDE

Name	Pin Name	I/O	Function	Signal Name Used in This Document
DU0 display data	DU0_DR0	Output	Digital 0 red 0	Digital RGB
	DU0_DR1	Output	Digital 0 red 1	
	DU0_DR2	Output	Digital 0 red 2	
	DU0_DR3	Output	Digital 0 red 3	
	DU0_DR4	Output	Digital 0 red 4	
	DU0_DR5	Output	Digital 0 red 5	
	DU0_DR6	Output	Digital 0 red 6	
	DU0_DR7	Output	Digital 0 red 7	
	DU0_DG0	Output	Digital 0 green 0	
	DU0_DG1	Output	Digital 0 green 1	
	DU0_DG2	Output	Digital 0 green 2	
	DU0_DG3	Output	Digital 0 green 3	
	DU0_DG4	Output	Digital 0 green 4	
	DU0_DG5	Output	Digital 0 green 5	
	DU0_DG6	Output	Digital 0 green 6	
	DU0_DG7	Output	Digital 0 green 7	
	DU0_DB0	Output	Digital 0 blue 0	
	DU0_DB1	Output	Digital 0 blue 1	
	DU0_DB2	Output	Digital 0 blue 2	
	DU0_DB3	Output	Digital 0 blue 3	
	DU0_DB4	Output	Digital 0 blue 4	
	DU0_DB5	Output	Digital 0 blue 5	
	DU0_DB6	Output	Digital 0 blue 6	
	DU0_DB7	Output	Digital 0 blue 7	

Notes: 1. When digital 0 and digital 1 are output at the same time, digital 0 and digital 1 are output in synchronization with rising and falling edges of the dot clock, respectively.
2. Synchronization for the output of digital 0 only is in accord with the setting of the output signal timing adjustment register (OTAR).
3. In this section, unless otherwise noted, "dot clock" refers to the output dot clock.
* These are expressed as EXHSYNC, EXVSYNC, and EXODDF in explanations of the functions as input signals and as HSYNC, VSYNC and ODDF otherwise.

14.2 Register Configuration

In the display unit (DU), register update methods include external update and internal update.

(1) External Update

An "external update" is an update which reflects the address-mapped register settings made by the CPU after the end of CPU access. Registers related to display control (for example, the display unit system control register) and the settings of which are updated through external updates can be overwritten without display flicker by using the VBK flag and FRM flag in the display unit status register (DSSR) indicating the start position of the vertical blanking interval.

(2) Internal Update

An "internal update" is an update that reflects the address-mapped register settings with the internal update timing of the display unit (DU). Hence in the case of a register with an internal update function, even when the CPU overwrites address-mapped registers related to display operation without being aware of the display timing, display flicker can be prevented.

An internal update is performed during the interval in which the display reset (DRES) bit in the display unit system control register (DSYSR) is 1 and at the beginning of each frame. The internal update performed at the beginning of each frame is disabled using the internal update disable (IUPD) bit in the display unit system control register (DSYSR).

Bits which are internally updated in response to setting of the display reset (DRES) bit in the display unit system control register (DSYSR) are listed in the column headed "Bit with Internal Update Function" in the lists of registers.

The registers for the X and Y start positions for plane n in the interlaced sync & video mode (PnSPXR, PnSPYR) are also internally updated at the beginning of a field.

Updates are performed at the falling edge of VSYNC output when the sync method of the display unit system control register (DSYSR) is master mode (TVM1 = 0, TVM0 = 0), or at the falling edge of EXVSYNC detected in TV sync mode (TVM1 = 1, TVM0 = 0). In sync transition mode (TVM1 = 0, TVM0 = 1), internal updates are not performed.

However, plane n display area start address 0 register, plane n display area start address 1 register, and plane n display area start address 2 register are internally updated in display operation and externally updated when the video data and rendering data are written to addresses specified for these registers.

The address-mapped registers with an internal update function are shown in tables 14.2 to 14.21. The initial settings for these registers should be made during the interval in which the DRES bit is 1.

(3) Display Control Register Configuration

Table 14.2 Display Control Register Configuration (1)

Register Name	Abbr.	R/W	Address in P4	Address in Area 7	Access Size	Bit with Internal Update Function
Display unit system control register	DSYSR	R/W	H'FFF80000	H'1FF80000	32 bits	DSEC (bit 20) DEN (bit 8)
Display unit mode register	DSMR	R/W	H'FFF80004	H'1FF80004	32 bits	All bits However, the following bits are updated with the DRES bit: VSPM (bit 28) ODPM (bit 27) DIPM (bits 26, 25) CSPM (bit 24) DIL (bit 19) VSL (bit 18) HSL (bit 17)
Display unit status register	DSSR	R	H'FFF80008	H'1FF80008	32 bits	None
Display unit status register clear register	DSRCR	W	H'FFF8000C	H'1FF8000C	32 bits	None
Display unit interrupt enable register	DIER	R/W	H'FFF80010	H'1FF80010	32 bits	None
Color palette control register	CPCR	R/W	H'FFF80014	H'1FF80014	32 bits	All bits
Display plane priority register	DPPR	R/W	H'FFF80018	H'1FF80018	32 bits	All bits
Display unit extensional function enable register	DEFR	R/W	H'FFF80020	H'1FF80020	32 bits	The following bits are updated with the DRES bit: EXSL (bit 12) EXUP (bit 5) VCUP (bit 4) DEFE (bit 0)

Register Name	Abbr.	R/W	Address in P4	Address in Area 7	Access Size	Bit with Internal Update Function
Display capture control register	DCPCR	R/W	H'FFF80028	H'1FF80028	32 bits	All bits
Display unit extensional function enable register 2	DEFR2	R/W	H'FFF80034	H'1FF80034	32 bits	All bits Updated with the DRES bit
Display unit extensional function enable register 3	DEFR3	R/W	H'FFF80038	H'1FF80038	32 bits	All bits However, the following bit is updated with the DRES bit: DEFE3 (bit 0)
Display unit extensional function enable register 4	DEFR4	R/W	H'FFF8003C	H'1FF8003C	32 bits	All bits However, the following bits are updated with the DRES bit: LRUO (bit 5) SPCE (bit 4)
Display unit video capture status register	DVCSR	R	H'FFF800D0	H'1FF800D0	32 bits	None

Table 14.3 Display Control Register Configuration (2)

Register Name	Abbr.	Power-on Reset/Deep Standby	Manual Reset	Sleep/ Module Standby/ Software Standby
Display unit system control register	DSYSR	H'00000280	H'00000280	Retained
Display unit mode register	DSMR	H'00000000	H'00000000	Retained
Display unit status register	DSSR	H'30000000	H'30000000	Retained
Display unit status register clear register	DSRCR	Undefined	Retained	Retained
Display unit interrupt enable register	DIER	H'00000000	H'00000000	Retained
Color palette control register	CPCR	H'00000000	H'00000000	Retained
Display plane priority register	DPPR	H'76543210	H'76543210	Retained
Display unit extensional function enable register	DEFR	H'00000000	H'00000000	Retained
Display capture control register	DCPCR	H'00000000	H'00000000	Retained
Display unit extensional function enable register 2	DEFR2	H'00000000	H'00000000	Retained
Display unit extensional function enable register 3	DEFR3	H'00000000	H'00000000	Retained
Display unit extensional function enable register 4	DEFR4	H'00000000	H'00000000	Retained
Display unit video capture status register	DVCSR	H'00000000	H'00000000	Retained

(4) Display Timing Generation Register Configuration

Table 14.4 Display Timing Generation Register Configuration (1)

Register Name	Abbr.	R/W	Address in P4	Address in Area 7	Access Size	Bit with Internal Update Function
Horizontal display start register	HDSR	R/W	H'FFF80040	H'1FF80040	32 bits	All bits
Horizontal display end register	HDER	R/W	H'FFF80044	H'1FF80044	32 bits	All bits
Vertical display start register	VDSR	R/W	H'FFF80048	H'1FF80048	32 bits	All bits
Vertical display end register	VDER	R/W	H'FFF8004C	H'1FF8004C	32 bits	All bits
Horizontal cycle register	HCR	R/W	H'FFF80050	H'1FF80050	32 bits	All bits
Horizontal sync width register	HSWR	R/W	H'FFF80054	H'1FF80054	32 bits	All bits
Vertical cycle register	VCR	R/W	H'FFF80058	H'1FF80058	32 bits	All bits
Vertical sync point register	VSPR	R/W	H'FFF8005C	H'1FF8005C	32 bits	All bits
Equal pulse width register	EQWR	R/W	H'FFF80060	H'1FF80060	32 bits	All bits
Serration width register	SPWR	R/W	H'FFF80064	H'1FF80064	32 bits	All bits
CLAMP signal start register	CLAMPSR	R/W	H'FFF80070	H'1FF80070	32 bits	All bits
CLAMP signal width register	CLAMPWR	R/W	H'FFF80074	H'1FF80074	32 bits	All bits
DE signal start register	DESR	R/W	H'FFF80078	H'1FF80078	32 bits	All bits
DE signal width register	DEWR	R/W	H'FFF8007C	H'1FF8007C	32 bits	All bits

Table 14.5 Display Timing Generation Register Configuration (2)

Register Name	Abbr.	Power-on Reset/Deep Standby	Manual Reset	Sleep/ Module Standby/ Software Standby
Horizontal display start register	HDSR	Undefined	Retained	Retained
Horizontal display end register	HDER	Undefined	Retained	Retained
Vertical display start register	VDSR	Undefined	Retained	Retained
Vertical display end register	VDER	Undefined	Retained	Retained
Horizontal cycle register	HCR	Undefined	Retained	Retained
Horizontal sync width register	HSWR	Undefined	Retained	Retained
Vertical cycle register	VCR	Undefined	Retained	Retained
Vertical sync point register	VSPR	Undefined	Retained	Retained
Equal pulse width register	EQWR	Undefined	Retained	Retained
Serration width register	SPWR	Undefined	Retained	Retained
CLAMP signal start register	CLAMPSR	Undefined	Retained	Retained
CLAMP signal width register	CLAMPWR	Undefined	Retained	Retained
DE signal start register	DESR	Undefined	Retained	Retained
DE signal width register	DEWR	Undefined	Retained	Retained

(5) Display Attribute Register Configuration

Table 14.6 Display Attribute Register Configuration (1)

Register Name	Abbr.	R/W	Address in P4	Address in Area 7	Access Size	Bit with Internal Update Function
Color palette 1 transparent color register	CP1TR	R/W	H'FFF80080	H'1FF80080	32 bits	All bits
Color palette 2 transparent color register	CP2TR	R/W	H'FFF80084	H'1FF80084	32 bits	All bits
Color palette 3 transparent color register	CP3TR	R/W	H'FFF80088	H'1FF80088	32 bits	All bits
Color palette 4 transparent color register	CP4TR	R/W	H'FFF8008C	H'1FF8008C	32 bits	All bits
Display off mode output register	DOOR	R/W	H'FFF80090	H'1FF80090	32 bits	All bits
Color detection register	CDER	R/W	H'FFF80094	H'1FF80094	32 bits	All bits
Background plane output register	BPOR	R/W	H'FFF80098	H'1FF80098	32 bits	All bits
Raster interrupt offset register	RINTOFSR	R/W	H'FFF8009C	H'1FF8009C	32 bits	All bits
Display SuperHyway priority register	DSHPR	R/W	H'FFF800C8	H'1FF800C8	32 bits	All bits Updated with the DRES bit

Table 14.7 Display Attribute Register Configuration (2)

Register Name	Abbr.	Power-on Reset/Deep Standby	Manual Reset	Sleep/ Module Standby/ Software Standby
Color palette 1 transparent color register	CP1TR	H'00000000	H'00000000	Retained
Color palette 2 transparent color register	CP2TR	H'00000000	H'00000000	Retained
Color palette 3 transparent color register	CP3TR	H'00000000	H'00000000	Retained
Color palette 4 transparent color register	CP4TR	H'00000000	H'00000000	Retained
Display off mode output register	DOOR	Undefined	Retained	Retained
Color detection register	CDER	Undefined	Retained	Retained
Background plane output register	BPOR	Undefined	Retained	Retained
Raster interrupt offset register	RINTOFSR	Undefined	Retained	Retained
Display SuperHyway priority register	DSHPR	H'000000A8	H'000000A8	Retained

(6) Display Plane Register Configuration

Table 14.8 Display Plane Register Configuration (1)

Register Name	Abbr.	R/W	Address in P4	Address in Area 7	Access Size	Bit with Internal Update Function
Plane 1 mode register	P1MR	R/W	H'FFF80100	H'1FF80100	32 bits	All bits
Plane 1 memory width register	P1MWR	R/W	H'FFF80104	H'1FF80104	32 bits	All bits
Plane 1 blending ratio register	P1ALPHAR	R/W	H'FFF80108	H'1FF80108	32 bits	All bits
Plane 1 display size X register	P1DSXR	R/W	H'FFF80110	H'1FF80110	32 bits	All bits
Plane 1 display size Y register	P1DSYR	R/W	H'FFF80114	H'1FF80114	32 bits	All bits
Plane 1 display position X register	P1DPXR	R/W	H'FFF80118	H'1FF80118	32 bits	All bits
Plane 1 display position Y register	P1DPYR	R/W	H'FFF8011C	H'1FF8011C	32 bits	All bits
Plane 1 display area start address 0 register	P1DSA0R	R/W	H'FFF80120	H'1FF80120	32 bits	All bits
Plane 1 display area start address 1 register	P1DSA1R	R/W	H'FFF80124	H'1FF80124	32 bits	All bits
Plane 1 display area start address 2 register	P1DSA2R	R/W	H'FFF80128	H'1FF80128	32 bits	All bits
Plane 1 start position X register	P1SPXR	R/W	H'FFF80130	H'1FF80130	32 bits	All bits
Plane 1 start position Y register	P1SPYR	R/W	H'FFF80134	H'1FF80134	32 bits	All bits
Plane 1 wrap around start position register	P1WASPR	R/W	H'FFF80138	H'1FF80138	32 bits	All bits
Plane 1 wrap around memory width register	P1WAMWR	R/W	H'FFF8013C	H'1FF8013C	32 bits	All bits
Plane 1 blinking time register	P1BTR	R/W	H'FFF80140	H'1FF80140	32 bits	All bits
Plane 1 transparent color 1 register	P1TC1R	R/W	H'FFF80144	H'1FF80144	32 bits	All bits
Plane 1 transparent color 2 register	P1TC2R	R/W	H'FFF80148	H'1FF80148	32 bits	All bits
Plane 1 memory length register	P1MLR	R/W	H'FFF80150	H'1FF80150	32 bits	All bits
Plane 1 swap control register	P1SWAPR	R/W	H'FFF80180	H'1FF80180	32 bits	All bits
Plane 1 display data control register	P1DDCR	R/W	H'FFF80184	H'1FF80184	32 bits	All bits

Register Name	Abbr.	R/W	Address in P4	Address in Area 7	Access Size	Bit with Internal Update Function
Plane 1 display data control register 2	P1DDCR2	R/W	H'FFF80188	H'1FF80188	32 bits	All bits
Plane 2 mode register	P2MR	R/W	H'FFF80200	H'1FF80200	32 bits	All bits
Plane 2 memory width register	P2MWR	R/W	H'FFF80204	H'1FF80204	32 bits	All bits
Plane 2 blending ratio register	P2ALPHAR	R/W	H'FFF80208	H'1FF80208	32 bits	All bits
Plane 2 display size X register	P2DSXR	R/W	H'FFF80210	H'1FF80210	32 bits	All bits
Plane 2 display size Y register	P2DSYR	R/W	H'FFF80214	H'1FF80214	32 bits	All bits
Plane 2 display position X register	P2DPXR	R/W	H'FFF80218	H'1FF80218	32 bits	All bits
Plane 2 display position Y register	P2DPYR	R/W	H'FFF8021C	H'1FF8021C	32 bits	All bits
Plane 2 display area start address 0 register	P2DSA0R	R/W	H'FFF80220	H'1FF80220	32 bits	All bits
Plane 2 display area start address 1 register	P2DSA1R	R/W	H'FFF80224	H'1FF80224	32 bits	All bits
Plane 2 display area start address 2 register	P2DSA2R	R/W	H'FFF80228	H'1FF80228	32 bits	All bits
Plane 2 start position X register	P2SPXR	R/W	H'FFF80230	H'1FF80230	32 bits	All bits
Plane 2 start position Y register	P2SPYR	R/W	H'FFF80234	H'1FF80234	32 bits	All bits
Plane 2 wrap around start position register	P2WASPR	R/W	H'FFF80238	H'1FF80238	32 bits	All bits
Plane 2 wrap around memory width register	P2WAMWR	R/W	H'FFF8023C	H'1FF8023C	32 bits	All bits
Plane 2 blinking time register	P2BTR	R/W	H'FFF80240	H'1FF80240	32 bits	All bits
Plane 2 transparent color 1 register	P2TC1R	R/W	H'FFF80244	H'1FF80244	32 bits	All bits
Plane 2 transparent color 2 register	P2TC2R	R/W	H'FFF80248	H'1FF80248	32 bits	All bits
Plane 2 memory length register	P2MLR	R/W	H'FFF80250	H'1FF80250	32 bits	All bits
Plane 2 swap control register	P2SWAPR	R/W	H'FFF80280	H'1FF80280	32 bits	All bits
Plane 2 display data control register	P2DDCR	R/W	H'FFF80284	H'1FF80284	32 bits	All bits

Register Name	Abbr.	R/W	Address in P4	Address in Area 7	Access Size	Bit with Internal Update Function
Plane 2 display data control register 2	P2DDCR2	R/W	H'FFF80288	H'1FF80288	32 bits	All bits
Plane 3 mode register	P3MR	R/W	H'FFF80300	H'1FF80300	32 bits	All bits
Plane 3 memory width register	P3MWR	R/W	H'FFF80304	H'1FF80304	32 bits	All bits
Plane 3 blending ratio register	P3ALPHAR	R/W	H'FFF80308	H'1FF80308	32 bits	All bits
Plane 3 display size X register	P3DSXR	R/W	H'FFF80310	H'1FF80310	32 bits	All bits
Plane 3 display size Y register	P3DSYR	R/W	H'FFF80314	H'1FF80314	32 bits	All bits
Plane 3 display position X register	P3DPXR	R/W	H'FFF80318	H'1FF80318	32 bits	All bits
Plane 3 display position Y register	P3DPYR	R/W	H'FFF8031C	H'1FF8031C	32 bits	All bits
Plane 3 display area start address 0 register	P3DSA0R	R/W	H'FFF80320	H'1FF80320	32 bits	All bits
Plane 3 display area start address 1 register	P3DSA1R	R/W	H'FFF80324	H'1FF80324	32 bits	All bits
Plane 3 display area start address 2 register	P3DSA2R	R/W	H'FFF80328	H'1FF80328	32 bits	All bits
Plane 3 start position X register	P3SPXR	R/W	H'FFF80330	H'1FF80330	32 bits	All bits
Plane 3 start position Y register	P3SPYR	R/W	H'FFF80334	H'1FF80334	32 bits	All bits
Plane 3 wrap around start position register	P3WASPR	R/W	H'FFF80338	H'1FF80338	32 bits	All bits
Plane 3 wrap around memory width register	P3WAMWR	R/W	H'FFF8033C	H'1FF8033C	32 bits	All bits
Plane 3 blinking time register	P3BTR	R/W	H'FFF80340	H'1FF80340	32 bits	All bits
Plane 3 transparent color 1 register	P3TC1R	R/W	H'FFF80344	H'1FF80344	32 bits	All bits
Plane 3 transparent color 2 register	P3TC2R	R/W	H'FFF80348	H'1FF80348	32 bits	All bits
Plane 3 memory length register	P3MLR	R/W	H'FFF80350	H'1FF80350	32 bits	All bits
Plane 3 swap control register	P3SWAPR	R/W	H'FFF80380	H'1FF80380	32 bits	All bits
Plane 3 display data control register	P3DDCR	R/W	H'FFF80384	H'1FF80384	32 bits	All bits

Register Name	Abbr.	R/W	Address in P4	Address in Area 7	Access Size	Bit with Internal Update Function
Plane 3 display data control register 2	P3DDCR2	R/W	H'FFF80388	H'1FF80388	32 bits	All bits
Plane 4 mode register	P4MR	R/W	H'FFF80400	H'1FF80400	32 bits	All bits
Plane 4 memory width register	P4MWR	R/W	H'FFF80404	H'1FF80404	32 bits	All bits
Plane 4 blending ratio register	P4ALPHAR	R/W	H'FFF80408	H'1FF80408	32 bits	All bits
Plane 4 display size X register	P4DSXR	R/W	H'FFF80410	H'1FF80410	32 bits	All bits
Plane 4 display size Y register	P4DSYR	R/W	H'FFF80414	H'1FF80414	32 bits	All bits
Plane 4 display position X register	P4DPXR	R/W	H'FFF80418	H'1FF80418	32 bits	All bits
Plane 4 display position Y register	P4DPYR	R/W	H'FFF8041C	H'1FF8041C	32 bits	All bits
Plane 4 display area start address 0 register	P4DSA0R	R/W	H'FFF80420	H'1FF80420	32 bits	All bits
Plane 4 display area start address 1 register	P4DSA1R	R/W	H'FFF80424	H'1FF80424	32 bits	All bits
Plane 4 display area start address 2 register	P4DSA2R	R/W	H'FFF80428	H'1FF80428	32 bits	All bits
Plane 4 start position X register	P4SPXR	R/W	H'FFF80430	H'1FF80430	32 bits	All bits
Plane 4 start position Y register	P4SPYR	R/W	H'FFF80434	H'1FF80434	32 bits	All bits
Plane 4 wrap around start position register	P4WASPR	R/W	H'FFF80438	H'1FF80438	32 bits	All bits
Plane 4 wrap around memory width register	P4WAMWR	R/W	H'FFF8043C	H'1FF8043C	32 bits	All bits
Plane 4 blinking time register	P4BTR	R/W	H'FFF80440	H'1FF80440	32 bits	All bits
Plane 4 transparent color 1 register	P4TC1R	R/W	H'FFF80444	H'1FF80444	32 bits	All bits
Plane 4 transparent color 2 register	P4TC2R	R/W	H'FFF80448	H'1FF80448	32 bits	All bits
Plane 4 memory length register	P4MLR	R/W	H'FFF80450	H'1FF80450	32 bits	All bits
Plane 4 swap control register	P4SWAPR	R/W	H'FFF80480	H'1FF80480	32 bits	All bits
Plane 4 display data control register	P4DDCR	R/W	H'FFF80484	H'1FF80484	32 bits	All bits
Plane 4 display data control register 2	P4DDCR2	R/W	H'FFF80488	H'1FF80488	32 bits	All bits

Register Name	Abbr.	R/W	Address in P4	Address in Area 7	Access Size	Bit with Internal Update Function
Plane 5 mode register	P5MR	R/W	H'FFF80500	H'1FF80500	32 bits	All bits
Plane 5 memory width register	P5MWR	R/W	H'FFF80504	H'1FF80504	32 bits	All bits
Plane 5 blending ratio register	P5ALPHAR	R/W	H'FFF80508	H'1FF80508	32 bits	All bits
Plane 5 display size X register	P5DSXR	R/W	H'FFF80510	H'1FF80510	32 bits	All bits
Plane 5 display size Y register	P5DSYR	R/W	H'FFF80514	H'1FF80514	32 bits	All bits
Plane 5 display position X register	P5DPXR	R/W	H'FFF80518	H'1FF80518	32 bits	All bits
Plane 5 display position Y register	P5DPYR	R/W	H'FFF8051C	H'1FF8051C	32 bits	All bits
Plane 5 display area start address 0 register	P5DSA0R	R/W	H'FFF80520	H'1FF80520	32 bits	All bits
Plane 5 display area start address 1 register	P5DSA1R	R/W	H'FFF80524	H'1FF80524	32 bits	All bits
Plane 5 display area start address 2 register	P5DSA2R	R/W	H'FFF80528	H'1FF80528	32 bits	All bits
Plane 5 start position X register	P5SPXR	R/W	H'FFF80530	H'1FF80530	32 bits	All bits
Plane 5 start position Y register	P5SPYR	R/W	H'FFF80534	H'1FF80534	32 bits	All bits
Plane 5 wrap around start position register	P5WASPR	R/W	H'FFF80538	H'1FF80538	32 bits	All bits
Plane 5 wrap around memory width register	P5WAMWR	R/W	H'FFF8053C	H'1FF8053C	32 bits	All bits
Plane 5 blinking time register	P5BTR	R/W	H'FFF80540	H'1FF80540	32 bits	All bits
Plane 5 transparent color 1 register	P5TC1R	R/W	H'FFF80544	H'1FF80544	32 bits	All bits
Plane 5 transparent color 2 register	P5TC2R	R/W	H'FFF80548	H'1FF80548	32 bits	All bits
Plane 5 memory length register	P5MLR	R/W	H'FFF80550	H'1FF80550	32 bits	All bits
Plane 5 swap control register	P5SWAPR	R/W	H'FFF80580	H'1FF80580	32 bits	All bits
Plane 5 display data control register	P5DDCR	R/W	H'FFF80584	H'1FF80584	32 bits	All bits
Plane 5 display data control register 2	P5DDCR2	R/W	H'FFF80588	H'1FF80588	32 bits	All bits

Register Name	Abbr.	R/W	Address in P4	Address in Area 7	Access Size	Bit with Internal Update Function
Plane 6 mode register	P6MR	R/W	H'FFF80600	H'1FF80600	32 bits	All bits
Plane 6 memory width register	P6MWR	R/W	H'FFF80604	H'1FF80604	32 bits	All bits
Plane 6 blending ratio register	P6ALPHAR	R/W	H'FFF80608	H'1FF80608	32 bits	All bits
Plane 6 display size X register	P6DSXR	R/W	H'FFF80610	H'1FF80610	32 bits	All bits
Plane 6 display size Y register	P6DSYR	R/W	H'FFF80614	H'1FF80614	32 bits	All bits
Plane 6 display position X register	P6DPXR	R/W	H'FFF80618	H'1FF80618	32 bits	All bits
Plane 6 display position Y register	P6DPYR	R/W	H'FFF8061C	H'1FF8061C	32 bits	All bits
Plane 6 display area start address 0 register	P6DSA0R	R/W	H'FFF80620	H'1FF80620	32 bits	All bits
Plane 6 display area start address 1 register	P6DSA1R	R/W	H'FFF80624	H'1FF80624	32 bits	All bits
Plane 6 display area start address 2 register	P6DSA2R	R/W	H'FFF80628	H'1FF80628	32 bits	All bits
Plane 6 start position X register	P6SPXR	R/W	H'FFF80630	H'1FF80630	32 bits	All bits
Plane 6 start position Y register	P6SPYR	R/W	H'FFF80634	H'1FF80634	32 bits	All bits
Plane 6 wrap around start position register	P6WASPR	R/W	H'FFF80638	H'1FF80638	32 bits	All bits
Plane 6 wrap around memory width register	P6WAMWR	R/W	H'FFF8063C	H'1FF8063C	32 bits	All bits
Plane 6 blinking time register	P6BTR	R/W	H'FFF80640	H'1FF80640	32 bits	All bits
Plane 6 transparent color 1 register	P6TC1R	R/W	H'FFF80644	H'1FF80644	32 bits	All bits
Plane 6 transparent color 2 register	P6TC2R	R/W	H'FFF80648	H'1FF80648	32 bits	All bits
Plane 6 memory length register	P6MLR	R/W	H'FFF80650	H'1FF80650	32 bits	All bits
Plane 6 swap control register	P6SWAPR	R/W	H'FFF80680	H'1FF80680	32 bits	All bits
Plane 6 display data control register	P6DDCR	R/W	H'FFF80684	H'1FF80684	32 bits	All bits
Plane 6 display data control register 2	P6DDCR2	R/W	H'FFF80688	H'1FF80688	32 bits	All bits

Register Name	Abbr.	R/W	Address in P4	Address in Area 7	Access Size	Bit with Internal Update Function
Plane 7 mode register	P7MR	R/W	H'FFF80700	H'1FF80700	32 bits	All bits
Plane 7 memory width register	P7MWR	R/W	H'FFF80704	H'1FF80704	32 bits	All bits
Plane 7 blending ratio register	P7ALPHAR	R/W	H'FFF80708	H'1FF80708	32 bits	All bits
Plane 7 display size X register	P7DSXR	R/W	H'FFF80710	H'1FF80710	32 bits	All bits
Plane 7 display size Y register	P7DSYR	R/W	H'FFF80714	H'1FF80714	32 bits	All bits
Plane 7 display position X register	P7DPXR	R/W	H'FFF80718	H'1FF80718	32 bits	All bits
Plane 7 display position Y register	P7DPYR	R/W	H'FFF8071C	H'1FF8071C	32 bits	All bits
Plane 7 display area start address 0 register	P7DSA0R	R/W	H'FFF80720	H'1FF80720	32 bits	All bits
Plane 7 display area start address 1 register	P7DSA1R	R/W	H'FFF80724	H'1FF80724	32 bits	All bits
Plane 7 display area start address 2 register	P7DSA2R	R/W	H'FFF80728	H'1FF80728	32 bits	All bits
Plane 7 start position X register	P7SPXR	R/W	H'FFF80730	H'1FF80730	32 bits	All bits
Plane 7 start position Y register	P7SPYR	R/W	H'FFF80734	H'1FF80734	32 bits	All bits
Plane 7 wrap around start position register	P7WASPR	R/W	H'FFF80738	H'1FF80738	32 bits	All bits
Plane 7 wrap around memory width register	P7WAMWR	R/W	H'FFF8073C	H'1FF8073C	32 bits	All bits
Plane 7 blinking time register	P7BTR	R/W	H'FFF80740	H'1FF80740	32 bits	All bits
Plane 7 transparent color 1 register	P7TC1R	R/W	H'FFF80744	H'1FF80744	32 bits	All bits
Plane 7 transparent color 2 register	P7TC2R	R/W	H'FFF80748	H'1FF80748	32 bits	All bits
Plane 7 memory length register	P7MLR	R/W	H'FFF80750	H'1FF80750	32 bits	All bits
Plane 7 swap control register	P7SWAPR	R/W	H'FFF80780	H'1FF80780	32 bits	All bits
Plane 7 display data control register	P7DDCR	R/W	H'FFF80784	H'1FF80784	32 bits	All bits
Plane 7 display data control register 2	P7DDCR2	R/W	H'FFF80788	H'1FF80788	32 bits	All bits

Register Name	Abbr.	R/W	Address in P4	Address in Area 7	Access Size	Bit with Internal Update Function
Plane 8 mode register	P8MR	R/W	H'FFF80800	H'1FF80800	32 bits	All bits
Plane 8 memory width register	P8MWR	R/W	H'FFF80804	H'1FF80804	32 bits	All bits
Plane 8 blending ratio register	P8ALPHAR	R/W	H'FFF80808	H'1FF80808	32 bits	All bits
Plane 8 display size X register	P8DSXR	R/W	H'FFF80810	H'1FF80810	32 bits	All bits
Plane 8 display size Y register	P8DSYR	R/W	H'FFF80814	H'1FF80814	32 bits	All bits
Plane 8 display position X register	P8DPXR	R/W	H'FFF80818	H'1FF80818	32 bits	All bits
Plane 8 display position Y register	P8DPYR	R/W	H'FFF8081C	H'1FF8081C	32 bits	All bits
Plane 8 display area start address 0 register	P8DSA0R	R/W	H'FFF80820	H'1FF80820	32 bits	All bits
Plane 8 display area start address 1 register	P8DSA1R	R/W	H'FFF80824	H'1FF80824	32 bits	All bits
Plane 8 display area start address 2 register	P8DSA2R	R/W	H'FFF80828	H'1FF80828	32 bits	All bits
Plane 8 start position X register	P8SPXR	R/W	H'FFF80830	H'1FF80830	32 bits	All bits
Plane 8 start position Y register	P8SPYR	R/W	H'FFF80834	H'1FF80834	32 bits	All bits
Plane 8 wrap around start position register	P8WASPR	R/W	H'FFF80838	H'1FF80838	32 bits	All bits
Plane 8 wrap around memory width register	P8WAMWR	R/W	H'FFF8083C	H'1FF8083C	32 bits	All bits
Plane 8 blinking time register	P8BTR	R/W	H'FFF80840	H'1FF80840	32 bits	All bits
Plane 8 transparent color 1 register	P8TC1R	R/W	H'FFF80844	H'1FF80844	32 bits	All bits
Plane 8 transparent color 2 register	P8TC2R	R/W	H'FFF80848	H'1FF80848	32 bits	All bits
Plane 8 memory length register	P8MLR	R/W	H'FFF80850	H'1FF80850	32 bits	All bits
Plane 8 swap control register	P8SWAPR	R/W	H'FFF80880	H'1FF80880	32 bits	All bits
Plane 8 display data control register	P8DDCR	R/W	H'FFF80884	H'1FF80884	32 bits	All bits
Plane 8 display data control register 2	P8DDCR2	R/W	H'FFF80888	H'1FF80888	32 bits	All bits

Table 14.9 Display Plane Register Configuration (2)

Register Name	Abbr.	Power-on Reset/Deep Standby	Manual Reset	Sleep/Module Standby/ Software Standby
Plane 1 mode register	P1MR	H'00000000	H'00000000	Retained
Plane 1 memory width register	P1MWR	Undefined	Retained	Retained
Plane 1 blending ratio register	P1ALPHAR	Undefined	Retained	Retained
Plane 1 display size X register	P1DSXR	Undefined	Retained	Retained
Plane 1 display size Y register	P1DSYR	Undefined	Retained	Retained
Plane 1 display position X register	P1DPXR	Undefined	Retained	Retained
Plane 1 display position Y register	P1DPYR	Undefined	Retained	Retained
Plane 1 display area start address 0 register	P1DSA0R	Undefined	Retained	Retained
Plane 1 display area start address 1 register	P1DSA1R	Undefined	Retained	Retained
Plane 1 display area start address 2 register	P1DSA2R	Undefined	Retained	Retained
Plane 1 start position X register	P1SPXR	Undefined	Retained	Retained
Plane 1 start position Y register	P1SPYR	Undefined	Retained	Retained
Plane 1 wrap around start position register	P1WASPR	Undefined	Retained	Retained
Plane 1 wrap around memory width register	P1WAMWR	Undefined	Retained	Retained
Plane 1 blinking time register	P1BTR	H'00000101	H'00000101	Retained
Plane 1 transparent color 1 register	P1TC1R	Undefined	Retained	Retained
Plane 1 transparent color 2 register	P1TC2R	Undefined	Retained	Retained
Plane 1 memory length register	P1MLR	H'00000000	H'00000000	Retained
Plane 1 swap control register	P1SWAPR	H'00000000	H'00000000	Retained
Plane 1 display data control register	P1DDCR	H'00000000	H'00000000	Retained
Plane 1 display data control register 2	P1DDCR2	H'00000000	H'00000000	Retained

Register Name	Abbr.	Power-on Reset/Deep Standby	Manual Reset	Sleep/Module Standby/ Software Standby
Plane 2 mode register	P2MR	H'00000000	H'00000000	Retained
Plane 2 memory width register	P2MWR	Undefined	Retained	Retained
Plane 2 blending ratio register	P2ALPHAR	Undefined	Retained	Retained
Plane 2 display size X register	P2DSXR	Undefined	Retained	Retained
Plane 2 display size Y register	P2DSYR	Undefined	Retained	Retained
Plane 2 display position X register	P2DPXR	Undefined	Retained	Retained
Plane 2 display position Y register	P2DPYR	Undefined	Retained	Retained
Plane 2 display area start address 0 register	P2DSA0R	Undefined	Retained	Retained
Plane 2 display area start address 1 register	P2DSA1R	Undefined	Retained	Retained
Plane 2 display area start address 2 register	P2DSA2R	Undefined	Retained	Retained
Plane 2 start position X register	P2SPXR	Undefined	Retained	Retained
Plane 2 start position Y register	P2SPYR	Undefined	Retained	Retained
Plane 2 wrap around start position register	P2WASPR	Undefined	Retained	Retained
Plane 2 wrap around memory width register	P2WAMWR	Undefined	Retained	Retained
Plane 2 blinking time register	P2BTR	H'00000101	H'00000101	Retained
Plane 2 transparent color 1 register	P2TC1R	Undefined	Retained	Retained
Plane 2 transparent color 2 register	P2TC2R	Undefined	Retained	Retained
Plane 2 memory length register	P2MLR	H'00000000	H'00000000	Retained
Plane 2 swap control register	P2SWAPR	H'00000000	H'00000000	Retained
Plane 2 display data control register	P2DDCR	H'00000000	H'00000000	Retained
Plane 2 display data control register 2	P2DDCR2	H'00000000	H'00000000	Retained

Register Name	Abbr.	Power-on Reset/Deep Standby	Manual Reset	Sleep/Module Standby/ Software Standby
Plane 3 mode register	P3MR	H'00000000	H'00000000	Retained
Plane 3 memory width register	P3MWR	Undefined	Retained	Retained
Plane 3 blending ratio register	P3ALPHAR	Undefined	Retained	Retained
Plane 3 display size X register	P3DSXR	Undefined	Retained	Retained
Plane 3 display size Y register	P3DSYR	Undefined	Retained	Retained
Plane 3 display position X register	P3DPXR	Undefined	Retained	Retained
Plane 3 display position Y register	P3DPYR	Undefined	Retained	Retained
Plane 3 display area start address 0 register	P3DSA0R	Undefined	Retained	Retained
Plane 3 display area start address 1 register	P3DSA1R	Undefined	Retained	Retained
Plane 3 display area start address 2 register	P3DSA2R	Undefined	Retained	Retained
Plane 3 start position X register	P3SPXR	Undefined	Retained	Retained
Plane 3 start position Y register	P3SPYR	Undefined	Retained	Retained
Plane 3 wrap around start position register	P3WASPR	Undefined	Retained	Retained
Plane 3 wrap around memory width register	P3WAMWR	Undefined	Retained	Retained
Plane 3 blinking time register	P3BTR	H'00000101	H'00000101	Retained
Plane 3 transparent color 1 register	P3TC1R	Undefined	Retained	Retained
Plane 3 transparent color 2 register	P3TC2R	Undefined	Retained	Retained
Plane 3 memory length register	P3MLR	H'00000000	H'00000000	Retained
Plane 3 swap control register	P3SWAPR	H'00000000	H'00000000	Retained
Plane 3 display data control register	P3DDCR	H'00000000	H'00000000	Retained
Plane 3 display data control register 2	P3DDCR2	H'00000000	H'00000000	Retained

Register Name	Abbr.	Power-on Reset/Deep Standby	Manual Reset	Sleep/Module Standby/ Software Standby
Plane 4 mode register	P4MR	H'00000000	H'00000000	Retained
Plane 4 memory width register	P4MWR	Undefined	Retained	Retained
Plane 4 blending ratio register	P4ALPHAR	Undefined	Retained	Retained
Plane 4 display size X register	P4DSXR	Undefined	Retained	Retained
Plane 4 display size Y register	P4DSYR	Undefined	Retained	Retained
Plane 4 display position X register	P4DPXR	Undefined	Retained	Retained
Plane 4 display position Y register	P4DPYR	Undefined	Retained	Retained
Plane 4 display area start address 0 register	P4DSA0R	Undefined	Retained	Retained
Plane 4 display area start address 1 register	P4DSA1R	Undefined	Retained	Retained
Plane 4 display area start address 2 register	P4DSA2R	Undefined	Retained	Retained
Plane 4 start position X register	P4SPXR	Undefined	Retained	Retained
Plane 4 start position Y register	P4SPYR	Undefined	Retained	Retained
Plane 4 wrap around start position register	P4WASPR	Undefined	Retained	Retained
Plane 4 wrap around memory width register	P4WAMWR	Undefined	Retained	Retained
Plane 4 blinking time register	P4BTR	H'00000101	H'00000101	Retained
Plane 4 transparent color 1 register	P4TC1R	Undefined	Retained	Retained
Plane 4 transparent color 2 register	P4TC2R	Undefined	Retained	Retained
Plane 4 memory length register	P4MLR	H'00000000	H'00000000	Retained
Plane 4 swap control register	P4SWAPR	H'00000000	H'00000000	Retained
Plane 4 display data control register	P4DDCR	H'00000000	H'00000000	Retained
Plane 4 display data control register 2	P4DDCR2	H'00000000	H'00000000	Retained

Register Name	Abbr.	Power-on Reset/Deep Standby	Manual Reset	Sleep/Module Standby/ Software Standby
Plane 5 mode register	P5MR	H'00000000	H'00000000	Retained
Plane 5 memory width register	P5MWR	Undefined	Retained	Retained
Plane 5 blending ratio register	P5ALPHAR	Undefined	Retained	Retained
Plane 5 display size X register	P5DSXR	Undefined	Retained	Retained
Plane 5 display size Y register	P5DSYR	Undefined	Retained	Retained
Plane 5 display position X register	P5DPXR	Undefined	Retained	Retained
Plane 5 display position Y register	P5DPYR	Undefined	Retained	Retained
Plane 5 display area start address 0 register	P5DSA0R	Undefined	Retained	Retained
Plane 5 display area start address 1 register	P5DSA1R	Undefined	Retained	Retained
Plane 5 display area start address 2 register	P5DSA2R	Undefined	Retained	Retained
Plane 5 start position X register	P5SPXR	Undefined	Retained	Retained
Plane 5 start position Y register	P5SPYR	Undefined	Retained	Retained
Plane 5 wrap around start position register	P5WASPR	Undefined	Retained	Retained
Plane 5 wrap around memory width register	P5WAMWR	Undefined	Retained	Retained
Plane 5 blinking time register	P5BTR	H'00000101	H'00000101	Retained
Plane 5 transparent color 1 register	P5TC1R	Undefined	Retained	Retained
Plane 5 transparent color 2 register	P5TC2R	Undefined	Retained	Retained
Plane 5 memory length register	P5MLR	H'00000000	H'00000000	Retained
Plane 5 swap control register	P5SWAPR	H'00000000	H'00000000	Retained
Plane 5 display data control register	P5DDCR	H'00000000	H'00000000	Retained
Plane 5 display data control register 2	P5DDCR2	H'00000000	H'00000000	Retained

Register Name	Abbr.	Power-on Reset/Deep Standby	Manual Reset	Sleep/Module Standby/ Software Standby
Plane 6 mode register	P6MR	H'00000000	H'00000000	Retained
Plane 6 memory width register	P6MWR	Undefined	Retained	Retained
Plane 6 blending ratio register	P6ALPHAR	Undefined	Retained	Retained
Plane 6 display size X register	P6DSXR	Undefined	Retained	Retained
Plane 6 display size Y register	P6DSYR	Undefined	Retained	Retained
Plane 6 display position X register	P6DPXR	Undefined	Retained	Retained
Plane 6 display position Y register	P6DPYR	Undefined	Retained	Retained
Plane 6 display area start address 0 register	P6DSA0R	Undefined	Retained	Retained
Plane 6 display area start address 1 register	P6DSA1R	Undefined	Retained	Retained
Plane 6 display area start address 2 register	P6DSA2R	Undefined	Retained	Retained
Plane 6 start position X register	P6SPXR	Undefined	Retained	Retained
Plane 6 start position Y register	P6SPYR	Undefined	Retained	Retained
Plane 6 wrap around start position register	P6WASPR	Undefined	Retained	Retained
Plane 6 wrap around memory width register	P6WAMWR	Undefined	Retained	Retained
Plane 6 blinking time register	P6BTR	H'00000101	H'00000101	Retained
Plane 6 transparent color 1 register	P6TC1R	Undefined	Retained	Retained
Plane 6 transparent color 2 register	P6TC2R	Undefined	Retained	Retained
Plane 6 memory length register	P6MLR	H'00000000	H'00000000	Retained
Plane 6 swap control register	P6SWAPR	H'00000000	H'00000000	Retained
Plane 6 display data control register	P6DDCR	H'00000000	H'00000000	Retained
Plane 6 display data control register 2	P6DDCR2	H'00000000	H'00000000	Retained

Register Name	Abbr.	Power-on Reset/Deep Standby	Manual Reset	Sleep/Module Standby/ Software Standby
Plane 7 mode register	P7MR	H'00000000	H'00000000	Retained
Plane 7 memory width register	P7MWR	Undefined	Retained	Retained
Plane 7 blending ratio register	P7ALPHAR	Undefined	Retained	Retained
Plane 7 display size X register	P7DSXR	Undefined	Retained	Retained
Plane 7 display size Y register	P7DSYR	Undefined	Retained	Retained
Plane 7 display position X register	P7DPXR	Undefined	Retained	Retained
Plane 7 display position Y register	P7DPYR	Undefined	Retained	Retained
Plane 7 display area start address 0 register	P7DSA0R	Undefined	Retained	Retained
Plane 7 display area start address 1 register	P7DSA1R	Undefined	Retained	Retained
Plane 7 display area start address 2 register	P7DSA2R	Undefined	Retained	Retained
Plane 7 start position X register	P7SPXR	Undefined	Retained	Retained
Plane 7 start position Y register	P7SPYR	Undefined	Retained	Retained
Plane 7 wrap around start position register	P7WASPR	Undefined	Retained	Retained
Plane 7 wrap around memory width register	P7WAMWR	Undefined	Retained	Retained
Plane 7 blinking time register	P7BTR	H'00000101	H'00000101	Retained
Plane 7 transparent color 1 register	P7TC1R	Undefined	Retained	Retained
Plane 7 transparent color 2 register	P7TC2R	Undefined	Retained	Retained
Plane 7 memory length register	P7MLR	H'00000000	H'00000000	Retained
Plane 7 swap control register	P7SWAPR	H'00000000	H'00000000	Retained
Plane 7 display data control register	P7DDCR	H'00000000	H'00000000	Retained
Plane 7 display data control register 2	P7DDCR2	H'00000000	H'00000000	Retained

Register Name	Abbr.	Power-on Reset/Deep Standby	Manual Reset	Sleep/Module Standby/ Software Standby
Plane 8 mode register	P8MR	H'00000000	H'00000000	Retained
Plane 8 memory width register	P8MWR	Undefined	Retained	Retained
Plane 8 blending ratio register	P8ALPHAR	Undefined	Retained	Retained
Plane 8 display size X register	P8DSXR	Undefined	Retained	Retained
Plane 8 display size Y register	P8DSYR	Undefined	Retained	Retained
Plane 8 display position X register	P8DPXR	Undefined	Retained	Retained
Plane 8 display position Y register	P8DPYR	Undefined	Retained	Retained
Plane 8 display area start address 0 register	P8DSA0R	Undefined	Retained	Retained
Plane 8 display area start address 1 register	P8DSA1R	Undefined	Retained	Retained
Plane 8 display area start address 2 register	P8DSA2R	Undefined	Retained	Retained
Plane 8 start position X register	P8SPXR	Undefined	Retained	Retained
Plane 8 start position Y register	P8SPYR	Undefined	Retained	Retained
Plane 8 wrap around start position register	P8WASPR	Undefined	Retained	Retained
Plane 8 wrap around memory width register	P8WAMWR	Undefined	Retained	Retained
Plane 8 blinking time register	P8BTR	H'00000101	H'00000101	Retained
Plane 8 transparent color 1 register	P8TC1R	Undefined	Retained	Retained
Plane 8 transparent color 2 register	P8TC2R	Undefined	Retained	Retained
Plane 8 memory length register	P8MLR	H'00000000	H'00000000	Retained
Plane 8 swap control register	P8SWAPR	H'00000000	H'00000000	Retained
Plane 8 display data control register	P8DDCR	H'00000000	H'00000000	Retained
Plane 8 display data control register 2	P8DDCR2	H'00000000	H'00000000	Retained

(7) Display Capture Register Configuration

Table 14.10 Display Capture Register Configuration (1)

Register Name	Abbr.	R/W	Address in P4	Address in Area 7	Access Size	Bit with Internal Update Function
Display capture 1 memory width register	DC1MWR	R/W	H'FFF8C104	H'1FF8C104	32 bits	All bits
Display capture 1 area start address register	DC1SAR	R/W	H'FFF8C120	H'1FF8C120	32 bits	All bits
Display capture 1 memory length register	DC1MLR	R/W	H'FFF8C150	H'1FF8C150	32 bits	All bits

Table 14.11 Display Capture Register Configuration (2)

Register Name	Abbr.	Power-on Reset/Deep Standby	Manual Reset	Sleep/ Module Standby/ Software Standby
Display capture 1 memory width register	DC1MWR	Undefined	Retained	Retained
Display capture 1 area start address register	DC1SAR	Undefined	Retained	Retained
Display capture 1 memory length register	DC1MLR	H'00000000	H'00000000	Retained

(8) Color Palette Register Configuration

Table 14.12 Color Palette Register Configuration (1)

Register Name	Abbr.	R/W	Address in P4	Address in Area 7	Access Size	Bit with Internal Update Function
Color palette 1 register 000	CP1_000R	R/W	H'FFF81000	H'1FF81000	32 bits	All bits
			~			
Color palette 1 register 255	CP1_255R	R/W	H'FFF813FC	H'1FF813FC	32 bits	All bits
Color palette 2 register 000	CP2_000R	R/W	H'FFF82000	H'1FF82000	32 bits	All bits
			~			
Color palette 2 register 255	CP2_255R	R/W	H'FFF823FC	H'1FF823FC	32 bits	All bits
Color palette 3 register 000	CP3_000R	R/W	H'FFF83000	H'1FF83000	32 bits	All bits
			~			
Color palette 3 register 255	CP3_255R	R/W	H'FFF833FC	H'1FF833FC	32 bits	All bits
Color palette 4 register 000	CP4_000R	R/W	H'FFF84000	H'1FF84000	32 bits	All bits
			~			
Color palette 4 register 255	CP4_255R	R/W	H'FFF843FC	H'1FF843FC	32 bits	All bits

Table 14.13 Color Palette Register Configuration (2)

Register Name	Abbr.	Power-on Reset/ Deep Standby	Manual Reset	Sleep/Module Standby/ Software Standby
Color palette 1 register 000	CP1_000R	Undefined	Retained	Retained
		~		
Color palette 1 register 255	CP1_255R	Undefined	Retained	Retained
Color palette 2 register 000	CP2_000R	Undefined	Retained	Retained
		~		
Color palette 2 register 255	CP2_255R	Undefined	Retained	Retained
Color palette 3 register 000	CP3_000R	Undefined	Retained	Retained
		~		
Color palette 3 register 255	CP3_255R	Undefined	Retained	Retained
Color palette 4 register 000	CP4_000R	Undefined	Retained	Retained
		~		
Color palette 4 register 255	CP4_255R	Undefined	Retained	Retained

(9) External Synchronization Control Register Configuration

Table 14.14 External Synchronization Control Register Configuration (1)

Register Name	Abbr.	R/W	Address in P4	Address in Area 7	Access Size	Bit with Internal Update Function
External synchronization control register	ESCR	R/W	H'FFF90000	H'1FF90000	32 bits	None
Output signal timing adjustment register	OTAR	R/W	H'FFF90004	H'1FF90004	32 bits	All bits updated with the DRES bit

Table 14.15 External Synchronization Control Register Configuration (2)

Register Name	Abbr.	Power-on Reset/Deep Standby	Manual Reset	Sleep/ Module Standby/ Software Standby
External synchronization control register	ESCR	H'00000000	H'00000000	Retained
Output signal timing adjustment register	OTAR	H'00000000	H'00000000	Retained

(10) Dual Display Output Control Register Configuration

Table 14.16 Dual Display Output Control Register Configuration (1)

Register Name	Abbr.	R/W	Address in P4	Address in Area 7	Access Size	Bit with Internal Update Function
Display unit output route control register	DORCR	R/W	H'FFF91000	H'1FF91000	32 bits	All bits updated with the DRES bit
Display superimpose 1 priority register	DS1PR	R/W	H'FFF91020	H'1FF91020	32 bits	All bits

Table 14.17 Dual Display Output Control Register Configuration (2)

Register Name	Abbr.	Power-on Reset/Deep Standby	Manual Reset	Sleep/ Module Standby/ Software Standby
Display unit output route control register	DORCR	H'00000000	H'00000000	Retained
Display superimpose 1 priority register	DS1PR	H'00000000	H'00000000	Retained

(11) YC-RGB Conversion Coefficient Register Configuration

Table 14.18 YC-RGB Conversion Coefficient Register Configuration (1)

Register Name	Abbr.	R/W	Address in P4	Address in Area 7	Access Size	Bit with Internal Update Function
Y normalization coefficient register	YNCR	R/W	H'FFF91080	H'1FF91080	32 bits	All bits
Y normalization offset register	YNOR	R/W	H'FFF91084	H'1FF91084	32 bits	All bits
Cr normalization offset register	CRNOR	R/W	H'FFF91088	H'1FF91088	32 bits	All bits
Cb normalization offset register	CBNOR	R/W	H'FFF9108C	H'1FF9108C	32 bits	All bits
Red Cr coefficient register	RCRCR	R/W	H'FFF91090	H'1FF91090	32 bits	All bits
Green Cr coefficient register	GCRCR	R/W	H'FFF91094	H'1FF91094	32 bits	All bits
Green Cb coefficient register	GCBCR	R/W	H'FFF91098	H'1FF91098	32 bits	All bits
Blue Cb coefficient register	BCBCR	R/W	H'FFF9109C	H'1FF9109C	32 bits	All bits

Table 14.19 YC-RGB Conversion Coefficient Register Configuration (2)

Register Name	Abbr.	Power-on Reset/Deep Standby	Manual Reset	Sleep/ Module Standby/ Software Standby
Y normalization coefficient register	YNCR	H'08000800	H'08000800	Retained
Y normalization offset register	YNOR	H'00000000	H'00000000	Retained
Cr normalization offset register	CRNOR	H'00800080	H'00800080	Retained
Cb normalization offset register	CBNOR	H'00800080	H'00800080	Retained
Red Cr coefficient register	RCRCR	H'0AF00AF0	H'0AF00AF0	Retained
Green Cr coefficient register	GCRCR	H'05900590	H'05900590	Retained
Green Cb coefficient register	GCBCR	H'02B002B0	H'02B002B0	Retained
Blue Cb coefficient register	BCBCR	H'0DE00DE0	H'0DE00DE0	Retained

(12) Display Output Compare Register Configuration

Table 14.20 Display Output Compare Register Configuration (1)

Register Name	Abbr.	R/W	Address in P4	Address in Area 7	Access Size	Bit with Internal Update Function
Display output compare control register	DOCMCR	R/W	H'FFF93000	H'1FF93000	32 bits	All bits
Display output compare status register	DOCMSTR	R	H'FFF93004	H'1FF93004	32 bits	None
Display output compare status clear register	DOCMCLSTR	—/W	H'FFF93008	H'1FF93008	32 bits	None
Display output compare interrupt enable register	DOCMIENR	R/W	H'FFF9300C	H'1FF9300C	32 bits	None
Display output compare mode register 1	DOCMMDR1	R/W	H'FFF93020	H'1FF93020	32 bits	None
Display output compare parameter register 1	DOCMPMR1	R/W	H'FFF93024	H'1FF93024	32 bits	All bits
Display output compare expected CRC value register 1	DOCMECRCR1	R/W	H'FFF93028	H'1FF93028	32 bits	All bits
Display output compare expected CRC value update register 1	DOCMECRCUR1	R	H'FFF9302C	H'1FF9302C	32 bits	None
Display output compare expected CRC value hold register 1	DOCMECRCHR1	R	H'FFF93030	H'1FF93030	32 bits	None
Display output compare calculated CRC value register 1	DOCMCCRCR1	R	H'FFF93034	H'1FF93034	32 bits	None
Display output compare start position X register 1	DOCMSPXR1	R/W	H'FFF93038	H'1FF93088	32 bits	All bits
Display output compare start position Y register 1	DOCMSPYR1	R/W	H'FFF9303C	H'1FF9303C	32 bits	All bits
Display output compare size X register 1	DOCMSZXR1	R/W	H'FFF93040	H'1FF93040	32 bits	All bits
Display output compare size Y register 1	DOCMSZYR1	R/W	H'FFF93044	H'1FF93044	32 bits	All bits
Display output compare CRC initial value register 1	DOCMCRCIR1	R/W	H'FFF93048	H'1FF93044	32 bits	None

Table 14.21 Display Output Compare Register Configuration (2)

Register Name	Abbr.	Power-on Reset/Deep Standby	Manual Reset	Sleep/ Module Standby/ Software Standby
Display output compare control register	DOCMCR	H'00000000	H'00000000	Retained
Display output compare status register	DOCMSTR	Undefined	Undefined	Retained
Display output compare status clear register	DOCMCLSTR	H'00000000	H'00000000	Retained
Display output compare interrupt enable register	DOCMIENR	H'00000000	H'00000000	Retained
Display output compare mode register 1	DOCMMDR1	H'00000000	H'00000000	Retained
Display output compare parameter register 1	DOCMPMR1	H'00000000	H'00000000	Retained
Display output compare expected CRC value register 1	DOCMECRCR1	H'00000000	H'00000000	Retained
Display output compare expected CRC value update register 1	DOCMECRCUR1	H'00000000	H'00000000	Retained
Display output compare expected CRC value hold register 1	DOCMECRCHR1	H'00000000	H'00000000	Retained
Display output compare calculated CRC value register 1	DOCMCCRCR1	H'00000000	H'00000000	Retained
Display output compare start position X register 1	DOCMSPXR1	H'00000000	H'00000000	Retained
Display output compare start position Y register 1	DOCMSPYR1	H'00000000	H'00000000	Retained
Display output compare size X register 1	DOCMSZXR1	H'00000000	H'00000000	Retained
Display output compare size Y register 1	DOCMSZYR1	H'00000000	H'00000000	Retained
Display output compare CRC initial value register 1	DOCMCRCIR1	H'FFFFFFFF	H'FFFFFFFF	Retained

14.3 Description of Display Unit Registers

The description of the registers within the display unit is given below. The symbols in the lower part of each register have the following meanings in the register setting:

Initial value: Register value after a power-on reset.

—: Undefined value

R/W: Can be read from or written to; the written value can be read.

R/WC0: Can be read from or written to; writing 0 initializes the bit but writing 1 is ignored.

R: Readable only, the write value should always be 0.

—/W: Write-only, an undefined value is read.

Plane n: Indicates plane 1 to plane 8.

Note: Do not access locations that appear empty in the address map. Operation is not guaranteed in case of access to change the value at any location other than those of the registers listed in tables 14.2 to 14.21.

14.3.1 Display Control Registers

(1) Display Unit System Control Register (DSYSR)

Address: H'FFF80000

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	ILTS	—	—	—	—	—	—	—	—	DSEC	—	—	—	IUPD
Initial value:	—	—	0	—	—	—	—	—	—	—	—	0	—	—	—	0
R/W:	R	R	R/W	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	DRES	DEN	TVM		SCM		—	—	—	—
Initial value:	—	—	—	—	—	—	1	0	1	0	0	0	—	—	—	—
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31, 30	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
29	ILTS	0	R/W	Not available	Input Pad Latch Timing Select To enable this bit, the DEFE bit in DEFR should be set to 1. In the initial state, this bit is fixed to 0. 0: A signal of the input pad is latched at the DCLKIN rising edge. (default) 1: A signal of the input pad is latched at the DCLKIN falling edge. Electrical characteristics do not apply.
28 to 21	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
20	DSEC	0	R/W	Available	Display Data Endian Change For details on data swapping, see section 14.4.7, Endian Conversion. 0: Display data on memory is not swapped in byte or word units. 1: Display data on memory is swapped in byte or word units.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
19 to 17	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
16	IUPD	0	R/W	Not available	Internal Updating Disable When DRES = 1, internal register updating occurs regardless of this bit setting. For details on internal updating, see section 14.2 (2), Internal Update. 0: Internal register updating occurs for each vertical synchronous signal (VSYNC) assertion. 1: If this bit is set to 1, internal register updating does not occur. If this bit is set to 0, register updating occurs according to the next vertical synchronous signal (VSYNC). If the display reset (DRES) is 1, internal updating occurs regardless of this bit.
15 to 10	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
9	DRES	1	R/W	Not available	Display Reset
8	DEN	0	R/W	Available	Display Enable 00: Display synchronous operation starts. If any register has not been set yet, the display unit may perform unexpected operation, thus set DRES to 0 after setting all registers in the display unit. For DEN = 0, the display data becomes values set with the display off mode output register (DOOR). 01: Display synchronous operation starts. If any register has not been set yet, the display unit may perform unexpected operation, thus set DRES = 0, DEN = 1 after setting all registers in the display unit. For DEN = 1, the display data becomes values stored on the unified memory from the next frame. 10: Display operation stops. Display operation and synchronization operation stop. The set values except for the following bits in the display unit status register (DSSR) are retained. In this setting, the following operations occur: 1. All 0s are output as display data. 2. The following bits of the display unit status register (DSSR) are cleared to 0: TV synchronous signal error flag (TVR) Frame flag (FRM) Vertical blanking flag (VBK) Raster interrupt flag (RINT) Horizontal blanking flag (HBK)

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
9	DRES	1	R/W	Not available	3. HSYNC, VSYNC, and ODDF pins function as inputs. However, the ODDF pin functions as a CLAMP output when the ODPM bit in the display unit mode register (DSMR) is 1. 11: Setting prohibited
8	DEN	0	R/W	Available	
7, 6	TVM	10	R/W	Not available	TV Synchronization Mode 00: Master mode HSYNC, VSYNC, and CSYNC are output. 01: Switching mode of synchronous mode Use this mode if necessary when switching from TV synchronous mode to master mode or vice versa is performed. In this mode, display operation forcibly stops and the DISP pin outputs the low level. Moreover, clock supply to DCLKIN can be stopped (input invalid) (the inside of the LSI is fixed to the high level). The EXHSYNC, EXVSYNC, and EXODDF pins are used as input pins. 10: TV synchronous mode EXHSYNC, EXVSYNC, and EXODDF are input. When the ODPM bit in the display unit mode register (DSMR) is set to 1, the ODDF pin functions as an output pin. 11: Setting prohibited
5, 4	SCM	00	R/W	Not available	Scan Mode 00: Non-interlaced mode 01: Setting prohibited 10: Interlace sync mode 11: Interlace sync & video mode
3 to 0	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

(2) Display Unit Mode Register (DSMR)

Address: H'FFF80004

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	VSPM	ODPM	DIPM		CSPM	—	—	—	—	DIL	VSL	HSL	DDIS
Initial value:	—	—	—	0	0	0	0	0	—	—	—	—	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CDEL	CDEM		CDED	—	—	—	ODEV	CSY		—	—	—	—	—	—
Initial value:	0	0	0	0	—	—	—	0	0	0	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 29	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
28	VSPM	0	R/W	DRES	VSYNC Pin Mode 0: The VSYNC signal is output to the VSYNC pin. 1: The CSYNC signal is output to the VSYNC pin. The VSYNC pin corresponds to the DU0_VSYNC/DU0_EXVSYNC pin in table 14.1, Pin Functions (DU0).
27	ODPM	0	R/W	DRES	ODDF Pin Mode 0: The ODDF signal is output to the ODDF pin. 1: The CLAMP signal is output to the ODDF pin. Even if the TVM bit in the display unit system control register (DSYSR) indicates TV synchronous mode, the ODDF pin becomes an output. The ODDF pin corresponds to the DU0_ODDF/DU0_EXODDF pin in table 14.1, Pin Functions (DU0).

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
26, 25	DIPM	00	R/W	DRES	DISP Pin Mode 00: The DISP signal is output to the DISP pin. 01: The CSYNC signal is output to the DISP pin. 10: Setting prohibited. (Fixed to 0.) 11: The DE signal is output to the DISP pin. The DISP pin corresponds to the DU0_DISP pin in table 14.1, Pin Functions (DU0).
24	CSPM	0	R/W	DRES	CSYNC Pin Mode 0: The CSYNC signal is output to the CSYNC pin. 1: The HSYNC signal is output to the CSYNC pin. The CSYNC pin corresponds to the DU0_HSYNC/DU0_EXHSYNC pin in table 14.1, Pin Functions (DU0).
23 to 20	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
19	DIL	0	R/W	DRES	DISP Polarity Select 0: The DISP signal becomes a high level during display period. 1: The polarity of the DISP signal is inverted.
18	VSL	0	R/W	DRES	VSYNC Polarity Select 0: The VSYNC signal becomes active low. 1: The polarity of VSYNC is inverted.
17	HSL	0	R/W	DRES	HSYNC Polarity Select 0: The HSYNC signal becomes active low. 1: The polarity of the HSYNC signal is inverted.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
16	DDIS	0	R/W	Available	DISP Disable 0: The DISP signal is output. 1: The DISP signal is not output.
15	CDEL	0	R/W	Available	CDE Polarity Select 0: The CDE signal becomes the high level when the output display data and color detection register (CDER) match. 1: The polarity of the CDE signal is inverted.
14, 13	CDEM	00	R/W	Available	CDE Output Mode 00: The CDE signal is output as is. (default) 01: The CDE signal is output as is. 10: The low level is output outside the display period. 11: The high level is output outside the display period.
12	CDED	0	R/W	Available	CDE Disable 0: The CDE signal is output. 1: The CDE signal output is prohibited.
11 to 9	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
8	ODEV	0	R/W	Available	ODD Even Select for ODDF Signal 0: In the same frame of the interlace display, the first half of the field is indicated with ODDF = Low level. 1: In the same frame of the interlace display, the first half of the field is indicated with ODDF = High level.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
7, 6	CSY	00	R/W	Available	CSYNC Mode 00: A waveform obtained by performing exclusive logical OR for VSYNC and HSYNC is output as CSYNC. 01: Setting prohibited 10: During intervals of 3 rasters from the VSYNC falling edge, the equalizing pulse is output. After that, during intervals of 3 rasters, the serration pulse is output. After that, during intervals of 3 rasters, the equalizing pulse is output. During other intervals, the HSYNC waveform is output as CSYNC. 11: After 1/2 rasters from the VSYNC falling edge, the equalizing pulse is output during intervals of 2.5 rasters. After that, during intervals of 2.5 rasters, the serration pulse is output. After that, during intervals of 2.5 rasters, the equalizing pulse is output. During other intervals, the HSYNC waveform is output as CSYNC.
5 to 0	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

(3) Display Unit Status Register (DSSR)

Address: H'FFF80008

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	VCFB		—	—	—	—	DFB8	DFB7	DFB6	DFB5	DFB4	DFB3	DFB2	DFB1
Initial value:	—	—	1	1	—	—	—	—	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TVR	FRM	—	—	VBK	CMPI	RINT	HBK	ADC8	ADC7	ADC6	ADC5	ADC4	ADC3	ADC2	ADC1
Initial value:	0	0	—	—	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31, 30	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
29, 28	VCFB	11	R	Not available	Video Capture Frame Buffer Flag 00: In the plane set for the video capture, the address indicated with PnDSA0R is in use as the display area start address. 01: In the plane set for the video capture, the address indicated with PnDSA1R is in use as the display area start address. 10: In the plane set for the video capture, the address indicated with PnDSA2R is in use as the display area start address. 11: The video capture module is in the initial status.
27 to 24	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
23	DFB8	0	R	Not available	Display Frame Buffer 8 Flag 0: In plane 8, the address indicated with P8DSA0R is in use as the display area start address. 1: In plane 8, the address indicated with P8DSA1R is in use as the display area start address.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
22	DFB7	0	R	Not available	Display Frame Buffer 7 Flag 0: In plane 7, the address indicated with P7DSA0R is in use as the display area start address. 1: In plane 7, the address indicated with P7DSA1R is in use as the display area start address.
21	DFB6	0	R	Not available	Display Frame Buffer 6 Flag 0: In plane 6, the address indicated with P6DSA0R is in use as the display area start address. 1: In plane 6, the address indicated with P6DSA1R is in use as the display area start address.
20	DFB5	0	R	Not available	Display Frame Buffer 5 Flag 0: In plane 5, the address indicated with P5DSA0R is in use as the display area start address. 1: In plane 5, the address indicated with P5DSA1R is in use as the display area start address.
19	DFB4	0	R	Not available	Display Frame Buffer 4 Flag 0: In plane 4, the address indicated with P4DSA0R is in use as the display area start address. 1: In plane 4, the address indicated with P4DSA1R is in use as the display area start address.
18	DFB3	0	R	Not available	Display Frame Buffer 3 Flag 0: In plane 3, the address indicated with P3DSA0R is in use as the display area start address. 1: In plane 3, the address indicated with P3DSA1R is in use as the display area start address.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
17	DFB2	0	R	Not available	Display Frame Buffer 2 Flag 0: In plane 2, the address indicated with P2DSA0R is in use as the display area start address. 1: In plane 2, the address indicated with P2DSA1R is in use as the display area start address.
16	DFB1	0	R	Not available	Display Frame Buffer 1 Flag 0: In plane 1, the address indicated with P1DSA0R is in use as the display area start address. 1: In plane 1, the address indicated with P1DSA1R is in use as the display area start address.
15	TVR	0	R	Not available	TV Synchronization Error Flag 0: Indicates that, after the TVR bit is cleared with the DRES bit of the display unit system control register (DSYSR) or the TVCL bit of the display unit status register clear register (DSRCR), the EXVSYNC rising edge is detected every time within the vertical cycle determined with the setting of the vertical cycle register (VCR). 1: Indicates that, in TV synchronous mode, the EXVSYNC rising edge was not detected within the vertical cycle determined with the setting of the vertical cycle register (VCR). The TVR bit retains its status until it is cleared with the DRES or TVCL bit.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
14	FRM	0	R	Not available	Frame Flag 0: Indicates the period from the time when the FRM bit is cleared by the DRES bit of the display unit system control register (DSYSR) or by the FRCL bit of the display unit status register clear register (DSRCR) to the time when display of the next field is completed in the non-interlaced mode, or when display of the next even field is completed in interlace sync mode or interlace sync & video mode. 1: Indicates the period (in units of frames) from the start of the vertical blanking period of the first even field after the FRM bit is cleared by the DRES or FRCL bit, to the time when the FRM bit is cleared again.
13, 12	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
11	VBK	0	R	Not available	Vertical Blanking Flag 0: Indicates the period from the time when the VBK bit is cleared by the DRES bit of the display unit system control register (DSYSR) or by the VBCL bit of the display unit status register clear register (DSRCR) to the time when display of the next field is completed. 1: Indicates the period (in units of fields) from the start of the first vertical blanking period after the VBK bit is cleared by the DRES or VBCL bit, to the time when the VBK bit is cleared again.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
10	CMPI	0	R	Not available	Display Output Compare Interrupt Flag 0: Indicates that no interrupt has occurred in display output compare operation. 1: Indicates that an interrupt has occurred in display output compare operation. Note: Interrupts from the display output compare function can be enabled through the display output compare interrupt enable register (DOCMIENR).
9	RINT	0	R	Not available	Raster Interrupt Flag 0: Indicates the period from the time when the RINT bit is cleared by the DRES bit of the display unit system control register (DSYSR) or by the RICL bit of the display unit status register clear register (DSRCR) to the time when the period of the rasters set in the raster interrupt offset register has elapsed after the beginning of the next field display. 1: Indicates the period from the time when the period of the rasters set in the raster interrupt offset register has elapsed after the beginning of the next field display after the RINT bit is cleared by the DRES or RICL bit, to the time when the RINT bit is cleared again.
8	HBK	0	R	Not available	Horizontal Blanking Flag 0: Indicates the period from the time when the HBK bit is cleared by the DRES bit of the display unit system control register (DSYSR) or by the HBCL bit of the display unit status register clear register (DSRCR) to the time of the next HSYNC assertion. 1: Indicates the period from the start of the first horizontal blanking period after the HBK bit is cleared by the DRES or HBCL bit, to the time when the HBK bit is cleared again.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
7	ADC8	0	R	Not available	Auto Rendering Display Change Flag 8 0: Indicates that the frame buffer for plane 8 has not been switched. 1: Indicates that the frame buffer for plane 8 has been switched. The ADC8 bit state is held until it is cleared.
6	ADC7	0	R	Not available	Auto Rendering Display Change Flag 7 0: Indicates that the frame buffer for plane 7 has not been switched. 1: Indicates that the frame buffer for plane 7 has been switched. The ADC7 bit state is held until it is cleared.
5	ADC6	0	R	Not available	Auto Rendering Display Change Flag 6 0: Indicates that the frame buffer for plane 6 has not been switched. 1: Indicates that the frame buffer for plane 6 has been switched. The ADC6 bit state is held until it is cleared.
4	ADC5	0	R	Not available	Auto Rendering Display Change Flag 5 0: Indicates that the frame buffer for plane 5 has not been switched. 1: Indicates that the frame buffer for plane 5 has been switched. The ADC5 bit state is held until it is cleared.
3	ADC4	0	R	Not available	Auto Rendering Display Change Flag 4 0: Indicates that the frame buffer for plane 4 has not been switched. 1: Indicates that the frame buffer for plane 4 has been switched. The ADC4 bit state is held until it is cleared.
2	ADC3	0	R	Not available	Auto Rendering Display Change Flag 3 0: Indicates that the frame buffer for plane 3 has not been switched. 1: Indicates that the frame buffer for plane 3 has been switched. The ADC3 bit state is held until it is cleared.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
1	ADC2	0	R	Not available	Auto Rendering Display Change Flag 2 0: Indicates that the frame buffer for plane 2 has not been switched. 1: Indicates that the frame buffer for plane 2 has been switched. The ADC2 bit state is held until it is cleared.
0	ADC1	0	R	Not available	Auto Rendering Display Change Flag 1 0: Indicates that the frame buffer for plane 1 has not been switched. 1: Indicates that the frame buffer for plane 1 has been switched. The ADC1 bit state is held until it is cleared.

- Description of the ADC bits

  For animation, although the display of moving images can be performed in auto rendering mode, at this time

  1. a check for a TRAP interrupt proceeds,
  2. a check for a VBK interrupt proceeds, and
  3. rendering starts after a VBK interrupt has occurred, so two rounds of flag checking are required before this.

  With these flag bits (ADC), an ADC interrupt is generated after a single flag check, and rendering can then start.

(4) Display Unit Status Register Clear Register (DSRCR)

Address: H'FFF8000C

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TVCL	FRCL	—	—	VBCL	CMPCL	RICL	HBCL	ADCL8	ADCL7	ADCL6	ADCL5	ADCL4	ADCL3	ADCL2	ADCL1
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	—/W	—/W	R	R	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 16	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
15	TVCL	—	—/W	Not available	TV Synchronous Signal Error Flag Clear 0: Does not change the TVR flag in the display unit status register (DSSR). 1: Clears the TVR flag in the display unit status register (DSSR).
14	FRCL	—	—/W	Not available	Flame Flag Clear 0: Does not change the FRM flag in the display unit status register (DSSR). 1: Clears the FRM flag in the display unit status register (DSSR).
13, 12	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
11	VBCL	—	—/W	Not available	Vertical Blinking Flag Clear 0: Does not change the VBK flag in the display unit status register (DSSR). 1: Clears the VBK flag in the display unit status register (DSSR).

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
10	CMPCL	—	—/W	—	Display Output Compare Interrupt Flag Clear 0: Does not change the CMPI flag in the display unit status register (DSSR). 1: Clears the CMPI flag in the display unit status register (DSSR).
9	RICL	—	—/W	Not available	Raster Interrupt flag clear 0: Does not change the RINT flag in the display unit status register (DSSR). 1: Clears the RINT flag in the display unit status register (DSSR).
8	HBCL	—	—/W	Not available	HBK flag clear 0: Does not change the HBK flag in the display unit status register (DSSR). 1: Clears the HBK flag in the display unit status register (DSSR).
7	ADCL8	—	—/W	Not available	Auto Rendering Display Change Flag Clear 8 0: Does not change the ADC flag 8 in the display unit status register (DSSR). 1: Clears the ADC flag 8 in the display unit status register (DSSR).
6	ADCL7	—	—/W	Not available	Auto Rendering Display Change Flag Clear 7 0: Does not change the ADC flag 7 in the display unit status register (DSSR). 1: Clears the ADC flag 7 in the display unit status register (DSSR).
5	ADCL6	—	—/W	Not available	Auto Rendering Display Change Flag Clear 6 0: Does not change the ADC flag 6 in the display unit status register (DSSR). 1: Clears the ADC flag 6 in the display unit status register (DSSR).
4	ADCL5	—	—/W	Not available	Auto Rendering Display Change Flag Clear 5 0: Does not change the ADC flag 5 in the display unit status register (DSSR). 1: Clears the ADC flag 5 in the display unit status register (DSSR).

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
3	ADCL4	—	—/W	Not available	Auto Rendering Display Change Flag Clear 4 0: Does not change the ADC flag 4 in the display unit status register (DSSR). 1: Clears the ADC flag 4 in the display unit status register (DSSR).
2	ADCL3	—	—/W	Not available	Auto Rendering Display Change Flag Clear 3 0: Does not change the ADC flag 3 in the display unit status register (DSSR). 1: Clears the ADC flag 3 in the display unit status register (DSSR).
1	ADCL2	—	—/W	Not available	Auto Rendering Display Change Flag Clear 2 0: Does not change the ADC flag 2 in the display unit status register (DSSR). 1: Clears the ADC flag 2 in the display unit status register (DSSR).
0	ADCL1	—	—/W	Not available	Auto Rendering Display Change Flag Clear 1 0: Does not change the ADC flag 1 in the display unit status register (DSSR). 1: Clears the ADC flag 1 in the display unit status register (DSSR).

(5) Display Unit Interrupt Enable Register (DIER)

Address: H'FFF80010

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TVE	FRE	—	—	VBE	—	RIE	HBE	ADCE8	ADCE7	ADCE6	ADCE5	ADCE4	ADCE3	ADCE2	ADCE1
Initial value:	0	0	—	—	0	—	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

The Display Unit Interrupt Enable Register (DIER) allows the generation of interrupts for the CPU by aspects of the internal state of the display unit as reflected in bits of the Display Unit Status Register (DSSR) as the sources. When a bit of this register is set to 1, setting the bit in the same position within the DSSR to 1 will lead to the generation of an interrupt for the CPU.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 16	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
15	TVE	0	R/W	Not available	TV Synchronous Signal Error Interrupt Enable 0: Disables the TVR flag interrupt of the display unit status register (DSSR). 1: Enables the TVR flag interrupt of the display unit status register (DSSR).
14	FRE	0	R/W	Not available	Flame Flag Interrupt Enable 0: Disables the FRM flag interrupt of the display unit status register (DSSR). 1: Enables the FRM flag interrupt of the display unit status register (DSSR).
13, 12	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
11	VBE	0	R/W	Not available	Vertical Blanking Flag Interrupt Enable 0: Disables the VBK flag interrupt of the display unit status register (DSSR). 1: Enables the VBK flag interrupt of the display unit status register (DSSR).
10	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
9	RIE	0	R/W	Not available	Raster Interrupt Flag Interrupt Enable 0: Disables the RINT flag interrupt of the display unit status register (DSSR). 1: Enables the RINT flag interrupt of the display unit status register (DSSR).
8	HBE	0	R/W	Not available	HBK Flag Interrupt Enable 0: Disables the HBK flag interrupt of the display unit status register (DSSR). 1: Enables the HBK flag interrupt of the display unit status register (DSSR).
7	ADCE8	0	R/W	Not available	Auto Rendering Display Change Flag 8 Interrupt Enable 0: Disables the ADC flag 8 interrupt of the display unit status register (DSSR). 1: Enables the ADC flag 8 interrupt of the display unit status register (DSSR).
6	ADCE7	0	R/W	Not available	Auto Rendering Display Change Flag 7 Interrupt Enable 0: Disables the ADC flag 7 interrupt of the display unit status register (DSSR). 1: Enables the ADC flag 7 interrupt of the display unit status register (DSSR).

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
5	ADCE6	0	R/W	Not available	Auto Rendering Display Change Flag 6 Interrupt Enable 0: Disables the ADC flag 6 interrupt of the display unit status register (DSSR). 1: Enables the ADC flag 6 interrupt of the display unit status register (DSSR).
4	ADCE5	0	R/W	Not available	Auto Rendering Display Change Flag 5 Interrupt Enable 0: Disables the ADC flag 5 interrupt of the display unit status register (DSSR). 1: Enables the ADC flag 5 interrupt of the display unit status register (DSSR).
3	ADCE4	0	R/W	Not available	Auto Rendering Display Change Flag 4 Interrupt Enable 0: Enables the ADC flag 4 interrupt of the display unit status register (DSSR). 1: Enables the ADC flag 4 interrupt of the display unit status register (DSSR).
2	ADCE3	0	R/W	Not available	Auto Rendering Display Change Flag 3 Interrupt Enable 0: Disables the ADC flag 3 interrupt of the display unit status register (DSSR). 1: Enables the ADC flag 3 interrupt of the display unit status register (DSSR).

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
1	ADCE2	0	R/W	Not available	Auto Rendering Display Change Flag 2 Interrupt Enable 0: Disables the ADC flag 2 interrupt of the display unit status register (DSSR). 1: Enables the ADC flag 2 interrupt of the display unit status register (DSSR).
0	ADCE1	0	R/W	Not available	Auto Rendering Display Change Flag 1 Interrupt Enable 0: Enables the ADC flag 1 interrupt of the display unit status register (DSSR). 1: Enables the ADC flag 1 interrupt of the display unit status register (DSSR).

The following are conditions, based on the display unit status register (DSSR) and display unit interrupt enable register (DIER), for issuing an interrupt to the CPU from the display unit (DU).

Conditions for issuing an interrupt = a + b + c + d + e + f + g + h + i + j + k+ l + m

- a = TVR • TVE
- b = FRM • FRE
- c = VBK • VBE
- d = RINT • RIE
- e = HBK • HBE
- f = ADC6 • ADCE6
- g = ADC5 • ADCE5
- h = ADC4 • ADCE4
- i = ADC3 • ADCE3
- j = ADC2 • ADCE2
- k = ADC1 • ADCE1
- l = ADC8 • ADCE8
- m = ADC7 • ADCE7

(6) Color Palette Control Register (CPCR)

Address: H'FFF80014

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	CP4CE	CP3CE	CP2CE	CP1CE
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 20	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
19	CP4CE	0	R/W	Available	Color Palette 4 Change Enable 0: The color palette 4 is not switched. 1: The color palette 4 is switched. Switching occurs when bit 9 (DRES) of the display unit system control register (DSYSR) is set to 0 from 1 or when internal updating is performed. This bit is valid only when it is set to 1. Setting this bit to 0 is invalid. This bit is cleared to 0 after the color palette 4 is switched. If setting of 1 and clearing to 0 occur at the same time, clearing to 0 is given priority.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
18	CP3CE	0	R/W	Available	Color Palette 3 Change Enable 0: The color palette 3 is not switched. 1: The color palette 3 is switched. Switching occurs when bit 9 (DRES) of the display unit system control register (DSYSR) is set to 0 from 1 or when internal updating is performed. This bit is valid only when it is set to 1. Setting this bit to 0 is invalid. This bit is cleared to 0 after the color palette 3 is switched. If setting of 1 and clearing to 0 occur at the same time, clearing to 0 is given priority.
17	CP2CE	0	R/W	Available	Color Palette 2 Change Enable 0: The color palette 2 is not switched. 1: The color palette 2 is switched. Switching occurs when bit 9 (DRES) of the display unit system control register (DSYSR) is set to 0 from 1 or when internal updating is performed. This bit is valid only when it is set to 1. Setting this bit to 0 is invalid. This bit is cleared to 0 after the color palette 2 is switched. If setting of 1 and clearing to 0 occur at the same time, clearing to 0 is given priority.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
16	CP1CE	0	R/W	Available	Color Palette 1 Change Enable 0: The color palette 1 is not switched. 1: The color palette 1 is switched. Switching occurs when bit 9 (DRES) of the display unit system control register (DSYSR) is set to 0 from 1 or when internal updating is performed. This bit is valid only when set to 1. Setting this bit to 0 is invalid. This bit is cleared to 0 after the color palette 1 is switched. If setting of 1 and clearing to 0 occur at the same time, clearing to 0 is given priority.
15 to 0	—	—	R	—	Reserved

(7) Display Plane Priority Register (DPPR)

Address: H'FFF80018

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DPE8	DPS8			DPE7	DPS7			DPE6	DPS6			DPE5	DPS5		
Initial value:	0	1	1	1	0	1	1	0	0	1	0	1	0	1	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DPE4	DPS4			DPE3	DPS3			DPE2	DPS2			DPE1	DPS1		
Initial value:	0	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- Defines the order of planes in image composition and turns the display of planes on or off
- When the setting of bit 0 in the Display Unit Output Route Control Register (DORCR) is 0, the display of planes from among planes 1 to 8 for which display is enabled is switched on.
- This register can only be used to set the priority order for superposition processor 1. To use superposition processor 2, set bit 0 in the Display Unit Output Route Control Register (DORCR) to 1 and make other settings as required in Display Superimpose 2 Priority Register.
- After setting a desired value in a register listed in section 14.3.4, Display Plane Registers, set the corresponding bit in DPPR (bit 31, 27, 23, 19, 15, 11, 7, or 3) to 1.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31	DPE8	0	R/W	Available	Display Plane Priority 8 Enable
30 to 28	DPS8	111	R/W	Available	Display Plane Priority 8 Select 1000: Assigns priority 8 to plane 1 and displays plane 1. 1001: Assigns priority 8 to plane 2 and displays plane 2. 1010: Assigns priority 8 to plane 3 and displays plane 3. 1011: Assigns priority 8 to plane 4 and displays plane 4. 1100: Assigns priority 8 to plane 5 and displays plane 5. 1101: Assigns priority 8 to plane 6 and displays plane 6. 1110: Assigns priority 8 to plane 7 and displays plane 7. 1111: Assigns priority 8 to plane 8 and displays plane 8. 0---: Priority 8 is not displayed.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
27	DPE7	0	R/W	Available	Display Plane Priority 7 Enable
26 to 24	DPS7	110	R/W	Available	Display Plane Priority 7 Select 1000: Assigns priority 7 to plane 1 and displays plane 1. 1001: Assigns priority 7 to plane 2 and displays plane 2. 1010: Assigns priority 7 to plane 3 and displays plane 3. 1011: Assigns priority 7 to plane 4 and displays plane 4. 1100: Assigns priority 7 to plane 5 and displays plane 5. 1101: Assigns priority 7 to plane 6 and displays plane 6. 1110: Assigns priority 7 to plane 7 and displays plane 7. 1111: Assigns priority 7 to plane 8 and displays plane 8. 0---: Priority 7 is not displayed.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
23	DPE6	0	R/W	Available	Display Plane Priority 6 Enable
22 to 20	DPS6	101	R/W	Available	Display Plane Priority 6 Select 1000: Assigns priority 6 to plane 1 and displays plane 1. 1001: Assigns priority 6 to plane 2 and displays plane 2. 1010: Assigns priority 6 to plane 3 and displays plane 3. 1011: Assigns priority 6 to plane 4 and displays plane 4. 1100: Assigns priority 6 to plane 5 and displays plane 5. 1101: Assigns priority 6 to plane 6 and displays plane 6. 1110: Assigns priority 6 to plane 7 and displays plane 7. 1111: Assigns priority 6 to plane 8 and displays plane 8. 0---: Priority 6 is not displayed.

Bit	Bit Name	Initial Value	R/W	Internal update Function	Description
19	DPE5	0	R/W	Available	Display Plane Priority 5 Enable
18 to 16	DPS5	100	R/W	Available	Display Plane Priority 5 Select 1000: Assigns priority 5 to plane 1 and displays plane 1. 1001: Assigns priority 5 to plane 2 and displays plane 2. 1010: Assigns priority 5 to plane 3 and displays plane 3. 1011: Assigns priority 5 to plane 4 and displays plane 4. 1100: Assigns priority 5 to plane 5 and displays plane 5. 1101: Assigns priority 5 to plane 6 and displays plane 6. 1110: Assigns priority 5 to plane 7 and displays plane 7. 1111: Assigns priority 5 to plane 8 and displays plane 8. 0---: Priority 5 is not displayed.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
15	DPE4	0	R/W	Available	Display Plane Priority 4 Enable
14 to 12	DPS4	011	R/W	Available	Display Plane Priority 4 Select 1000: Assigns priority 4 to plane 1 and displays plane 1. 1001: Assigns priority 4 to plane 2 and displays plane 2. 1010: Assigns priority 4 to plane 3 and displays plane 3. 1011: Assigns priority 4 to plane 4 and displays plane 4. 1100: Assigns priority 4 to plane 5 and displays plane 5. 1101: Assigns priority 4 to plane 6 and displays plane 6. 1110: Assigns priority 4 to plane 7 and displays plane 7. 1111: Assigns priority 4 to plane 8 and displays plane 8. 0---: Priority 4 is not displayed.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
11	DPE3	0	R/W	Available	Display Plane Priority 3 Enable
10 to 8	DPS3	010	R/W	Available	Display Plane Priority 3 Select 1000: Assigns priority 3 to plane 1 and displays plane 1. 1001: Assigns priority 3 to plane 2 and displays plane 2. 1010: Assigns priority 3 to plane 3 and displays plane 3. 1011: Assigns priority 3 to plane 4 and displays plane 4. 1100: Assigns priority 3 to plane 5 and displays plane 5. 1101: Assigns priority 3 to plane 6 and displays plane 6. 1110: Assigns priority 3 to plane 7 and displays plane 7. 1111: Assigns priority 3 to plane 8 and displays plane 8. 0---: Priority 3 is not displayed.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
7	DPE2	0	R/W	Available	Display Plane Priority 2 Enable
6 to 4	DPS2	001	R/W	Available	Display Plane Priority 2 Select 1000: Assigns priority 2 to plane 1 and displays plane 1. 1001: Assigns priority 2 to plane 2 and displays plane 2. 1010: Assigns priority 2 to plane 3 and displays plane 3. 1011: Assigns priority 2 to plane 4 and displays plane 4. 1100: Assigns priority 2 to plane 5 and displays plane 5. 1101: Assigns priority 2 to plane 6 and displays plane 6. 1110: Assigns priority 2 to plane 7 and displays plane 7. 1111: Assigns priority 2 to plane 8 and displays plane 8. 0---: Priority 2 is not displayed.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
3	DPE1	0	R/W	Available	Display Plane Priority 1 Enable
2 to 0	DPS1	000	R/W	Available	Display Plane Priority 1 Select 1000: Assigns priority 1 to plane 1 and displays plane 1. 1001: Assigns priority 1 to plane 2 and displays plane 2. 1010: Assigns priority 1 to plane 3 and displays plane 3. 1011: Assigns priority 1 to plane 4 and displays plane 4. 1100: Assigns priority 1 to plane 5 and displays plane 5. 1101: Assigns priority 1 to plane 6 and displays plane 6. 1110: Assigns priority 1 to plane 7 and displays plane 7. 1111: Assigns priority 1 to plane 8 and displays plane 8. 0---: Priority 1 is not displayed.

(8) Display Unit Extensional Function Enable Register (DEFR)

Address: H'FFF80020

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CODE															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	EXSL	EXVL	—	—	—	—	—	EXUP	VCUP	—	—	—	DEFE
Initial value:	—	—	—	0	0	—	—	—	—	—	0	0	—	—	—	0
R/W:	R	R	R	R/W	R/W	R	R	R	R	R	R/W	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 16	CODE	—	—/W	Not available	DEFR Enabling Code (register available code) For a value written to DEFR to be effective, the value must include H'7773 in these bits.
15 to 13	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
12	EXSL	0	R/W	DRES	External Sync Signal Select 0: External SYNC signals (EXVSYNC, EXHSYNC) allow signals from the pins to be directly read in terms of post-division clocks. 1: External SYNC signals (EXVSYNC, EXHSYNC) allow the signals that were read in terms of pre-division clock to be read again as post-division clocks.
11	EXVL	0	R/W	Not available	External Vsync Latch Select 0: External VSYNC signal (EXVSYNC) allows the signal from the pin to be latched every clock cycle. 1: External VSYNC signal (EXVSYNC) allows the signal from the pin to be latched at the rising edge of the external HSYNC signal.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
10 to 6	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
5	EXUP	0	R/W	DRES	External Updating Mode 0: Internally updates the internal update function bit. 1: Externally updates the internal update function bit without updating it internally. This bit takes precedence over the display unit system control register (DSYSR)/ IUPD.
4	VCUP	0	R/W	DRES	Vertical Cycle Register Update Timing Select 0: The internal updating is based on the falling VSYNC. 1: The internal updating is based on the rising VSYNC. By setting the internal updating of the vertical scanning cycle register as a VSYNC rise, any disturbance of VSYNC signals during vertical scanning cycle register switching can be prevented.
3 to 1	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
0	DEFE	0	R/W	DRES	Display Unit Extensional Function Enable 0: Disables the expanded functions 1: Enables the following expanded functions: Enables bits 31 to 29 of the display area start address register (DSAR) and display capture area start address register (DCSAR) and bit 29 of the display unit system control register (DSYSR). Enables bits 26 to 24 of the plane n mode register (PnMR). Enables bits 31 to 29 of the display area starting address. Enables bits 24 and 5 in the external synchronization control register (ESCR). Enables bit 10 in the plane n blending ratio register (PnALPHAR).

(9) Display Capture Control Register (DCPCR)

Address: H'FFF80028

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CODE															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CAB	CDF	—	—	—	DCE
Initial value:	—	—	—	—	—	—	—	—	—	—	0	0	—	—	—	0
R/W:	R	R	R/W	R/W	R	R	R	R/W	R	R	R/W	R/W	R	R	R	R/W

- After setting a desired value in a register listed in section 14.3.5, Display Capture Registers, set the corresponding bit in DCPCR (bit 8 or 0) to 1.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 16	CODE	—	—/W	Not available	DCPCR Enabling Code (register available code) For a value written to DCPCR to be effective, the value must include H'7773 in these bits.
15, 14	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
13, 12	—	—	R/W	—	Reserved The read value is undefined. The write value should always be 0.
11 to 9	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
8	—	—	R/W	—	Reserved The read value is undefined. The write value should always be 0.
7, 6	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
5	CAB	0	R/W	Available	Display Capture A Bit Function Select To enable bit 5, the DEFE2G bit in DEFR2 should be set to 1. In the initial state, bit 5 is fixed to 0. 0: When the display capture data format is ARGB1555, the A value is 0. 1: When the display capture data format is ARGB1555, the A value is 1.
4	CDF	0	R/W	Available	Display Capture Data Format To enable bit 4, the DEFE2G bit in DEFR2 should be set to 1. In the initial state, bit 4 is fixed to 0. 0: Display capture data format is RGB565. 1: Display capture data format is ARGB1555. The A value is determined by bit 5 in this register.
3 to 1	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
0	DCE	0	R/W	Available	Display Capture Enable 0: Display data is not captured. 1: Display data is captured when the DRES and DEN bits in the display unit system control register (DSYSR) are 01. After this bit is set to 1, data capture is started in the subsequent frame.

(10) Display Unit Extensional Function Enable Register 2 (DEFR2)

Address: H'FFF80034

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CODE															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DEFE2G
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 16	CODE	—	—/W	Not available	DEFR2 Enabling Code (register available code) For a value written to DEFR2 to be effective, the value must include H'7775 in these bits.
15 to 1	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
0	DEFE2G	0	R/W	DRES	Display Unit Extensional Function Enable from 2G 0: Extensional functions are disabled. 1: Extensional functions are enabled. The following extensional functions are enabled. Bits 13 and 12 in the plane n blending ratio register (PnALPHAR) Bits 5 and 4 in the display capture control register (DCPCR)

(11) Display Unit Extensional Function Enable Register 3 (DEFR3)

Address: H'FFF80038

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CODE															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	EVDM		—	—	—	—	—	—	VMSM1		—	—	—	DEFE3
Initial value:	—	—	0	0	—	—	—	—	—	—	0	0	—	—	—	0
R/W:	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 16	CODE	—	—/W	Not available	DEFR3 Enabling Code (register available code) For a value written to DEFR3 to be effective, the value must include H'7776 in these bits.
15, 14	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
13, 12	EVDM	00	R/W	Available	Extensional Video Display Mode with Planes 7 and 8 00: The extensional video display function is not used. 01: Extensional video display is performed using planes 7 and 8. The image data captured in odd/even field capture mode by the video input module is sorted into planes 7 and 8, and α blending using the blending ratio register values is performed at all times. 10: Extensional video display is performed using planes 7 and 8. The image data captured in odd/even field capture mode by the video input module is sorted into planes 7 and 8, and blending is performed based on the phase difference between the vertical sync signal of the video input and that of the display unit. 11: Extensional video display is performed using planes 7 and 8. The image data captured in odd/even field capture mode by the video input module is sorted into planes 7 and 8, and α blending using the blending ratio register values is performed only when the same field is displayed based on alternating field display.
11 to 6	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
5, 4	VMSM1	00	R/W	DRES	Video Input Master Mode 1 To enable video input master mode, set the TVM bits in the display unit system control register (DSYSR) to 00 (master mode). Otherwise, master mode is not enabled even if these bits are set. 00: The video input master mode of timing generator 1 is disabled. The TV sync mode is that specified by the TVM bits in DSYSR. 01: The video input master mode of timing generator 1 is enabled. The HSYNC, VSYNC, and CSYNC signals are output. The VSYNC signal is output in synchronization with the VSYNC signal output by the video input module. The HSYNC and ODDF signal outputs follow the register settings. 10: Setting prohibited 11: Setting prohibited
3 to 1	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
0	DEFE3	0	R/W	DRES	Display Unit Extensional Function Enable from SH-Navi3 0: Extensional functions are disabled. 1: Extensional functions are enabled. The following extensional functions are enabled. Bits 13, 12, and 8 in the display capture control register (DCPCR) Display unit video capture status register (DVCSR)

(12) Display Unit Extensional Function Enable Register 4 (DEFR4)

Address: H'FFF8003C

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CODE															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	LRUO	SPCE	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	0	0	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 16	CODE	—	—/W	Not available	DEFR4 Enabling Code (register available code) For a value written to DEFR4 to be effective, the value must include H'7777 in these bits.
15 to 6	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
5	LRUO	0	R/W	DRES	LRU Function Off 0: Requests for the SHwy from the individual planes are arbitrated by the LRU system. 1: Requests for the SHwy from the individual planes are arbitrated according to the following decreasing order of priority: (High) plane 1 > plane 2 > plane 3 > plane 4 > plane 5 > plane 6 > plane 7 > plane 8 > display capture 1 (Low) When the display is 32 bits per pixel, set this bit to 1.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
4	SPCE	0	R/W	DRES	SHwy Priority Change Enable 0: When bit 10 in the plane n display data control register (PnDDCR) is 1, the bit is fixed to 1 for a low priority level for SHwy access. 1: Bit 0 of the low priority level for SHwy access becomes bit 0 of the display SuperHyway priority register (DSHPR).
3 to 0	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

(13) Display Unit Video Capture Status Register (DVCSR)

Address: H'FFF800D0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	V1CFB		—	—	—	—	VCFB	
Initial value:	—	—	—	—	—	—	—	—	0	0	—	—	—	—	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

To enable this register, set the DEFE3 bit in the display unit extensional function enable register 3 (DEFR3) to 1. The value read from this register in its initial state is fixed to 0.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 8	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
7, 6	V1CFB	00	R	Not available	Video 1 Capture Frame Buffer Flag These bits indicate the status of the display start address in the VIN1. The status is updated with the same timing as the internal updating of timing generator 1. 00: In the plane set for video capture, the display area starts at the address indicated by PnDSA0R. 01: In the plane set for video capture, the display area starts at the address indicated by PnDSA1R. 10: In the plane set for video capture, the display area starts at the address indicated by PnDSA2R. 11: Video input module is in its initial state.
5 to 2	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
1, 0	VCFB	00	R	Not available	Video Capture Frame Buffer Flag These bits are updated with the same timing as internal updating of timing generator 1. 00: In the plane set for video capture 0, the display area starts at the address indicated by PnDSA0R. 01: In the plane set for video capture 0, the display area starts at the address indicated by PnDSA1R. 10: In the plane set for video capture 0, the display area starts at the address indicated by PnDSA2R. 11: Video input 0 module is in its initial state.

14.3.2 Display Timing Generation Registers

(1) Horizontal Display Start Register (HDSR)

Address: H'FFF80040

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	HDS								
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 9	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
8 to 0	HDS	—	R/W	Available	Horizontal Display Start These bits are used to set the horizontal display start position in dot clock units. The set value is retained at a reset.

(2) Horizontal Display End Register (HDER)

Address: H'FFF80044

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	HDE										
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 11	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
10 to 0	HDE	—	R/W	Available	Horizontal Display End These bits are used to set the horizontal display end position in dot clock units. The set value is retained at a reset.

(3) Vertical Display Start Register (VDSR)

Address: H'FFF80048

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	VDS								
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 9	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
8 to 0	VDS	—	R/W	Available	Vertical Display Start These bits are used to set the vertical display start position in raster line units. The set value is retained at a reset.

(4) Vertical Display End Register (VDER)

Address: H'FFF8004C

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	VDE									
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 10	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
9 to 0	VDE	—	R/W	Available	Vertical Display End These bits are used to set the vertical display end position in raster line units. The set value is retained at a reset.

(5) Horizontal Cycle Register (HCR)

Address: H'FFF80050

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	HC										
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 11	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
10 to 0	HC	—	R/W	Available	Horizontal Cycle These bits are used to set one horizontal scan cycle including the horizontal blanking period in dot clock units. In TV synchronous mode, set this register so that the HSYNC cycle set with this register is equal to or greater than the EXHSYNC cycle. The set value is retained at a reset.

(6) Horizontal Sync Width Register (HSWR)

Address: H'FFF80054

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	HSW								
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 9	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
8 to 0	HSW	—	R/W	Available	Horizontal Sync Width These bits are used to set the low-level pulse width of the horizontal synchronous signal in dot clock units. The set value is retained at a reset.

(7) Vertical Cycle Register (VCR)

Address: H'FFF80058

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	VC									
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 10	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
9 to 0	VC	—	R/W	Available	Vertical Cycle These bits are used to set the vertical scan cycle including the vertical blanking period in raster line units. In TV synchronous mode, set the time limit of the EXVSYNC rising edge detection. If the EXVSYNC rising edge is not detected within the time limit, the result is reflected in the TVR flag. The set value is retained at a reset.

(8) Vertical Sync Point Register (VSPR)

Address: H'FFF8005C

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	VSP									
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 10	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
9 to 0	VSP	—	R/W	Available	Vertical Sync Point These bits are used to set the vertical synchronous signal start position in raster line units. In TV synchronous mode, set this register so that the VSYNC falling edge setting position set with this register is the same as or comes after that of the EXVSYNC falling edge. The set value is retained at a reset.

(9) Equal Pulse Width Register (EQWR)

Address: H'FFF80060

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	EQW						
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 7	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
6 to 0	EQW	—	R/W	Available	Equal Pulse Width These bits are used to set the low-level equalizing pulse width of the CSYNC signal in dot clock units. To validate this setting, set bit 7 in the display unit mode register (DSMR)/CSYNC mode (CSY) to 1. The set value is retained at a reset.

(10) Serration Width Register (SPWR)

Address: H'FFF80064

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SPW									
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 10	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
9 to 0	SPW	—	R/W	Available	Serration Width These bits are used to set the low-level serration pulse width of the CSYNC signal in dot clock units. Set a value smaller than 1/2 of HC. To validate this setting, set bit 7 in the display unit mode register (DSMR)/CSYNC mode (CSY) to 1. The set value is retained at a reset.

(11) CLAMP Signal Start Register (CLAMPSR)

Address: H'FFF80070

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	CLAMPS										
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 11	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
10 to 0	CLAMPS	—	R/W	Available	CLAMP Signal Start These bits are used to set the CLAMP signal rising edge position in dot clock units, using as the reference the HSYNC signal falling edge. The CLAMP signal rises (setting value + 1) cycle after the HSYNC signal falls. Therefore, the CLAMP signal cannot rise in the same cycle as the HSYNC signal falling edge. The set value is retained at a reset.

(12) CLAMP Signal Width Register (CLAMPWR)

Address: H'FFF80074

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	CLAMPW										
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 11	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
10 to 0	CLAMPW	—	R/W	Available	CLAMP Signal Width These bits are used to set the high-level width of the CLAMP signal in dot clock units. When the CLAMP signal is high level, if the HSYNC signal falls, the CLAMP signal falls. The set value is retained at a reset.

(13) DE Signal Start Register (DESR)

Address: H'FFF80078

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	DES										
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 11	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
10 to 0	DES	—	R/W	Available	DE Signal Start These bits are used to set the DE signal rising edge position in dot clock units, using as the reference the HSYNC signal falling edge. The DE signal rises (setting value + 1) cycle after the HSYNC signal falls. Therefore, the DE signal cannot rise in the same cycle as the HSYNC signal falling edge. During a vertical blanking period, the DE signal is fixed to the low level. The set value is retained at a reset.

(14) DE Signal Width Register (DEWR)

Address: H'FFF8007C

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	DEW										
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 11	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
10 to 0	DEW	—	R/W	Available	DE Signal Width These bits are used to set the high-level width of the DE signal in dot clock units. When the DE signal is high level, if the HSYNC signal falls, the DE signal falls. The set value is retained at a reset.

14.3.3 Display Attribute Registers

(1) Color Palette 1 Transparent Color Register (CP1TR)

Address: H'FFF80080

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CP1IF	CP1IE	CP1ID	CP1IC	CP1IB	CP1IA	CP1I9	CP1I8	CP1I7	CP1I6	CP1I5	CP1I4	CP1I3	CP1I2	CP1I1	CP1I0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 16	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
15	CP1IF	0	R/W	Available	Color Palette 1 Index F This bit is used to specify the color palette 1 transparent color. 0: Does not set the color of the color palette 1 index F to the transparent color. 1: Sets the color of the color palette 1 index F to the transparent color.
14	CP1IE	0	R/W	Available	Color Palette 1 Index E This bit is used to specify the color palette 1 transparent color. 0: Does not set the color of the color palette 1 index E to the transparent color. 1: Sets the color of the color palette 1 index E to the transparent color.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
13	CP1ID	0	R/W	Available	Color Palette 1 Index D This bit is used to specify the color palette 1 transparent color. 0: Does not set the color of the color palette 1 index D to the transparent color. 1: Sets the color of the color palette 1 index D to the transparent color.
12	CP1IC	0	R/W	Available	Color Palette 1 Index C This bit is used to specify the color palette 1 transparent color. 0: Does not set the color of the color palette 1 index C to the transparent color. 1: Sets the color of the color palette 1 index C to the transparent color.
11	CP1IB	0	R/W	Available	Color Palette 1 Index B This bit is used to specify the color palette 1 transparent color. 0: Does not set the color of the color palette 1 index B to the transparent color. 1: Sets the color of the color palette 1 index B to the transparent color.
10	CP1IA	0	R/W	Available	Color Palette 1 Index A This bit is used to specify the color palette 1 transparent color. 0: Does not set the color of the color palette 1 index A to the transparent color. 1: Sets the color of the color palette 1 index A to the transparent color.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
9	CP1I9	0	R/W	Available	Color Palette 1 Index 9 This bit is used to specify the color palette 1 transparent color. 0: Does not set the color of the color palette 1 index 9 to the transparent color. 1: Sets the color of the color palette 1 index 9 to the transparent color.
8	CP1I8	0	R/W	Available	Color Palette 1 Index 8 This bit is used to specify the color palette 1 transparent color. 0: Does not set the color of the color palette 1 index 8 to the transparent color. 1: Sets the color of the color palette 1 index 8 to the transparent color.
7	CP1I7	0	R/W	Available	Color Palette 1 Index 7 This bit is used to specify the color palette 1 transparent color. 0: Does not set the color of the color palette 1 index 7 to the transparent color. 1: Sets the color of the color palette 1 index 7 to the transparent color.
6	CP1I6	0	R/W	Available	Color Palette 1 Index 6 This bit is used to specify the color palette 1 transparent color. 0: Does not set the color of the color palette 1 index 6 to the transparent color. 1: Sets the color of the color palette 1 index 6 to the transparent color.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
5	CP1I5	0	R/W	Available	Color Palette 1 Index 5 This bit is used to specify the color palette 1 transparent color. 0: Does not set the color of the color palette 1 index 5 to the transparent color. 1: Sets the color of the color palette 1 index 5 to the transparent color.
4	CP1I4	0	R/W	Available	Color Palette 1 Index 4 This bit is used to specify the color palette 1 transparent color. 0: Does not set the color of the color palette 1 index 4 to the transparent color. 1: Sets the color of the color palette 1 index 4 to the transparent color.
3	CP1I3	0	R/W	Available	Color Palette 1 Index 3 This bit is used to specify the color palette 1 transparent color. 0: Does not set the color of the color palette 1 index 3 to the transparent color. 1: Sets the color of the color palette 1 index 3 to the transparent color.
2	CP1I2	0	R/W	Available	Color Palette 1 Index 2 This bit is used to specify the color palette 1 transparent color. 0: Does not set the color of the color palette 1 index 2 to the transparent color. 1: Sets the color of the color palette 1 index 2 to the transparent color.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
1	CP1I1	0	R/W	Available	Color Palette 1 Index F This bit is used to specify the color palette 1 transparent color. 0: Does not set the color of the color palette 1 index 1 to the transparent color. 1: Sets the color of the color palette 1 index 1 to the transparent color.
0	CP1I0	0	R/W	Available	Color Palette 1 Index 0 This bit is used to specify the color palette 1 transparent color. 0: Does not set the color of the color palette 1 index 0 to the transparent color. 1: Sets the color of the color palette 1 index 0 to the transparent color.

(2) Color Palette 2 Transparent Color Register (CP2TR)

Address: H'FFF80084

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CP2IF	CP2IE	CP2ID	CP2IC	CP2IB	CP2IA	CP2I9	CP2I8	CP2I7	CP2I6	CP2I5	CP2I4	CP2I3	CP2I2	CP2I1	CP2I0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 16	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
15	CP2IF	0	R/W	Available	Color Palette 2 Index F This bit is used to specify the color palette 2 transparent color. 0: Does not set the color of the color palette 2 index F to the transparent color. 1: Sets the color of the color palette 2 index F to the transparent color.
14	CP2IE	0	R/W	Available	Color Palette 2 Index E This bit is used to specify the color palette 2 transparent color. 0: Does not set the color of the color palette 2 index E to the transparent color. 1: Sets the color of the color palette 2 index E to the transparent color.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
13	CP2ID	0	R/W	Available	Color Palette 2 Index D This bit is used to specify the color palette 2 transparent color. 0: Does not set the color of the color palette 2 index D to the transparent color. 1: Sets the color of the color palette 2 index D to the transparent color.
12	CP2IC	0	R/W	Available	Color Palette 2 Index C This bit is used to specify the color palette 2 transparent color. 0: Does not set the color of the color palette 2 index C to the transparent color. 1: Sets the color of the color palette 2 index C to the transparent color.
11	CP2IB	0	R/W	Available	Color Palette 2 Index B This bit is used to specify the color palette 2 transparent color. 0: Does not set the color of the color palette 2 index B to the transparent color. 1: Sets the color of the color palette 2 index B to the transparent color.
10	CP2IA	0	R/W	Available	Color Palette 2 Index A This bit is used to specify the color palette 2 transparent color. 0: Does not set the color of the color palette 2 index A to the transparent color. 1: Sets the color of the color palette 2 index A to the transparent color.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
9	CP2I9	0	R/W	Available	Color Palette 2 Index 9 This bit is used to specify the color palette 2 transparent color. 0: Does not set the color of the color palette 2 index 9 to the transparent color. 1: Sets the color of the color palette 2 index 9 to the transparent color.
8	CP2I8	0	R/W	Available	Color Palette 2 Index 8 This bit is used to specify the color palette 2 transparent color. 0: Does not set the color of the color palette 2 index 8 to the transparent color. 1: Sets the color of the color palette 2 index 8 to the transparent color.
7	CP2I7	0	R/W	Available	Color Palette 2 Index 7 This bit is used to specify the color palette 2 transparent color. 0: Does not set the color of the color palette 2 index 7 to the transparent color. 1: Sets the color of the color palette 2 index 7 to the transparent color.
6	CP2I6	0	R/W	Available	Color Palette 2 Index 6 This bit is used to specify the color palette 2 transparent color. 0: Does not set the color of the color palette 2 index 6 to the transparent color. 1: Sets the color of the color palette 2 index 6 to the transparent color.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
5	CP2I5	0	R/W	Available	Color Palette 2 Index 5 This bit is used to specify the color palette 2 transparent color. 0: Does not set the color of the color palette 2 index 5 to the transparent color. 1: Sets the color of the color palette 2 index 5 to the transparent color.
4	CP2I4	0	R/W	Available	Color Palette 2 Index 4 This bit is used to specify the color palette 2 transparent color. 0: Does not set the color of the color palette 2 index 4 to the transparent color. 1: Sets the color of the color palette 2 index 4 to the transparent color.
3	CP2I3	0	R/W	Available	Color Palette 2 Index 3 This bit is used to specify the color palette 2 transparent color. 0: Does not set the color of the color palette 2 index 3 to the transparent color. 1: Sets the color of the color palette 2 index 3 to the transparent color.
2	CP2I2	0	R/W	Available	Color Palette 2 Index 2 This bit is used to specify the color palette 2 transparent color. 0: Does not set the color of the color palette 2 index 2 to the transparent color. 1: Sets the color of the color palette 2 index 2 to the transparent color.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
1	CP2I1	0	R/W	Available	Color Palette 2 Index 1 This bit is used to specify the color palette 2 transparent color. 0: Does not set the color of the color palette 2 index 1 to the transparent color. 1: Sets the color of the color palette 2 index 1 to the transparent color.
0	CP2I0	0	R/W	Available	Color Palette 2 Index 0 This bit is used to specify the color palette 2 transparent color. 0: Does not set the color of the color palette 2 index 0 to the transparent color. 1: Sets the color of the color palette 2 index 0 to the transparent color.

(3) Color Palette 3 Transparent Color Register (CP3TR)

Address: H'FFF80088

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CP3IF	CP3IE	CP3ID	CP3IC	CP3IB	CP3IA	CP3I9	CP3I8	CP3I7	CP3I6	CP3I5	CP3I4	CP3I3	CP3I2	CP3I1	CP3I0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 16	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
15	CP3IF	0	R/W	Available	Color Palette 3 Index F This bit is used to specify the color palette 3 transparent color. 0: Does not set the color of the color palette 3 index F to the transparent color. 1: Sets the color of the color palette 3 index F to the transparent color.
14	CP3IE	0	R/W	Available	Color Palette 3 Index E This bit is used to specify the color palette 3 transparent color. 0: Does not set the color of the color palette 3 index E to the transparent color. 1: Sets the color of the color palette 3 index E to the transparent color.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
13	CP3ID	0	R/W	Available	Color Palette 3 Index D This bit is used to specify the color palette 3 transparent color. 0: Does not set the color of the color palette 3 index D to the transparent color. 1: Sets the color of the color palette 3 index D to the transparent color.
12	CP3IC	0	R/W	Available	Color Palette 3 Index C This bit is used to specify the color palette 3 transparent color. 0: Does not set the color of the color palette 3 index C to the transparent color. 1: Sets the color of the color palette 3 index C to the transparent color.
11	CP3IB	0	R/W	Available	Color Palette 3 Index B This bit is used to specify the color palette 3 transparent color. 0: Does not set the color of the color palette 3 index B to the transparent color. 1: Sets the color of the color palette 3 index B to the transparent color.
10	CP3IA	0	R/W	Available	Color Palette 3 Index A This bit is used to specify the color palette 3 transparent color. 0: Does not set the color of the color palette 3 index A to the transparent color. 1: Sets the color of the color palette 3 index A to the transparent color.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
9	CP3I9	0	R/W	Available	Color Palette 3 Index 9 This bit is used to specify the color palette 3 transparent color. 0: Does not set the color of the color palette 3 index 9 to the transparent color. 1: Sets the color of the color palette 3 index 9 to the transparent color.
8	CP3I8	0	R/W	Available	Color Palette 3 Index 8 This bit is used to specify the color palette 3 transparent color. 0: Does not set the color of the color palette 3 index 8 to the transparent color. 1: Sets the color of the color palette 3 index 8 to the transparent color.
7	CP3I7	0	R/W	Available	Color Palette 3 Index 7 This bit is used to specify the color palette 3 transparent color. 0: Does not set the color of the color palette 3 index 7 to the transparent color. 1: Sets the color of the color palette 3 index 7 to the transparent color.
6	CP3I6	0	R/W	Available	Color Palette 3 Index 6 This bit is used to specify the color palette 3 transparent color. 0: Does not set the color of the color palette 3 index 6 to the transparent color. 1: Sets the color of the color palette 3 index 6 to the transparent color.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
5	CP3I5	0	R/W	Available	Color Palette 3 Index 5 This bit is used to specify the color palette 3 transparent color. 0: Does not set the color of the color palette 3 index 5 to the transparent color. 1: Sets the color of the color palette 3 index 5 to the transparent color.
4	CP3I4	0	R/W	Available	Color Palette 3 Index 4 This bit is used to specify the color palette 3 transparent color. 0: Does not set the color of the color palette 3 index 4 to the transparent color. 1: Sets the color of the color palette 3 index 4 to the transparent color.
3	CP3I3	0	R/W	Available	Color Palette 3 Index 3 This bit is used to specify the color palette 3 transparent color. 0: Does not set the color of the color palette 3 index 3 to the transparent color. 1: Sets the color of the color palette 3 index 3 to the transparent color.
2	CP3I2	0	R/W	Available	Color Palette 3 Index 2 This bit is used to specify the color palette 3 transparent color. 0: Does not set the color of the color palette 3 index 2 to the transparent color. 1: Sets the color of the color palette 3 index 2 to the transparent color.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
1	CP3I1	0	R/W	Available	Color Palette 3 Index 1 This bit is used to specify the color palette 3 transparent color. 0: Does not set the color of the color palette 3 index 1 to the transparent color. 1: Sets the color of the color palette 3 index 1 to the transparent color.
0	CP3I0	0	R/W	Available	Color Palette 3 Index 0 This bit is used to specify the color palette 3 transparent color. 0: Does not set the color of the color palette 3 index 0 to the transparent color. 1: Sets the color of the color palette 3 index 0 to the transparent color.

(4) Color Palette 4 Transparent Color Register (CP4TR)

Address: H'FFF8008C

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CP4IF	CP4IE	CP4ID	CP4IC	CP4IB	CP4IA	CP4I9	CP4I8	CP4I7	CP4I6	CP4I5	CP4I4	CP4I3	CP4I2	CP4I1	CP4I0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 16	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
15	CP4IF	0	R/W	Available	Color Palette 4 Index F This bit is used to specify the color palette 4 transparent color. 0: Does not set the color of the color palette 4 index F to the transparent color. 1: Sets the color of the color palette 4 index F to the transparent color.
14	CP4IE	0	R/W	Available	Color Palette 4 Index E This bit is used to specify the color palette 4 transparent color. 0: Does not set the color of the color palette 4 index E to the transparent color. 1: Sets the color of the color palette 4 index E to the transparent color.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
13	CP4ID	0	R/W	Available	Color Palette 4 Index D This bit is used to specify the color palette 4 transparent color. 0: Does not set the color of the color palette 4 index D to the transparent color. 1: Sets the color of the color palette 4 index D to the transparent color.
12	CP4IC	0	R/W	Available	Color Palette 4 Index C This bit is used to specify the color palette 4 transparent color. 0: Does not set the color of the color palette 4 index C to the transparent color. 1: Sets the color of the color palette 4 index C to the transparent color.
11	CP4IB	0	R/W	Available	Color Palette 4 Index B This bit is used to specify the color palette 4 transparent color. 0: Does not set the color of the color palette 4 index B to the transparent color. 1: Sets the color of the color palette 4 index B to the transparent color.
10	CP4IA	0	R/W	Available	Color Palette 4 Index A This bit is used to specify the color palette 4 transparent color. 0: Does not set the color of the color palette 4 index A to the transparent color. 1: Sets the color of the color palette 4 index A to the transparent color.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
9	CP4I9	0	R/W	Available	Color Palette 4 Index 9 This bit is used to specify the color palette 4 transparent color. 0: Does not set the color of the color palette 4 index 9 to the transparent color. 1: Sets the color of the color palette 4 index 9 to the transparent color.
8	CP4I8	0	R/W	Available	Color Palette 4 Index 8 This bit is used to specify the color palette 4 transparent color. 0: Does not set the color of the color palette 4 index 8 to the transparent color. 1: Sets the color of the color palette 4 index 8 to the transparent color.
7	CP4I7	0	R/W	Available	Color Palette 4 Index 7 This bit is used to specify the color palette 4 transparent color. 0: Does not set the color of the color palette 4 index 7 to the transparent color. 1: Sets the color of the color palette 4 index 7 to the transparent color.
6	CP4I6	0	R/W	Available	Color Palette 4 Index 6 This bit is used to specify the color palette 4 transparent color. 0: Does not set the color of the color palette 4 index 6 to the transparent color. 1: Sets the color of the color palette 4 index 6 to the transparent color.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
5	CP4I5	0	R/W	Available	Color Palette 4 Index 5 This bit is used to specify the color palette 4 transparent color. 0: Does not set the color of the color palette 4 index 5 to the transparent color. 1: Sets the color of the color palette 4 index 5 to the transparent color.
4	CP4I4	0	R/W	Available	Color Palette 4 Index 4 This bit is used to specify the color palette 4 transparent color. 0: Does not set the color of the color palette 4 index 4 to the transparent color. 1: Sets the color of the color palette 4 index 4 to the transparent color.
3	CP4I3	0	R/W	Available	Color Palette 4 Index 3 This bit is used to specify the color palette 4 transparent color. 0: Does not set the color of the color palette 4 index 3 to the transparent color. 1: Sets the color of the color palette 4 index 3 to the transparent color.
2	CP4I2	0	R/W	Available	Color Palette 4 Index 2 This bit is used to specify the color palette 4 transparent color. 0: Does not set the color of the color palette 4 index 2 to the transparent color. 1: Sets the color of the color palette 4 index 2 to the transparent color.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
1	CP4I1	0	R/W	Available	Color Palette 4 Index 1 This bit is used to specify the color palette 4 transparent color. 0: Does not set the color of the color palette 4 index 1 to the transparent color. 1: Sets the color of the color palette 4 index 1 to the transparent color.
0	CP4I0	0	R/W	Available	Color Palette 4 Index 0 This bit is used to specify the color palette 4 transparent color. 0: Does not set the color of the color palette 4 index 0 to the transparent color. 1: Sets the color of the color palette 4 index 0 to the transparent color.

(5) Display Off Mode Output Register (DOOR)

Address: H'FFF80090

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	DOR						—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DOG						—	—	DOB						—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 24	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
23 to 18	DOR	—	R/W	Available	Display Off Mode Output Red These bits are used to set the red display data to be output when the display is off (DRES and DEN bits in the display unit system control register (DSYSR) are 00). The set value is retained at a reset.
17, 16	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
15 to 10	DOG	—	R/W	Available	Display Off Mode Output Green These bits are used to set the green display data to be output when the display is off (DRES and DEN bits in the display unit system control register (DSYSR) are 00). The set value is retained at a reset.
9, 8	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
7 to 2	DOB	—	R/W	Available	Display Off Mode Output Blue These bits are used to set the blue display data to be output when the display is off (DRES and DEN bits in the display unit system control register (DSYSR) are 00). The set value is retained at a reset.
1, 0	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

(6) Color Detection Register (CDER)

Address: H'FFF80094

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	CDR						—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CDG						—	—	CDB						—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 24	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
23 to 18	CDR	—	R/W	Available	Color Detection Red These bits are used to set the red data for color detection. The set value is retained at a reset.
17, 16	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
15 to 10	CDG	—	R/W	Available	Color Detection Green These bits are used to set the green data for color detection. The set value is retained at a reset.
9, 8	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
7 to 2	CDB	—	R/W	Available	Color Detection Blue These bits are used to set the blue data for color detection. The set value is retained at a reset.
1, 0	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

When output data matches the setting value of this register, the high level is output from the CDE pin. For details about the output color data format, see section 14.4.6, Data Formats for Output and Display Capture.

(7) Background Plane Output Register (BPOR)

Address: H'FFF80098

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	BPOR						—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BPOG						—	—	BPOB						—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 24	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
23 to 18	BPOR	—	R/W	Available	Background Plane Output Red These bits are used to set the red color to be displayed if no plane to be displayed exists due to a display size, transparent color, and so on. The set value is retained at a reset.
17, 16	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
15 to 10	BPOG	—	R/W	Available	Background Plane Output Green These bits are used to set the green color to be displayed if no plane to be displayed exists due to a display size, transparent color, and so on. The set value is retained at a reset.
9, 8	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
7 to 2	BPOB	—	R/W	Available	Background Plane Output Blue These bits are used to set the blue color to be displayed if no plane to be displayed exists due to a display size, transparent color, and so on. The set value is retained at a reset.
1, 0	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

(8) Raster Interrupt Offset Register (RINTOFSR)

Address: H'FFF8009C

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RINTOFS									
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 10	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
9 to 0	RINTOFS	—	R/W	Available	Raster Interrupt Offset These bits are used to set the raster offset value (number of Hs) that is based on the number of rasters set with the vertical display start register (VDSR). If the offset value is assumed to be n, RINT in the display unit status register (DSSR) is set to 1 at the HSYNC falling edge after the horizontal display period of (VDS + n-th raster). The set value is retained at a reset.

(9) Display SuperHyway Priority Register (DSHPR)

Address: H'FFF800C8

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CODE															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	PRIH				PRIL			
Initial value:	—	—	—	—	—	—	—	—	1	0	1	0	1	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 16	CODE	—	—/W	Not available	DSHPR Enabling Code (register available code) For a value written to DSHPR to be effective, the value must include H'7776 in these bits.
15 to 8	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
7 to 4	PRIH	1010	R/W	DRES	Priority High These bits specify a high-priority value. In normal usage, do not change these bits from its initial value. Otherwise, correct operation cannot be guaranteed.
3 to 0	PRIL	1000	R/W	DRES	Priority Low These bits specify a low-priority value. In normal usage, do not change these bits from its initial value. Only set these bits to H'9 when display is to have 32 bits/pixel. Correct operation cannot be guaranteed if these bits are set to H'9 when the display is not to have 32 bits/pixel, or when these bits are set to a value other than its initial value or H'9.

14.3.4 Display Plane Registers

In descriptions of registers that are common to planes 1 to 8, the planes are generically referred to as plane n. The meanings of characters n and # are given below.

n: 1, 2, 3, 4, 5, 6, 7, or 8
#: Replaces n (in hexadecimal) in addresses. For example, address H'FFF80#00 for the plane 3 mode register corresponds to H'FFF80300.

(1) Plane n Mode Register (PnMR)

Address: H'FFF80#00

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	PnVISL			—	—	—	—	—	PnYCDF	—	—	PnTC	PnWAE
Initial value:	—	—	—	0	0	0	—	—	—	—	—	0	—	—	0	0
R/W:	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PnSPIM			—	PnCPSL			PnDC	—	PnBM		—	—	PnDDF	
Initial value:	—	0	0	0	—	0	0	0	0	—	0	0	—	—	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 29	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
28 to 26	PnVISL	000	R/W	Available	Plane n Video Input Select To enable bits 28 to 26, the DEFE bit in DEFR should be set to 1. In the initial state, these bits are fixed to 000. 000: Video input 0 (VIN0) is selected. 001: Setting prohibited 010: Setting prohibited 011: Setting prohibited 100: Setting prohibited 101: Setting prohibited 110: Setting prohibited 111: Setting prohibited
25 to 21	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
20	PnYCDF	0	R/W	Available	Plane n YC Data Format 0: Sets the alignment of YC data to UYVY format. 1: Sets the alignment of YC data to YUYV format.
19, 18	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
17	PnTC	0	R/W	Avail-able	Plane n Transparent Color 0: When an 8-bit/pixel display has been selected, the transparent color is that indicated by the value of the plane n transparent color 1 register (PnTC1R) 1: When an 8-bit/pixel display has been selected, the transparent color is that indicated by the values of color palette 1 to 4 transparent color registers (CPT1R to CPT4R) The setting of this bit is invalid when an 8-bit/pixel display has not been selected. For details, refer to section 14.4.9 (2), Transparent Colors.
16	PnWAE	0	R/W	Avail-able	Plane n Wrap Around Enable 0: Wrapping-around for plane n is disabled. 1: Wrapping-around for plane n is enabled.
15	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
14 to 12	PnSPIM	000	R/W	Avail-able	Plane n Super Impose Mode 000: Transparent color processing is performed for plane n. When plane n is in the transparent color, the lower plane is displayed. 001: Blending of plane n and the lower plane is performed. When plane n is the transparent color blending is not performed, and the lower plane is displayed. 010: An EOR operation is performed on plane n and the lower plane. When plane n is the transparent color the EOR operation is not performed, and the lower plane is displayed. 011: Setting prohibited 100: Transparent color processing is not performed for plane n. Plane n is displayed. 101: Blending of plane n and the lower plane is performed. The transparent color specification for plane n is ignored, and blending is performed between all the pixels of plane n and the lower plane. 110: An EOR operation is performed on plane n and the lower plane. The transparent color specification for plane n is ignored, and EOR operation is performed on all the pixels of plane n and the lower plane. 111: Setting prohibited Transparent color processing for YC data or 32-bit/pixel data is not possible.
11	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
10 to 8	PnCPSL	000	R/W	Avail-able	Plane n Color Palette Select These bits indicate whether the color palette is to be used when the value of the PnDDF bits is 00 (i.e. the plane n display data format is 8 bits/pixel). 000: Use color palette 1 001: Use color palette 2 010: Use color palette 3 011: Use color palette 4 100: Setting prohibited 101: Setting prohibited 110: Setting prohibited 111: Setting prohibited
7	PnDC	0	R/W	Avail-able	Plane n Display Area Change 0: In manual display change mode, switching of the frame buffer is not performed. 1: In manual display change mode, switching of the frame buffer is performed. When the PnDC bit is 0, this bit can be set to 1. Switching is performed in frame units. After frame buffer switching (after vertical blanking detection), this bit is cleared to 0.
6	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
5, 4	PnBM	00	R/W	Avail-able	Plane n Buffer Mode 00: Manual display change mode 01: Auto rendering mode 10: Auto display change mode (blinking mode) 11: Video capture mode In manual display change mode, auto rendering mode, or auto display change mode (blinking mode), double-buffering control is performed using addresses 0 and 1, respectively indicated by the PnDSA0 and PnDSA1 bits in PnDSA0R and PnDSA1R. In video capture mode, triple-buffering control is performed using addresses 0 to 2 indicated by the VCFB bits in DSSR.
3, 2	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
1, 0	PnDDF	00	R/W	Avail-able	Plane n Display Data Format 00: 8 bits/pixel 01: 16 bits/pixel 10: ARGB (ARGB1555) 11: YC (4:2:2 YUV is converted to 8:8:8 RGB) Set these bits to 01 for 32-bit/pixel data.

(2) Plane n Memory Width Register (PnMWR)

Address: H'FFF80#04

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	PnMWX									—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 13	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
12 to 4	PnMWX	—	R/W	Avail-able	Plane n Memory Width X The plane n memory width should be set in the range 16 pixels to 4096 pixels, in 16-pixel units. The value is retained during a reset.
3 to 0	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

(3) Plane n Blending Ratio Register (PnALPHAR)

Address: H'FFF80#08

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PnABIT		—	PnBRSL			PnALPHA							
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 14	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
13, 12	PnABIT	—	R/W	Avail-able	Plane n A Bit Function Select This function is not available for 32-bit/pixel data. To enable these bits, set the DEFE2G bit in DEFR2 to 1. In the initial state, these bits are fixed to 0. 00: In ARGB mode (i.e. when the value of the PnDDF bits in PnMR is 10), blending is performed when the value of A is 1. 01: In ARGB mode (i.e. when the value of the PnDDF bits in PnMR is 10), blending is performed when the value of A is 0. 1-: In ARGB mode (i.e. when the value of the PnDDF bits in PnMR is 10), blending is performed regardless of the value of A. The value is retained during a reset.
11	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
10 to 8	PnBRSL	—	R/W	Avail-able	Plane n Blending Ratio Select This bit is valid when the value of the PnSPIM bits in PnMR allows blending. This function is not available for 32-bit/pixel data. To enable bit 10, set the DEFE bit in DEFR to 1. In the initial state, bit 10 is fixed to 0. -00: The value of bits 7 to 0 in this register is taken to be the blending ratio. -01: Setting prohibited -10: Bits 31 to 24 of the color palette register specified by the PnCPSL bits in PnMR are taken to be the blending ratio. Note: This setting is only effective when the display data format specified by the PnDDF bits in PnMR is 8 bits/pixel. For formats other than 8 bits/pixel, the value of bits 7 to 0 in this register is taken to be the blending ratio. 011: The display data for the plane specified by bits 2 to 0 in this register is taken to be the blending ratio. Bits 2 to 0 = 000: Display data for plane 1 is the blending ratio Bits 2 to 0 = 001: Display data for plane 2 is the blending ratio Bits 2 to 0 = 010: Display data for plane 3 is the blending ratio Bits 2 to 0 = 011: Display data for plane 4 is the blending ratio Bits 2 to 0 = 100: Display data for plane 5 is the blending ratio Bits 2 to 0 = 101: Display data for plane 6 is the blending ratio Bits 2 to 0 = 110: Display data for plane 7 is the blending ratio Bits 2 to 0 = 111: Display data for plane 8 is the blending ratio

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
10 to 8	PnBRSL	—	R/W	Avail-able	Notes: 1. When the register's own plane is specified, the value of bits 7 to 0 in this register is taken to be the blending ratio. 2. The specified plane should satisfy the following conditions. If the conditions are not satisfied, the blending ratio is undefined. - The display should be turned on by using DPPR. - The display data format should be set to 8 bits/pixel. - The display size should be greater than or equal to the size of the plane for this register. - Ensure that display positions (X and Y) are the same as those in the plane for this register. 111: Setting prohibited The value is retained during a reset.
7 to 0	PnALPHA	—	R/W	Avail-able	Plane n Blending Ratio These bits indicate the alpha value (α), which determines the blending ratio for plane n. This function is not available for 32-bit/pixel data. Blending result $\approx$ (plane n $\times$ α/255) + lower plane $\times$ (1 $-$ α/255) (Approximation) Note: Blending result, α, plane n, and lower plane in the above formula are all 8-bit data. The value is retained during a reset.

(4) Plane n Display Size X Register (PnDSXR)

Address: H'FFF80#10

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	PnDSX										
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 11	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
10 to 0	PnDSX	—	R/W	Avail-able	Plane n Display Size X The horizontal-direction display size of plane n should be set in dot clock units. Note: When the YC mode has been selected by the PnDDF bits in PnMR, this value should be set to an even number. The value is retained during a reset.

(5) Plane n Display Size Y Register (PnDSYR)

Address: H'FFF80#14

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PnDSY									
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 10	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
9 to 0	PnDSY	—	R/W	Avail-able	Plane n Display Size Y The vertical-direction display size of plane n should be set in raster line units. The value is retained during a reset.

(6) Plane n Display Position X Register (PnDPXR)

Address: H'FFF80#18

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	PnDPX										
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 11	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
10 to 0	PnDPX	—	R/W	Avail-able	Plane n Display Position X The horizontal start position on the display monitor of plane n should be set in dot clock units, taking as the origin the upper-left corner of the display monitor. The value is retained during a reset.

(7) Plane n Display Position Y Register (PnDPYR)

Address: H'FFF80#1C

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PnDPY									
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 10	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
9 to 0	PnDPY	—	R/W	Avail-able	Plane n Display Position Y The vertical start position on the display monitor of plane n should be set in raster line units, taking as the origin the upper-left corner of the display monitor. The value is retained during a reset.

(8) Plane n Display Area Start Address 0 Register (PnDSA0R)

Address: H'FFF80#20

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PnDSA0															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PnDSA0												—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 4	PnDSA0	—	R/W	Avail-able	Plane n Display Area Start Address 0 To enable bits 31 to 29, the DEFE bit in DEFR must be set to 1. In the initial state, these bits are not enabled. When the buffer mode for plane n is manual display, auto rendering, auto display change, or video capture, the area indicated by this register is used as frame buffer 0. Note: When bits 31 to 29 are disabled in 32-bit address mode, out of the lower-order 29 bits in a 32-bit memory location to be specified, specify a 25-bit address (A28 to A4) in bits 28 to 4. The value is retained during a reset.
3 to 0	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

(9) Plane n Display Area Start Address 1 Register (PnDSA1R)

Address: H'FFF80#24

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PnDSA1															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PnDSA1												—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 4	PnDSA1	—	R/W	Avail-able	Plane n Display Area Start Address 1 To enable bits 31 to 29, the DEFE bit in DEFR must be set to 1. In the initial state, these bits are not enabled. When the buffer mode for plane n is manual display, auto rendering, auto display change, or video capture, the area indicated by this register is used as frame buffer 1. Note: When bits 31 to 29 are disabled in 32-bit address mode, out of the lower-order 29 bits in a 32-bit memory location to be specified, specify a 25-bit address (A28 to A4) in bits 28 to 4. The value is retained during a reset.
3 to 0	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

(10) Plane n Display Area Start Address 2 Register (PnDSA2R)

Address: H'FFF80#28

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PnDSA2															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PnDSA2												—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 4	PnDSA2	—	R/W	Avail-able	Plane n Display Area Start Address 2 To enable bits 31 to 29, the DEFE bit in DEFR must be set to 1. In the initial state, these bits are not enabled. When the buffer mode for plane n is video capture, the area indicated by this register is used as frame buffer 2. Note: When bits 31 to 29 are disabled in 32-bit address mode, out of the lower-order 29 bits in a 32-bit memory location to be specified, specify a 25-bit address (A28 to A4) in bits 28 to 4. The value is retained during a reset.
3 to 0	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

(11) Plane n Start Position X Register (PnSPXR)

Address: H'FFF80#30

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	PnSPX											
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 12	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
11 to 0	PnSPX	—	R/W	Avail-able	Plane n Start Position X Specify the distance in the X direction to the position where plane n starts in memory. Notes: 1. When the YC mode has been selected by the PnDDF bits in PnMR, the value should be even. 2. Setting of a value greater than twice the value of the PnMWX bits is prohibited. The value is retained during a reset.

(12) Plane n Start Position Y Register (PnSPYR)

Address: H'FFF80#34

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PnSPY															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 16	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
15 to 0	PnSPY	—	R/W	Avail-able	Plane n Start Position Y Specify the distance in the Y direction to the position where plane n starts in memory. Note: Setting of a value greater than twice {the value of the PnWASPY bits + the value of the PnWAMWY bits} is prohibited. The value is retained during a reset.

(13) Plane n Wrap-Around Start Position Register (PnWASPR)

Address: H'FFF80#38

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PnWASPY										—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 14	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
13 to 4	PnWASPY	—	R/W	Avail-able	Plane n Wrap Around Start Position Y The Y direction start position of one wrap-around area should be set with reference to the address specified in PnDSA0R and PnDSA1R. The position where wrapping-around is to start can be set in 16-pixel units (bits 3 to 0: Fixed to 0). The value is retained during a reset.
3 to 0	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

(14) Plane n Wrap-Around Memory Width Register (PnWAMWR)

Address: H'FFF80#3C

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	PnWAMWY											
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 12	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
11 to 0	PnWAMWY	—	R/W	Avail-able	Plane n Wrap Around Memory Width Y The memory width for wrap-around in the Y-direction should be set to a number corresponding to a value in the range from 240 to 4095 raster lines. The value is retained during a reset.

(15) Plane n Blinking Time Register (PnBTR)

Address: H'FFF80#40

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PnBTA								PnBTB							
Initial value:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 16	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
15 to 8	PnBTA	00000001	R/W	Avail-able	Plane n Blinking Time A
7 to 0	PnBTB	00000001	R/W	Avail-able	Plane n Blinking Time B

When the PnBM bits in PnMR are set to select the auto display change mode (blinking mode), specify, as numbers of fields, the times over which PnDSA0 and PnDSA1 are to be displayed. Blinking operation employs the settings in PnDSA0 and PnDSA1.

Setting this register to 1 (the value should be other than 0) leads to switching between the buffers at the addresses indicated by the plane n display area start addresses 0 and 1 (PnDSA0 and PnDSA1) for each field.

(16) Plane n Transparent Color 1 Register (PnTC1R)

Address: H'FFF80#44

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	PnTC1							
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 8	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
7 to 0	PnTC1	—	R/W	Avail-able	Plane n Transparent Color 1 for 8 Bits/Pixel This setting is for a transparent color for plane n in the 8 bits/pixel data format. To enable the transparent color setting in this register, the PnTC bit in PnMR must be set to 0. The value is retained during a reset.

(17) Plane n Transparent Color 2 Register (PnTC2R)

Address: H'FFF80#48

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PnTC2															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 16	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
15 to 0	PnTC2	—	R/W	Avail-able	Plane n Transparent Color 2 for 16 Bits/Pixel This setting is for a transparent color for plane n in the 16 bits/pixel or ARGB data format. In the case of ARGB, comparison is with bits 14 to 0 of this register, i.e. bit 15 is ignored. The value is retained during a reset.

(18) Plane n Memory Length Register (PnMLR)

Address: H'FFF80#50

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PnMLY
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PnMLY															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 17	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
16 to 0	PnMLY	All 0	R/W	Available	Plane n Memory Length Y These bits indicate the length in memory (memory area in the Y-direction) of plane n. When the actual display is beyond this area, the data selected in BPOR will be displayed. When the value is 0 (initial value), the area is handled as an infinite area. Thus the data selected in BPOR will never be displayed.

(19) Plane n Swap Control Register (PnSWAPR)

Address: H'FFF80#80

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CODE															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	PnDIGN	PnSPQW	PnSPLW	PnSPWD	PnSPBY
Initial value:	—	—	—	—	—	—	—	—	—	—	—	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 16	CODE	—	—/W	Not available	PnSWAPR Enabling Code (register available code) For a value written to PnSWAPR to be effective, the value must include H'7775 in these bits.
15 to 5	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
4	PnDIGN	0	R/W	Available	Plane n Display Data Format Invalid (Plane n Display Data Format IGNore) This bit is used in combination with bit 0 to control swapping of data in byte (8-bit) units. Also see the description of bit 0. The setting of this bit does not affect the values of bits 3 to 1.
3	PnSPQW	0	R/W	Available	Plane n Quadword Swap Enable (Plane n SwaP Quad Word) 0: Data are not swapped. 1: Data are quadword-swapped (each quadword contains 64 bits) when bit 20 (DSE) of the display unit system control register (DSYSR) is 0.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
2	PnSPLW	0	R/W	Avail-able	Plane n Longword Swap Enable (Plane n SwaP Long Word) 0: Data are not swapped. 1: Data are longword-swapped (each longword contains 32 bits) when bit 20 (DSEC) of the display unit system control register (DSYSR) is 0.
1	PnSPWD	0	R/W	Avail-able	Plane n Word Swap Enable (Plane n SwaP WorD) 0: Data are not swapped. 1: Data are word-swapped (each word contains 16 bits) when bit 20 (DSEC) of the display unit system control register (DSYSR) is 0.
0	PnSPBY	0	R/W	Avail-able	Plane n Byte Swap Enable (Plane n SwaP BYte) This bit is used in combination with bit 4 to control swapping of data in byte (8-bit) units. The following combinations of values are possible when the DSEC bit in the display unit system control register (DSYSR) is 0. [Bits 0 and 4] 00: Data are not swapped. 01: Data are not swapped. 10: Data are swapped in byte (8-bit) units if the value of the PnDDF bits in the plane n mode register (PnMR) is 00 or 11. In other cases, data are not swapped. 11: Data are swapped in byte (8-bit) units.

The functions of the DSEC bit in the display unit system control register (DSYSR), the PnDDF bits in the plane n mode register (PnMR), and this register are summarized below.

Table 14.22 Summary of Endian Conversion

DSEC	PnSPQW	PnSPLW	PnSPWD	PnSPBY	PnDIGN	PnDDF	Data Format	Swap Unit
1	—	—	—	—	—	00	8 bits/pixel	128 bits in byte units
						01	16 bits/pixel (RGB data)	128 bits in word units
						10	ARGB	128 bits in word units
						11	YC	128 bits in byte units
0	0	0	0	0	—	—	—	No conversion
	1	—	—	—	—	—	—	64 bits
	—	1	—	—	—	—	—	32 bits
	—	—	1	—	—	—	—	16 bits
	—	—	—	1	0	00	8 bits/pixel	8 bits
						01	16 bits/pixel (RGB data)	No conversion
						10	ARGB	No conversion
						11	YC	8 bits
					1	—	—	8 bits

(20) Plane n Display Data Control Register (PnDDCR)

Address: H'FFF80#84

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CODE															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	PnLR GB1	PnLR GB0	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	0	0	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 16	CODE	—	—/W	Not available	PnDDCR Enabling Code (register available code) For a value written to PnDDCR to be effective, the value must include H'7775 in these bits.
15 to 12	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
11	PnLRGB1	0	R/W	Available	Plane n 32-Bit/Pixel Display Control 1 (Plane n Long (32-bit) RGB1) 0: Data are not handled as 32 bits/pixel. 1: Data in the plane are used as the lower-order bits (G: 8 bits and B: 8 bits) of 32-bit/pixel data. Bit 11 must be set to 1 in the plane immediately after the plane in which bit 10 was set to 1. For example, if bit 10 of this register in plane 1 is set to 1, bit 11 of this register in plane 2 must be set to 1. Also, the value of the PnDDF bits in PnMR must be 01. Do not set bits 11 and 10 to 1 at the same time. When bit 10 is set to 1, data are handled as 32 bits/pixel even if bit 11 is set to 0, and thus the function that bit 10 is set to 1 is realized. Use planes 1 and 2 for a 32-bit/pixel display. Since transparent color processing is not possible with 32-bit/pixel data, set bit 14 among the PnSPIM bits in PnMR to 1. α blending or EOR operations for 32-bit/pixel data and a lower plane must be performed in an area where the lower plane exists. If there is no lower plane in the area, display of 32-bit/pixel data will not be possible after α blending or an EOR operation.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
10	PnLRGB0	0	R/W	Available	Plane n 32-Bit/Pixel Display Control 0 (Plane n Long (32-bit) RGB0) 0: Data are not handled as 32 bits/pixel. 1: Data in the plane are used as the higher-order bits (A: 8 bits and R: 8 bits) of 32-bit/pixel data. The A value is an 8-bit blending ratio, so the blending ratio from the plane n blending ratio register is not selectable. Also, the value of the PnDDF bits in PnMR must be 01. Do not set bits 11 and 10 to 1 at the same time. When bit 11 is set to 1, data are handled as 32 bits/pixel even if bit 10 is set to 0, and thus the function that bit 11 is set to 1 is realized. Use planes 1 and 2 for a 32-bit/pixel display. Since transparent color processing is not possible with 32-bit/pixel data, set bit 14 among the PnSPIM bits in PnMR to 1.
9 to 0	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

Use planes 1 and 2 if the display is to have 32 bits per pixel. Register settings for using planes 1 and 2 in a 32-bit/pixel display are given below.

(a) Set bits 4 and 5 in the display unit extensional function enable register 4 (DEFR4) to 01.

(b) The settings for this register are shown below.

Bit:	15	14	13	12	11	10	9	8
					PnLRGB 1	PnLRGB 0		
Plane 1 (A, R)	—	—	—	—	0	1	—	—
Plane 2 (G, B)	—	—	—	—	1	0	—	—

(c) The same settings should be made in the plane 1 and plane 2 registers of display plane registers other than this register.

(d) Turn on the display of planes 1 and 2, and give plane 2 the next order of priority for processing after plane 1. For details on turning on the display, see section 14.4.2, Display On/Off.

(e) A 32-bit pixel display should always be specified in lower-numbered planes. For example, when the 32- and 16-bit pixel display planes are to be superimposed, the 32-bit pixel display should be specified in planes 1 and 2, and the 16-bit pixel display should be specified in plane 3 or in higher-numbered planes.

(f) Set H'A9 to bits 7 to 0 in the display SuperHyway priority register (DSHPR).

(g) When the video input module is to be used, set H'0A090008 to the memory transfer control register (MTC) of the video input module.

(h) If the DU operates in big-endian, set bits 3 and 2 in the plane n swap control register (PnSWAP) to 1 and bits 1 and 0 to 0. Bit 4 may be set to either 0 or 1. Set the DSEC bit in the display unit system control register (DSYSR) to 0.

(21) Plane n Display Data Control Register 2 (PnDDCR2)

Address: H'FFF80#88

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CODE															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W	—/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	PnNV21	PnY420	—	—	Pn DIVU	Pn DIVY
Initial value:	—	—	—	—	—	—	—	—	—	—	0	0	—	—	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 16	CODE	—	—/W	Not available	PnDDCR2 Enabling Code (register available code) For a value written to PnDDCR2 to be effective, the value must include H'7776 in these bits.
15 to 6	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
5	PnNV21	0	R/W	Available	Plane n NV21 Data Format This bit is only effective when bits 1 and 4 have been set to 1. 0: When YUV420 data are divided up for storage in memory, the order of the UV portion is NV12. 1: When YUV420 data are divided up for storage in memory, the order of the UV portion is NV21.
4	PnY420	0	R/W	Available	Plane n YUV420 Data Format This bit is only effective if bit 1 has been set to 1. When bit 1 is 0, the data format will be YUV422. 0: The YUV data to be divided up for storage in memory is YUV422. 1: The YUV data to be divided up for storage in memory is YUV420. For a divided YUV display, see section 14.4.19, Divided YUV Display.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
3, 2	—	—	R	—	Reserved
1	PnDIVU	0	R/W	Available	Plane n UV Data from Divided YUV 0: The data format will be determined by the setting of the PnDDF bits in PnMR. 1: The setting of the PnDDF bits in PnMR is ineffective and the plane contains the UV data from YUV data that have been divided up for storage in memory. However, to validate the YC→RGB color space conversion functions, set the PnDDF bits in PnMR to 11 (YC format). Do not set bits 1 and 0 in this register to 1 at the same time. When bit 0 is set to 1, the data format will not be determined by the setting of the PnDDF bits in PnMR even if this bit is set to 0. The data format will be determined by bit 0 in this register being set to 1. For a divided YUV display, see section 14.4.19, Divided YUV Display.
0	PnDIVY	0	R/W	Available	Plane n Y Data from Divided YUV 0: The data format will be determined by the setting of the PnDDF bits in PnMR. 1: The setting of the PnDDF bits in PnMR is ineffective and the data format is the Y data from YUV data that have been divided up for storage in memory. However, to validate the YC→RGB color space conversion functions, set the PnDDF bits in PnMR to 11 (YC format). Do not set bits 1 and 0 in this register to 1 at the same time. When bit 1 is set to 1, the data format will not be determined by the setting of the PnDDF bits in PnMR even if this bit is set to 0. The data format will be determined by bit 1 in this register being set to 1. For a divided YUV display, see section 14.4.19, Divided YUV Display.

14.3.5 Display Capture Registers

n: 1

#: Corresponds to n but represents a hexadecimal digit in an address. For example, address H'FFF8C#04 for the display capture 1 memory width register corresponds to H'FFF8C104.

(1) Display Capture n Memory Width Register (DCnMWR)

Address: H'FFF8C#04

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	DCnMWX									—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 13	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
12 to 4	DCnMWX	—	R/W	Avail-able	Display Capture n Memory Width X Select the memory width for display capture n to a value between 16 and 4096 pixels, in 16-pixel units. When the width of captured data exceeds this size, the excess data will not be captured. The value is retained during a reset.
3 to 0	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

(2) Display Capture n Area Start Address Register (DCnSAR)

Address: H'FFF8C#20

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DCnSA															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCnSA												—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 4	DCnSA	—	R/W	Avail-able	Display Capture n Area Start Address To enable bits 31 to 29, the DEFE bit in DEFR must be set to 1. In the initial state, these bits are not enabled. Note: When bits 31 to 29 are disabled in 32-bit address mode, out of the lower-order 29 bits in a 32-bit memory location to be specified, specify a 25-bit address (A28 to A4) in bits 28 to 4. The value is retained during a reset.
3 to 0	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

(3) Display Capture n Memory Length Register (DCnMLR)

Address: H'FFF8C#50

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DCnMLY
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCnMLY															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 17	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
16 to 0	DCnMLY	H'00000	R/W	Avail-able	Display Capture n Memory Length Y Select the memory length (Y-direction memory area) for display capture n. When the length of captured data exceeds this size, the excess data will not be captured. When the value of these bits is 0 (initial value), the size is unlimited.

14.3.6 Color Palette Registers

(1) Color Palette 1 Register 000 to 255 (CP1_000R to CP1_255R)

Addresses: FFF81000 to FFF813FC

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CP1_000A								CP1_000R						—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CP1_000G						—	—	CP1_000B						—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R

.
.
.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CP1_255A								CP1_255R						—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CP1_255G						—	—	CP1_255B						—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 24	CP1_000A to CP1_255A	—	R/W	Avail-able	Color Palette 1_000 to 255 Blending Ratio When the PnBRSL bits in PnALPHAR are 10, the value is the alpha value, which is the blending ratio. The value is retained during a reset.
23 to 18	CP1_000R to CP1_255R	—	R/W	Avail-able	Color Palette 1_000 to 255 Red Red-color data of color palette 1. The value is retained during a reset.
17, 16	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
15 to 10	CP1_000G to CP1_255G	—	R/W	Avail-able	Color Palette 1_000 to 255 Green Green-color data of color palette 1. The value is retained during a reset.
9, 8	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
7 to 2	CP1_000B to CP1_255B	—	R/W	Avail-able	Color Palette 1_000 to 255 Blue Blue-color data of color palette 1. The value is retained during a reset.
1, 0	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

CP1_000R to CP1_255R form a group of 256 registers in which six bits are set for each of the R, G, and B components of a color, setting up the color palette of 256 colors from among the 260,000 possible colors for display. The values are valid for 8-bit/pixel data display.

After the CP1CE bit in CPCR has been set to 1, settings for CP1_000R to CP1_255R become effective on the next falling edge of VSYNC (timing of an internal update) or when the display is reset. When you update the color palette, rewrite the entire palette. Reading of the color palette by the CPU must proceed before the CP1CE bit has been set to 1. The color palette is accessible in longword units.

(2) Color Palette 2 Register 000 to 255 (CP2_000R to CP2_255R)

Addresses: FFF82000 to FFF823FC

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CP2_000A								CP2_000R						—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CP2_000G						—	—	CP2_000B						—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R

.
.
.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CP2_255A								CP2_255R						—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CP2_255G						—	—	CP2_255B						—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 24	CP2_000A to CP2_255A	—	R/W	Avail-able	Color Palette 2_000 to 255 Blending Ratio When the PnBRSL bits in PnALPHAR are 10, the value is the alpha value, which is the blending ratio. The value is retained during a reset.
23 to 18	CP2_000R to CP2_255R	—	R/W	Avail-able	Color Palette 2_000 to 255 Red Red-color data of color palette 2. The value is retained during a reset.
17, 16	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
15 to 10	CP2_000G to CP2_255G	—	R/W	Avail-able	Color Palette 2_000 to 255 Green Green-color data of color palette 2. The value is retained during a reset.
9, 8	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
7 to 2	CP2_000B to CP2_255B	—	R/W	Avail-able	Color Palette 2_000 to 255 Blue Blue-color data of color palette 2. The value is retained during a reset.
1, 0	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

CP2_000R to CP2_255R form a group of 256 registers in which six bits are set for each of the R, G, and B components of a color, setting up the color palette of 256 colors from among the 260,000 possible colors for display. The values are valid for 8-bit/pixel data display.

After the CP2CE bit in CPCR has been set to 1, settings for CP2_000R to CP2_255R become effective on the next falling edge of VSYNC (timing of an internal update) or when the display is reset. When you update the color palette, rewrite the entire palette. Reading of the color palette by the CPU must proceed before the CP2CE bit has been set to 1. The color palette is accessible in longword units.

(3) Color Palette 3 Register 000 to 255 (CP3_000R to CP3_255R)

Addresses: FFF83000 to FFF833FC

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CP3_000A								CP3_000R						—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CP3_000G						—	—	CP3_000B						—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R

.
.
.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CP3_255A								CP3_255R						—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CP3_255G						—	—	CP3_255B						—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 24	CP3_000A to CP3_255A	—	R/W	Avail-able	Color Palette 3_000 to 255 Blending Ratio When the PnBRSL bits in PnALPHAR are 10, the value is the alpha value, which is the blending ratio. The value is retained during a reset.
23 to 18	CP3_000R to CP3_255R	—	R/W	Avail-able	Color Palette 3_000 to 255 Red Red-color data of color palette 3. The value is retained during a reset.
17, 16	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
15 to 10	CP3_000G to CP3_255G	—	R/W	Avail-able	Color Palette 3_000 to 255 Green Green-color data of color palette 3. The value is retained during a reset.
9, 8	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
7 to 2	CP3_000B to CP3_255B	—	R/W	Avail-able	Color Palette 3_000 to 255 Blue Blue-color data of color palette 3. The value is retained during a reset.
1, 0	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

CP3_000R to CP3_255R form a group of 256 registers in which six bits are set for each of the R, G, and B components of a color, setting up the color palette of 256 colors from among the 260,000 possible colors for display. The values are valid for 8-bit/pixel data display.

After the CP3CE bit in CPCR has been set to 1, settings for CP3_000R to CP3_255R become effective on the next falling edge of VSYNC (timing of an internal update) or when the display is reset. When you update the color palette, rewrite the entire palette. Reading of the color palette by the CPU must proceed before the CP3CE bit has been set to 1. The color palette is accessible in longword units.

(4) Color Palette 4 Register 000 to 255 (CP4_000R to CP4_255R)

Addresses: FFF84000 to FFF843FC

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CP4_000A								CP4_000R						—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CP4_000G						—	—	CP4_000B						—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R

.
.
.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CP4_255A								CP4_255R						—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CP4_255G						—	—	CP4_255B						—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 24	CP4_000A to CP4_255A	—	R/W	Avail-able	Color Palette 4_000 to 255 Blending Ratio When the PnBRSL bits in PnALPHAR are 10, the value is the alpha value, which is the blending ratio. The value is retained during a reset.
23 to 18	CP4_000R to CP4_255R	—	R/W	Avail-able	Color Palette 4_000 to 255 Red Red-color data of color palette 4. The value is retained during a reset.
17, 16	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
15 to 10	CP4_000G to CP4_255G	—	R/W	Avail-able	Color Palette 4_000 to 255 Green Green-color data of color palette 4. The value is retained during a reset.
9, 8	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
7 to 2	CP4_000B to CP4_255B	—	R/W	Avail-able	Color Palette 4_000 to 255 Blue Blue-color data of color palette 4. The value is retained during a reset.
1, 0	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

CP4_000R to CP4_255R form a group of 256 registers in which six bits are set for each of the R, G, and B components of a color, setting up the color palette of 256 colors from among the 260,000 possible colors for display. The values are valid for 8-bit/pixel data display.

After the CP4CE bit in CPCR has been set to 1, settings for CP4_000R to CP4_255R become effective on the next falling edge of VSYNC (timing of an internal update) or when the display is reset. When you update the color palette, rewrite the entire palette. Reading of the color palette by the CPU must proceed before the CP4CE bit has been set to 1. The color palette is accessible in longword units.

14.3.7 External Synchronization Control Registers

(1) External Synchronization Control Register (ESCR)

Address: H'FFF90000

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	DCLKSEL	—	—	—	DCLKDIS
Initial value:	—	—	—	—	—	—	—	—	—	—	—	0	—	—	—	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SYNCSEL		—	—	FRQSEL					
Initial value:	—	—	—	—	—	—	0	0	—	—	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 21	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
20	DCLKSEL	0	R/W	None	DCLKIN Select 0: The input dot clock source is the DCLKIN pin 1: The input dot clock source is the CLKS clock. If this setting is made, ensure that the frequency of the input dot clock is divided by two or a greater value (so that the result of dividing the frequency is more than or equal to half the CLKS frequency).
19 to 17	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
16	DCLKDIS	0	R/W	None	DCLKOUT Disable 0: DCLKOUT is output. 1: DCLKOUT is not output. DCLKOUT is fixed to low level.
15 to 10	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
9, 8	SYNCSEL	00	R/W	None	SYNC Select 00: Phases are not synchronized 01: Phases are not synchronized 10: Phases are synchronized by using the EXVSYNC signal 11: Phases are synchronized by using the EXHSYNC signal
7, 6	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
5 to 0	FRQSEL	000000	R/W	None	Frequency Select To enable bit 5, the DEFE bit in DEFR must be set to 1. In the initial state, bit 5 is fixed to 0. If frequency division is by an odd number, the duty cycle of the frequency divided dot clock will be below 50%. 000000: Frequency division of the input dot clock (clock for division) is not performed. 000001: Division by 2 of the input dot clock (clock for division) 000010: Division by 3 of the input dot clock (clock for division) 000011: Division by 4 of the input dot clock (clock for division) 000100: Division by 5 of the input dot clock (clock for division) 000101: Division by 6 of the input dot clock (clock for division) 000110: Division by 7 of the input dot clock (clock for division) 000111: Division by 8 of the input dot clock (clock for division) 001000: Division by 9 of the input dot clock (clock for division) 001001: Division by 10 of the input dot clock (clock for division) 001010: Division by 11 of the input dot clock (clock for division) 001011: Division by 12 of the input dot clock (clock for division) 001100: Division by 13 of the input dot clock (clock for division) 001101: Division by 14 of the input dot clock (clock for division) 001110: Division by 15 of the input dot clock (clock for division) 001111: Division by 16 of the input dot clock (clock for division) 010000: Division by 17 of the input dot clock (clock for division) 010001: Division by 18 of the input dot clock (clock for division) 010010: Division by 19 of the input dot clock (clock for division) 010011: Division by 20 of the input dot clock (clock for division) 010100: Division by 21 of the input dot clock (clock for division) 010101: Division by 22 of the input dot clock (clock for division) 010110: Division by 23 of the input dot clock (clock for division) 010111: Division by 24 of the input dot clock (clock for division) 011000: Division by 25 of the input dot clock (clock for division) 011001: Division by 26 of the input dot clock (clock for division) 011010: Division by 27 of the input dot clock (clock for division) 011011: Division by 28 of the input dot clock (clock for division) 011100: Division by 29 of the input dot clock (clock for division) 011101: Division by 30 of the input dot clock (clock for division)

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
5 to 0	FRQSEL	000000	R/W	None	011110: Division by 31 of the input dot clock (clock for division) 011111: Division by 32 of the input dot clock (clock for division) 100000: Division by 33 of the input dot clock (clock for division) 100001: Division by 34 of the input dot clock (clock for division) 100010: Division by 35 of the input dot clock (clock for division) 100011: Division by 36 of the input dot clock (clock for division) 100100: Division by 37 of the input dot clock (clock for division) 100101: Division by 38 of the input dot clock (clock for division) 100110: Division by 39 of the input dot clock (clock for division) 100111: Division by 40 of the input dot clock (clock for division) 101000: Division by 41 of the input dot clock (clock for division) 101001: Division by 42 of the input dot clock (clock for division) 101010: Division by 43 of the input dot clock (clock for division) 101011: Division by 44 of the input dot clock (clock for division) 101100: Division by 45 of the input dot clock (clock for division) 101101: Division by 46 of the input dot clock (clock for division) 101110: Division by 47 of the input dot clock (clock for division) 101111: Division by 48 of the input dot clock (clock for division) 110000: Division by 49 of the input dot clock (clock for division) 110001: Division by 50 of the input dot clock (clock for division) 110010: Division by 51 of the input dot clock (clock for division) 110011: Division by 52 of the input dot clock (clock for division) 110100: Division by 53 of the input dot clock (clock for division) 110101: Division by 54 of the input dot clock (clock for division) 110110: Division by 55 of the input dot clock (clock for division) 110111: Division by 56 of the input dot clock (clock for division) 111000: Division by 57 of the input dot clock (clock for division) 111001: Division by 58 of the input dot clock (clock for division) 111010: Division by 59 of the input dot clock (clock for division) 111011: Division by 60 of the input dot clock (clock for division) 111100: Division by 61 of the input dot clock (clock for division) 111101: Division by 62 of the input dot clock (clock for division) 111110: Division by 63 of the input dot clock (clock for division) 111111: Division by 64 of the input dot clock (clock for division)

(2) Output Signal Timing Adjustment Register (OTAR)

Address: H'FFF90004

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	DEA			—	CLAMPA			—	DRGBA			—	—	—	—
Initial value:	—	0	0	0	—	0	0	0	—	0	0	0	—	—	—	—
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	CDEA			—	DISPA			—	SYNCA		
Initial value:	—	—	—	—	—	0	0	0	—	0	0	0	—	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
30 to 28	DEA	000	R/W	DRES	DE Output Timing Adjustment 000: Adjustment of output timing is not performed. The DE signal is output on the rising edge of the dot clock, with the reference timing. 001: The DE signal is output on the rising edge, delayed one dot clock cycle relative to the reference timing. 010: The DE signal is output on the rising edge, delayed two dot clock cycles relative to the reference timing. 011: The DE signal is output on the rising edge, delayed three dot clock cycles relative to the reference timing. 100: The DE signal is output on the falling edge, preceding the reference timing by 1/2 dot clock cycle. 101: The DE signal is output on the falling edge, delayed 1/2 dot clock cycle relative to the reference timing. 110: The DE signal is output on the falling edge, delayed (1+1/2) dot clock cycles relative to the reference timing. 111: The DE signal is output on the falling edge, delayed (2+1/2) dot clock cycles relative to the reference timing.
27	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
26 to 24	CLAMPA	000	R/W	DRES	CLAMP Output Timing Adjustment 000: Adjustment of output timing is not performed. The CLAMP signal is output on the rising edge of the dot clock, with the reference timing. 001: The CLAMP signal is output on the rising edge, delayed one dot clock cycle relative to the reference timing. 010: The CLAMP signal is output on the rising edge, delayed two dot clock cycles relative to the reference timing. 011: The CLAMP signal is output on the rising edge, delayed three dot clock cycles relative to the reference timing. 100: The CLAMP signal is output on the falling edge, preceding the reference timing by 1/2 dot clock cycle. 101: The CLAMP signal is output on the falling edge, delayed 1/2 dot clock cycle relative to the reference timing. 110: The CLAMP signal is output on the falling edge, delayed (1+1/2) dot clock cycles relative to the reference timing. 111: The CLAMP signal is output on the falling edge, delayed (2+1/2) dot clock cycles relative to the reference timing.
23	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
22 to 20	DRGBA	000	R/W	DRES	Digital RGB Output Timing Adjustment 000: Adjustment of output timing is not performed. The digital RGB signal is output on the rising edge of the dot clock, with the reference timing. 001: The digital RGB signal is output on the rising edge, delayed one dot clock cycle relative to the reference timing. 010: The digital RGB signal is output on the rising edge, delayed two dot clock cycles relative to the reference timing. 011: The digital RGB signal is output on the rising edge, delayed three dot clock cycles relative to the reference timing. 100: The digital RGB signal is output on the falling edge, preceding the reference timing by 1/2 dot clock cycle. 101: The digital RGB signal is output on the falling edge, delayed 1/2 dot clock cycle relative to the reference timing. 110: The digital RGB signal is output on the falling edge, delayed (1+1/2) dot clock cycles relative to the reference timing. 111: The digital RGB signal is output on the falling edge, delayed (2+1/2) dot clock cycles relative to the reference timing. If you intend to use the output dot clock to control the timing of the switching of data for display, do not select the falling edge (i.e. do not set bit 22 to 1). The value of bits 22 to 20 must be the same as that of bits 22 to 20 in OTAR2.
19 to 11	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
10 to 8	CDEA	000	R/W	DRES	CDE Output Timing Adjustment 000: Adjustment of output timing is not performed. The CDE signal is output on the rising edge of the dot clock, with the reference timing. 001: The CDE signal is output on the rising edge, delayed one dot clock cycle relative to the reference timing. 010: The CDE signal is output on the rising edge, delayed two dot clock cycles relative to the reference timing. 011: The CDE signal is output on the rising edge, delayed three dot clock cycles relative to the reference timing. 100: The CDE signal is output on the falling edge, preceding the reference timing by 1/2 dot clock cycle. 101: The CDE signal is output on the falling edge, delayed 1/2 dot clock cycle relative to the reference timing. 110: The CDE signal is output on the falling edge, delayed (1+1/2) dot clock cycles relative to the reference timing. 111: The CDE signal is output on the falling edge, delayed (2+1/2) dot clock cycles relative to the reference timing.
7	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
6 to 4	DISPA	000	R/W	DRES	DISP Output Timing Adjustment 000: Adjustment of output timing is not performed. The DISP signal is output on the rising edge of the dot clock, with the reference timing. 001: The DISP signal is output on the rising edge, delayed one dot clock cycle relative to the reference timing. 010: The DISP signal is output on the rising edge, delayed two dot clock cycles relative to the reference timing. 011: The DISP signal is output on the rising edge, delayed three dot clock cycles relative to the reference timing. 100: The DISP signal is output on the falling edge, preceding the reference timing by 1/2 dot clock cycle. 101: The DISP signal is output on the falling edge, delayed 1/2 dot clock cycle relative to the reference timing. 110: The DISP signal is output on the falling edge, delayed (1+1/2) dot clock cycles relative to the reference timing. 111: The DISP signal is output on the falling edge, delayed (2+1/2) dot clock cycles relative to the reference timing.
3	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
2 to 0	SYNCA	000	R/W	DRES	SYNC* Output Timing Adjustment 000: Adjustment of output timing is not performed. The SYNC* signal is output on the rising edge of the dot clock, with the reference timing. 001: The SYNC* signal is output on the rising edge, delayed one dot clock cycle relative to the reference timing. 010: The SYNC* signal is output on the rising edge, delayed two dot clock cycles relative to the reference timing. 011: The SYNC* signal is output on the rising edge, delayed three dot clock cycles relative to the reference timing. 100: The SYNC* signal is output on the falling edge, preceding the reference timing by 1/2 dot clock cycle. 101: The SYNC* signal is output on the falling edge, delayed 1/2 dot clock cycle relative to the reference timing. 110: The SYNC* signal is output on the falling edge, delayed (1+1/2) dot clock cycles relative to the reference timing. 111: The SYNC* signal is output on the falling edge, delayed (2+1/2) dot clock cycles relative to the reference timing. Note: * HSYNC, VSYNC, CSYNC, and ODDF signals

When signals are to be output on the falling edge, the electrical characteristics do not apply.

14.3.8 Dual Display Output Control Registers

(1) Display Unit Output Route Control Register (DORCR)

Address: H'FFF91000

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PG1D	
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DPRS
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 18	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
17, 16	PG1D	00	R/W	DRES	Pin Generate 1 Input Data Select 00: Data from superposition processor 1 are input to pin controller 1. The setting of bit 2 in this register indicates the timing of superposition processor 1. 01: Setting prohibited 10: Input to pin controller 1 is fixed to 0. The value of the CDE pin is fixed to 0. 11: The value of the display off mode output register (DOOR) is input to pin controller 1. The value of the CDE pin is fixed to 0. The combination of the DRES and DEN bits of the display unit system control register (DSYSR) determines the data to be superposed. DRES/DEN = 00: Value of DOOR DRES/DEN = 01: Data in unified memory DRES/DEN = 10: 0 DRES/DEN = 11: Setting prohibited (data: 0)

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
15 to 1	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
0	DPRS	0	R/W	DRES	Display Priority Register Select 0: The order of priority for the planes should be set in the display plane priority register. Superposition processor 1 can be used to superpose planes 1 to 8. 1: The order of priority for the planes should be set in the display superimpose 1 priority register. Superposition processor 1 can be used to superpose planes 1 to 8.

(2) Display Superimpose 1 Priority Register (DS1PR)

Address: H'FFF91020

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S1S8				S1S7				S1S6				S1S5			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	S1S4				S1S3				S1S2				S1S1			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Notes: 1. The setting of this register is valid when the DPRS bit in the display unit output route control register (DORCR) is 1.

2. After setting a desired value in a register listed in section 14.3.4, Display Plane Registers, set DS1PR to the value indicating the relevant plane number (0001 to 1000).

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 28	S1S8	0000	R/W	Avail-able	Display Superimposition 1 Priority 8 Select 0000: Priority level 8 is not used in display generation by superposition processor 1. 0001: Plane 1 has priority level 8 in display generation by superposition processor 1. 0010: Plane 2 has priority level 8 in display generation by superposition processor 1. 0011: Plane 3 has priority level 8 in display generation by superposition processor 1. 0100: Plane 4 has priority level 8 in display generation by superposition processor 1. 0101: Plane 5 has priority level 8 in display generation by superposition processor 1. 0110: Plane 6 has priority level 8 in display generation by superposition processor 1. 0111: Plane 7 has priority level 8 in display generation by superposition processor 1. 1000: Plane 8 has priority level 8 in display generation by superposition processor 1. 1001: Setting prohibited 1010: Setting prohibited 1011: Setting prohibited 1100: Setting prohibited 1101: Setting prohibited 1110: Setting prohibited 1111: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
27 to 24	S1S7	0000	R/W	Avail-able	Display Superimposition 1 Priority 7 Select 0000: Priority level 7 is not used in display generation by superposition processor 1. 0001: Plane 1 has priority level 7 in display generation by superposition processor 1. 0010: Plane 2 has priority level 7 in display generation by superposition processor 1. 0011: Plane 3 has priority level 7 in display generation by superposition processor 1. 0100: Plane 4 has priority level 7 in display generation by superposition processor 1. 0101: Plane 5 has priority level 7 in display generation by superposition processor 1. 0110: Plane 6 has priority level 7 in display generation by superposition processor 1. 0111: Plane 7 has priority level 7 in display generation by superposition processor 1. 1000: Plane 8 has priority level 7 in display generation by superposition processor 1. 1001: Setting prohibited 1010: Setting prohibited 1011: Setting prohibited 1100: Setting prohibited 1101: Setting prohibited 1110: Setting prohibited 1111: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
23 to 20	S1S6	0000	R/W	Avail-able	Display Superimposition 1 Priority 6 Select 0000: Priority level 6 is not used in display generation by superposition processor 1. 0001: Plane 1 has priority level 6 in display generation by superposition processor 1. 0010: Plane 2 has priority level 6 in display generation by superposition processor 1. 0011: Plane 3 has priority level 6 in display generation by superposition processor 1. 0100: Plane 4 has priority level 6 in display generation by superposition processor 1. 0101: Plane 5 has priority level 6 in display generation by superposition processor 1. 0110: Plane 6 has priority level 6 in display generation by superposition processor 1. 0111: Plane 7 has priority level 6 in display generation by superposition processor 1. 1000: Plane 8 has priority level 6 in display generation by superposition processor 1. 1001: Setting prohibited 1010: Setting prohibited 1011: Setting prohibited 1100: Setting prohibited 1101: Setting prohibited 1110: Setting prohibited 1111: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
19 to 16	S1S5	0000	R/W	Avail-able	Display Superimposition 1 Priority 5 Select 0000: Priority level 5 is not used in display generation by superposition processor 1. 0001: Plane 1 has priority level 5 in display generation by superposition processor 1. 0010: Plane 2 has priority level 5 in display generation by superposition processor 1. 0011: Plane 3 has priority level 5 in display generation by superposition processor 1. 0100: Plane 4 has priority level 5 in display generation by superposition processor 1. 0101: Plane 5 has priority level 5 in display generation by superposition processor 1. 0110: Plane 6 has priority level 5 in display generation by superposition processor 1. 0111: Plane 7 has priority level 5 in display generation by superposition processor 1. 1000: Plane 8 has priority level 5 in display generation by superposition processor 1. 1001: Setting prohibited 1010: Setting prohibited 1011: Setting prohibited 1100: Setting prohibited 1101: Setting prohibited 1110: Setting prohibited 1111: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
15 to 12	S1S4	0000	R/W	Avail-able	Display Superimposition 1 Priority 4 Select 0000: Priority level 4 is not used in display generation by superposition processor 1. 0001: Plane 1 has priority level 4 in display generation by superposition processor 1. 0010: Plane 2 has priority level 4 in display generation by superposition processor 1. 0011: Plane 3 has priority level 4 in display generation by superposition processor 1. 0100: Plane 4 has priority level 4 in display generation by superposition processor 1. 0101: Plane 5 has priority level 4 in display generation by superposition processor 1. 0110: Plane 6 has priority level 4 in display generation by superposition processor 1. 0111: Plane 7 has priority level 4 in display generation by superposition processor 1. 1000: Plane 8 has priority level 4 in display generation by superposition processor 1. 1001: Setting prohibited 1010: Setting prohibited 1011: Setting prohibited 1100: Setting prohibited 1101: Setting prohibited 1110: Setting prohibited 1111: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
11 to 8	S1S3	0000	R/W	Avail-able	Display Superimposition 1 Priority 3 Select 0000: Priority level 3 is not used in display generation by superposition processor 1. 0001: Plane 1 has priority level 3 in display generation by superposition processor 1. 0010: Plane 2 has priority level 3 in display generation by superposition processor 1. 0011: Plane 3 has priority level 3 in display generation by superposition processor 1. 0100: Plane 4 has priority level 3 in display generation by superposition processor 1. 0101: Plane 5 has priority level 3 in display generation by superposition processor 1. 0110: Plane 6 has priority level 3 in display generation by superposition processor 1. 0111: Plane 7 has priority level 3 in display generation by superposition processor 1. 1000: Plane 8 has priority level 3 in display generation by superposition processor 1. 1001: Setting prohibited 1010: Setting prohibited 1011: Setting prohibited 1100: Setting prohibited 1101: Setting prohibited 1110: Setting prohibited 1111: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
7 to 4	S1S2	0000	R/W	Avail-able	Display Superimposition 1 Priority 2 Select 0000: Priority level 2 is not used in display generation by superposition processor 1. 0001: Plane 1 has priority level 2 in display generation by superposition processor 1. 0010: Plane 2 has priority level 2 in display generation by superposition processor 1. 0011: Plane 3 has priority level 2 in display generation by superposition processor 1. 0100: Plane 4 has priority level 2 in display generation by superposition processor 1. 0101: Plane 5 has priority level 2 in display generation by superposition processor 1. 0110: Plane 6 has priority level 2 in display generation by superposition processor 1. 0111: Plane 7 has priority level 2 in display generation by superposition processor 1. 1000: Plane 8 has priority level 2 in display generation by superposition processor 1. 1001: Setting prohibited 1010: Setting prohibited 1011: Setting prohibited 1100: Setting prohibited 1101: Setting prohibited 1110: Setting prohibited 1111: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
3 to 0	S1S1	0000	R/W	Avail-able	Display Superimposition 1 Priority 1 Select 0000: Priority level 1 is not used in display generation by superposition processor 1. 0001: Plane 1 has priority level 1 in display generation by superposition processor 1. 0010: Plane 2 has priority level 1 in display generation by superposition processor 1. 0011: Plane 3 has priority level 1 in display generation by superposition processor 1. 0100: Plane 4 has priority level 1 in display generation by superposition processor 1. 0101: Plane 5 has priority level 1 in display generation by superposition processor 1. 0110: Plane 6 has priority level 1 in display generation by superposition processor 1. 0111: Plane 7 has priority level 1 in display generation by superposition processor 1. 1000: Plane 8 has priority level 1 in display generation by superposition processor 1. 1001: Setting prohibited 1010: Setting prohibited 1011: Setting prohibited 1100: Setting prohibited 1101: Setting prohibited 1110: Setting prohibited 1111: Setting prohibited

14.3.9 YC-RGB Conversion Coefficient Registers

(1) Y Normalization Coefficient Register (YNCR)

Address: H'FFF91080

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	YNC1											
Initial value:	—	—	—	—	1	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 12	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
11 to 0	YNC1	H'800	R/W	Avail-able	Y Normalization Coefficient 1 This coefficient is for normalization of Y values in YC-RGB conversion by the YC-RGB conversion generation by superposition processor 1. The coefficient is a fixed-point 12-bit value. The 11th bit represents the integer one or zero and the 10th to 0th bits represent the value below the fixed point. The default value is 1.

(2) Y Normalization Offset Register (YNOR)

Address: H'FFF91084

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	YNO1							
Initial value:	—	—	—	—	—	—	—	—	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 8	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
7 to 0	YNO1	H'00	R/W	Avail-able	Y Normalization Offset 1 This offset is to be subtracted from Y values before they are normalized in YC-RGB conversion by the YC-RGB conversion generation by superposition processor 1. The offset is an unsigned 8-bit integer. Its default value is 0.

(3) Cr Normalization Offset Register (CRNOR)

Address: H'FFF91088

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CRNO1							
Initial value:	—	—	—	—	—	—	—	—	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 8	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
7 to 0	CRNO1	H'80	R/W	Avail-able	Cr Normalization Offset 1 This offset is to be subtracted from Cr values before they are normalized in YC-RGB conversion by the YC-RGB conversion generation by superposition processor 1. The offset is an unsigned 8-bit integer. Its default value is 128.

(4) Cb Normalization Offset Register (CBNOR)

Address: H'FFF9108C

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CBNO1							
Initial value:	—	—	—	—	—	—	—	—	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 8	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
7 to 0	CBNO1	H'80	R/W	Avail-able	Cb Normalization Offset 1 This offset is to be subtracted from Cb values before they are normalized in YC-RGB conversion by the YC-RGB conversion generation by superposition processor 1. The offset is an unsigned 8-bit integer. Its default value is 128.

(5) Red Cr Coefficient Register (RCRCR)

Address: H'FFF91090

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	RCRC1											
Initial value:	—	—	—	—	1	0	1	0	1	1	1	1	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 12	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
11 to 0	RCRC1	H'AF0	R/W	Avail-able	Red Cr Coefficient 1 Cr is multiplied by this coefficient to create the red component in YC-RGB conversion by the YC-RGB conversion generation by superposition processor 1. The coefficient is a fixed-point 12-bit value. The 11th bit represents the integer one or zero and the 10th to 0th bits represent the value below the fixed point. The default value is 1.37.

(6) Green Cr Coefficient Register (GCRCR)

Address: H'FFF91094

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	GCRC1											
Initial value:	—	—	—	—	0	1	0	1	1	0	0	1	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 12	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
11 to 0	GCRC1	H'590	R/W	Avail-able	Green Cr Coefficient 1 Cr is multiplied by this coefficient to create the green component in YC-RGB conversion by the YC-RGB conversion generation by superposition processor 1. The coefficient is a fixed-point 12-bit value. The 11th bit represents the integer one or zero and the 10th to 0th bits represent the value below the fixed point. The default value is 0.698.

(7) Green Cb Coefficient Register (GCBCR)

Address: H'FFF91098

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	GCBC1											
Initial value:	—	—	—	—	0	0	1	0	1	0	1	1	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 12	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
11 to 0	GCBC1	H'2B0	R/W	Avail-able	Green Cb Coefficient 1 Cb is multiplied by this coefficient to create the green component in YC-RGB conversion by the YC-RGB conversion generation by superposition processor 1. The coefficient is a fixed-point 12-bit value. The 11th bit represents the integer one or zero and the 10th to 0th bits represent the value below the fixed point. The default value is 0.336.

(8) Blue Cb Coefficient Register (BCBCR)

Address: H'FFF9109C

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	BCBC1											
Initial value:	—	—	—	—	1	1	0	1	1	1	1	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 12	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
11 to 0	BCBC1	H'DE0	R/W	Avail-able	Blue Cb Coefficient 1 Cb is multiplied by this coefficient to create the blue component in YC-RGB conversion by the YC-RGB conversion generation by superposition processor 1. The coefficient is a fixed-point 12-bit value. The 11th bit represents the integer one or zero and the 10th to 0th bits represent the value below the fixed point. The default value is 1.73.

14.3.10 Display Output Compare Registers

(1) Display Output Compare Control Register (DOCMCR)

Address: H'FFF93000

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CMPR1U
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CMPR1
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 17	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
16	CMPR1U	0	R	—	Updated Value of Display Output Compare Execution 1 This bit reflects the internally-updated value of the display output compare execution 1 (CMPR1) bit.
15 to 1	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
0	CMPR1	0	R/W	Available	Display Output Compare Execution 1 Starts operation of display output compare 1. 0: Display output compare operation is stopped (initial value). 1: Display output compare operation is started.

(2) Display Output Compare Status Register (DOCMSTR)

Address: H'FFF93004

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CMPVS	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CMPST1
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31	CMPVS	—	R	—	Vsync Status 0: No falling edge has been detected in Vsync (internal signal). 1: A falling edge has been detected in Vsync (internal signal).
30 to 1	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
0	CMPST1	0	R	Not available	Display Output Compare Status 1 0: No difference has been detected in display output compare 1. 1: A difference has been detected in display output compare 1.

(3) Display Output Compare Status Clear Register (DOCMCLSTR)

Address: H'FFF93008

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CMPCLVS	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	—/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CMCLST1
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	—/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31	CMPCLVS	—	—/W	Not available	Vsync Status Clear 0: Does not change the CMPVS flag in the display output compare status register (DOCMSTR). 1: Clears the CMPVS flag in the display output compare status register (DOCMSTR).
30 to 1	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
0	CMPCLST1	—	—/W	Not available	Display Output Compare Status 1 Clear 0: Does not change the CMPST1 flag in the display output compare status register (DOCMSTR). 1: Clears the CMPST1 flag in the display output compare status register (DOCMSTR).

(4) Display Output Compare Interrupt Enable Register (DOCMIENR)

Address: H'FFF9300C

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CMPI EN1
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 1	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
0	CMPIEN1	—	R/W	Not available	Display Output Compare Interrupt Enable 1 0: Disables the CMPST1 flag interrupt of the display output compare status register (DOCMSTR). 1: Enables the CMPST1 flag interrupt of the display output compare status register (DOCMSTR).

(5) Display Output Compare Mode Register (DOCMMDR1)

Address: H'FFF93020

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CRC UMD	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 2	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
1	CRCUMD	0	R/W	Not available	Internal Update Mode for Expected CRC Value in Display Output Compare (Update modes other than that for the expected CRC value are specified through CMPUMD) 0: The expected CRC value is internally updated with the Vsync timing. 1: The expected CRC value is internally updated with the register write timing.
0	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.

(6) Display Output Compare Parameter Register (DOCMPMR1)

Address: H'FFF93024

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CMPDFAU								CMPD AUFU	CMPCVFU		CMPD FFU	CMPSELPU			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMPDFA								CMPD AUF	CMPCVF		CMP DFF	CMPSELP			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 24	CMPDFAU	H'00	R	—	Updated Default Alpha Value for Display Output Compare These bits reflect the internally-updated default alpha value (CMPDFA) for display output compare.
23	CMPDAUFU	0	R	—	Updated Value of Default Alpha Value Use Flag for Display Output Compare This bit reflects the internally-updated value of the default alpha value use flag (CMPDAUF) for display output compare.
22, 21	CMPCVFU	00	R	—	Updated Value of Data Conversion Flag for Display Output Compare These bits reflect the internally-updated value of the data conversion flag (CMPCVF) for display output compare.
20	CMPDFFU	0	R	—	Updated Value of Data Format Flag for Display Output Compare This bit reflects the internally-updated value of the data format flag (CMPDFF) for display output compare.
19 to 16	CMPSELPU	H'0	R	—	Updated Value of Plane Select for Display Output Compare These bits reflect the internally-updated value of the plane select bits (CMPSELP) for display output compare.

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
15 to 8	CMPDFA	H'00	R/W	Available	Default Alpha Value for Display Output Compare These bits specify the alpha value (value used for CRC calculation) used when CMPDAUF = 1.
7	CMPDAUF	0	R/W	Available	Default Alpha Value Use Flag for Display Output Compare 0: The default alpha value (CMPDFA) is not used; the value read from memory is used. 1: The default alpha value (CMPDFA) is used.
6, 5	CMPCVF	00	R/W	Available	Data Conversion Flag for Display Output Compare These bits should be used when CRC is calculated in the RGB666 format. 00: The lower two bits are filled with 0 in conversion to RGB888. 01: The lower two bits are filled with 1 in conversion to RGB888. 10: The lower bits are filled with the value of bit 3 in conversion to RGB888. 11: The lower bits are filled with the value of the MSB in conversion to RGB888.
4	CMPDFF	0	R/W	Available	Data Format Flag for Display Output Compare Specifies the data format when the 32-bit/pixel format is used. 0: RGB888 or ARGB8888. 1: RGB666.
3 to 0	CMPSELP	H'0	R/W	Available	Plane Select for Display Output Compare These bits select the target plane for comparison. H'0: No plane is selected. H'1 to H'8: Display data in planes 1 to 8 are selected, respectively. H'9: Display data after blending is selected.

(7) Display Output Compare Expected CRC Value Register (DOCMECRCR1)

Address: H'FFF93028

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CMPECRC															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMPECRC															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 0	CMPECRC	H'00000000	R/W	Available	Display Output Compare Expected CRC Value These bits specify the expected CRC value for the selected area in the selected plane.

(8) Display Output Compare Expected CRC Value Update Register (DOCMECRCUR1)

Address: H'FFF9302C

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CMPECRCU															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMPECRCU															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 0	CMPECRCU	H'00000000	R	Not available	Updated Expected CRC Value for Display Output Compare These bits reflect the internally-updated value of the display output compare expected CRC value register (DOCMECRCR).

(9) Display Output Compare Expected CRC Value Hold Register (DOCMECRCHR1)

Address: H'FFF93030

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CMPECRCH															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMPECRCH															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 0	CMPECRCH	H'00000000	R	Not Available	Expected CRC Value Hold for Display Output Compare These bits hold the expected CRC value that was used for display output compare.

(10) Display Output Compare Calculated CRC Value Register (DOCMCCRCR1)

Address: H'FFF93034

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CMPCCRC															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMPCCRC															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 0	CMPCCRC	H'00000000	R	Not Available	Calculated CRC Value for Display Output Compare These bits show the CRC value calculated for the selected area in the selected plane.

(11) Display Output Compare Start Position X Register (DOCMSPXR1)

Address: H'FFF93038

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	CMPSPXU										
Initial value:	—	—	—	—	—	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	CMPSPX										
Initial value:	—	—	—	—	—	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 27	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
26 to 16	CMPSPXU	H'000	R	Not Available	Updated Value of Display Output Compare Start Position X These bits reflect the internally-updated value of the display output compare start position X (CMPSPX).
15 to 11	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
10 to 0	CMPSPX	H'000	R/W	Available	Display Output Compare Start Position X These bits specify the horizontal start position of the target area for display output compare operation. Specify a value not greater than the value in the selected plane n display size X register (PnDSXR).

(12) Display Output Compare Start Position Y Register (DOCMSPYR1)

Address: H'FFF9303C

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CMPSPYU									
Initial value:	—	—	—	—	—	—	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CMPSPY									
Initial value:	—	—	—	—	—	—	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 26	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
25 to 16	CMPSPYU	H'000	R	Not Available	Updated Value of Display Output Compare Start Position Y These bits reflect the internally-updated value of the display output compare start position Y (CMPSPY).
15 to 10	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
9 to 0	CMPSPY	H'000	R/W	Available	Display Output Compare Start Position Y These bits specify the vertical start position of the target area for display output compare operation. Specify a value not greater than the value in the selected plane n display size Y register (PnDSYR).

(13) Display Output Compare Size X Register (DOCMSZXR1)

Address: H'FFF93040

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	CMPSZXU										
Initial value:	—	—	—	—	—	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	CMPSZX										
Initial value:	—	—	—	—	—	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 27	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
26 to 16	CMPSZXU	H'000	R	Not Available	Updated Value of Display Output Compare Size X These bits reflect the internally-updated value of the display output compare size X (CMPSZX).
15 to 11	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
10 to 0	CMPSZX	H'000	R/W	Available	Display Output Compare Size X These bits specify the horizontal size of the target area for display output compare operation. Specify an appropriate value so that the sum of the values in the display output compare start position X (CMPSPX) and the display output compare size X (CMPSZX) bits does not become greater than the value in the selected plane n display size X register (PnDSXR).

(14) Display Output Compare Size Y Register (DOCMSZYR1)

Address: H'FFF93044

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CMPSZYU									
Initial value:	—	—	—	—	—	—	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CMPSZY									
Initial value:	—	—	—	—	—	—	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 26	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
25 to 16	CMPSZYU	H'000	R	Not Available	Updated Value of Display Output Compare Size Y These bits reflect the internally-updated value of the display output compare size Y (CMPSZY).
15 to 10	—	—	R	—	Reserved The read value is undefined. The write value should always be 0.
9 to 0	CMPSZY	H'000	R/W	Available	Display Output Compare Size Y These bits specify the vertical size of the target area for display output compare operation. Specify an appropriate value so that the sum of the values in the display output compare start position Y (CMPSPY) and the display output compare size Y (CMPSZY) bits does not become greater than the value in the selected plane n display size Y register (PnDSYR).

(15) Display Output Compare Initial CRC Value Register (DOCMCRCIR1)

Address: H'FFF93048

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRCINI															
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMPCCRC															
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Internal Update	Description
31 to 0	CRCINI	H'FFFFFFFF	R/W	Not available	Initial CRC Value for Display Output Compare These bits specify the initial CRC value for the selected area in the selected plane.

14.4 Display Function

14.4.1 Configuration of Output Screen

The display unit (DU) executes window displays with up to a maximum of eight window layers. Each of these windows is called a "plane", and the order of stacking of the planes can be set arbitrarily. For each plane, display can be turned on and off, and the display data format (8 bits/pixel, 16 bits/pixel, ARGB, YC, 32 bits/pixel), blending functions, and other settings can be changed independently.

Each plane has a double-buffer configuration (triple buffer only for a video capture plane), so that smooth display is possible.

Note: In cases of high-resolution display, the unified memory traffic volume may be considerable depending on the number of combined planes and display size, and constraints may arise owing to the traffic volume; but there are no constraints on display functions.

Table 14.23 Display Functions of Planes

	Display On/Off	Display Data Format: 8 bits/pixel	16 bits/pixel	ARGB	YC	32 bits/pixel	Super-positioning	Blinking	Size	Scroll-ing	Wrap-around	α-Ratio Plane
Plane 1	√	√*1	√	√	√*2	√*4	α blending/ transparent color/ EOR operation	√	X, Y (as desired)	√	√	√
Plane 2	√	√*1	√	√	√*2	√*4	Same as above	√	Same as above	√	√	√
Plane 3	√	√*1	√	√	√*2	√*4	Same as above	√	Same as above	√	√	√
Plane 4	√	√*1	√	√	√*2	√*4	Same as above	√	Same as above	√	√	√
Plane 5	√	√*1	√	√	√*2	√*4	Same as above	√	Same as above	√	√	√
Plane 6	√	√*1	√	√	√*2	√*4	Same as above	√	Same as above	√	√	√
Plane 7	√	√*1	√	√	√*2	√*4	Same as above	√	Same as above	√	√	√
Plane 8	√	√*1	√	√	√*2	√*4	Same as above	√	Same as above	√	√	√
Back-ground color*3	—	—	—	—	—	—	—	—	—	—	—	—

Notes: 1. Any among the color palettes 1, 2, 3, 4 is selected.
2. YC→RGB conversion can be performed only for the YUV data of the uppermost plane.
3. The data format for background color is RGB: 6·6·6.
4. 32-bit/pixel display involves the use of two planes. For example, planes 1 and 2 are combined to display 32 bits/pixel.
5. Transparent color processing for 32-bit/pixel data is not possible.

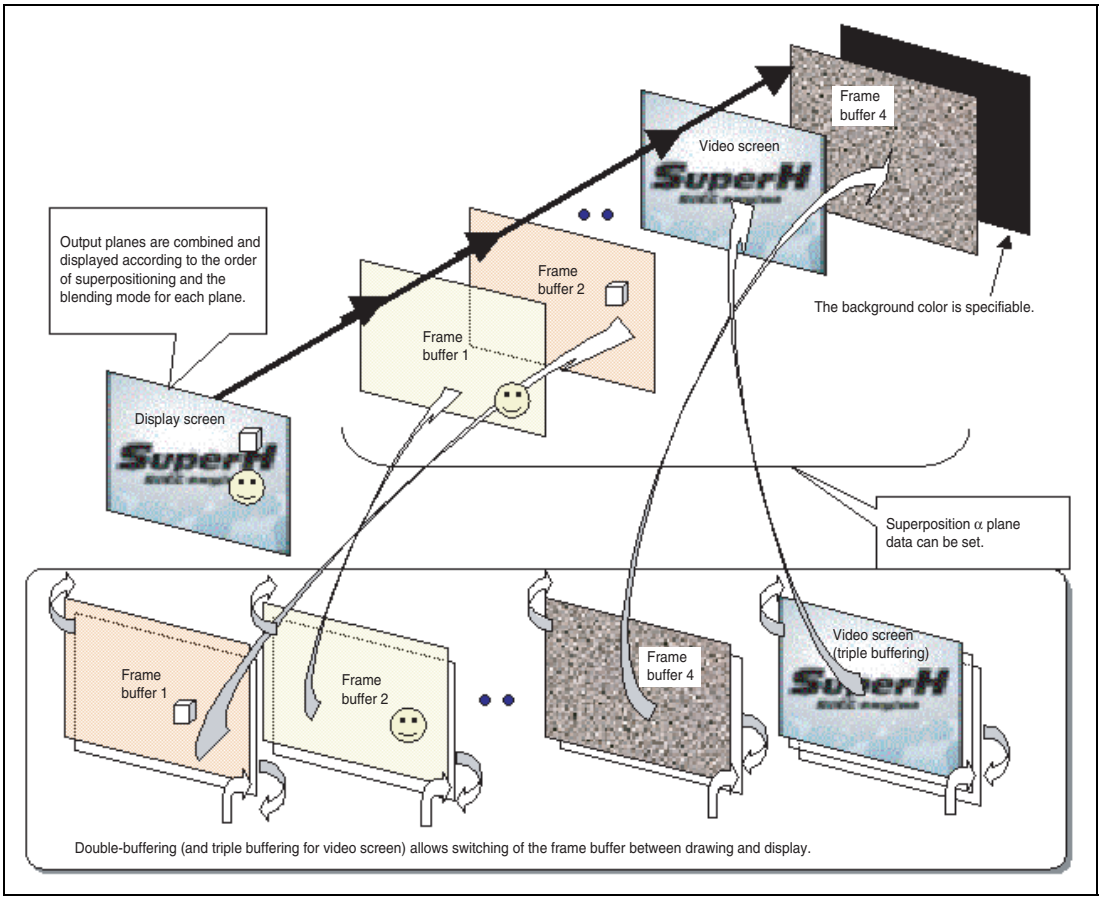


Figure 14.2 Block Diagram of Plane Configuration and Superpositioning

14.4.2 Display On/Off

All plane display can be turned on and off using the DEN bit in DSYSR. When the DEN bit is 0, the display data set in DOOR is displayed.

When the value of bit 0 in DORCR is 0, DPPR is used to turn the display of planes 1 to 8 on and off. When the value of bit 0 in DORCR is 1, on the other hand, DS1PR and DS2PR are used to turn the display of planes 1 to 8 on and off. Under the following display conditions, display data set in BPOR is displayed.

1. When display of all planes 1 to 8 is turned off
2. In an area with no plane for display, due to the display size and display position
3. When the pixels in a plane for display are all a transparent color

Table 14.24 Turning the Display of Planes 1 to 8 On and Off (when Bit 0 of DORCR is 0)

Display Plane	Display Plane Priority Register (DPPR)
Plane 1	Plane 1 is selected in one among priority positions 1 to 8, and the corresponding enable bit is set to 1
Plane 2	Plane 2 is selected in one among priority positions 1 to 8, and the corresponding enable bit is set to 1
Plane 3	Plane 3 is selected in one among priority positions 1 to 8, and the corresponding enable bit is set to 1
Plane 4	Plane 4 is selected in one among priority positions 1 to 8, and the corresponding enable bit is set to 1
Plane 5	Plane 5 is selected in one among priority positions 1 to 8, and the corresponding enable bit is set to 1
Plane 6	Plane 6 is selected in one among priority positions 1 to 8, and the corresponding enable bit is set to 1
Plane 7	Plane 7 is selected in one among priority positions 1 to 8, and the corresponding enable bit is set to 1
Plane 8	Plane 8 is selected in one among priority positions 1 to 8, and the corresponding enable bit is set to 1

Table 14.25 Turning the Display of Planes 1 to 8 On and Off (when Bit 0 of DORCR is 1)

Display Plane	Display Superimpose 1 Priority Register (DS1PR) Display Superimpose 2 Priority Register (DS2PR)
Plane 1	Plane 1 is selected in one among priority positions 1 to 8 (setting to select the plane: 0001)
Plane 2	Plane 2 is selected in one among priority positions 1 to 8 (setting to select the plane: 0010)
Plane 3	Plane 3 is selected in one among priority positions 1 to 8 (setting to select the plane: 0011)
Plane 4	Plane 4 is selected in one among priority positions 1 to 8 (setting to select the plane: 0100)
Plane 5	Plane 5 is selected in one among priority positions 1 to 8 (setting to select the plane: 0101)
Plane 6	Plane 6 is selected in one among priority positions 1 to 8 (setting to select the plane: 0110)
Plane 7	Plane 7 is selected in one among priority positions 1 to 8 (setting to select the plane: 0111)
Plane 8	Plane 8 is selected in one among priority positions 1 to 8 (setting to select the plane: 1000)

Note: Even if display on is set using DPPR, DS1PR, or DS2PR under the following conditions, the setting is handled as display off, and the corresponding plane is not displayed.

- Planes for which the value set in PnDPXR is greater than the screen size (horizontal display end register (HDE) - horizontal display start register (HDS))
- Planes for which the value set in PnDPYR is greater than the screen size (vertical display end register (VDE) - vertical display start register (VDS))
- Planes for which the value set in PnDSXR is 0
- Planes for which the value set in PnDSYR is 0
- Planes for which the value set in PnMWR is 0
- Planes for which the value set in PnSPXR is equal to or greater than twice the value set in PnMWR

14.4.3 Plane Parameter

For each plane, a display area start position, memory width, display start position, and display size are set using registers.

The followings are the schematic diagram of start positions and sizes related to planes and the registers used for setting start positions and sizes.

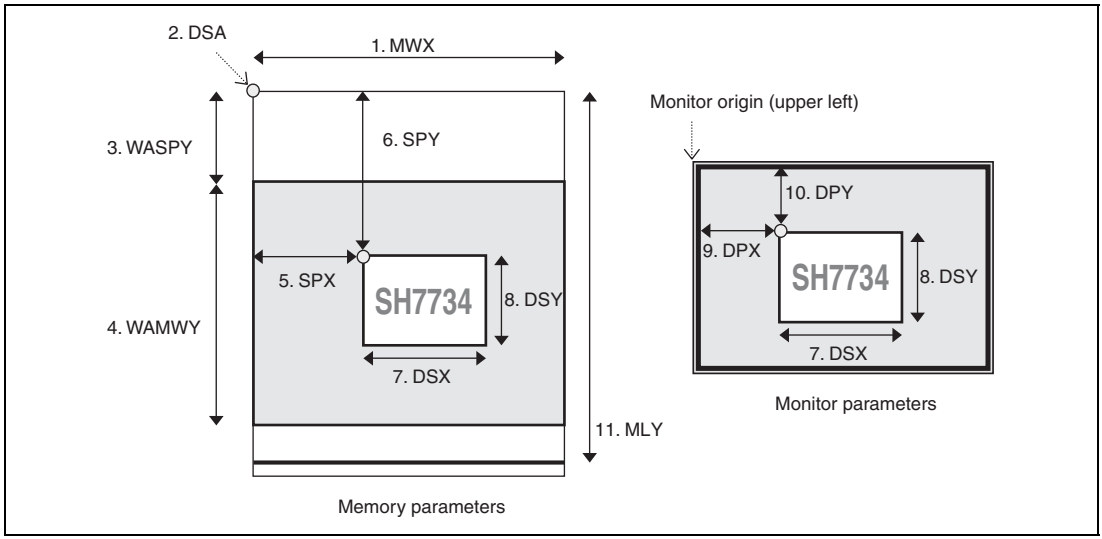


Figure 14.3 Parameters

Table 14.26 Memory Parameter/Monitor Parameter Setting Registers

No.	Name Used in the Figure	Register Abbreviation	Description
1	MWX (Plane memory width)	PnMWR	The plane X-direction memory width is set between 16 and 4096 pixels, in 16 pixel units.
2	DSA (Display area start address)	PnDSA0R to PnDSA2R	The start address in memory area is set for plane.
3	WASPY (Wrap-around start position)	PnWASPR	The Y direction start position of the wrap-around area is set in line units, with the address set by DSA as reference.
4	WAMWY (Wrap-around memory width)	PnWAMWR	The wrap-around Y-direction memory width is set to a desired value in the range from 240 to 4095 (representing line units).
5	SPX (Start position X)	PnSPXR	The distance in the X direction to the display start position is set in pixel units, taking the address set by DSA as the origin.
6	SPY (Start position Y)	PnSPYR	The distance in the Y direction to the display start position is set in raster line units, taking the address set by DSA as the origin.
7	DSX (Display size X)	PnDSXR	The X-direction display size of plane is set in dot-clock units.
8	DSY (Display size Y)	PnDSYR	The Y-direction display size of plane is set in raster line units.
9	DPX (Display position X)	PnDPXR	The X-direction distance to the display position is set in dot-clock units, taking the upper-left corner of the monitor as the origin.
10	DPY (Display position Y)	PnDPYR	The Y-direction distance to the display position is set in raster line units, taking the upper-left corner of the monitor as the origin.
11	MLY (Memory length Y)	PnMLR	The Y-direction memory area of plane is set in line units.

14.4.4 Memory Allocation

A display start address for the display screen, drawing screen 1, and drawing screen 2 used for video display can be set individually for each plane. The start addresses for the memory areas used are set in each of the display area start address registers.

In the display unit (DU), when the display plane is video captured, the display area start addresses 0, 1, and 2 for each plane are used to perform triple-buffering control and display the plane. When the display plane is not video captured, the display area start addresses 0 and 1 for each plane are used to perform double-buffering control and display the plane.

In display capture, only a single area can be set to store the data.

Below is a list of display area start address registers used for each of the planes.

Table 14.27 Memory Allocation Registers

Display Screen	Register Name	
Plane n	Plane n display area start address 0 register	PnDSA0R
	Plane n display area start address 1 register	PnDSA1R
	Plane n display area start address 2 register	PnDSA2R
Display capture	Display capture area start address register	DCSAR

Note: n = 1, 2, 3, 4, 5, 6, 7, or 8

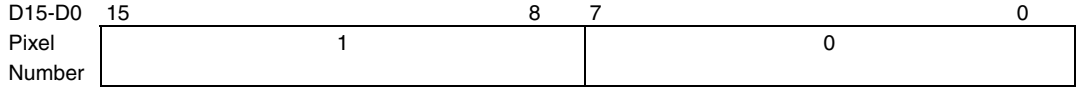
14.4.5 Display Data Format

The following format is used for color data used in display. A data configuration is shown in which data is allocated to the unified memory in little endian.

(1) 8 bits/pixel

A color palette index is used. The color palette is used to convert and display image data into RGB data with 6 bits for each RGB color (RGB666).

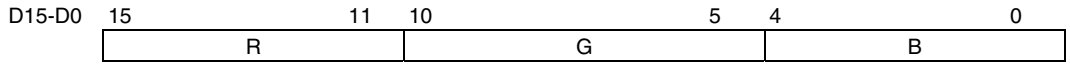
8 bits/pixel data (index color)



(2) 16 bits/pixel: RGB

The RGB levels are represented using 5 bits for R, 6 bits for G, and 5 bits for B (RGB565).

16 bits/pixel data (RGB data) format

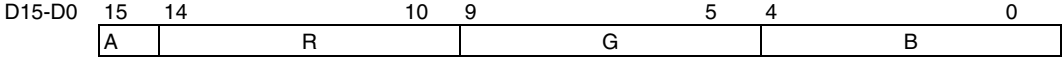


(3) 16 bits/pixel: ARGB

Levels for A, R, G, and B are represented by 1, 5, 5, and 5 bits respectively (ARGB1555). In addition to the R, G, and B values, this format includes a bit that represents α blending control. In this section, "ARGB" indicates ARGB1555 and "ARGB8888" indicates the 32-bit/pixel format unless otherwise specified.

α blending control using the A value is valid when the PnSPIM bits in the plane n mode register (PnMR) are set to perform blending. When the PnABIT bits of the plane n blending ratio register (PnALPHAR) are 00, α blending is performed when A = 1. When the PnABIT bits are 01, α blending is performed when A = 0. When the PnABIT bits are 10 or 11, α blending is performed regardless of the A value. When the PnSPIM bits are not set to perform blending, blending is not performed regardless of the A value.

16 bits/pixel data (ARGB data) format



(4) YC: YUV422

Image data has the format YC (YCbCr) = 4:2:2. A calculation circuit is used to convert each of the 8 bits of the RGB colors (RGB888) of image data. Transparent color processing is not possible.

The YC data order corresponds to the UYVY format and YUYV format. The UYVY format and YUYV format can be selected using the PnYCDF bits in PnMR.

The formulae for YC-RGB conversion are given below. The underlined coefficients are defined by the settings in the corresponding registers.

$$R = \underline{YNC} \times (Y - \underline{YNO}) + \underline{RCRC} \times (Cr - \underline{CRNO})$$
$$G = \underline{YNC} \times (Y - \underline{YNO}) + \underline{GCRCR} \times (Cr - \underline{CRNO}) - \underline{GCBC} \times (Cb - \underline{CBNO})$$
$$B = \underline{YNC} \times (Y - \underline{YNO}) + \underline{BCBC} \times (Cb - \underline{CBNO})$$

The formulae for YC-RGB conversion in the default state are thus as follows.

$$R = Y + 1.37 \times (Cr - 128)$$
$$G = Y - 0.698 \times (Cr - 128) - 0.336 \times (Cb - 128)$$
$$B = Y + 1.73 \times (Cb - 128)$$

(a) UYVY format

	63	56	55	48
D63-D48				
Image data 3 and 4	Y3		V2	

	47	40	39	32
D47-D32				
Image data 3 and 4	Y2		U2	

	31	24	23	16
D31-D16				
Image data 1 and 2	Y1		V0	

	15	8	7	0
D15-D0				
Image data 1 and 2	Y0		U0	

(b) YUYV format

	63	56	55	48
D63-D48				
Image data 3 and 4	V2		Y3	

	47	40	39	32
D47-D32				
Image data 3 and 4	U2		Y2	

	31	24	23	16
D31-D16				
Image data 1 and 2	V0		Y1	

D15-D0	15	8	7	0
Image data 1 and 2	U0		Y0	

(c) RGB color space conversion format

	23	16	15	8	7	0
Image data 4	Y3		U2		V2	

	23	16	15	8	7	0
Image data 3	Y2		U2		V2	

	23	16	15	8	7	0
Image data 2	Y1		U0		V0	

	23	16	15	8	7	0
Image data 1	Y0		U0		V0	

(5) YC: YUV420

Image data has the format YC (YCbCr) = 4:2:0. A calculation circuit is used to convert each of the 8 bits of the RGB colors (RGB888) of image data. Transparent color processing is not possible.

The UV data order corresponds to the NV12 format and NV21 format. The NV12 format and NV21 format can be selected using the PnNV21 bit in PnDDCR2.

The formulas for YC-RGB conversion are the same as those for YC: YUV422.

(a) Y data

D63-D48	63	56	55	48
Image data 8 and 7	Y7		Y6	
D47-D32	47	40	39	32
Image data 6 and 5	Y5		Y4	
D31-D16	31	24	23	16
Image data 4 and 3	Y3		Y2	
D15-D0	15	8	7	0
Image data 2 and 1	Y1		Y0	

(b) UV data (NV12)

	63	56	55	48
D63-D48				
Image data 8 and 7	V6		U6	

	47	40	39	32
D47-D32				
Image data 6 and 5	V4		U4	

	31	24	23	16
D31-D16				
Image data 4 and 3	V2		U2	

	15	8	7	0
D15-D0				
Image data 2 and 1	V0		U0	

(c) UV data (NV21)

	63	56	55	48
D63-D48				
Image data 8 and 7	U6		V6	

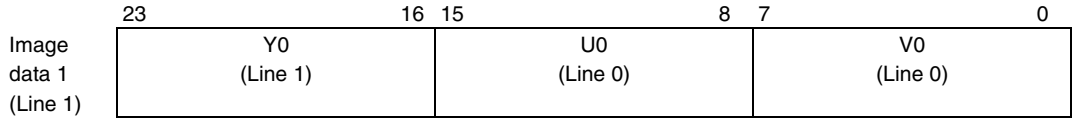
	47	40	39	32
D47-D32				
Image data 6 and 5	U4		V4	

	31	24	23	16
D31-D16				
Image data 4 and 3	U2		V2	

D15-D0	15	8	7	0
Image data 2 and 1	U0		V0	

(d) RGB color space conversion format

	23	16	15	8	7	0
Image data 4 (Line 0)	Y3 (Line 0)		U2 (Line 0)		V2 (Line 0)	
Image data 3 (Line 0)	Y2 (Line 0)		U2 (Line 0)		V2 (Line 0)	
Image data 2 (Line 0)	Y1 (Line 0)		U0 (Line 0)		V0 (Line 0)	
Image data 1 (Line 0)	Y0 (Line 0)		U0 (Line 0)		V0 (Line 0)	
Image data 4 (Line 1)	Y3 (Line 1)		U2 (Line 0)		V2 (Line 0)	
Image data 3 (Line 1)	Y2 (Line 1)		U2 (Line 0)		V2 (Line 0)	
Image data 2 (Line 1)	Y1 (Line 1)		U0 (Line 0)		V0 (Line 0)	



(6) 32 bits/pixel: ARGB8888

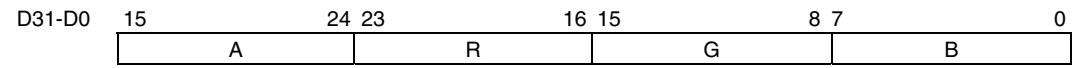
Levels for A, R, G, and B are represented by 8 bits each (ARGB8888). Note that the format includes a setting for alpha ratio in addition to the R, G, and B values. In this section, "32-bit/pixel" refers to the ARGB8888 format unless otherwise specified.

α blending using the A value as the blending ratio is performed when the PnSPIM bits in the plane n mode register (PnMR) are set. This means that the alpha value cannot be selected by the plane n blending ratio register (PnALPHAR).

Transparent color processing for 32-bit/pixel data is not possible.

α blending or EOR operations must be performed in an area where a lower plane exists. If the setting of the PnSPIM bits allows α blending or an EOR operation but there is no lower plane, display of 32-bit/pixel data will not be possible without superposition.

32-bit/pixel data (ARGB8888 data) format



14.4.6 Data Formats for Output and Display Capture

In the case of digital RGB output from a display unit (DU), superpositioning in the form of α blending and EOR operations is performed after the display data format has been expanded into the RGB888 format, and then the data are output. For data capture, the five higher-order bits each from the red and blue pins and the six bits from the green pins are stored as captured display data. The supplementary formats and data formats in the case of expansion to R:G:B = 8:8:8 are as indicated in the following table.

Table 14.28 Output Data Format

		Red								Green								Blue							
		7	**6**	**5**	**4**	**3**	**2**	**1**	**0**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Expanded data	8 bits/pixel (after conversion from a color palette)	R (6 bits)						0	0	G (6 bits)						0	0	B (6 bits)						0	0
	16 bits/pixel	R (5 bits)					0	0	0	G (6 bits)						0	0	B (5 bits)					0	0	0
	ARGB	R (5 bits)					0	0	0	G (5 bits)					0	0	0	B (5 bits)					0	0	0
	YC after RGB conversion	R (8 bits)								G (8 bits)								B (8 bits)							
	32 bits/pixel	R (8 bits)								G (8 bits)								B (8 bits)							
After superpositioning		R (8 bits)								G (8 bits)								B (8 bits)							
DU0 display data		R (8 bits)								G (8 bits)								B (8 bits)							
Display capture		R (5 bits)								G (6 bits)								B (5 bits)							
		R (5 bits)								G (5 bits)								B (5 bits)							

14.4.7 Endian Conversion

The display unit (DU) can perform big-endian/little-endian conversion according to the setting of the DSEC bit in DSYSR. If image data are in 32-bits/pixel format, big-endian/little-endian conversion for the data involves the use of the plane n swap control register (PnSWAP). Then set the DSEC bit in DSYSR to 0.

The internal data format in the display unit (DU) is fixed to little-endian; by setting the DSEC bit in DSYSR to 1, image data arranged in big-endian format in memory are converted into little-endian format and read.

The unit for endian conversion (byte/word) is determined by the setting of the PnDDF bits in PnMR.

Table 14.29 Endian Conversion

PnMR/PnDDF	Data Format	Units for Endian Conversion
00	8 bits/pixel	Byte
01	16 bits/pixel	Word
10	ARGB	Word
11	YC	Byte

Endian conversion in each of the units indicated below is shown in figure 14.4.

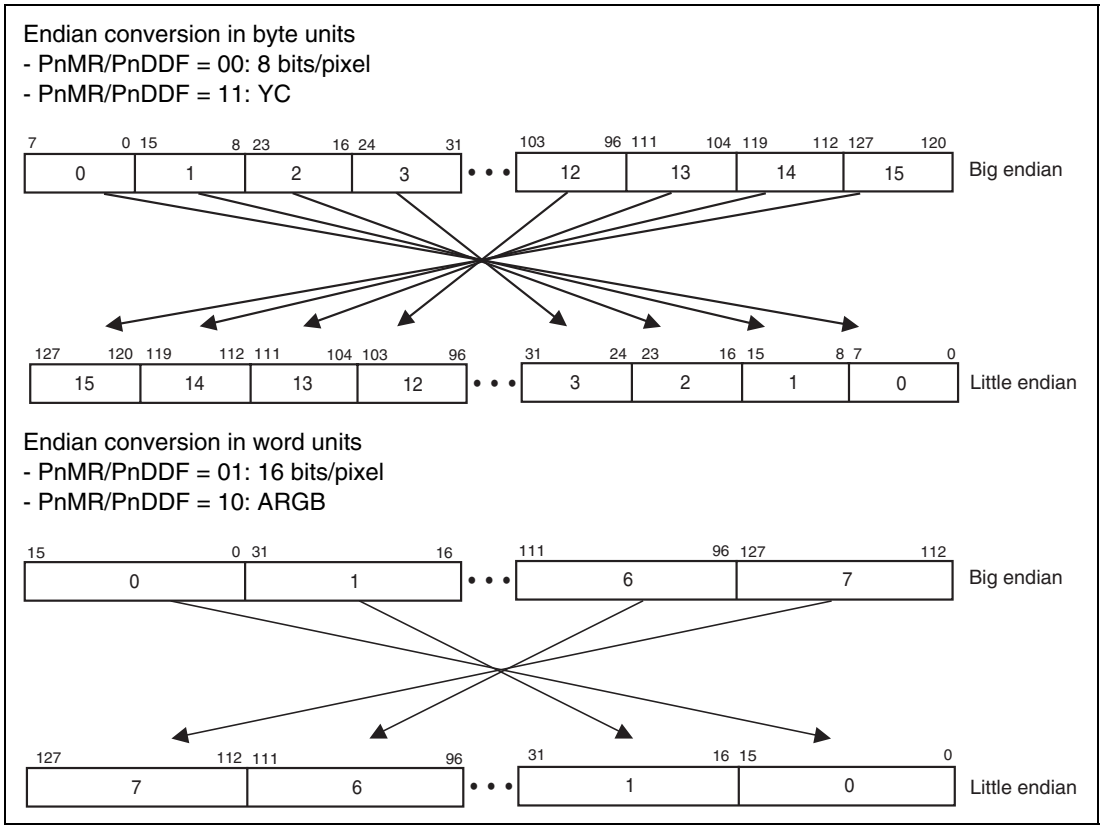


Figure 14.4 Endian Conversion

Since bits 2 and 3 in the plane n swap control register (PnSWAP) are set to 1 and bits 1 and 0 in PnSWAP are set to 0 in the 32-bit/pixel case, the endian must be converted as shown in figure 14.5. Bit 4 in PnSWAP may be set to either 1 or 0. At this time, set the DSEC bit in the display unit system control register (DSYSR) to 0.

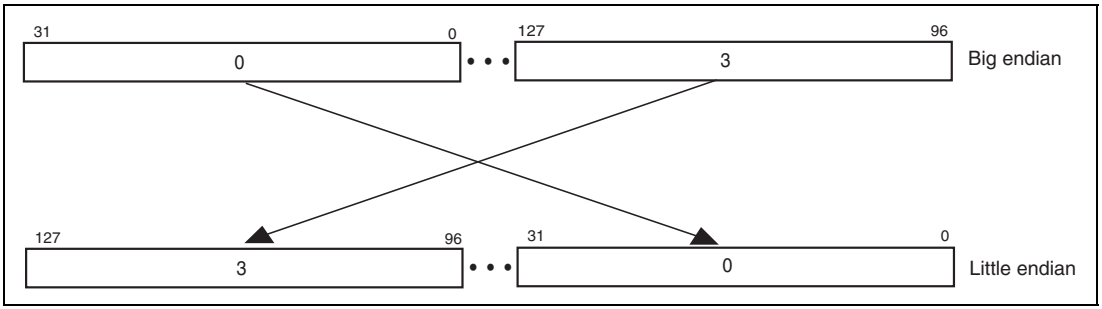


Figure 14.5 Endian Conversion for ARGB888

14.4.8 Color Palette

8 bits/pixel data employs color palettes. The DU has four color palettes which are only accessible to superposition processor 1.

The color palette for use in plane n is determined by the setting of the PnCPSL bits in PnMR. Each of the color palettes consists of two alternate buffers; one serves as a display buffer, and the other is for CPU access. After setting each color palette, by setting the color palette switching enable bit CP4CE, CP3CE, CP2CE, or CP1CE in CPCR to 1, the color palette thus set becomes valid at the next VSYNC falling edge (internal update timing), or upon release from the display reset (when the DRES bit in DSYSR is changed from 1 to 0).

(1) Notes on Use of Color Palettes:

1. Because palettes consist of alternate buffers, complete overwriting is necessary upon a color palette update. However, when the details of color palette updates are being managed, there is no problem with overwriting only the relevant part.
2. Upon completion of color palette settings, the switching enable bit must always be set to 1.
3. When reading a color palette from the CPU, reading should be performed before setting the switching enabled bit to 1.

(2) Procedure for Setting a Color Palette:

(a) Procedure for switching from the initial state

The initial state (after power-on reset) is the display reset state.

1. Set the display control register.
2. Set color palette 1, 2, 3, or 4.
3. After setting the color palette, set the color palette switching enable bit to 1.
4. Cancel the display reset.

(b) Procedure for switching from display state

In the display state, the DRES bit and DEN bit in the DSYSR are 0 and 1 respectively.

1. Confirm that the color palette switching enable bit is 0.
2. Set color palette 1, 2, 3, or 4.
3. After setting the color palette, set the color palette switching enable bit to 1.

14.4.9 Superpositioning of Planes

For each plane, three types of combined superpositioning are possible: α blending, transparent colors, and EOR operations. By setting the PnSPIM bits in PnMR, the superpositioned display type can be selected.

However, α blending and EOR operation cannot be performed simultaneously on the same plane.

Transparent color processing for YC data is not possible.

The following restrictions apply to superpositioning of 32-bit/pixel data.

- Transparent color processing is not possible. Bit 2 of the PnSPIM bits must be set to 1.
- When α blending or an EOR operation has been specified but all of the lower planes are off, the display of 32-bit/pixel data is not possible.
- The alpha value can only be the higher-order eight bits (A value) of the 32-bit/pixel data.

Table 14.30 Superpositioning

PnSPIM	Superpositioning	YC Data or 32-Bit/Pixel Data
000	Transparency processing is performed for the specified plane. When the specified plane is a transparent color, the lower plane is displayed. (Initial value)	Prohibited
001	Blending of the specified plane with the lower plane is performed. When the specified plane is a transparent color, blending is not performed and the lower plane is displayed.	Prohibited
010	EOR operation of the specified plane and the lower plane is performed. When the specified plane is a transparent color, EOR operation is not performed and the lower plane is displayed.	Prohibited
011	Setting prohibited (The lower plane is displayed.)	Prohibited
100	Transparency processing is not performed for the specified plane. The specified plane is displayed.	Possible
101	Blending of the specified plane with the lower plane is performed. Transparent color specification for the specified plane is ignored, and blending of all the pixels in the specified plane with the lower plane is performed.	Possible*
110	EOR operation of the specified plane and the lower plane is performed. Transparent color specification for the specified plane is ignored, and EOR operation of all the pixels in the specified plane and the lower plane is performed.	Possible*
111	Setting prohibited (The lower plane is displayed.)	Prohibited

Note: * Display of 32-bit/pixel data is not possible when all of the lower planes are off.

After the image data format has been expanded to R:G:B = 8:8:8, α blending or EOR operation is performed. The complementary format of each display data format is shown in table 14.28.

α blending and EOR operation are performed in the sequence of the lower plane to the upper plane. The block diagram is shown in figure 14.6.

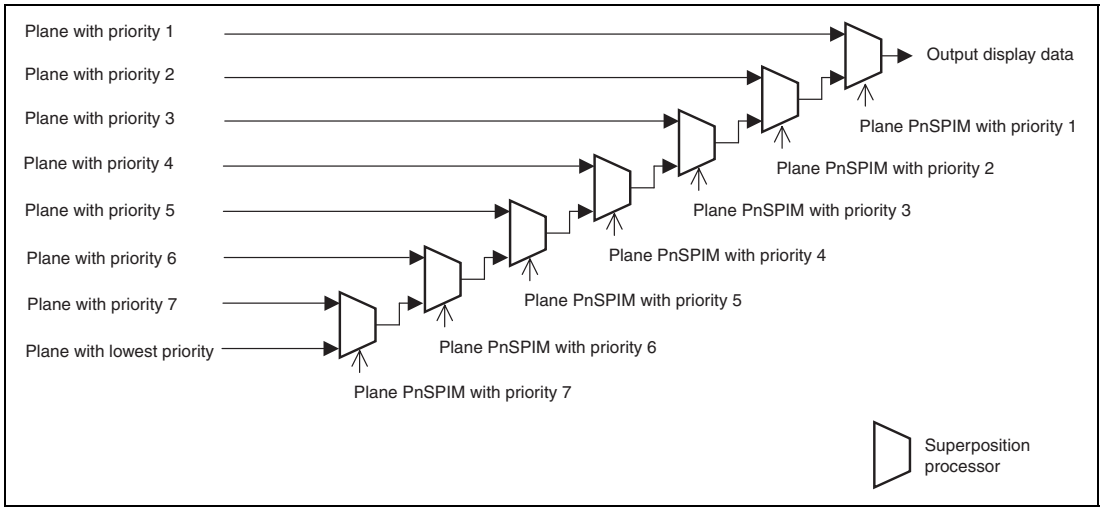


Figure 14.6 Plane Processing Sequence in α Blending and EOR Operation

When the format of display data for α blending or EOR operation is 8 bits/pixel, after selection in advance of the color palette to be used, the α blending or EOR operation on/off should be specified. At this time, when both planes for α blending or for EOR operation have the same color palette selected (color palette contention), only the specified plane is displayed, with no α blending or EOR operation performed. When display of all lower planes is turned off, the specified plane is displayed. That is, blending or EOR operation of the specified plane with the image data specified in BPOR is not performed.

(1) α Blending

In α blending, blending processing is performed according to the alpha (α) ratio set by the PnALPHA bits in PnALPHAR, the alpha (α) ratio set by the blending ratio bits in the color palette, or the alpha (α) ratio of the image data of the display plane. In the case of 32-bit/pixel data, the alpha value can only be the higher-order eight bits (A value) of the data.

$$\text{Result of blending} \approx (\text{specified plane} \times \alpha/255 + \text{lower plane} \times (1 - \alpha/255) \quad \text{(Approximation)}$$

Note: In the above formula, the blending result, α, the specified plane, and the lower plane are all given as 8-bit data.

When H'00 is set as the alpha ratio, only the lower plane is displayed. When H'FF is set as the alpha ratio, only the specified plane is displayed. When the PnDDF bits in PnMR are set to ARGB, and moreover the PnSPIM bits in PnMR are set to perform blending, α blending is performed according to the A value of the ARGB data format and by the alpha value specified by the PnALPHA bits in PnALPHAR.

When the PnABIT bits in PnALPHAR are 00, α blending is performed when the A value is 1. When the PnABIT bits are 01, α blending is performed when the A value is 0. When the PnABIT bits are 10 or 11, α blending is performed regardless of the A value.

(2) Transparent Colors

For each plane, transparent color processing can be performed between the specified plane and the lower plane by setting bit 2 of the PnSPIM bits in PnMR to 0. However, transparent color processing cannot be performed for YC data or 32-bit/pixel data.

- 8 bits/pixel

  When the PnTC bit in PnMR is 0 (initial value), transparent color processing is performed according to the setting in the plane n transparent color 1 register (PnTC1R). When the PnTC bit in PnMR is 1, CPT1R to CPT4R can be used to set up to 16 colors in each of color palettes 1 to 4 as transparent colors. Only the indexes H'00 to H'0F can be specified as transparent colors; H'10 to H'FF cannot be specified as transparent colors.

  The color palette transparent color registers 1 to 4 can be selected using the PnCPSL bits in PnMR.
- 16 bits/pixel or ARGB

  Transparent color processing is performed according to PnTC2R, regardless of the setting of the PnTC bit in PnMR.

  In the case of ARGB, bits 14 to 0 of PnTC2R are compared, and bit 15 is ignored.

The above is summarized in table 14.31, which indicates the transparent color specification registers which are valid when the PnTC bit is 0 or 1.

Table 14.31 Transparent Color Specification Registers

Data Format	Transparent Color Specification Bit	Color Palette Select Bit	Transparent Color Specification Register
—	(PnMR)/PnTC	(PnMR)/PnCPSL	
8 bits/pixel	0	—	PnTC1R
	1	000	CP1TR
	1	001	CP2TR
	1	010	CP3TR
	1	011	CP4TR
16 bits/pixel	—	—	PnTC2R
ARGB	—	—	PnTC2R

(3) EOR Operation

EOR operation of the specified plane with the lower plane is performed.

14.4.10 Contention

(1) Contention for a Color Palette

When the same color palette has been specified for two or more planes in the 8 bits/pixel image data format, contention for that color palette between the planes may arise in α blending and EOR operations. Whether or not contention has occurred is judged on a per-pixel rather than per-plane basis.

In the figure below, planes 1 to 3 are in 8 bits/pixel format with α blending specified for plane 1 and an EOR operation specified for planes 2 and 3. The figure shows the results in cases where contention has arisen because the same color palette has been specified for all of the planes (we assume that none of these planes includes transparent-color pixels).

In the event of contention, α blending and EOR operations do not proceed and the uppermost plane is displayed.

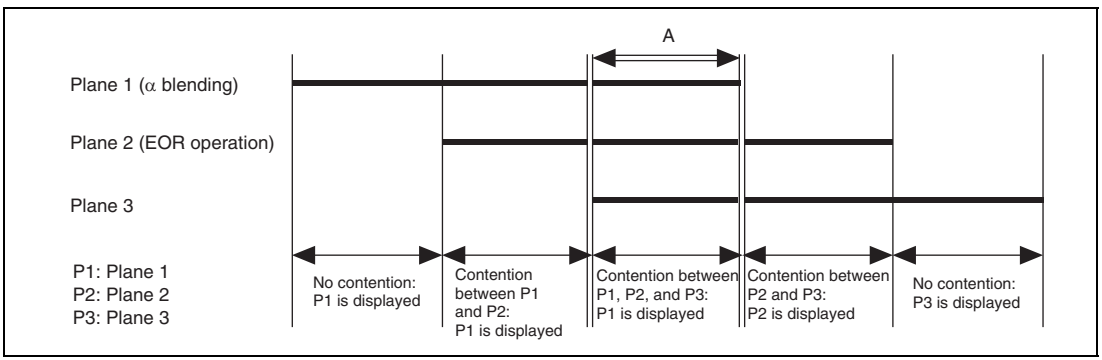


Figure 14.7 Contention for a Color Palette (when Multiple Planes Have the Same Color Palette)

Figure 14.8 shows the results of all patterns of contention and the presence of transparent colors to be displayed during period A in the above figure.

- P1 α P2 indicates α blending of planes 1 and 2.
- P2 ⊙ P3 indicates an EOR operation for planes 2 and 3.
- P1 α (P2 ⊙ P3) indicates α blending of plane 1 with the result of the EOR operation for planes 2 and 3.
- BPOR indicates data specified by the background plane output register (BPOR).

○: Transparent, ●: Non-transparent				✓: Contention, —: No contention				
				P1	✓	✓	✓	—	—
				P2	✓	✓	—	✓	—
				P3	✓	—	✓	✓	—
	P1	P2	P3						
	●	●	●		P1	P1	P1αP2	P1αP2	P1α (P2⊙P3)
	●	●	○		P1	P1	P1αP2	P1αP2	P1αP2
	●	○	●		P1	P1αP3	P1	P1αP3	P1αP3
	●	○	○		P1	P1	P1	P1	P1
	○	●	●		P2	P2⊙P3	P2⊙P3	P2	P2⊙P3
	○	●	○		P2	P2	P2	P2	P2
	○	○	●		P3	P3	P3	P3	P3
	○	○	○		BPOR	BPOR	BPOR	BPOR	BPOR

Figure 14.8 Contention for a Color Palette and Transparent Colors

(2) YC Data Contention

The display unit (DU) has only one YC-RGB conversion circuit internally, and so YC-RGB conversion cannot be performed simultaneously for two or more planes. When there are pixels requiring YC-RGB conversion on two or more planes simultaneously, the pixels on the uppermost plane are YC-RGB converted, and the lower plane is not displayed.
Figure 14.9 describes YC-RGB conversion when the data for three planes is in YC format.

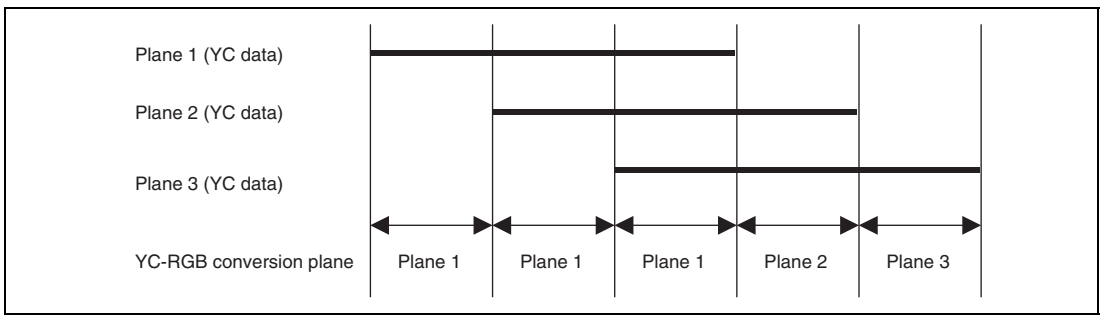


Figure 14.9 YC Data Contention

(3) Plane Priority Order

The display priority order for planes is set using the display plane priority register (DPPR) or the display superimpose 1 priority register (DS1PR) and display superimpose 2 priority register (DS2PR); if one plane is set in two or more places in the priority order, the place with highest priority is selected.
For example, if the setting in DPPR is H'00CB_D888, then the results of the priority order and display on/off settings are as follows.

Plane with priority 1	Plane 1
Plane with priority 2	No corresponding plane
Plane with priority 3	No corresponding plane
Plane with priority 4	Plane 6
Plane with priority 5	Plane 4
Plane with lowest priority	Plane 5
Display off planes	Plane 2, plane 3, plane 7, and plane 8

14.4.11 Blinking

For each plane, blinking operation can be performed by using the display area start addresses 0 and 1.

Usually, double-buffering control is performed for each plane according to the setting of the PnBM bit in PnMR. However, blinking is performed with the period specified by the PnBTA and PnBTB bits in PnBTR by setting the PnBM bits in PnMR to 10 (auto display change mode (blinking mode)). When the blinking period is set to 1, the display area start addresses 0 and 1 can be switched for every VSYNC; the same function as the auto display change mode can be achieved.

Note: Set a value other than 0 to the PnBTA and PnBTB bits in PnBTR.

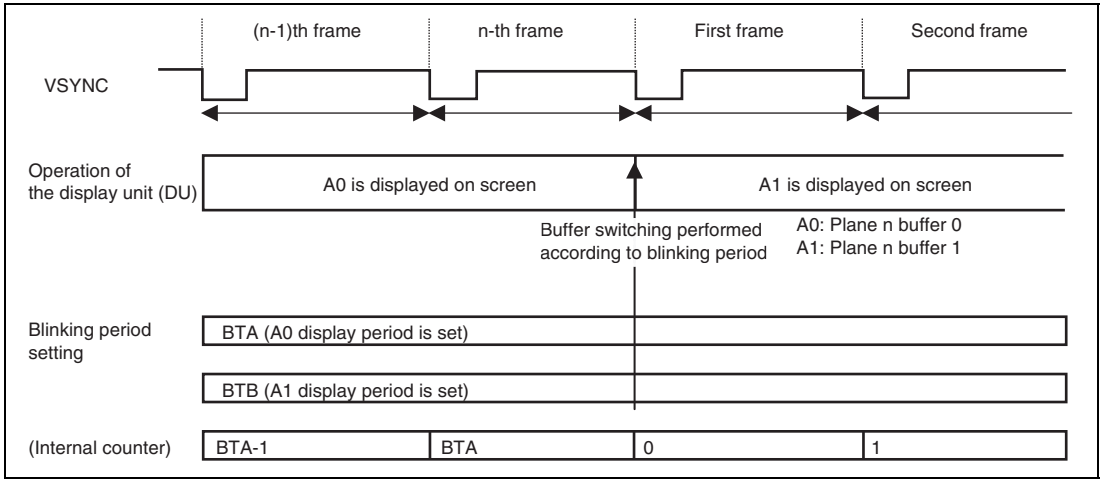


Figure 14.10 Blinking

14.4.12 Scroll Display

By setting display area and display screen sizes and start positions independently for each plane, smooth scroll processing can be performed independently for each plane.

The display can be scrolled by cyclically setting the plane n display start position (coordinates specified by PnSPXR and PnSPYR), taking as the origin the start address in memory specified by PnDSA0R to PnDSA2R for each plane.

Figure 14.11 summarizes display scrolling. The display is scrolled by setting the display start position from A to B.

Note: Display sizes and other area settings for each plane should be set such that there is no data display outside the memory configuration area.

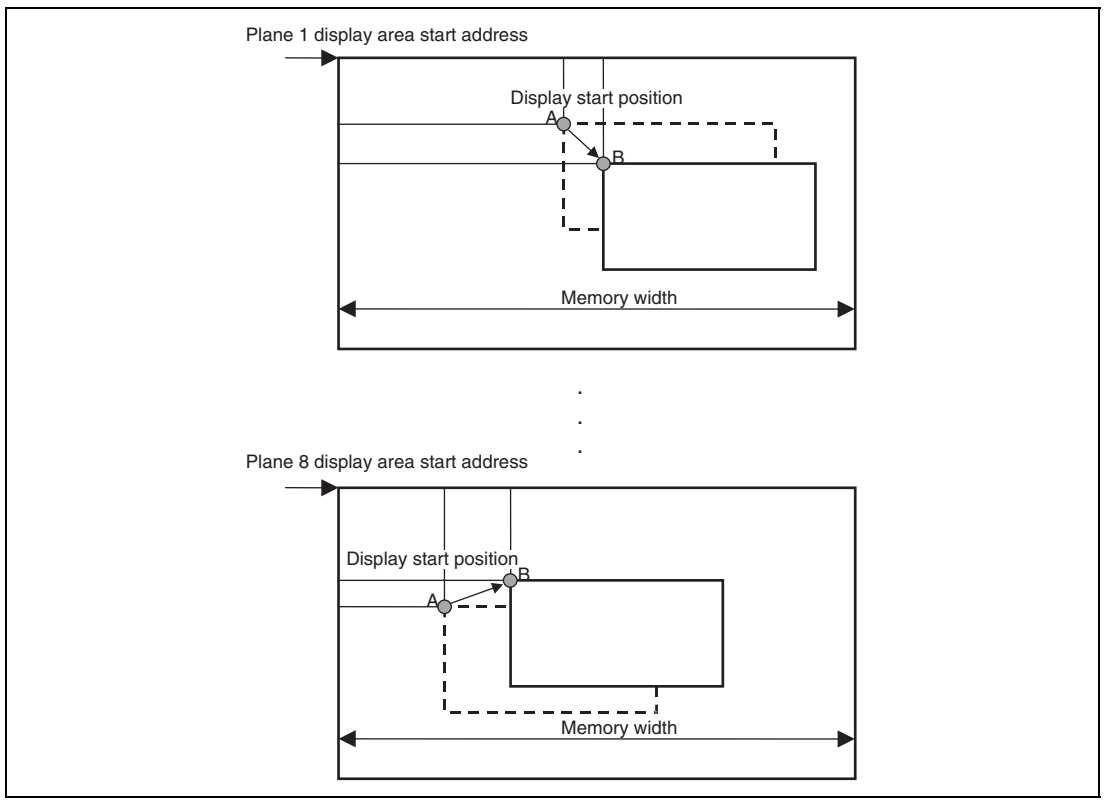


Figure 14.11 Schematic Diagram of Scroll Display

14.4.13 Wrap-Around Display

In addition to display scrolling, wrap-around display, which can be used in spherical scrolling, is possible for each plane. When enabling wrap-around display, the PnWAE bit in the plane n mode register (PnMR) is set to 1. As a result of changing the plane n display start position (coordinates specified by the plane n start position X register (PnSPXR) and plane n start position Y register (PnSPYR)) in order to scroll the display, even when plane n overflows the wrap-around area, the wrap-around area is seen as a spherical surface in wrap-around display, as in figure 14.12, and the part overflowing is complemented and displayed. The method used to specify the wrap-around area is described below.

1. The start address of the memory used for plane n is specified in PnDSA0R to PnDSA2R.
2. With the beginning of the specified memory as origin, the upper-left coordinates of the wrap-around area are specified in PnWASPR. The X-direction width of the wrap-around area is the memory width set in PnMWR.
3. The Y-direction width of the wrap-around area is set in PnWAMWR.

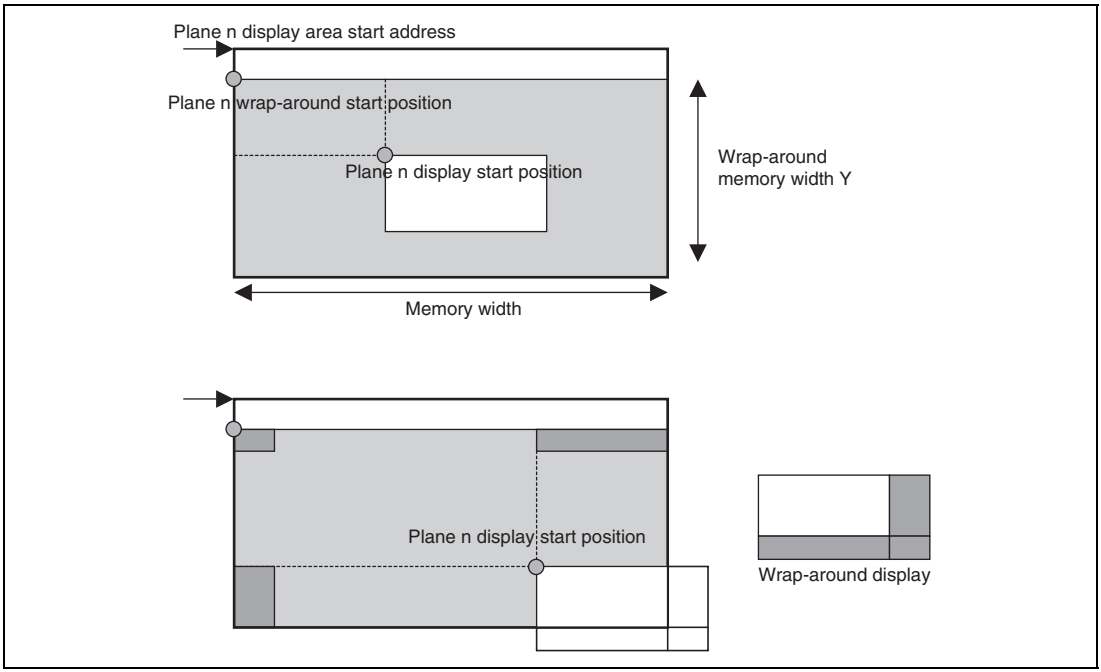


Figure 14.12 Schematic Diagram of Wrap-Around Display

Note: When wrap-around display is disabled (when the PnWAE bit in PnMR is 0), the part overflowing the wrap-around area becomes the color specified by BPOR, and superposition processing using this color is performed.

14.4.14 Upper-Left Overflow Display

For each plane, a display start position in memory (PnSPXR, PnSPYR) and display size (PnDSXR, PnDSYR) can be set arbitrarily, so that by combining and using these registers, areas overflowing the upper-left relative to the monitor origin (upper-left corner) can be displayed without overwriting display data in memory.

For a picture of size (DSX, DSY) and with start position (SPX, SPY), by setting the size to (DSX-ΔX, DSY-ΔY) and the start position to (SPX+ΔX, SPY+ΔY), the ΔX part overflowing on the left side and the ΔY part overflowing on top can be displayed. At this time, the display position (PnDPXR, PnDPYR) is fixed at 0.

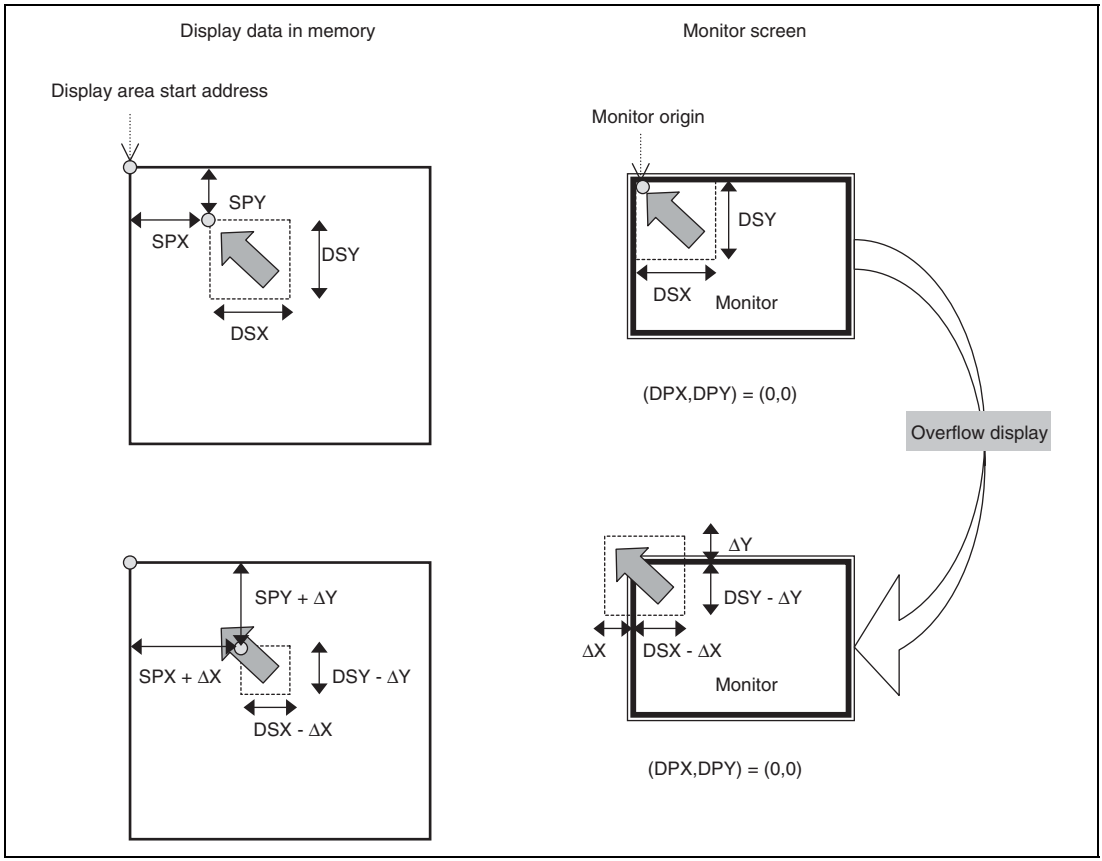


Figure 14.13 Schematic Diagram of Upper-Left Overflow Display

14.4.15 Double-Buffering Control

The double-buffering control of the display unit (DU) includes four types of functions, which are a auto rendering mode that does not switch the display until drawing is completed, a manual display change mode in which display or drawing switching is all controlled by software, an auto display change mode that realize blinking, and a video capture mode that is based on a frame ID of the video input (VIN0, VIN1) module.

In the case of auto rendering mode and manual display change mode, the display change is performed in frame units for non-interlaced and interlaced sync display, and in field units for interlaced sync & video display. In the case of auto display change mode, all switching is performed in field units. For video capture mode, all switching is performed in frame units.

Auto Rendering Mode: In auto rendering mode, display is not switched until drawing is completed. Even if drawing is not completed within a single frame period, drawing operation continues as it is.

Manual Display Change Mode: In manual display change mode, display frame switching and start of drawing are controlled by software. Display switching can either be performed by software using the PnDC bit in PnMR, or by setting the buffer 0 or buffer 1 start address in PnDSA0R and PnDSA1R indicated by the DFBn bit in DSSR.

Start of drawing is controlled by the rendering start bit. The control timing is determined by an interrupt set up by using the VBK bit in the display unit status register (DSSR) or the trap flag. When making a transition from this mode to another mode, the PnDC bit should always be set to 1 first.

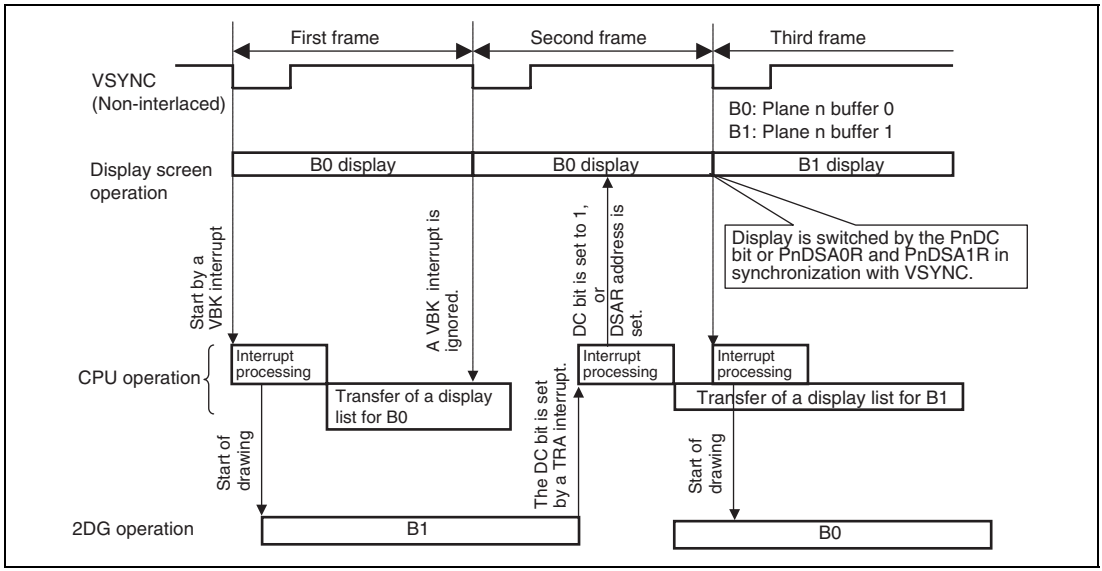


Figure 14.14 Manual Display Change Mode

Auto Display Change Mode: For information on the auto display change mode, refer to section 14.4.11, Blinking.

Video Capture Mode: In video capture mode, a display frame buffer is switched in frame units according to a frame ID, indicating the latest capture frame, output from the video input (VIN0, VIN1) module.

14.4.16 Sync Mode

In order to facilitate synchronization with external equipment, in addition to master mode, a TV synchronization function is provided. Selection of master mode and TV sync mode is performed using the TVM bit in DSYSR. In master mode (internal sync mode), the position of the falling edge of the vertical sync signal (VSYNC) set by VSPR is detected. In TV sync mode (external sync mode), the position of the falling edge of the EXVYNC signal is detected. The results are then reflected in the FRM and VBK bits of DSSR.

Master Mode (Internal Sync Mode): By setting the period and pulse width of the horizontal and vertical sync signals (HSYNC, VSYNC) in the display timing generation registers, the corresponding waveforms are output. Also, display data is output in sync with these signals.

In interlaced sync mode and interlaced sync & video mode, a signal is output to the ODDF pin indicating odd/even fields.

TV Sync Mode (External Sync Mode): In TV sync mode, display data is output in sync with a horizontal sync signal and vertical sync signal (EXHSYNC, EXVSYNC) input from a TV, video, or other external sync signal generation circuit. Display data is output with reference to the falling edge of the EXHSYNC signal and the rising edge of the EXVSYNC signal.

The horizontal sync signal, vertical sync signal, and clock signal from the external sync signal generation circuit are input to the EXHSYNC, EXVSYNC, and DCLKIN pins, respectively. CSYNC is at high level. In interlaced sync mode and interlaced sync & video mode, a signal should be input to the EXODDF pin indicating odd/even fields. In non-interlaced mode, the input to the EXODDF pin should be fixed at low level or at high level.

When operating the unit in TV sync mode also, values must be set in HCR, HSWR, VCR, and VSPR (display timing generation registers given in section 14.3.2, Display Timing Generation Registers).

When the EXVSYNC signal is input, either before or after completion of display of the display size portion set in the display unit (DU), the display unit (DU) performs vertical display completion operation and transitions to control for the next screen. When the EXVSYNC signal is not input, the unit continues to wait for the EXVSYNC signal while remaining in the vertical blanking interval (auto-control is not performed). Similarly, when the EXHSYNC signal is input the display unit (DU) performs horizontal display completion operation and transitions to control for the next raster line; but if the EXHSYNC signal is not input, the unit continues to wait for the EXHSYNC signal while remaining in the horizontal blanking interval (auto-control is not performed).

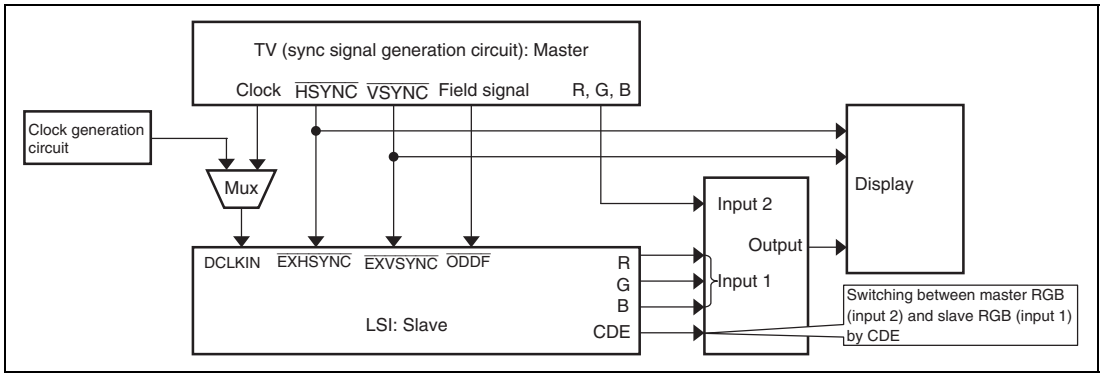


Figure 14.15 Signal Flow in TV Sync Mode

Sync Method Switching Mode: When switching from master mode into TV sync mode, or from TV sync mode into master mode, when necessary this mode should be switched into first. Even if a transition to this mode is not made first, switching of the synchronization method is possible.

In this mode, input/output pins connected to the display unit (DU) are for input, and so collision of pin signals can be avoided. Also, in this mode the internal dot clock is stopped, so that disorder in the input dot clock has no effect on display operation.

14.4.17 Display Capture

In display capture, display data (RGB666) that have been composed for output on the relevant pins are converted to RGB565 or ARGB1555 data, stored in a buffer having the same configuration as the read buffer, and then stored in the area specified by DCSAR via the SuperHyway. To capture the display data, set bit 0 of DCPCR to 1. Capturing starts from the next frame after this setting has been made. The specifications for display capturing are as follows:

Data formats: Refer to table 14.28, Output Data Format. In the case of ARGB1555, the A value is determined by the display capture control register (DCPCR).

Capture area start address: Only a single address can be specified.

Capture size X: The same as the monitor size (horizontal display end position (HDE) – horizontal display start position (HDS)

Capture size Y: The same as the monitor size (vertical display end position (VDE) – vertical display start position (VDS)

Memory width X: Specified by a register. Writing to the buffer ends when the memory width is exceeded.

Memory length Y: Specified by a register. Storage of data in the memory ends when the memory length (in lines) is exceeded.

14.4.18 Display Output Compare

The display output compare function calculates a CRC value for a specified plane area and compares it with the expected CRC value calculated in advance. From this result, whether the current display output is as expected can be checked. The following items can be specified for this function.

- A desired plane from planes 1 to 8
- A desired rectangular area in the selected plane
- 32-bit/pixel format (ARGB8888, RGB888, or RGB666) or 16-bit/pixel format (RGB565) (Other formats are not supported)
- Interrupt to be issued when an unmatched value is found as a result of comparison

Figure 14.16 shows the structure of the display output compare unit. As shown in the figure, plane data is compared before superposition.

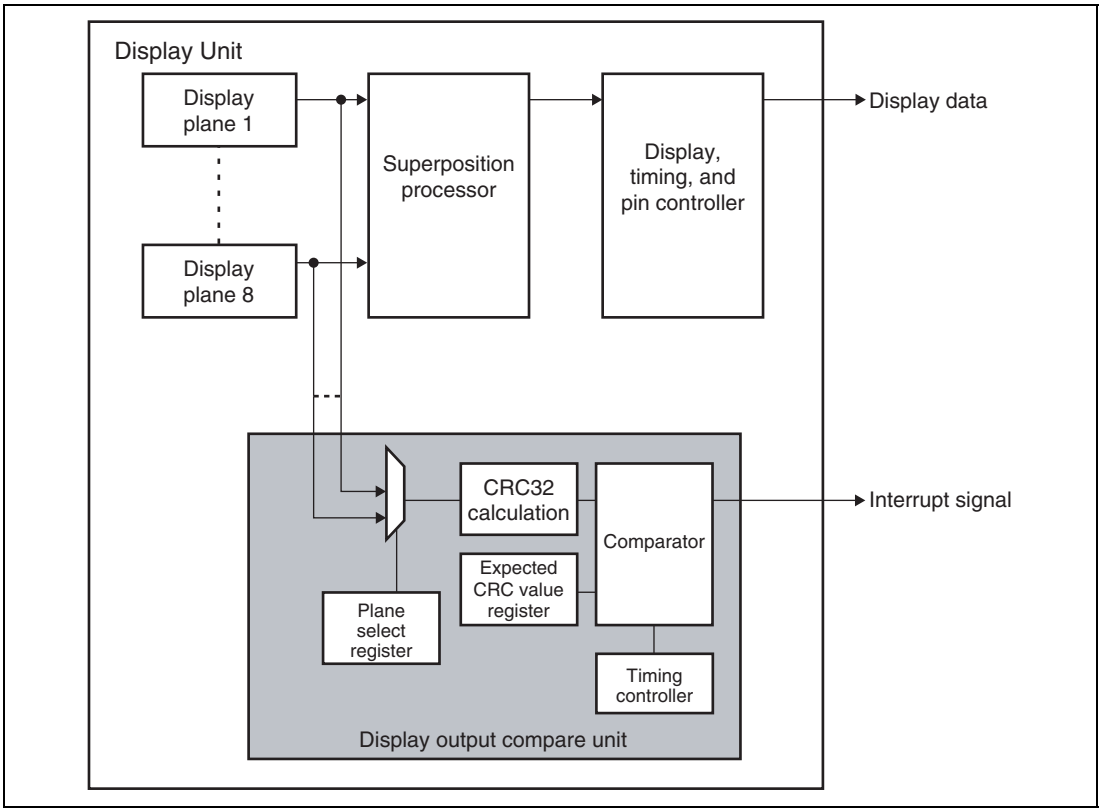


Figure 14.16 Block Diagram of Display Output Compare Unit

(1) CRC Calculation Method

The display output compare unit calculates a 32-bit CRC value through the following expression (the polynomial defined in RFC2083).

$$x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$$

Initial value: H'FFFFFFFF; the bits are inverted at output (after all data calculation has finished).

A CRC is calculated in pixel units assuming that bits are input in order from the LSB to the MSB; that is, a CRC is calculated in 32-bit units in the 32-bit/pixel format and in 16-bit units in the 16-bit/pixel format. Pixels are processed for CRC calculation in the order of scanning (top left to bottom right).

(a) Specifying Plane and Display Color

Use the display output compare parameter register (DOCMPMR1 or DOCMPMR2) to specify a plane. Table 14.32 shows a sample setting for each display color format in the display output compare parameter register (DOCMPMR1 or DOCMPMR2). (Note that the display unit does not support RGB888 and RGB666, so use the settings shown in table 14.32.)

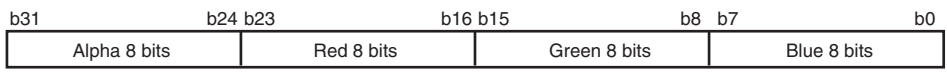
Table 14.32 Sample Settings for Display Output Compare Parameter Register

		Bits in DOCMPMR1 or DOCMPMR2			
Format		**CMPDFF**	**CMPCVF**	**CMPDAUF**	**CMPDFA**
32 bits/pixel	ARGB8888	0	—	0	—
	RGB888	0	—	0 or 1*	Any value
	RGB666	1	Any value	0 or 1*	Any value
16 bits/pixel	RGB565	—	—	—	—

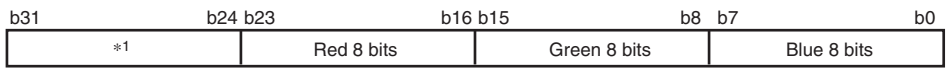
Note: * When CMPDAUF = 0, the value read from RAM is used as the alpha value.
For the bits where "Any value" is indicated, specify the value used to calculate the expected CRC value.

(b) Pixel Format Used for CRC Calculation

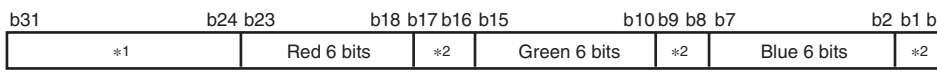
• ARGB8888 (32 bits/pixel)



• RGB888 (32 bits/pixel)

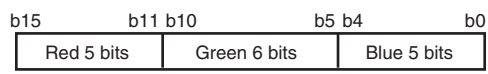


• RGB666 (32 bits/pixel)



Notes: 1. When CMPDAUF = 0, the value read from RAM is used.
When CMPDAUF = 1, the CMPDFA value is used.
2. Value specified in CMPCVF.

• RGB565 (16 bits/pixel)



(c) Specifying Target Area

The target area to be compared should be specified by setting the comparison start positions and comparison sizes in the respective registers.

The following shows the relationship between the area-specifying parameters and the parameter-specifying registers.

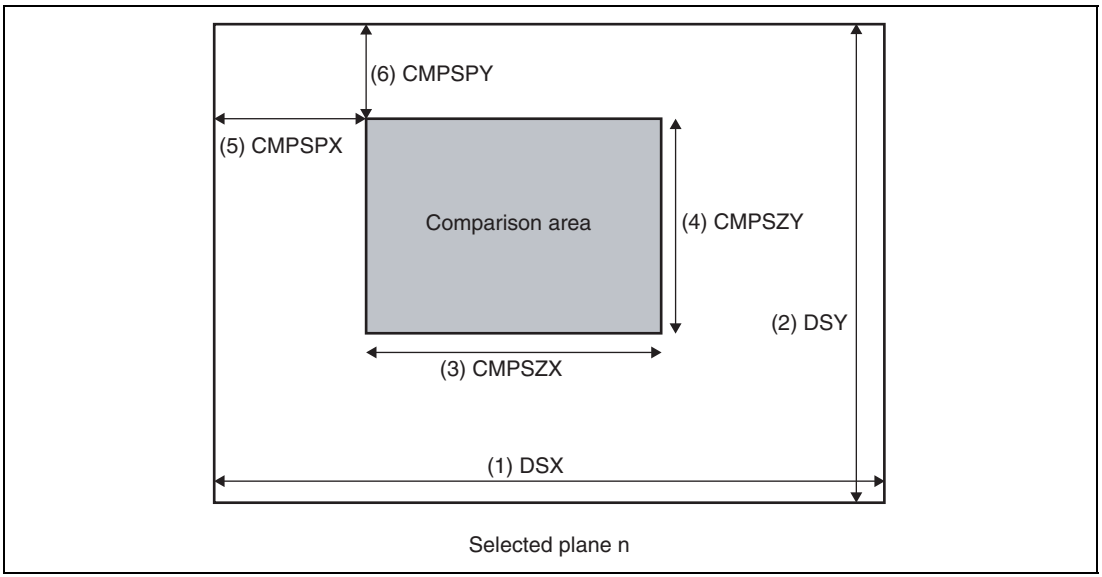


Figure 14.17 Parameters for Specifying Comparison Area

Table 14.33 Registers for Specifying Comparison Area Parameters

No.	Parameter Name in Figure	Register Name	Description
(1)	DSX (display size X)	PnDSXR	Specifies the X-direction display size of the plane in dot clock cycle units. (This is a display plane register.)
(2)	DSY (display size Y)	PnDSYR	Specifies the Y-direction display size of the plane in raster line units. (This is a display plane register.)
(3)	CMPSZX (comparison size X)	DOCMSZXR1 DOCMSZXR2	Specifies the X-direction distance from the top left corner (origin) of the plane to the comparison start position in dot clock cycle units. Satisfy DSX ≥ CMPSPX + CMPSZX.
(4)	CMPSZY (comparison size Y)	DOCMSZYR1 DOCMSZYR2	Specifies the Y-direction distance from the top left corner (origin) of the plane to the comparison start position in line units. Satisfy DSY ≥ CMPSPY + CMPSZY.
(5)	CMPSPX (comparison start position X)	DOCMSPXR1 DOCMSPXR2	Specifies the X-direction size of the comparison area in dot clock cycle units.
(6)	CMPSPY (comparison start position Y)	DOCMSPYR1 DOCMSPYR2	Specifies the Y-direction size of the comparison area in line units.

(2) CRC Comparison Timing

The following shows an example of comparison execution timing.

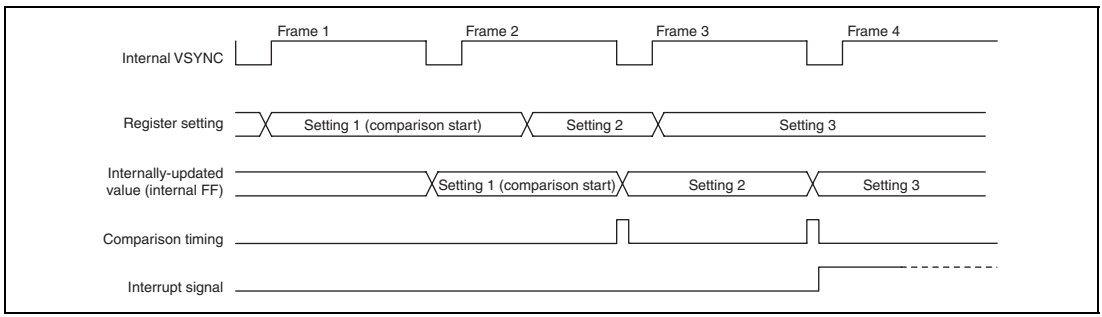


Figure 14.18 Example of CRC Comparison Timing

The display output compare unit operates as follows in the example shown in figure 14.18.

- Frame 1

  When the start of comparison is specified (CMPR1 = 1) in frame 1, comparison is not done within frame 1. This is because the display output compare unit provides a double control for register values; the values are latched into the internal FFs at the falling edge of the internal VSYNC signal and then the latched values are used for actual operation (the double control is provided in the display output control registers having the internal update function).

  Note that only the values in the display output compare expected CRC value register (DOCMECRCR1 or DOCMECRCR2) can be latched into the corresponding internal FF with the register write timing by setting the CRCUMD bit to 1 in the display output compare mode register (DOCMMDR1 or DOCMMDR2). In this case, be sure to complete settings by one line before the VSYNC falling edge.
- Frame 2

  In frame 2, since the display output compare unit uses the values latched at the VSYNC falling edge in frame 1, it calculates a CRC according to setting 1 and compares CRCs at the VSYNC falling edge in frame 2. Values are latched into the internal FFs after the comparison timing, so comparison is not done at the VSYNC falling edge in frame 1. In this example, the display output compare unit latches setting 2 into the internal FFs after CRC comparison.
- Frame 3

  In frame 3, the display output compare unit calculates a CRC according to setting 2 and compares CRCs at the VSYNC falling edge in frame 3. Here, when interrupts are enabled in setting 2 and a CRC difference is detected as a result of comparison, an interrupt signal is asserted. This assertion continues until the status register or interrupt enable register is cleared.

This double control of set values enables change of the plane and comparison area in every frame.

14.4.19 Divided YUV Display

An image that has been divided up into separate Y and UV data for separate storage in memory can be displayed as YUV422 or YUV420 data by using two planes.

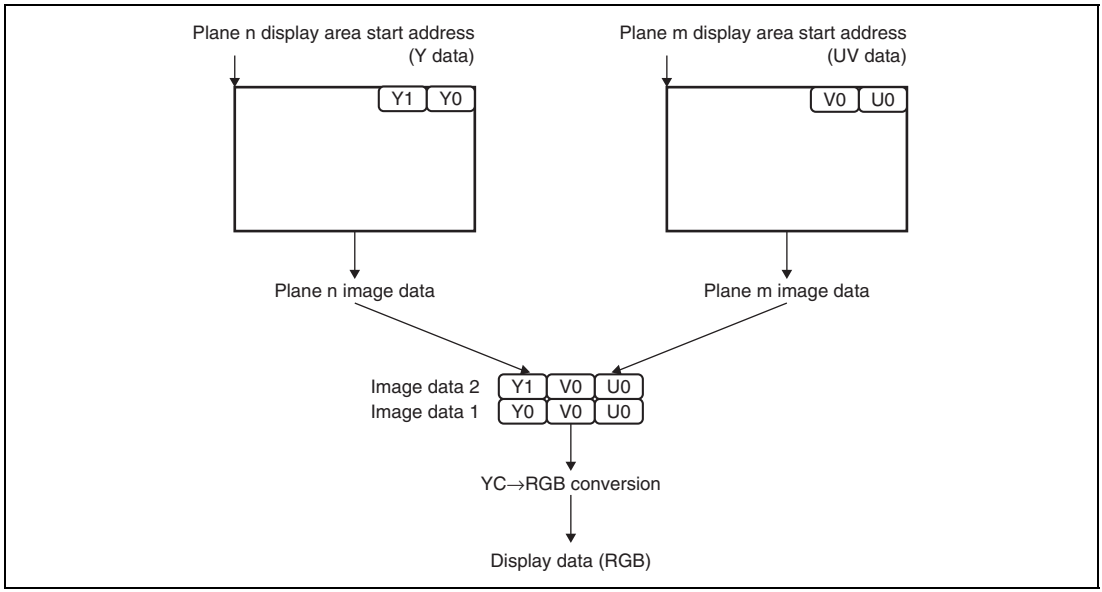


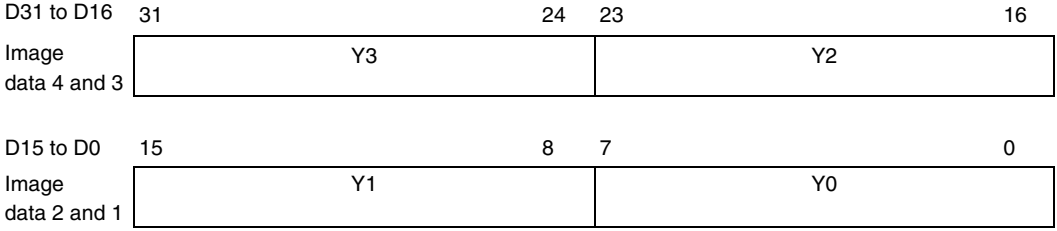
Figure 14.19 Overview of Divided YUV Display Function (Little Endian)

(1) Image Data Format

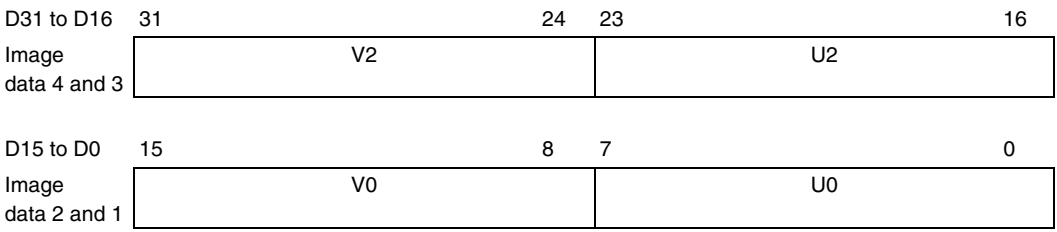
The following figures show the configuration of YUV422 data divided for separate storage in little endian in memory. The configuration of YUV420 data is the same as that of YC: YUV420.

The formulas for YC-RGB conversion are the same as those for YC: YUV422 (section 14.4.5 (4)).

(a) Y data



(b) UV data



(2) Register Setting Examples

Two planes are used to display the divided YUV data. The methods for specifying Y data in plane 1 and UV data in plane 2 to display the divided YUV data are described below.

1. The following is the setting made by the plane n display data control register 2 (PnDDCR2).

- YUV422 data

Bit:	7	6	5	4	3	2	1	0
			PnNV21	PnY420			PnDIVU	PnDIVY
Plane 1 (Y data)	–	–	0	0	–	–	0	1
Plane 2 (UV data)	–	–	0	0	–	–	1	0

- YUV420 data

Bit:	7	6	5	4	3	2	1	0
			PnNV21	PnY420			PnDIVU	PnDIVY
Plane 1 (Y data)	–	–	0	0	–	–	0	1
Plane 2 (UV data)	–	–	0/1	1	–	–	1	0

2. Set the start address of Y data in the plane 1 display area start address 0 to 2 registers (P1DSA0R to P1DSA2R).
3. Set the start address of UV data in the plane 2 display area start address 0 to 2 registers (P2DSA0R to P2DSA2R).
4. The same settings should be made in the plane 1 and plane 2 registers of display plane registers other than the registers described above. Set the PnDDF bits in the plane n mode register (PnMR) to 11 (YC).
5. Set the DSEC bit in the display unit system control register (DSYSR) to 1 if data is stored in big endian in memory.
6. Turn on the display of planes 1 and 2, and give plane 2 the next order of priority for processing after plane 1. For details on turning on the display, see section 14.4.2, Display On/Off.

(3) Combinations of Planes

The combinations of planes that can specify Y data and UV data are listed below. UV data should be on the plane immediately below that for the Y data.

Table 14.34 Possible Combinations of Planes for Specifying Y Data and UV Data

Y Data Obtained by Dividing up YUV	UV Data Obtained by Dividing up YUV
Plane 1	Plane 2
Plane 2	Plane 3
Plane 3	Plane 4
Plane 4	Plane 5
Plane 5	Plane 6
Plane 6	Plane 7
Plane 7	Plane 8
Plane 8	Plane 1

14.5 Display Control

14.5.1 Display Timing Generation

In the display unit (DU), display timing is generated for the horizontal direction and vertical direction of the display screen. Display timing is set by using display timing generation registers in section 14.3.2, Display Timing Generation Registers. Figure 14.20 shows the display timing in non-interlaced mode. Here, the display screen is defined in terms of the variables of table 14.35.

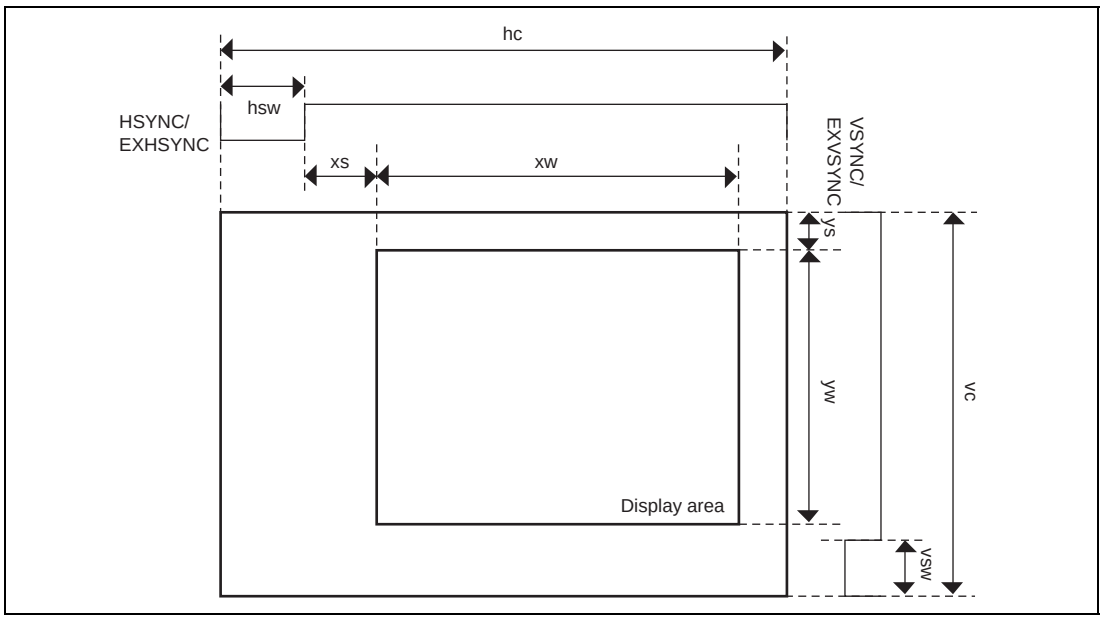


Figure 14.20 Display Timing Generation for Horizontal Direction and Vertical Direction of Display Screen

Table 14.35 Variables Defined in Display Screen

Variables	Contents	Units
hc*[1]	Horizontal scan period	Dot clock
hsw	Horizontal sync pulse width	Dot clock
xs	From rise of HSYNC to display start position in the horizontal direction of the display screen	Dot clock
xw	Display width per 1 raster of display screen	Dot clock
vc*[2]	Vertical scan period	Raster line
vsw	Vertical sync pulse width	Raster line
ys	From rise of VSYNC to display start position in the vertical direction of the display screen	Raster line
yw	Vertical display period of display screen	Raster line

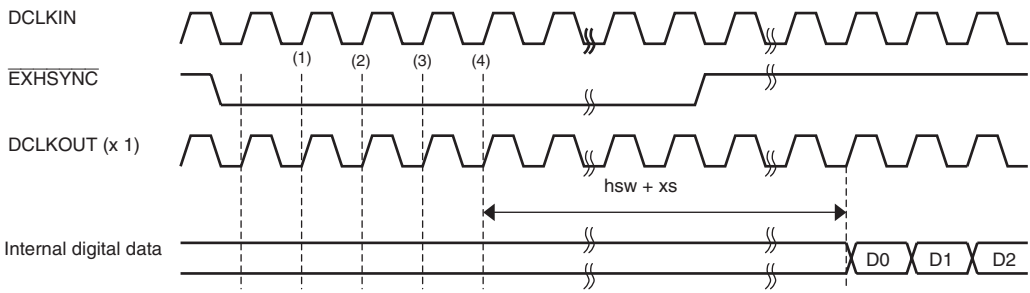
Notes: 1. Should be set such that hsw + xs + xw < hc
2. Should be set such that vsw + ys + yw < vc

The display timing generation register settings are different depending on the scan method and synchronization method. Hence the value of the display timing generation registers should be set after performing calculations like those indicated in table 14.36.

Table 14.36 Correspondence Table of Settings of Display Timing Generation Registers

		Synchronization Method	
Register Name	**Bit Name**	**Master Mode**	**TV Sync Mode**
Horizontal display start register (HDSR)	HDS	hsw + xs - 19	hsw + xs – 25*2
Horizontal display end register (HDER)	HDE	hsw + xs – 19 + xw	hsw + xs – 25 + xw*2
Vertical display start register (VDSR)	VDS	ys – 2*3	ys – 2*3
Vertical display end register (VDER)	VDE	ys – 2 + yw	ys – 2 + yw
Horizontal sync width register (HSWR)	HSW	hsw - 1	hsw - 1
Horizontal cycle register (HCR)	HC	hc - 1	hc - 1
Vertical sync point register (VSPR)	VSP	vc – vsw - 1	vc – vsw - 1
Vertical cycle register (VCR)	VC	vc - 1	vc - 1

Notes: 1. In all scan modes, VDS, VDE, VSP, VC settings are in single-field units.
2. The values of HDS and HDE are from the fourth rising edge of DCLKOUT after detection of the falling edge of EXHSYNC through the rising edge of DCLKOUT.



3. VDS should be set to 1 or greater.
4. HC should be set so as to satisfy HC > HDE.

14.5.2 CSYNC

In master mode, a CSYNC (composite sync) signal is output. EQWR is used to set the low-level pulse width of the CSYNC equivalent pulse. SPWR is used to set the low-level pulse width of the CSYNC serration pulse.

The CSYNC waveform is selected using the CSY (CSYNC mode) bit in DSMR.

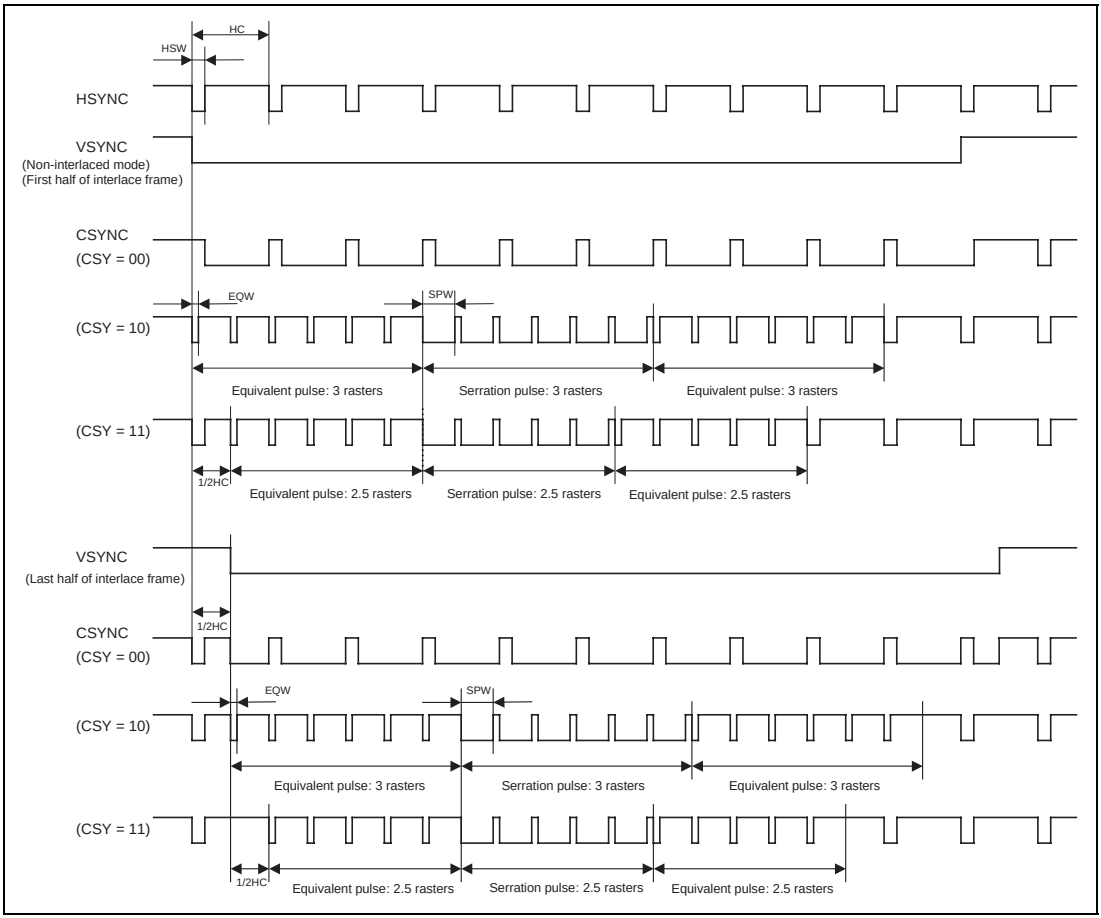


Figure 14.21 CSYNC Timing Chart

14.5.3 Scan Method

The scan method can be selected from among non-interlaced mode, interlaced sync mode, and interlaced sync & video mode. The mode is selected using the SCM bit in DSYSR.

- Non-interlaced mode
  In this scan method, one frame consists of a single field.
- Interlaced sync mode
  In this scan method, one frame consists of two fields. The two fields are an even field and an odd field, displaying the same data.
- Interlaced sync & video mode
  In this scan method, one frame consists of two fields. The two fields are an even field and an odd field, displaying different data.

The ODEV bit in DSMR is used to set the display order of fields in interlaced sync mode and in interlaced sync & video mode. When the ODEV bit is 0, the display order for one frame is odd field, then even field; when the ODEV bit is 1, the order for one frame is even field, then odd field.

In master mode, high level is output from the ODDF pin during even field display, and low level is output during odd field display. In TV sync mode, high level is input to the EXODDF pin to cause display of the even field, and low level is input to cause display of the odd field.

Note: When non-interlaced mode is selected in TV sync mode, the EXODDF pin should be fixed at low level or high level.

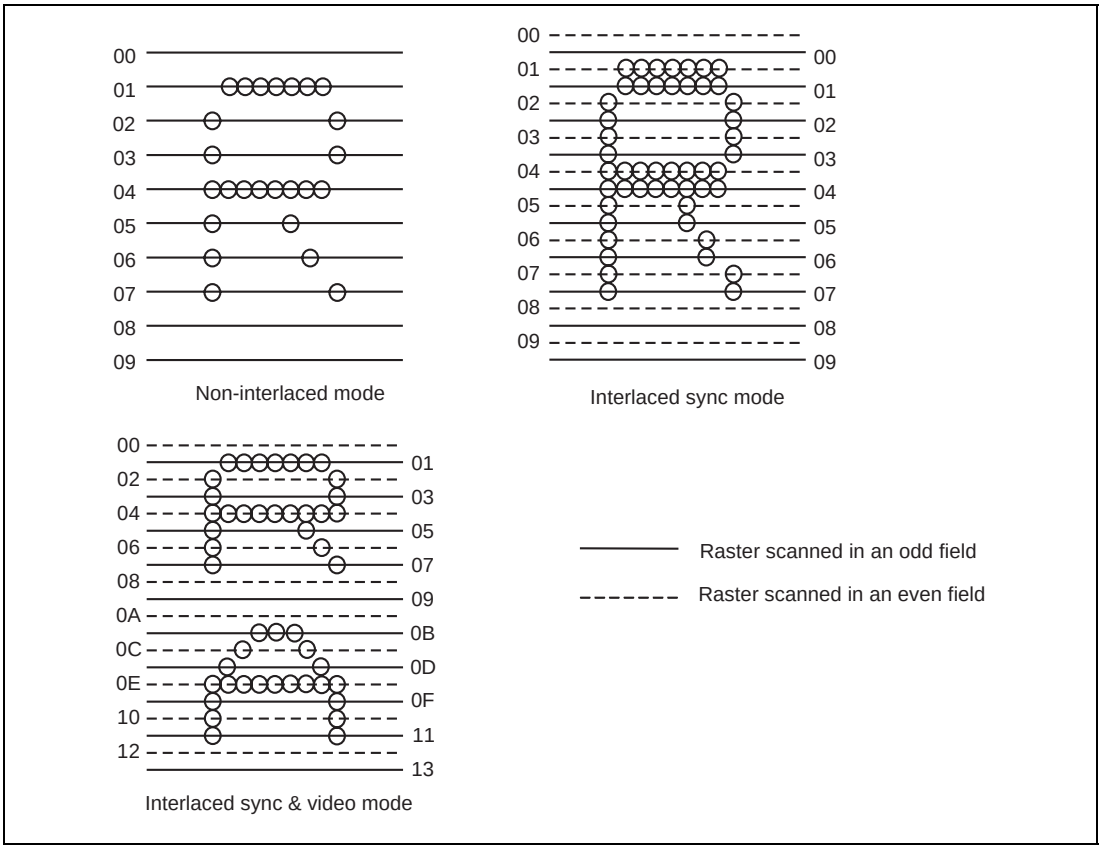


Figure 14.22 Example of Display in Each Scan Mode

- Example of vertical scan period

  Non-interlaced mode: 1/60 second/field, 1/30 second/field

  Interlaced sync mode: 1/30 second/frame

  Interlaced sync & video mode: 1/30 second/frame

- Display in non-interlaced method

  In this method, all lines are displayed at once without providing intervals between input video signals.

  This input method is for monitors capable of high-resolution display.

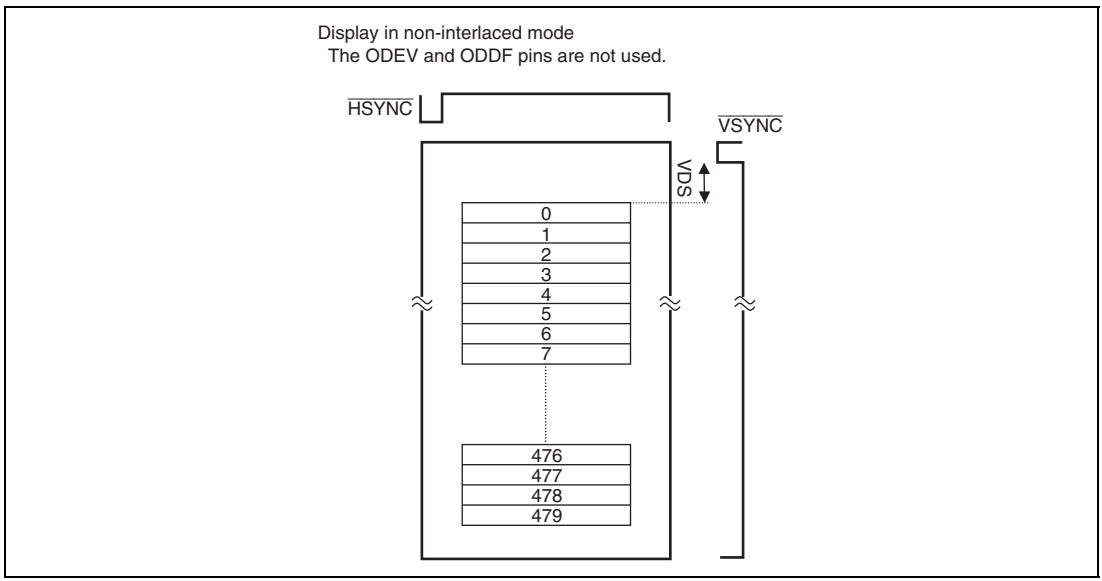


Figure 14.23 Display in Non-Interlaced Method

- Display in interlaced method

  At every scan period VC of the input video signal, even lines and odd lines are switched and displayed in alternation, and a single screen (one frame) is combined and displayed (with the afterimage of the preceding VC) with a period of 2VC. This is a basic TV input method.

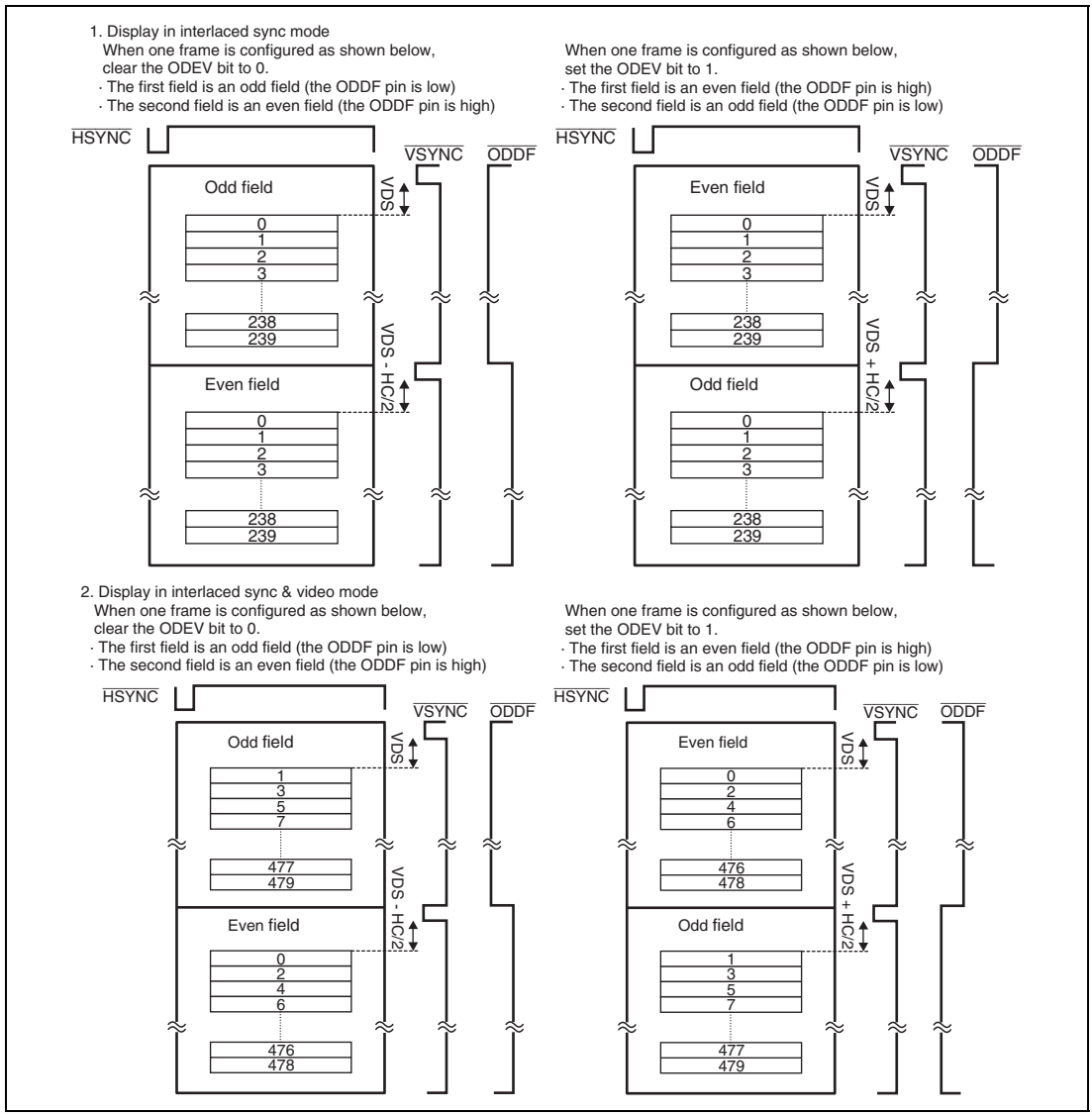


Figure 14.24 Display in Interlaced Method

14.5.4 Color Detection

When output display data matches a color set in CDER, high level is output from the CDE pin. The CDEM bit in DSMR can be used to fix the level outside display intervals. Also, the CDEL bit in DSMR can be used to select the polarity of the output level.

Table 14.37 Output Level of the CDE Pin

		CDE Pin in Display Intervals		CDE Pin Outside Display Intervals	
		Result of Comparison of Output Display Data and Color Detection Register		Value of Color Detection Register*	
CDEL	CDEM	Same	Different	0	Other than 0
0	00	High level	Low level	High level	Low level
0	01	High level	Low level	High level	Low level
0	10	High level	Low level	Low level	Low level
0	11	High level	Low level	High level	High level
1	00	Low level	High level	Low level	High level
1	01	Low level	High level	Low level	High level
1	10	Low level	High level	High level	High level
1	11	Low level	High level	Low level	Low level

Note: * Output display data is 0 outside display intervals.

14.5.5 External Sync Control

In TV-sync mode, the display unit (DU) is capable of using an externally input dot clock (clock signal for frequency multiplication: DCLKIN) to generate a dot clock (output dot clock: DCLKOUT) that is in accord with external synchronizing signals (EXHSYNC, EXVSYNC). Supply the externally input dot clock (multiplication clock: DCLKIN) and set the following parameters in ESCR.

Table 14.38 External Sync Control Parameters

Variable	Function
ESCR/SYNCSEL	Selects the sync signal (EXHSYNC or EXVSYNC) to use in phase matching of the dot clock.
ESCR/FRQSEL	Selects the dot-clock division ratio.

1. Use the SYNCSEL bits of ESCR to set the sync timing of the dot clock (output dot clock: DCLKOUT) generated from the internal dot clock.
2. Use the FRQSEL bits of ESCR to set the division ratio for generation of the internal dot clock.

The following figure shows the internal dot clock timing where the input dot clock has been synchronized with EXHSYNC and then divided by four.

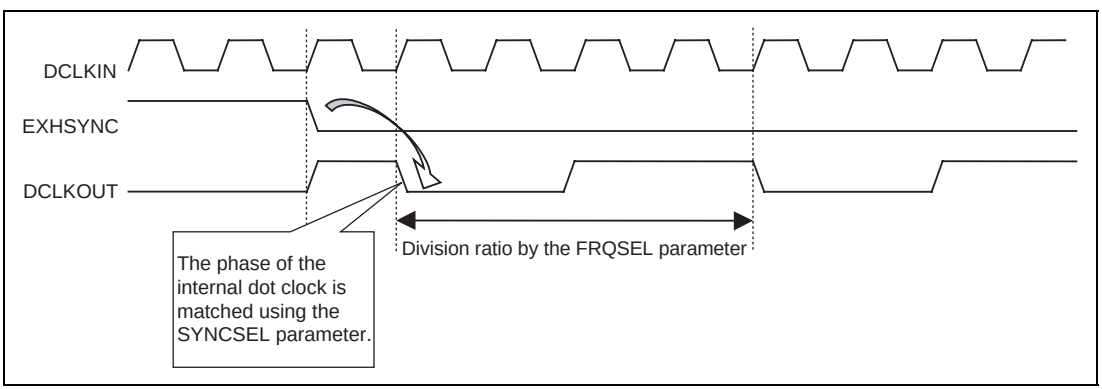


Figure 14.25 DCLKOUT Timing Chart where DCLKIN Synchronized with EXHSYNC is Divided by Four

14.5.6 Output Signal Timing Adjustment

The display unit (DU) enables selection of output timing, with respect to the output dot clock, of the various output signals (the four sync signals HSYNC, VSYNC, CSYNC, ODDF, as well as DISP, CDE, CLAMP, DE, digital RGB signals). Timing is selected by setting OTAR.

Table 14.39 Output Signal Timing Setting Parameters

Variable	Description
SYNCA	Sets output timing of the HSYNC, VSYNC, CSYNC, ODDF signal
DISPA	Sets output timing of the DISP signal
CDEA	Sets output timing of the CDE signal
DRGBA	Sets output timing of digital RGB signal
CLAMPA	Sets output timing of the CLAMP signal
DEA	Sets output timing of the DE signal

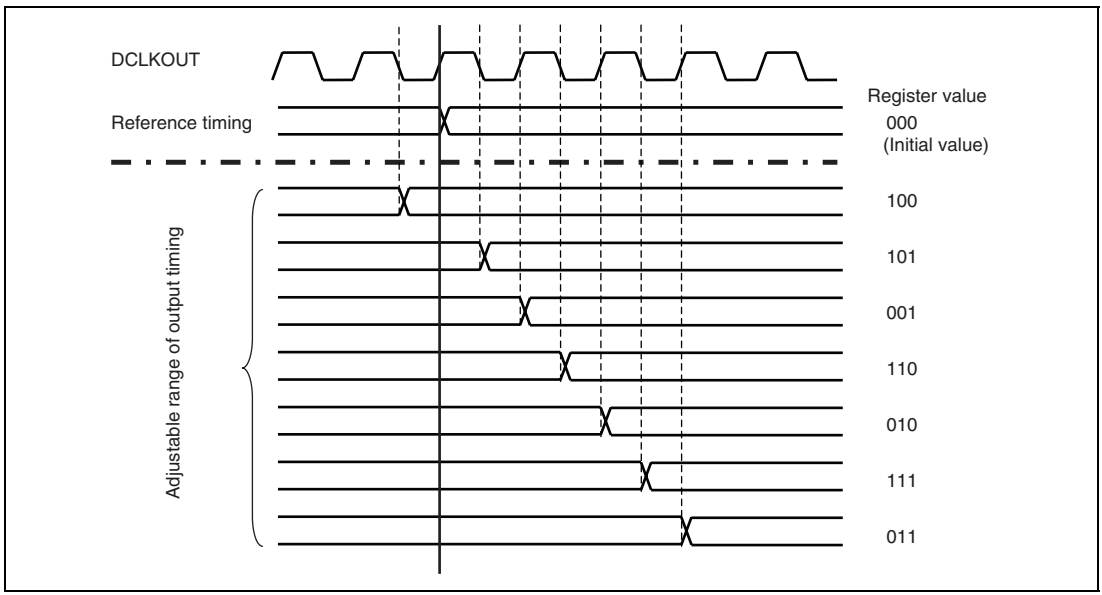


Figure 14.26 Adjustable Range of Output Timing

14.6 Notes on Usage

14.6.1 Module Standby Mode

Module standby mode, in which supply of the clock signal to the display unit (DU) is stopped, is supported.

Even when the display unit (DU) enters the standby mode, the register values are retained. Do not access the display unit (DU) during periods in standby mode.

14.6.2 Transition to Module Standby Mode

1. Turn off the display by setting both the DEN and DRES bits in DSYSR to 0.
2. Test the VBK bit in DSSR to confirm the next VBK flag (because the display is turned off with the timing of VBK).
3. Stop the clock.

14.6.3 Release from Module Standby Mode and Restarting Display

1. Start the clock.
2. Make settings to turn the display on by setting the DEN and DRES bits in DSYSR to 1 and 0, respectively.

14.6.4 Acquisition of External Sync Signal

The following three ways of acquiring the external SYNC signal are available.

(a) The SYNC signal is acquired on rising edges of DCLKIN (or the frequency-divided clock signal derived from this if division has been set up).
(b) The SYNC signal is acquired on falling edges of DCLKIN (or the frequency-divided clock signal derived from this if division has been set up).
(c) Assuming that division has been set up, the SYNC signal that is acquired by the pre-division clock signal is latched on edges of the frequency-divided clock signal.

The electrical characteristics (AC spec.) are only guaranteed for case (a) above. There is no guarantee of electrical characteristics (AC spec.) for cases (b) and (c).

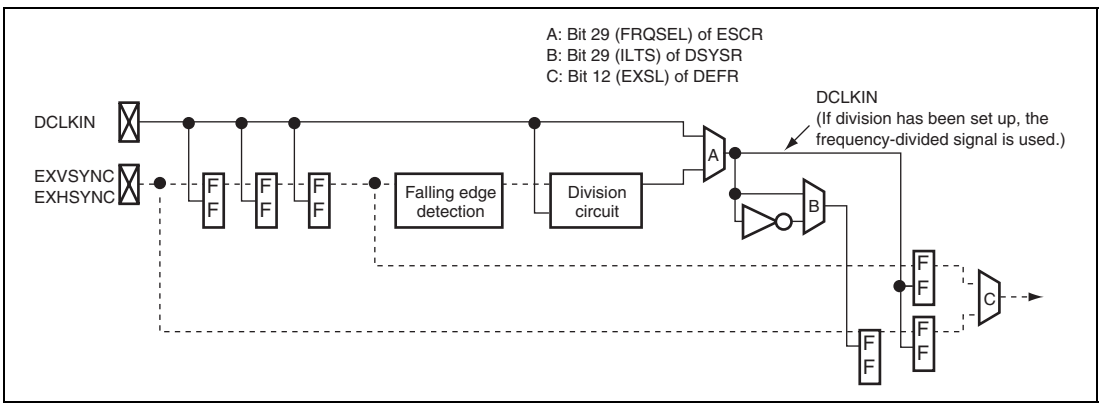


Figure 14.27 Diagram of Circuit that Generates the Display Timing from an External Sync Signal

14.6.5 Restrictions on Changing the Synchronization of the External SYNC Signal

When electrical characteristics (AC spec.) related to the acquisition of an external signal are not satisfied, ensure an interval of at least two cycles between changes of EXHSYNC or EXVSYNC, i.e. the external SYNC signals.

The frequency-divided dot clock is used as the basis of the cycle period when EXSL (bit 12) of DEFR is 0. The pre-division dot clock is used when EXSL is 1.

14.6.6 Note on Register Settings

At the time of a reset, some registers of the DU have fixed initial values but others do not. The initial values of the latter when the power for the LSI is turned on can be either 0 or 1.

From the initial state after power is switched on, be sure to set all of the registers in the DU to the desired values before starting synchronization of the display*[1].

If turning on the display*[2] or capturing*[3] is attempted while values in registers are unknown, operation may be incorrect due to access by the DU to areas other than those intended (the DU is capable of access to any area).

Notes: 1. Setting (a) or (b) starts synchronization of the display.
(a) Setting the DRES and DEN bits in DSYSR to 00.
(b) Setting the DRES and DEN bits in DSYSR to 01.

2. Setting (a) or (b) turns the display on.
(a) When the DPRS bit in DORCR is 0:
With setting (b) in note 1, setting any of bits DPE8 to DPE1 in DPPR to 1.
(b) When the DPRS bit in DORCR is 1:
With setting (b) in note 1, setting bits S1S8 to S1S1 in DS1PR or bits S2S8 to S2S1 in DS2PR to any value from 0001 to 1000.

3. Capturing is turned on by the combination of setting (b) in note 1 and the DC2E or DCE bit in DCPCR being set to 1.

Section 15 LCD Controller (LCDC)

A unified memory architecture is adopted for the LCD controller (LCDC) so that the image data for display is stored in system memory. The LCDC module reads data from system memory, uses the palette memory to determine the colors, then puts the display on the LCD panel. It is impossible to connect the LCDC to the LCD modules*, such as those of microcomputer bus interface types and NTSC/PAL types and those that apply the LVDS interface.

Note: * LCD module can be connected to the LVDS interface by using the LSI with LVDS conversion LSI.

15.1 Features

The LCDC has the following features.

- Panel interface
  - Supports data formats for STN/dual-STN/TFT panels (8/12/16/18-bit bus width)*[1]
- Supports 4/8/15/16-bpp (bits per pixel) color modes
- Supports 1/2/4/6-bpp grayscale modes
- Supports LCD-panel sizes from 16 × 1 to 1024 × 1024*[2]
- 24-bit color palette memory (16 of the 24 bits are valid; R:5/G:6/B:5)
- STN/DSTN panels are prone to flicker and shadowing. The controller applies 65536-color control by 24-bit space-modulation FRC with 8-bit RGB values for reduced flicker.
- Dedicated display memory is unnecessary using part of the memory (area 2 or 3) connected to the CPU as the VRAM to store display data of the LCDC.
- The display is stable because of the large 2.4-Kbyte line buffer
- Supports the inversion of the output signal to suit the LCD panel's signal polarity
- Supports the selection of data formats (the endian setting for bytes, backed pixel method) by register settings
- An interrupt can be generated at the user specified position (controlling the timing of VRAM update start prevents flicker)

Notes: 1. When connecting the LCDC to a TFT panel with an unwired 18-bit bus, the lower bit lines should be connected to GND or to the lowest bit from which data is output.

2. For details, see section 15.4.1, LCD Module Sizes which can be Displayed in this LCDC.

Figure 15.1 shows a block diagram of the LCDC.

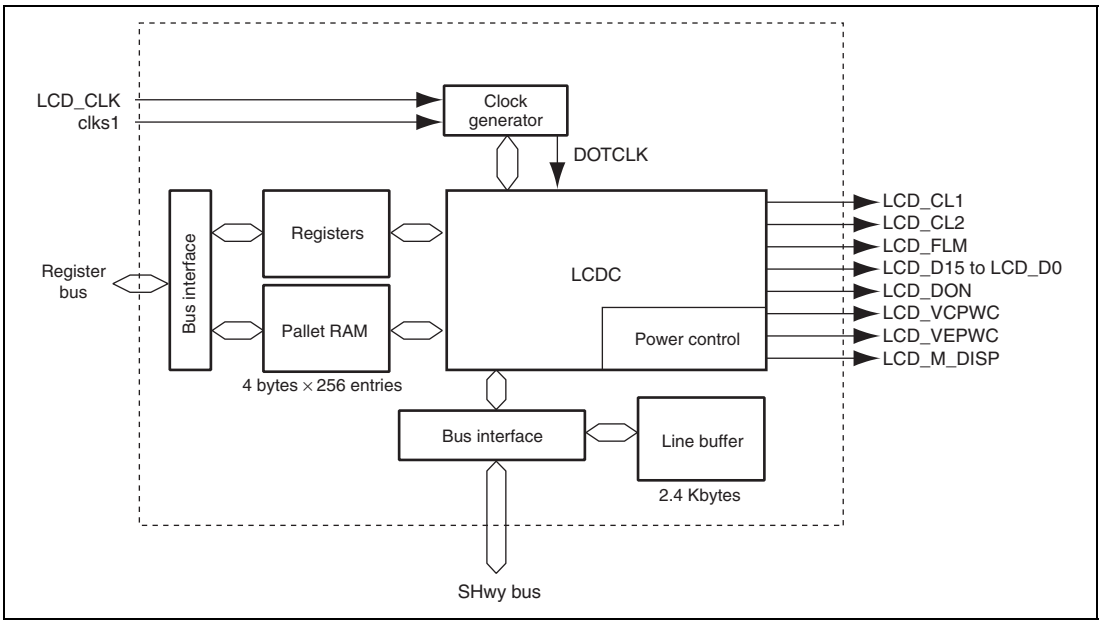


Figure 15.1 LCDC Block Diagram

15.2 Input/Output Pins

Table 15.1 summarizes the LCDC's pin configuration.

The LCDC pins are selected by the pin multiplexing settings of the PFC.

Table 15.1 Pin Configuration

Pin Name	I/O	Function
LCD_D15_* to LCD_D0_*	Output	LCD_DATA15 to LCD_DATA0 Data for LCD panel. Abbreviated to LCD15_* to LCD_D0_*.
LCD_DON_*	Output	Display-on signal (DON)
LCD_CL1_*	Output	Shift-clock 1 (STN/DSTN)/horizontal sync signal (HSYNC)
LCD_CL2_*	Output	Shift-clock 2 (STN/DSTN)/dot clock (DOTCLK)
LCD_M_DISP_*	Output	LCD current-alternating signal/DISP signal
LCD_FLM_*	Output	First line marker/vertical sync signal (VSYNC) (TFT)
LCD_VCPWC_*	Output	LCD-module power control (VCC)
LCD_VEPWC_*	Output	LCD-module power control (VEE)
LCD_CLK_*	Input	External LCD clock-source input Input an external clock. Do not connect this pin to a crystal resonator.

[Legend] *: A or B (mirror pin A or mirror pin B)

Note: Check the LCD module specifications carefully in section 15.5, Clock and LCD Data Signal Examples, before deciding on the wiring specifications for the LCD module.

15.3 Register Configuration

Table 15.2 shows the LCDC register configuration. Table 15.3 shows the register status in each operating mode.

Table 15.2 Register Configuration

Register Name	Abbreviation	R/W	Area P4 Address*	Area 7 Address*	Access Size
Palette data register 00 to FF	LDPR00 to LDPRFF	R/W	H'FFC6 0000 to H'FFC6 03FC	H'1FC6 0000 to H'1FC6 03FC	32
LCDC input clock register	LDICKR	R/W	H'FFC6 0400	H'1FC6 0400	16
LCDC module type register	LDMTR	R/W	H'FFC6 0402	H'1FC6 0402	16
LCDC data format register	LDDFR	R/W	H'FFC6 0404	H'1FC6 0404	16
LCDC data fetch start address register for upper display panel	LDSARU	R/W	H'FFC6 0408	H'1FC6 0408	32
LCDC data fetch start address register for lower display panel	LDSARL	R/W	H'FFC6 040C	H'1FC6 040C	32
LCDC fetch data line address offset register for display panel	LDLAOR	R/W	H'FFC6 0410	H'1FC6 0410	16
LCDC palette control register	LDPALCR	R/W	H'FFC6 0412	H'1FC6 0412	16
LCDC horizontal character number register	LDHCNR	R/W	H'FFC6 0414	H'1FC6 0414	16
LCDC horizontal synchronization signal register	LDHSYNR	R/W	H'FFC6 0416	H'1FC6 0416	16
LCDC vertical displayed line number register	LDVDLNR	R/W	H'FFC6 0418	H'1FC6 0418	16
LCDC vertical total line number register	LDVTLNR	R/W	H'FFC6 041A	H'1FC6 041A	16
LCDC vertical synchronization signal register	LDVSYNR	R/W	H'FFC6 041C	H'1FC6 041C	16
LCDC AC modulation signal toggle line number register	LDACLNR	R/W	H'FFC6 041E	H'1FC6 041E	16
LCDC interrupt control register	LDINTR	R/W	H'FFC6 0420	H'1FC6 0420	16
LCDC power management mode register	LDPMMR	R/W	H'FFC6 0424	H'1FC6 0424	16
LCDC power supply sequence period register	LDPSPR	R/W	H'FFC6 0426	H'1FC6 0426	16

Register Name	Abbreviation	R/W	Area P4 Address*	Area 7 Address*	Access Size
LCDC control register	LDCNTR	R/W	H'FFC6 0428	H'1FC6 0428	16
LCDC user specified interrupt control register	LDUINTR	R/W	H'FFC6 0434	H'1FC6 0434	16
LCDC user specified interrupt line number register	LDUINTLNR	R/W	H'FFC6 0436	H'1FC6 0436	16
LCDC memory access interval number register	LDLIRNR	R/W	H'FFC6 0440	H'1FC6 0440	16

Table 15.3 Register State in Each Operating Mode

Abbreviation	Power-On Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
LDPR00 to LDPRFF	Undefined	Undefined	Retained	Retained	Retained	Undefined
LDICKR	H'1101	H'1101	Retained	Retained	Retained	H'1101
LDMTR	H'0109	H'0109	Retained	Retained	Retained	H'0109
LDDFR	H'000C	H'000C	Retained	Retained	Retained	H'000C
LDSARU	H'04000000	H'04000000	Retained	Retained	Retained	H'04000000
LDSARL	H'04000000	H'04000000	Retained	Retained	Retained	H'04000000
LDLAOR	H'0280	H'0280	Retained	Retained	Retained	H'0280
LDPALCR	H'0000	H'0000	Retained	Retained	Retained	H'0000
LDHCNR	H'4F52	H'4F52	Retained	Retained	Retained	H'4F52
LDHSYNR	H'0050	H'0050	Retained	Retained	Retained	H'0050
LDVDLNR	H'01DF	H'01DF	Retained	Retained	Retained	H'01DF
LDVTLNR	H'01DF	H'01DF	Retained	Retained	Retained	H'01DF
LDVSYNR	H'01DF	H'01DF	Retained	Retained	Retained	H'01DF
LDACLNR	H'000C	H'000C	Retained	Retained	Retained	H'000C
LDINTR	H'0000	H'0000	Retained	Retained	Retained	H'0000
LDPMMR	H'0010	H'0010	Retained	Retained	Retained	H'0010
LDPSPR	H'F60F	H'F60F	Retained	Retained	Retained	H'F60F
LDCNTR	H'0000	H'0000	Retained	Retained	Retained	H'0000
LDUINTR	H'0000	H'0000	Retained	Retained	Retained	H'0000
LDUINTLNR	H'004F	H'004F	Retained	Retained	Retained	H'004F
LDLIRNR	H'0000	H'0000	Retained	Retained	Retained	H'0000

15.3.1 LCDC Input Clock Register (LDICKR)

This LCDC can select bus clock, the peripheral clock, or the external clock as its operation clock source. The selected clock source can be divided using an internal divider into a clock of 1/1 to 1/32 and be used as the LCDC operating clock (DOTCLK). The clock output from the LCDC is used to generate the synchronous clock output (LCD_CL2) for the LCD panel from the operating clock selected in this register. For a TFT panel, LCD_CL2 = DOTCLK, and for an STN or DSTN panel, LCD_CL2 = a clock with a frequency of (DOTCLK/data bus width of output to LCD panel). The LDICKR must be set so that the clock input to the LCDC is less than or equal to the peripheral clock (clks1) regardless of the LCD_CL2.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	ICKSEL[1:0]		—	—	—	—	—	—	DCDR[5:0]					
Initial value:	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	1
R/W:	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13, 12	ICKSEL[1:0]	00	R/W	Input Clock Select Set the clock source for DOTCLK. 00: Setting prohibited 01: Peripheral clock (clks1) is selected 10: External clock (LCD_CLK) is selected 11: Setting prohibited
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5 to 0	DCDR[5:0]	000001	R/W	Clock Division Ratio Set the input clock division ratio. For details on the setting, refer to table 15.4.

Table 15.4 I/O Clock Frequency and Clock Division Ratio

DCDR[5:0]	Clock Division Ratio	I/O Clock Frequency (MHz)	
		100.000*1	54.000*2
000001	1/1	100.000	54.000
000010	1/2	50.000	27.000
000011	1/3	33.333	18.000
000100	1/4	25.000	13.500
000110	1/6	16.666	9.000
001000	1/8	12.500	6.750
001100	1/12	8.333	4.500
010000	1/16	6.250	3.375
011000	1/24	4.166	2.250
100000	1/32	3.125	1.688

Notes: Any setting other than above is handled as a clock division ratio of 1/1 (initial value).

1. Case of peripheral clock = 100 MHz (changes depending on the EXTAL frequency and clock mode settings)
2. Case of external clock = 54 MHz

15.3.2 LCDC Module Type Register (LDMTR)

LDMTR sets the control signals output from this LCDC and the polarity of the data signals, according to the polarity of the signals for the LCD module connected to the LCDC.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FLM POL	CL1 POL	DISP POL	DPOL	—	MCNT	CL1CNT	CL2CNT	—	—	MIFTYP[5:0]					
Initial value:	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	1
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	FLMPOL	0	R/W	FLM (Vertical Sync Signal) Polarity Select Selects the polarity of the LCD_FLM (vertical sync signal, first line marker) for the LCD module. 0: LCD_FLM pulse is high active 1: LCD_FLM pulse is low active
14	CL1POL	0	R/W	CL1 (Horizontal Sync Signal) Polarity Select Selects the polarity of the LCD_CL1 (horizontal sync signal) for the LCD module. 0: LCD_CL1 pulse is high active 1: LCD_CL1 pulse is low active
13	DISPPOL	0	R/W	DISP (Display Enable) Polarity Select Selects the polarity of the LCD_M_DISP (display enable) for the LCD module. 0: LCD_M_DISP is high active 1: LCD_M_DISP is low active

Bit	Bit Name	Initial Value	R/W	Description
12	DPOL	0	R/W	Display Data Polarity Select Selects the polarity of the LCD_D (display data) for the LCD module. This bit supports inversion of the LCD module. 0: LCD_D is high active, transparent-type LCD panel 1: LCD_D is low active, reflective-type LCD panel
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	MCNT	0	R/W	M Signal Control Sets whether or not to output the LCD's current-alternating signal of the LCD module. 0: M (AC line modulation) signal is output 1: M signal is not output
9	CL1CNT	0	R/W	CL1 (Horizontal Sync Signal) Control Sets whether or not to enable CL1 output during the vertical retrace period. 0: CL1 is output during vertical retrace period 1: CL1 is not output during vertical retrace period
8	CL2CNT	1	R/W	CL2 (Dot Clock of LCD Module) Control Sets whether or not to enable CL2 output during the vertical and horizontal retrace period. 0: CL2 is output during vertical and horizontal retrace period 1: CL2 is not output during vertical and horizontal retrace period
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
5 to 0	MIFTYP [5:0]	001001	R/W	Module Interface Type Select Set the LCD panel type and data bus width to be output to the LCD panel. There are three LCD panel types: STN, DSTN, and TFT. There are four data bus widths for output to the LCD panel: 4, 8, 12, and 16 bits. When the required data bus width for a TFT panel is 16 bits or more, connect the LCDC and LCD panel according to the data bus size of the LCD panel. Unlike in a TFT panel, in an STN or DSTN panel, the data bus width setting does not have a 1:1 correspondence with the number of display colors and display resolution, e.g., an 8-bit data bus can be used for 16 bpp, and a 12-bit data bus can be used for 4 bpp. This is because the number of display colors in an STN or DSTN panel is determined by how data is placed on the bus, and not by the number of bits. For data specifications for an STN or DSTN panel, see the specifications of the LCD panel used. The output data bus width should be set according to the mechanical interface specifications of the LCD panel. If an STN or DSTN panel is selected, display control is performed using a 24-bit space-modulation FRC consisting of the 8-bit R, G, and B included in the LCDC, regardless of the color and gradation settings. Accordingly, the color and gradation specified by DSPCOLOR is selected from 16 million colors in an STN or DSTN panel. If a palette is used, the color specified in the palette is displayed. 000000: STN monochrome 4-bit data bus module 000001: STN monochrome 8-bit data bus module 001000: STN color 4-bit data bus module 001001: STN color 8-bit data bus module 001010: STN color 12-bit data bus module 001011: STN color 16-bit data bus module 010001: DSTN monochrome 8-bit data bus module 010011: DSTN monochrome 16-bit data bus module 011001: DSTN color 8-bit data bus module 011010: DSTN color 12-bit data bus module 011011: DSTN color 16-bit data bus module 101011: TFT color 16-bit data bus module Settings other than above: Setting prohibited

15.3.3 LCDC Data Format Register (LDDFR)

LDDFR sets the bit alignment for pixel data in one byte and selects the data type and number of colors used for display so as to match the display driver software specifications.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	PABD	—	DSPCOLOR[6:0]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	PABD	0	R/W	Byte Data Pixel Alignment Sets the pixel data alignment type in one byte of data. The contents of aligned data per pixel are the same regardless of this bit's setting. For example, data H'05 should be expressed as B'0101 which is the normal style handled by a MOV instruction of the this CPU, and should not be selected between B'0101 and B'1010. 0: Big endian for byte data 1: Little endian for byte data
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
6 to 0	DSPCOLOR [6:0]	0001100	R/W	Display Color Select Set the number of display colors for the display (0 is written to upper bits of 4 to 6 bpp). For display colors to which the description (via palette) is added below, the color set by the color palette is actually selected by the display data and displayed. The number of colors that can be supported in rotation mode is restricted by the display resolution. 0000000: Monochrome, 2 grayscales, 1 bpp (via palette) 0000001: Monochrome, 4 grayscales, 2 bpp (via palette) 0000010: Monochrome, 16 grayscales, 4 bpp (via palette) 0000100: Monochrome, 64 grayscales, 6 bpp (via palette) 0001010: Color, 16 colors, 4 bpp (via palette) 0001100: Color, 256 colors, 8 bpp (via palette) 0011101: Color, 32K colors (RGB: 555), 15 bpp 0101101: Color, 64K colors (RGB: 565), 16 bpp Settings other than above: Setting prohibited

15.3.4 LCDC Start Address Register for Upper Display Data Fetch (LDSARU)

LDSARU sets the start address from which data is fetched by the LCDC for display of the LCDC panel. When a DSTN panel is used, this register specifies the fetch start address for the upper side of the panel.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	SAU[27:16]											
Initial value:	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SAU[15:4]												—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27	SAU[27]	0	R/W	Start Address for Upper Display Data Fetch The start address for data fetch of the display data must be set within the memory area of area 2 or 3. Setting the SAU[27] bit to 0 is prohibited (the initial value should be changed before usage).
26	SAU[26]	1	R/W	
25 to 4	SAU[25:4]	All 0	R/W	
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: The minimum alignment unit of LDSARU is 512 bytes. Write 0 to the lower nine bits.

15.3.5 LCDC Start Address Register for Lower Display Data Fetch (LDSARL)

When a DSTN panel is used, LDSARL specifies the fetch start address for the lower side of the panel.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	SAL[27:16]											
Initial value:	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SAL[15:4]												—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27	SAL[27]	0	R/W	Start Address for Lower Panel Display Data Fetch The start address for data fetch of the display data must be set within the memory area of area 2 or 3. STN and TFT: Cannot be used DSTN: Start address for fetching display data corresponding to the lower panel Setting the SAL[27] bit to 0 is prohibited (the initial value should be changed before usage).
26	SAL[26]	1	R/W	
25 to 4	SAL[25:4]	All 0	R/W	
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: The minimum alignment unit of LDSARU is 32 bytes. Write 0 to the lower five bits.

15.3.6 LCDC Line Address Offset Register for Display Data Fetch (LDLAOR)

LDLAOR sets the address width of the Y-coordinates increment used for LCDC to read the image recognized by the graphics driver. This register specifies how many bytes the address from which data is to be read should be moved when the Y coordinates have been incremented by 1. This register does not have to be equal to the horizontal width of the LCD panel. When the memory address of a point (X, Y) in the two-dimensional image is calculated by Ax + By+ C, this register becomes equal to B in this equation.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LAO[15:0]															
Initial value:	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	LAO [15:0]	H'0280	R/W	Line Address Offset The minimum alignment unit of LDLAOR is 32 bytes. Because the LCDC handles these values as 32-byte data, the values written to the lower five bits of the register are always treated as 0. The lower five bits of the register are always read as 0. The initial values (× resolution = 640) will continuously and accurately place the VGA (640 × 480 dots) display data without skipping an address between lines. A binary exponential at least as large as the horizontal width of the image is recommended for the LDLAOR value, taking into consideration the software operation speed.

15.3.7 LCDC Palette Control Register (LDPALCR)

LDPALCR selects whether the CPU or LCDC accesses the palette memory. When the palette memory is being used for display operation, display mode should be selected. When the palette memory is being written to, color-palette setting mode should be selected.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	PALS	—	—	—	PALEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 5	—	All 0	R	Reserved These bits always read as 0. The write value should always be 0.
4	PALS	0	R	Palette State Indicates the access right state of the palette. 0: Normal display mode: LCDC uses the palette 1: Color-palette setting mode: The host (CPU) uses the palette
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	PALEN	0	R/W	Palette Read/Write Enable Requests the access right to the palette. 0: Request for transition to normal display mode 1: Request for transition to color palette setting mode

15.3.8 Palette Data Registers 00 to FF (LDPR00 to LDPRFF)

LDPR registers are for accessing palette data directly allocated (4 bytes x 256 addresses) to the memory space. To access the palette memory, access the corresponding register among this register group (LDPR00 to LDPRFF). Each palette register is a 32-bit register including three 8-bit areas for R, G, and B. For details on the color palette specifications, see section 15.4.2, Color Palette Specification.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	PALDnn[23:16]							
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PALDnn[15:0]															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	—	R	Reserved
23 to 0	PALDnn[23:0]	—	R/W	Palette Data Bits 18 to 16, 9, 8, and 2 to 0 are reserved within each RGB palette and cannot be set. However, these bits can be extended according to the upper bits.

Note: nn = H'00 to H'FF

15.3.9 LCDC Horizontal Character Number Register (LDHCNR)

LDHCNR specifies the LCD module's horizontal size (in the scan direction) and the entire scan width including the horizontal retrace period.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HDCN[7:0]								HTCN[7:0]							
Initial value:	0	1	0	0	1	1	1	1	0	1	0	1	0	0	1	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	HDCN [7:0]	01001111	R/W	Horizontal Display Character Number Set the number of horizontal display characters (unit: character = 8 dots). Specify to the value of (the number of display characters) -1. Example: For a LCD module with a width of 640 pixels. HDCN = (640/8) -1 = 79 = H'4F
7 to 0	HTCN [7:0]	01010010	R/W	Horizontal Total Character Number Set the number of total horizontal characters (unit: character = 8 dots). Specify to the value of (the number of total characters) - 1. However, the minimum horizontal retrace period is three characters (24 dots). Example: For a LCD module with a width of 640 pixels. HTCN = [(640/8)-1] + 3 = 82 = H'52 In this case, the number of total horizontal dots is 664 dots and the horizontal retrace period is 24 dots.

Notes: 1. The values set in HDCN and HTCN must satisfy the relationship of HTCN ≥ HDCN. Also, the total number of characters of HTCN must be an even number. (The set value will be an odd number, as it is one less than the actual number.)

2. Set HDCN according to the display resolution as follows:
1 bpp: (multiple of 16) – 1 [1 line is multiple of 128 pixel]
2 bpp: (multiple of 8) – 1 [1 line is multiple of 64 pixel]
4 bpp: (multiple of 4) – 1 [1 line is multiple of 32 pixel]
6 bpp/8 bpp: (multiple of 2) – 1 [1 line is multiple of 16 pixel]

15.3.10 LCDC Horizontal Sync Signal Register (LDHSYNR)

LDHSYNR specifies the timing of the generation of the horizontal (scan direction) sync signals for the LCD module.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HSYNW[3:0]				—	—	—	—	HSYNP[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	HSYNW [3:0]	0000	R/W	Horizontal Sync Signal Width Set the width of the horizontal sync signals (CL1 and Hsync) (unit: character = 8 dots). Specify to the value of (the number of horizontal sync signal width) -1. Example: For a horizontal sync signal width of 8 dots. HSYNW = (8 dots/8 dots/character) -1 = 0 = H'0
11 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	HSYNP [7:0]	01010000	R/W	Horizontal Sync Signal Output Position Set the output position of the horizontal sync signals (unit: character = 8 dots). Specify to the value of (the number of horizontal sync signal output position) -1. Example: For a LCD module with a width of 640 pixels. HSYNP = [(640/8) + 1] -1 = 80 = H'50 In this case, the horizontal sync signal is active from the 648th through the 655th dot.

Note: The following conditions must be satisfied:
HTCN ≥ HSYNP + HSYNW + 1
HSYNP ≥ HDCN + 1

15.3.11 LCDC Vertical Display Line Number Register (LDVDLNR)

LDVDLNR specifies the LCD module's vertical size (for both scan direction and vertical direction). For a DSTN panel, specify an even number at least as large as the LCD panel's vertical size regardless of the size of the upper and lower panels, e.g. 480 for a 640 × 480 panel.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	VDLN[10:0]										
Initial value:	0	0	0	0	0	0	0	1	1	1	0	1	1	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	VDLN[10:0]	00111011111	R/W	Vertical Display Line Number Set the number of vertical display lines (unit: line). Specify to the value of (the number of display line) -1. Example: For an 480-line LCD module VDLN = 480-1 = 479 = H'1DF

15.3.12 LCDC Vertical Total Line Number Register (LDVTLNR)

LDVTLNR specifies the LCD panel's entire vertical size including the vertical retrace period.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	VTLN[10:0]										
Initial value:	0	0	0	0	0	0	0	1	1	1	0	1	1	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	VTLN[10:0]	00111011111	R/W	Vertical Total Line Number Set the total number of vertical display lines (unit: line). Specify to the value of (the number of total line) -1. The minimum for the total number of vertical lines is 2 lines. The following conditions must be satisfied: VTLN ≥ VDLN, VTLN ≥ 1. Example: For a 480-line LCD module and a vertical period of 0 lines: VTLN = (480 + 0) –1 = 479 = H'1DF

15.3.13 LCDC Vertical Sync Signal Register (LDVSYNR)

LDVSYNR specifies the vertical (scan direction and vertical direction) sync signal timing of the LCD module.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VSYNW[3:0]				—	VSYNP[10:0]										
Initial value:	0	0	0	0	0	0	0	1	1	1	0	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	VSYNW[3:0]	0000	R/W	Vertical Sync Signal Width Set the width of the vertical sync signals (FLM and Vsync) (unit: line). Specify to the value of (the vertical sync signal width) -1. Example: For a vertical sync signal width of 1 line. VSYNW = (1 - 1) = 0 = H'0
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10 to 0	VSYNP[10:0]	00111011111	R/W	Vertical Sync Signal Output Position Set the output position of the vertical sync signals (FLM and Vsync) (unit: line). Specify to the value of (the number of vertical sync signal output position) -2. DSTN should be set to an odd number value. It is handled as (setting value+1)/2. Example: For an 480-line LCD module and a vertical retrace period of 0 lines (in other words, VTLN=479 and the vertical sync signal is active for the first line): • Single display VSYNP = [(1-1)+VTLN]mod(VTLN+1) = [(1-1)+479]mod(479+1) = 479mod480 = 479 =H'1DF • Dual displays VSYNP = [(1-1)×2+VTLN]mod(VTLN+1) = [(1-1)×2+479]mod(479+1) = 479mod480 = 479 =H'1DF

15.3.14 LCDC AC Modulation Signal Toggle Line Number Register (LDACLNR)

LDACLNR specifies the timing to toggle the AC modulation signal (LCD current-alternating signal) of the LCD module.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	ACLN[4:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4 to 0	ACLN[4:0]	01100	R/W	AC Line Number Set the number of lines where the LCD current-alternating signal of the LCD module is toggled (unit: line). Specify to the value of (the number of toggle line) - 1. Example: For toggling every 13 lines. ACLN = 13-1 = 12= H'0C

Note: When the total line number of the LCD panel is even, set an even number so that toggling is performed at an odd line.

15.3.15 LCDC Interrupt Control Register (LDINTR)

LDINTR specifies where to control the Vsync interrupt of the LCD module. See also section 15.3.19, LCDC User Specified Interrupt Control Register (LDUINTR) and section 15.3.20, LCDC User Specified Interrupt Line Number Register (LDUINTLNR) for interrupts. Note that operations by this register setting and the LCDC user specified interrupt control register (LDUINTR) setting are independent.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MINTEN	FINTEN	VSINTEN	VEINTEN	MINTS	FINTS	VSINTS	VEINTS	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	MINTEN	0	R/W	Memory Access Interrupt Enable Enables or disables an interrupt generation at the start point of each vertical retrace line period for VRAM access by LCDC. 0: Disables an interrupt generation at the start point of each vertical retrace line period for VRAM access 1: Enables an interrupt generation at the start point of each vertical retrace line period for VRAM access
14	FINTEN	0	R/W	Frame End Interrupt Enable Enables or disables the generation of an interrupt after the last pixel of a frame is output to LDC panel. 0: Disables an interrupt generation when the last pixel of the frame is output 1: Enables an interrupt generation when the last pixel of the frame is output
13	VSINTEN	0	R/W	Vsync Starting Point Interrupt Enable Enables or disables the generation of an interrupt at the start point of LCDC's Vsync. 0: Interrupt at the start point of the Vsync is disabled 1: Interrupt at the start point of the Vsync is enabled

Bit	Bit Name	Initial Value	R/W	Description
12	VEINTEN	0	R/W	Vsync Ending Point Interrupt Enable Enables or disables the generation of an interrupt at the end point of LCDC's Vsync. 0: Interrupt at the end point of the Vsync signal is disabled 1: Interrupt at the end point of the Vsync signal is enabled
11	MINTS	0	R/W	Memory Access Interrupt State Indicates the memory access interrupt handling state. This bit indicates 1 when the LCDC memory access interrupt is generated (set state). During the memory access interrupt handling routine, this bit should be cleared by writing 0. 0: LCDC did not generate a memory access interrupt or has been informed that the generated memory access interrupt has completed 1: LCDC has generated a memory access end interrupt and not yet been informed that the generated memory access interrupt has completed
10	FINTS	0	R/W	Flame End Interrupt State Indicates the flame end interrupt handling state. This bit indicates 1 at the time when the LCDC flame end interrupt is generated (set state). During the flame end interrupt handling routine, this bit should be cleared by writing 0. 0: LCDC did not generate a flame end interrupt or has been informed that the generated flame end interrupt has completed 1: LCDC has generated a flame end interrupt and not yet been informed that the generated flame end interrupt has completed

Bit	Bit Name	Initial Value	R/W	Description
9	VSINTS	0	R/W	Vsync Start Interrupt State Indicates the LCDC's Vsync start interrupt handling state. This bit is set to 1 at the time a Vsync start interrupt is generated. During the Vsync start interrupt handling routine, this bit should be cleared by writing 0 to it. 0: LCDC did not generate a Vsync start interrupt or has been informed that the generated Vsync start interrupt has completed 1: LCDC has generated a Vsync start interrupt and has not yet been informed that the generated Vsync start interrupt has completed
8	VEINTS	0	R/W	Vsync End Interrupt State Indicates the LCDC's Vsync end interrupt handling state. This bit is set to 1 at the time a Vsync end interrupt is generated. During the Vsync end interrupt handling routine, this bit should be cleared by writing 0. 0: LCDC did not generate a Vsync end interrupt or has been informed that the generated Vsync end interrupt has completed 1: LCDC has generated a Vsync end interrupt and has not yet been informed that the generated Vsync interrupt has completed
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

15.3.16 LCDC Power Management Mode Register (LDPMMR)

LDPMMR controls the power supply circuit that provides power to the LCD module. The usage of two types of power-supply control pins, LCD_VCPWC and LCD_VEPWC, and turning on or off the power supply function are selected.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ONC[3:0]				OFFD[3:0]				—	VCPE	VEPE	DONE	—	—	LPS[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	ONC[3:0]	0000	R/W	LCDC Power-On Sequence Period Set the period from LCD_VEPWC assertion to LCD_DON assertion in the power-on sequence of the LCD module in frame units. Specify to the value of (the period) -1. This period is the (c) period in figures 15.4 to 15.7, Power-Supply Control Sequence and States of the LCD Module. For details on setting this register, see table 15.5. (The setting method is common for ONA, ONB, OFFD, OFFE, and OFFF.)
11 to 8	OFFD[3:0]	0000	R/W	LCDC Power-Off Sequence Period Set the period from LCD_DON negation to LCD_VEPWC negation in the power-off sequence of the LCD module in frame units. Specify to the value of (the period) -1. This period is the (d) period in figures 15.4 to 15.7, Power-Supply Control Sequence and States of the LCD Module.
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
6	VCPE	0	R/W	LCD_VCPWC Pin Enable Sets whether or not to enable a power-supply control sequence using the LCD_VCPWC pin. 0: Disabled: LCD_VCPWC pin is masked and fixed low 1: Enabled: LCD_VCPWC pin output is asserted and negated according to the power-on or power-off sequence
5	VEPE	0	R/W	LCD_VEPWC Pin Enable Sets whether or not to enable a power-supply control sequence using the LCD_VEPWC pin. 0: Disabled: LCD_VEPWC pin is masked and fixed low 1: Enabled: LCD_VEPWC pin output is asserted and negated according to the power-on or power-off sequence
4	DONE	1	R/W	LCD_DON Pin Enable Sets whether or not to enable a power-supply control sequence using the LCD_DON pin. 0: Disabled: LCD_DON pin is masked and fixed low 1: Enabled: LCD_DON pin output is asserted and negated according to the power-on or power-off sequence
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	LPS[1:0]	00	R	LCD Module Power-Supply Input State Indicates the power-supply input state of the LCD module when using the power-supply control function. 00: LCD module power off 11: LCD module power on

15.3.17 LCDC Power-Supply Sequence Period Register (LDPSPR)

LDPSPR controls the power supply circuit that provides power to the LCD module. The timing to start outputting the timing signals to the LCD_VEPWC and LCD_VCPWC pins is specified.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ONA[3:0]				ONB[3:0]				OFFE[3:0]				OFFF[3:0]			
Initial value:	1	1	1	1	0	1	1	0	0	0	0	0	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	ONA[3:0]	1111	R/W	LCDC Power-On Sequence Period Set the period from LCD_VCPWC assertion to starting output of the display data (LCD_D) and timing signals (LCD_FLM, LCD_CL1, LCD_CL2, and LCD_M_DISP) in the power-on sequence of the LCD module in frame units. Specify to the value of (the period)-1. This period is the (a) period in figures 15.4 to 15.7, Power-Supply Control Sequence and States of the LCD Module.
11 to 8	ONB[3:0]	0110	R/W	LCDC Power-On Sequence Period Set the period from starting output of the display data (LCD_D) and timing signals (LCD_FLM, LCD_CL1, LCD_CL2, and LCD_M_DISP) to the LCD_VEPWC assertion in the power-on sequence of the LCD module in frame units. Specify to the value of (the period)-1. This period is the (b) period in figures 15.4 to 15.7, Power-Supply Control Sequence and States of the LCD Module.

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	OFFE[3:0]	0000	R/W	LCDC Power-Off Sequence Period Set the period from LCD_VEPWC negation to stopping output of the display data (LCD_D) and timing signals (LCD_FLM, LCD_CL1, LCD_CL2, and LCD_M_DISP) in the power-off sequence of the LCD module in frame units. Specify to the value of (the period)-1. This period is the (e) period in figures 15.4 to 15.7, Power-Supply Control Sequence and States of the LCD Module.
3 to 0	OFFF[3:0]	1111	R/W	LCDC Power-Off Sequence Period Set the period from stopping output of the display data (LCD_D) and timing signals (LCD_FLM, LCD_CL1, LCD_CL2, and LCD_M_DISP) to LCD_VCPWC negation to in the power-off sequence of the LCD module in frame units. Specify to the value of (the period)-1. This period is the (f) period in figures 15.4 to 15.7, Power-Supply Control Sequence and States of the LCD Module.

15.3.18 LCDC Control Register (LDCNTR)

LDCNTR specifies start and stop of display by the LCDC.

When 1s are written to the DON2 bit and the DON bit, the LCDC starts display. Turn on the LCD module following the sequence set in the LDPMMR and LDCNTR. The sequence ends when the LPS[1:0] value changes from B'00 to B'11. Do not make any action to the DON bit until the sequence ends.

When 0 is written to the DON bit, the LCDC stops display. Turn off the LCD module following the sequence set in the LDPMMR and LDCNTR. The sequence ends when the LPS[1:0] value changes from B'11 to B'00. Do not make any action to the DON bit until the sequence ends.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	DON2	—	—	—	DON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	DON2	0	R/W	Display On 2 Specifies the start of the LCDC display operation. 0: LCDC is being operated or stopped 1: LCDC starts operation When this bit is read, always read as 0. Write 1 to this bit only when starting display. If a value other than 0 is written when starting display, the operation is not guaranteed. When 1 is written to, it resumes automatically to 0. Accordingly, this bit does not need to be cleared by writing 0.
3 to 1	—	All 0	R	Reserved. These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
0	DON	0	R/W	Display On Specifies the start and stop of the LCDC display operation. The control sequence state can be checked by referencing the LPS[1:0] of LDPMMR. 0: Display-off mode: LCDC is stopped 1: Display-on mode: LCDC operates

Notes: 1. Write H'0011 to LDCNTR when starting display and H'0000 when completing display. Data other than H'0011 and H'0000 must not be written to.
2. Setting bit DON2 to 1 makes the contents of the palette RAM undefined. Before writing to the palette RAM, set bit DON2 to 1.
3. To access to other registers of the LCDC after writing to LDCNTR, keep an interval of at least four cycles of the peripheral clock (PCLK) or perform dummy read operation of a 32-bit register in a module other than the LCDC.

15.3.19 LCDC User Specified Interrupt Control Register (LDUINTR)

LDUINTR sets whether the user specified interrupt is generated, and indicates its processing state. This interrupt is generated at the time when image data which is set by the line number register (LDUINTLNR) in LCDC is read from VRAM.

This LCDC issues the interrupts (LCDCI): user specified interrupt by this register, memory access interrupt by the LCDC interrupt control register (LDINTR), and OR of Vsync interrupt output. This register and LCDC interrupt control register (LDINTR) settings affect the interrupt operation independently.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	UINTEN	—	—	—	—	—	—	—	UINTS
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
8	UINTEN	0	R/W	User Specified Interrupt Enable Sets whether generate an LCDC user specified interrupt. 0: LCDC user specified interrupt is not generated 1: LCDC user specified interrupt is generated
7 to 1	—	All 0	R	Reserved. These bits are always read as 0. The write value should always be 0.
0	UINTS	0	R/W	User Specified Interrupt State This bit is set to 1 at the time an LCDC user specified interrupt is generated (set state). During the user specified interrupt handling routine, this bit should be cleared by writing 0 to it. 0: LCDC did not generate a user specified interrupt or has been informed that the generated user specified interrupt has completed 1: LCDC has generated a user specified interrupt and has not yet been notified that the generated user specified interrupt has completed

Note: Interrupt processing flow:

1. Interrupt signal is input
2. LDINTR is read
3. If MINTS, FINTS, VSINTS, or VEINTS is 1, a generated interrupt is memory access interrupt, flame end interrupt, Vsync rising edge interrupt, or Vsync falling edge interrupt. Processing for each interrupt is performed.
4. If MINTS, FINTS, VSINTS, or VEINTS is 0, a generated interrupt is not memory access interrupt, flame end interrupt, Vsync rising edge interrupt, or Vsync falling edge interrupt.
5. UINTS is read.
6. If UINTS is 1, a generated interrupt is a user specified interrupt. Process for user specified interrupt is carried out.
7. If UINTS is 0, a generated interrupt is not a user specified interrupt. Other processing is performed.

15.3.20 LCDC User Specified Interrupt Line Number Register (LDUINTLNR)

LDUINTLNR sets the point where the user specified interrupt is generated. Setting is done in horizontal line units.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	UINTLN[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	1	0	0	1	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	UINTLN [10:0]	00001001111	R/W	User Specified Interrupt Generation Line Number Specifies the line in which the user specified interrupt is generated (line units). Set (the number of lines in which interrupts are generated) −1 Example: Generate the user specified interrupt in the 80th line. UINTLN = 160/2 − 1 = 79 = H'04F

Notes: 1. When using the LCD module with STN/TFT display, the setting value of this register should be equal to lower than the vertical display line number (VDLN) in LDVDLNR.

2. When using the LCD module with DSTN display, the setting value of this register should be equal to or lower than half the vertical display line number (VDLN) in LDVDLNR. The user specified interrupt is generated at the point when the LCDC read the specified piece of image data in lower display from VRAM.

15.3.21 LCDC Memory Access Interval Number Register (LDLIRNR)

LDLIRNR controls the bus cycle interval when the LCDC reads VRAM. When LDLIRNR is set to a value other than H'00, the LCDC does not access VRAM until clock count of the SDRAM matches the value set in LDLIRNR. When LDLIRNR is set to H'00 (initial value), the LCDC accesses VRAM one clock after the LCDC accessed VRAM.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	LIRN[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	LIRN[7:0]	All 0	R/W	VRAM Read Clock Cycle Interval Specifies the number of the SDRAM clock cycles which can be performed during the bus cycle for reading of the VRAM by the LCDC. H'00: 1 clock cycle H'01: 1 clock cycle H'02: 2 clock cycle : H'FE: 254 clock cycles H'FF: 255 clock cycles

15.4 Operation

15.4.1 LCD Module Sizes which can be Displayed in this LCDC

This LCDC is capable of controlling displays with up to 1024 × 1024 dots and 16 bpp (bits per pixel). The image data for display is stored in VRAM, which is shared with the CPU. This LCDC should read the data from VRAM before display.

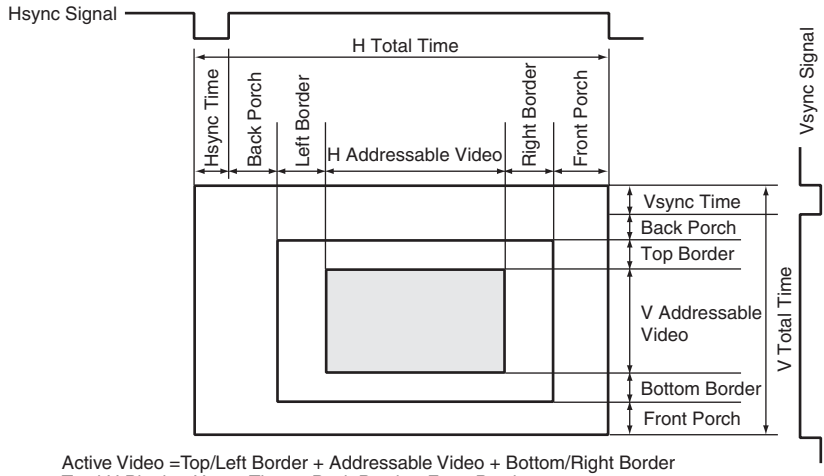
This LSI has a maximum 64-byte memory read operation and a 2.4-Kbyte line buffer, so although a complete breakdown of the display is unlikely, there may be some problems with the display depending on the combination. A recommended size at the frame rate of 60 Hz is 320 × 240 dots in 16 bpp or 640 × 480 dots in 8 bpp.

The bus occupation ratio can be calculated as follows:

$$\text{Bus occupation ratio (\%)} = \frac{\text{Overhead coefficient} \times \text{Total number of display pixels ((HDCN + 1)} \times 8 \times \text{(VDLN + 1))} \times \text{Frame rate (Hz)} \times \text{Number of colors (bpp)}}{\text{MCK of DDR-SDRAM} \times \text{Bus width (16 bits)}} \times 100$$

The overhead coefficient becomes 1.75 when the CL (CAS latency) of DDR-SDRAM is 3, 1.8125 when 4, and 1.875 when 5 (each value is ideal under the best conditions).

Figure 15.2 shows the valid display and the retrace period.



Active Video =Top/Left Border + Addressable Video + Bottom/Right Border
Total H Blank = Hsync Time + Back Porch + Front Porch
Total V Blank = Vsync Time + Back Porch + Front Porch

HTCN = H Total Time
HDCN = H Addressable Video
HSYNP = H Addressable Video + Right Border + Front Porch
HSYNW = Hsync Time

VTLN = V Total Time
CDLN = V Addressable Video
VSYNP = V Addressable Video + Bottom Border + Front porch
VSYNW = Vsync Time

Figure 15.2 Valid Display and the Retrace Period

15.4.2 Color Palette Specification

(1) Color Palette Register

This LCDC has a color palette which outputs 24 bits of data per entry and is able to simultaneously hold 256 entries. The color palette thus allows the simultaneous display of 256 colors chosen from among 16-M colors.

The procedure below may be used to set up color palettes at any time.

1. The PALEN bit in the LDPALCR is 0 (initial value); normal display operation
2. Access LDPALCR and set the PALEN bit to 1; enter color-palette setting mode after three cycles of peripheral clock.
3. Access LDPALCR and confirm that the PALS bit is 1.
4. Access LDPR00 to LDPRFF and write the required values to the PALD00 to PALDFF bits.
5. Access LDPALCR and clear the PALEN bit to 0; return to normal display mode after a cycle of peripheral clock.

A 0 is output on the LCDC display data output (LCD_D) while the PALS bit in LDPALCR is set to 1.

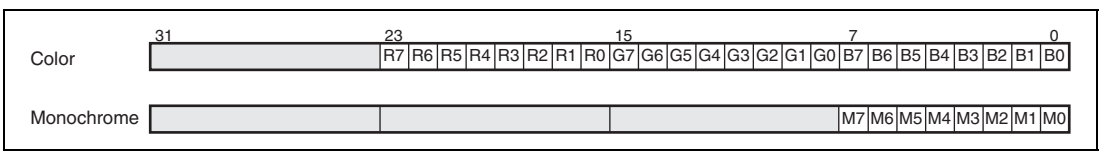


Figure 15.3 Color-Palette Data Format

PALDnn color and gradation data should be set as above.

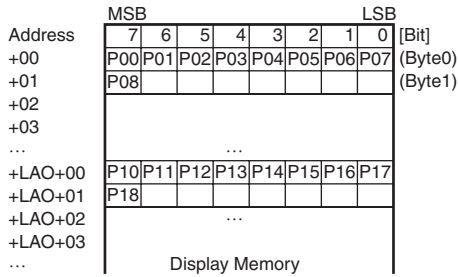
For a color display, PALDnn[23:16], PALDnn[15:8], and PALDnn[7:0] respectively hold the R, G, and B data. Although the bits PALDnn[18:16], PALDnn[9:8], and PALDnn[2:0] exist, no memory is associated with these bits. PALDnn[18:16], PALDnn[9:8], and PALDnn[2:0] are thus not available for storing palette data. The numbers of valid bits are thus R: 5, G: 6, and B: 5. A 24-bit (R: 8 bits, G: 8 bits, and B: 8 bits) data should, however, be written to the palette-data registers. When the values for PALDnn[23:19], PALDnn[15:10], or PALDnn[7:3] are not 0, 1 or 0 should be written to PALDnn[18:16], PALDnn[9:8], or PALDnn[2:0], respectively. When the values of PALDnn[23:19], PALDnn[15:10], or PALDnn[7:3] are 0, 0s should be written to PALDnn[18:16], PALDnn[9:8], or PALDnn[2:0], respectively. Then 24 bits are extended.

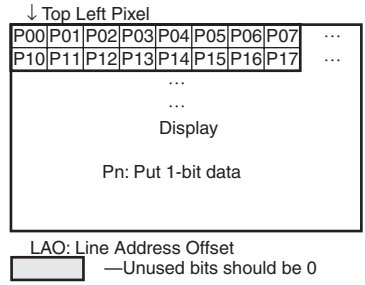
Grayscale data for a monochromatic display should be set in PALDnn[7:3]. PALDnn[23:8] are all "don't care". When the value in PALDnn[7:3] is not 0, 1s should be written to PALDnn[2:0].

When the value in PALDnn[7:3] is 0, 0s should be written to PALDnn[2:0]. Then 8 bits are extended.

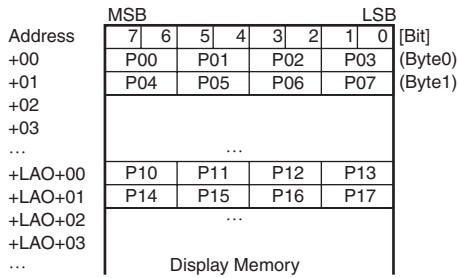
15.4.3 Data Format

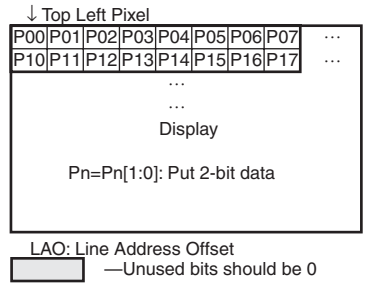
1. Packed 1bpp (Pixel Alignment in Byte is Big Endian) [Windows CE Recommended Format]



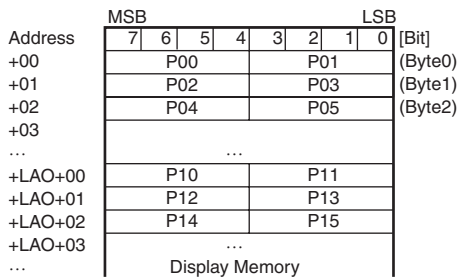


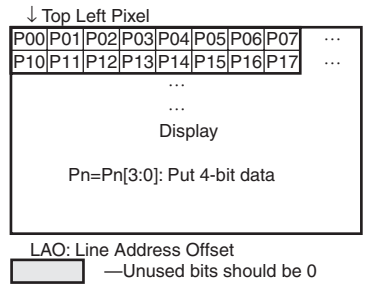
2. Packed 2bpp (Pixel Alignment in Byte is Big Endian) [Windows CE Recommended Format]



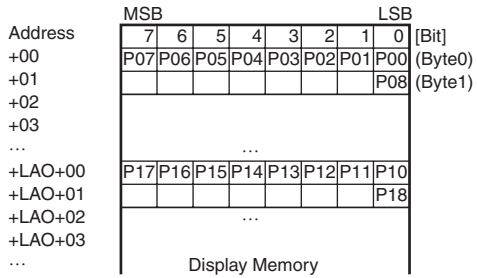


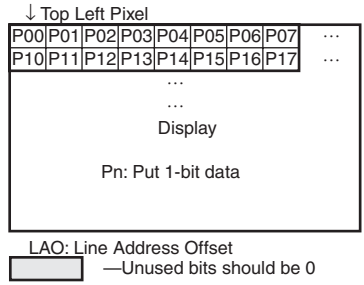
3. Packed 4bpp (Pixel Alignment in Byte is Big Endian) [Windows CE Recommended Format]



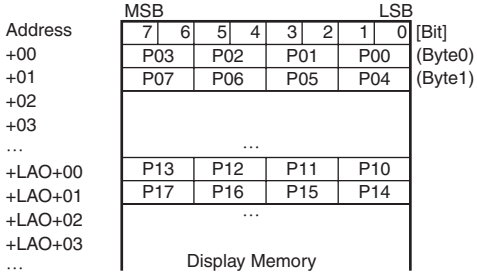


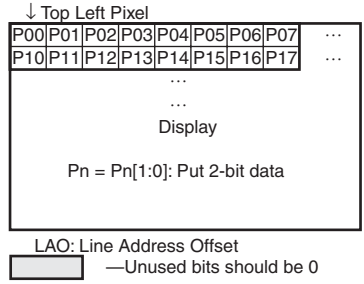
4. Packed 1bpp (Pixel Alignment in Byte is Little Endian)



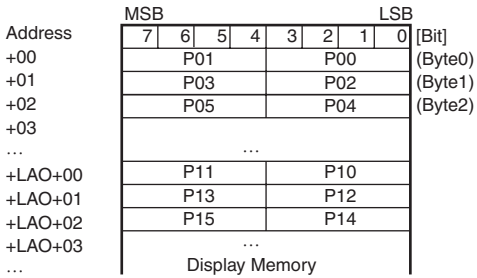


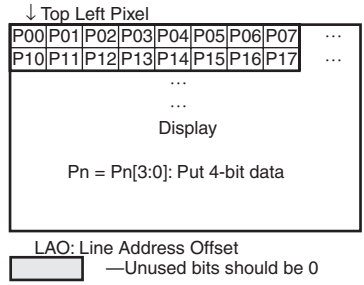
5. Packed 2bpp (Pixel Alignment in Byte is Little Endian)



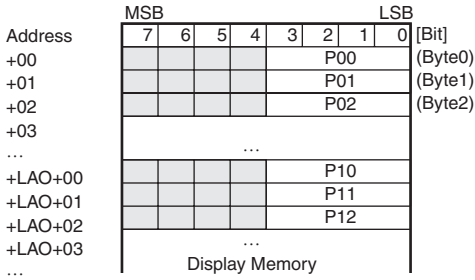


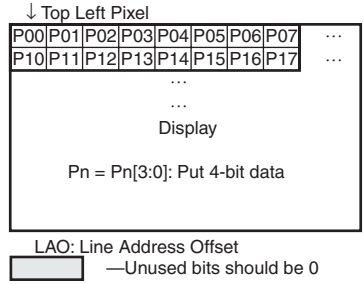
6. Packed 4bpp (Pixel Alignment in Byte is Little Endian)



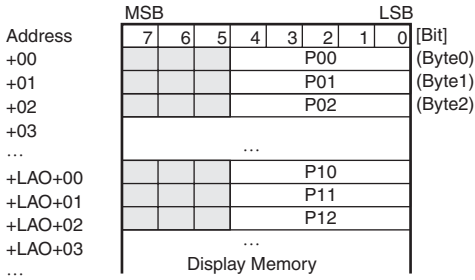


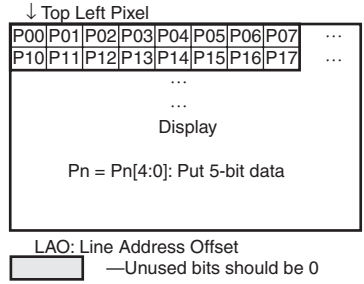
7. Unpacked 4bpp [Windows CE Recommended Format]



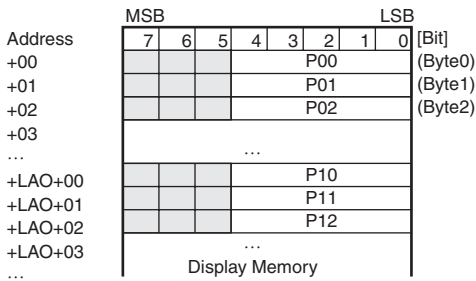


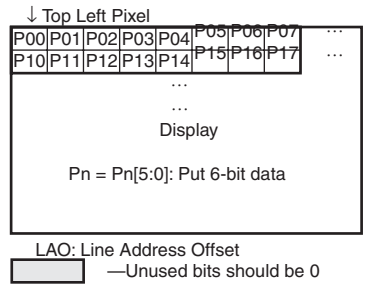
8. Unpacked 5bpp [Windows CE Recommended Format]



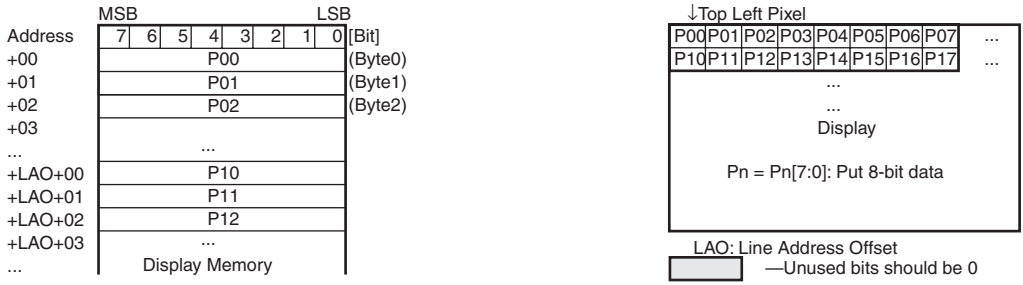


9. Unpacked 6bpp [Windows CE Recommended Format]

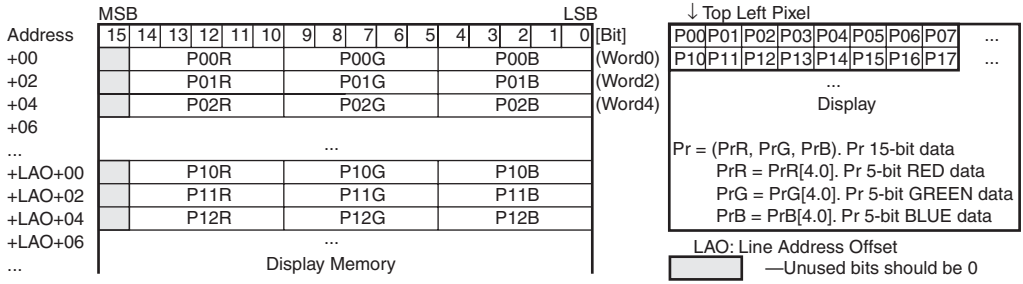




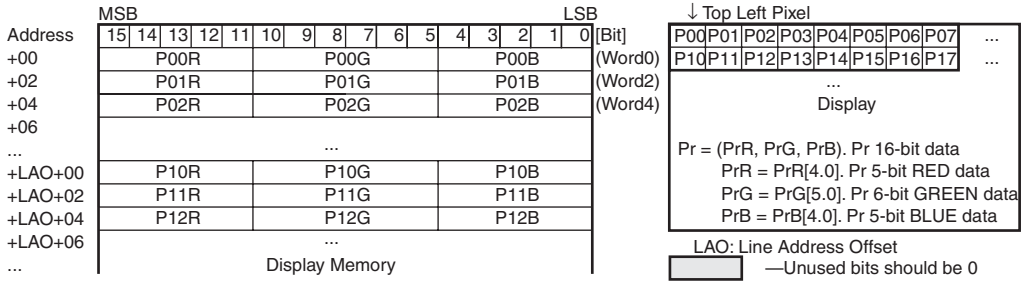
10. Packed 8bpp [Windows CE Recommended Format]



11. Unpacked color 15bpp (RGB 555) [Windows CE Recommended Format]



12. Packed color 16bpp (RGB 565) [Windows CE Recommended Format]



15.4.4 Setting the Display Resolution

The display resolution is set up in LDHCNR, LDHSYNR, LDVDLNR, LDVTLNR, and LDVSYNR. The LCD current-alternating period for an STN or DSTN display is set by using the LDACLNR. The initial values in these registers are typical settings for VGA (640 × 480 dots) on an STN or DSTN display.

The clock to be used is set with the LDICKR. The LCD module frame rate is determined by the display interval + retrace line interval (non-display interval) for one screen set in a size related register and the frequency of the clock used.

This LCDC has a Vsync interrupt function so that it is possible to issue an interrupt at the beginning of each vertical retrace line period (to be exact, at the beginning of the line after the last line of the display). This function is set up by using the LDINTR.

15.4.5 Power-Supply Control Sequence

An LCD module normally requires a specific sequence for processing to do with the cutoff of the input power supply. Settings in LDPMMR, LDPSPR, and LDCNTR, in conjunction with the LCD power-supply control pins (LCD_VCPWC, LCD_VEPWC, and LCD_DON), are used to provide processing of power-supply control sequences that suits the requirements of the LCD module.

Figures 15.4 to 15.7 are timing charts that show outlines of power-supply control sequences and table 15.5 is a summary of available power-supply control sequence periods.

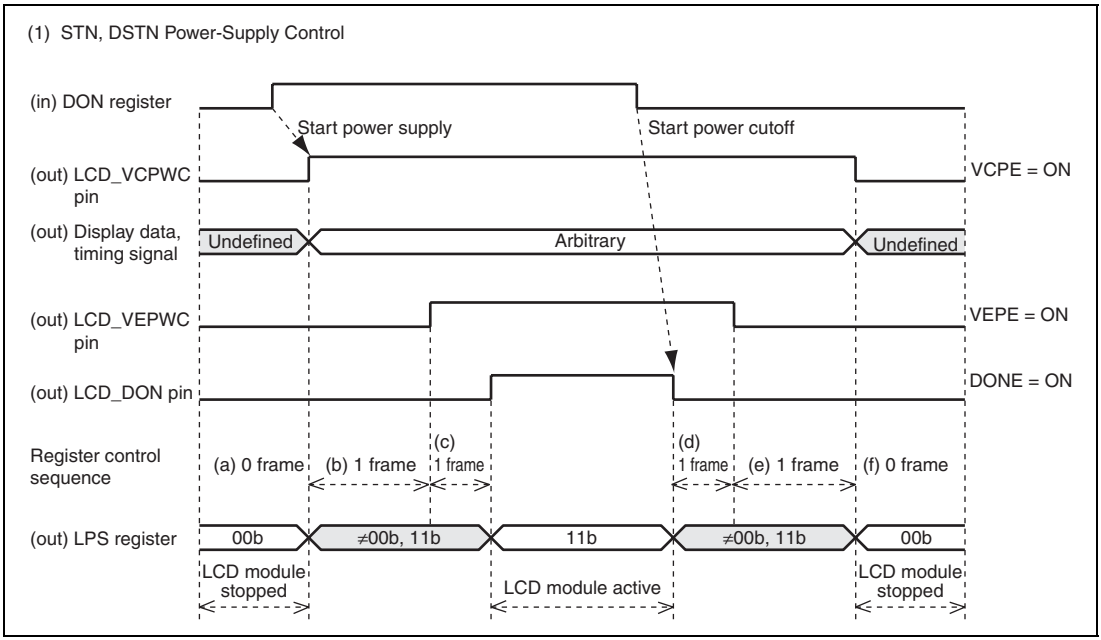


Figure 15.4 Power-Supply Control Sequence and States of the LCD Module

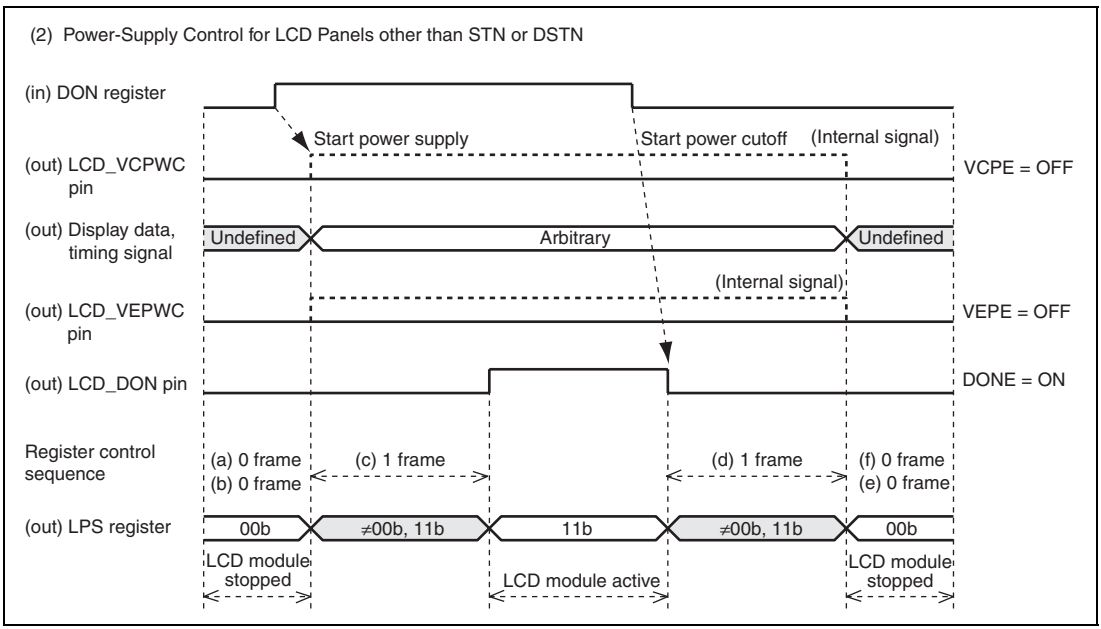


Figure 15.5 Power-Supply Control Sequence and States of the LCD Module

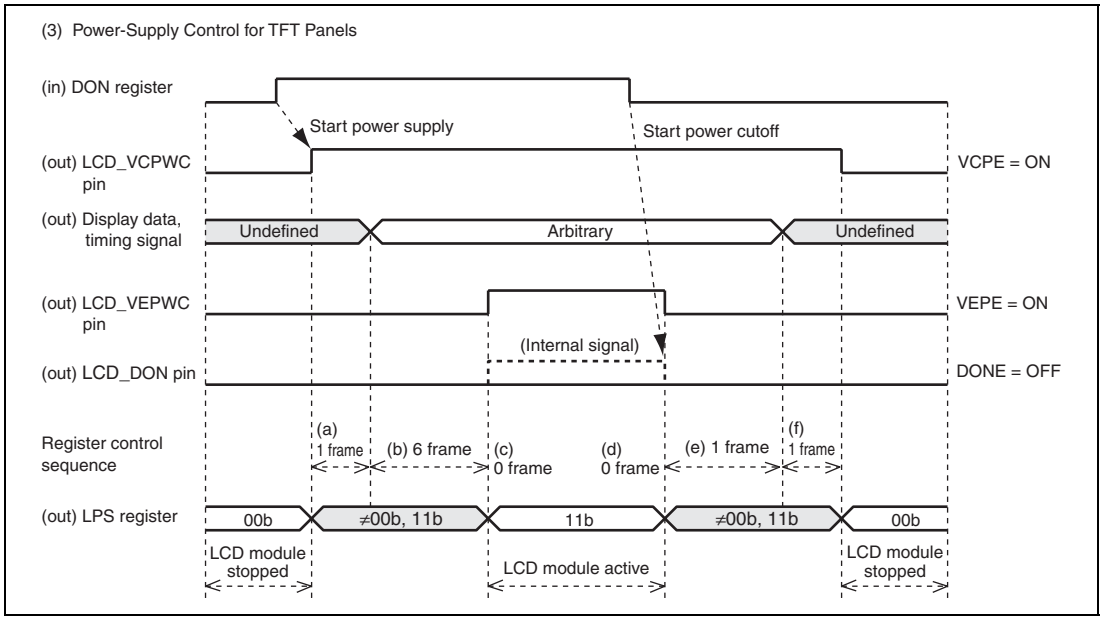


Figure 15.6 Power-Supply Control Sequence and States of the LCD Module

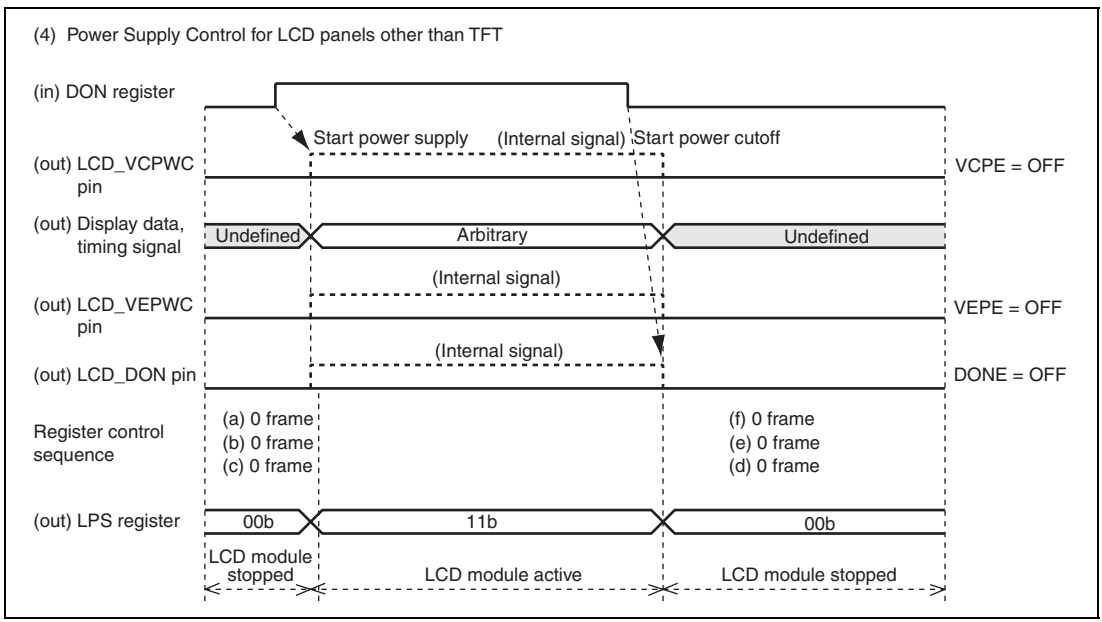


Figure 15.7 Power-Supply Control Sequence and States of the LCD Module

Table 15.5 Available Power-Supply Control-Sequence Periods at Typical Frame Rates

ONX, OFFX Register Value	Frame Rate	
	120 Hz	60 Hz
H'F	(–1+1)/120 = 0.00 (ms)	(–1+1)/60 = 0.00 (ms)
H'0	(0+1)/120 = 8.33 (ms)	(0+1)/60 = 16.67 (ms)
H'1	(1+1)/120 = 16.67 (ms)	(1+1)/60 = 33.33 (ms)
H'2	(2+1)/120 = 25.00 (ms)	(2+1)/60 = 50.00 (ms)
H'3	(3+1)/120 = 33.33 (ms)	(3+1)/60 = 66.67 (ms)
H'4	(4+1)/120 = 41.67 (ms)	(4+1)/60 = 83.33 (ms)
H'5	(5+1)/120 = 50.00 (ms)	(5+1)/60 = 100.00 (ms)
H'6	(6+1)/120 = 58.33 (ms)	(6+1)/60 = 116.67 (ms)
H'7	(7+1)/120 = 66.67 (ms)	(7+1)/60 = 133.33 (ms)
H'8	(8+1)/120 = 75.00 (ms)	(8+1)/60 = 150.00 (ms)
H'9	(9+1)/120 = 83.33 (ms)	(9+1)/60 = 166.67 (ms)
H'A	(10+1)/120 = 91.67 (ms)	(10+1)/60 = 183.33 (ms)
H'B	(11+1)/120 = 100.00 (ms)	(11+1)/60 = 200.00 (ms)
H'C	(12+1)/120 = 108.33 (ms)	(12+1)/60 = 216.67 (ms)
H'D	(13+1)/120 = 116.67 (ms)	(13+1)/60 = 233.33 (ms)
H'E	(14+1)/120 = 125.00 (ms)	(14+1)/60 = 250.00 (ms)

ONA, ONB, ONC, OFFD, OFFE, and OFFF are used to set the power-supply control-sequence periods, in units of frames, from 0 to 15. 1 is subtracted from each register. H'0 to H'E settings select from 1 to15 frames. The setting H'F selects 0 frames.

Actual sequence periods depend on the register values and the frame frequency of the display. The following table gives power-supply control-sequence periods for display frame frequencies used by typical LCD modules.

- When ONB is set to H'6 and display's frame frequency is 120 Hz
  The display's frame frequency is 120 Hz. 1 frame period is thus 8.33 (ms) = 1/120 (sec).
  The power-supply input sequence period is 7 frames because ONB setting is subtracted by 1.
  As a result, the sequence period is 58.33 (ms) = 8.33 (ms) × 7.

Table 15.6 LCDC Operating Modes

Mode		Function
Display on (LCDC active)	Register setting: DON = 1	Fixed resolution, the format of the data for display is determined by the number of colors, and timing signals are output to the LCD module.
Display off (LCDC stopped)	Register setting: DON = 0	Register access is enabled. Fixed resolution, the format of the data for display is determined by the number of colors, and timing signals are not output to the LCD module.

Table 15.7 LCD Module Power-Supply States

(STN, DSTN module)

State	Power Supply for Logic	Display Data, Timing Signal	Power Supply for High-Voltage Systems	DON Signal
Control Pin	LCD_VCPWC	LCD_CL2, LCD_CL1, LCD_FLM, LCD_M_DISP, LCD_D	LCD_VEPWC	LCD_DON
Operating State	Supply	Supply	Supply	Supply
(Transitional State)	Supply	Supply	Supply	
	Supply	Supply		
	Supply			
Stopped State				

(TFT module)

State	Power Supply for Logic	Display Data, Timing Signal	Power Supply for High-Voltage Systems
Control Pin	LCD_VCPWC	LCD_CL2, LCD_CL1, LCD_FLM, LCD_M_DISP, LCD_D	LCD_VEPWC
Operating State	Supply	Supply	Supply
(Transitional State)	Supply	Supply	
	Supply		
Stopped State			

The table above shows the states of the power supply, display data, and timing signals for the typical LCD module in its active and stopped states. Some of the supply voltages described may not be necessary, because some modules internally generate the power supply required for high-voltage systems from the logic-level power-supply voltage.

Notes on display-off mode (LCDC stopped):

If LCD module power-supply control-sequence processing is in use by the LCDC or the supply of power is cut off while the LCDC is in its display-on mode, normal operation is not guaranteed. In the worst case, the connected LCD module may be damaged.

15.5 Clock and LCD Data Signal Examples

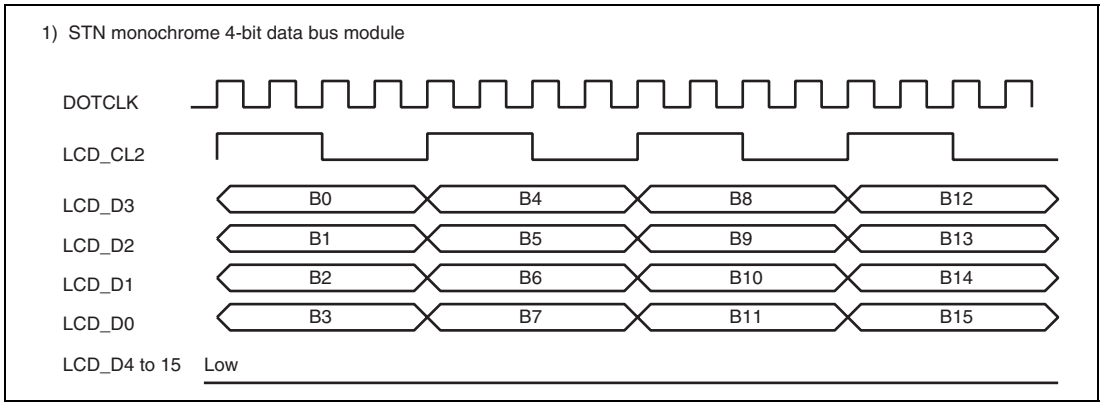


Figure 15.8 Clock and LCD Data Signal Example

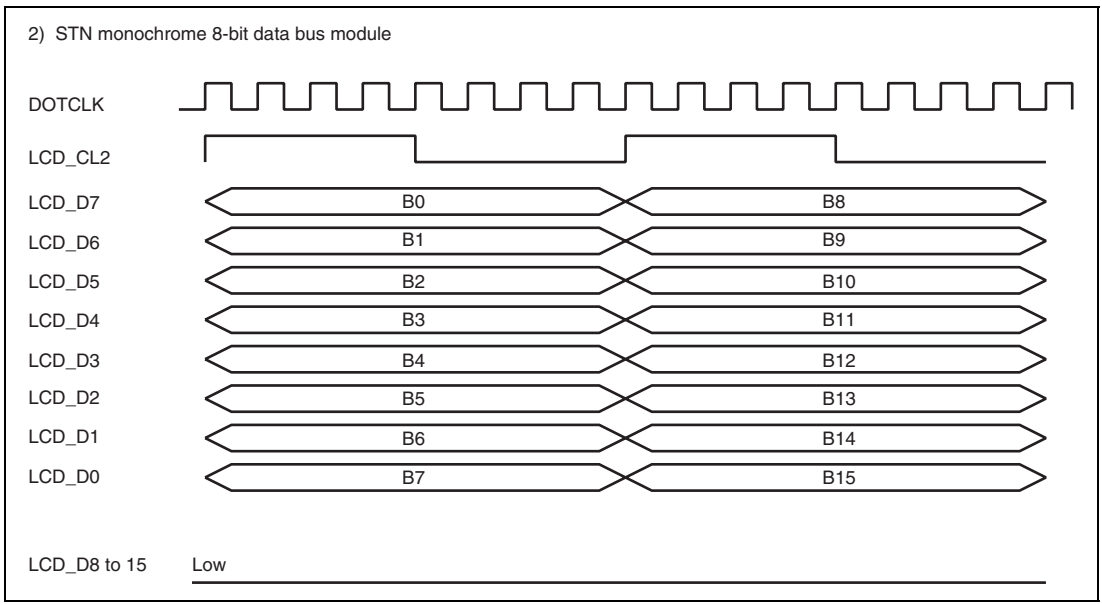


Figure 15.9 Clock and LCD Data Signal Example (STN Monochrome 8-Bit Data Bus Module)

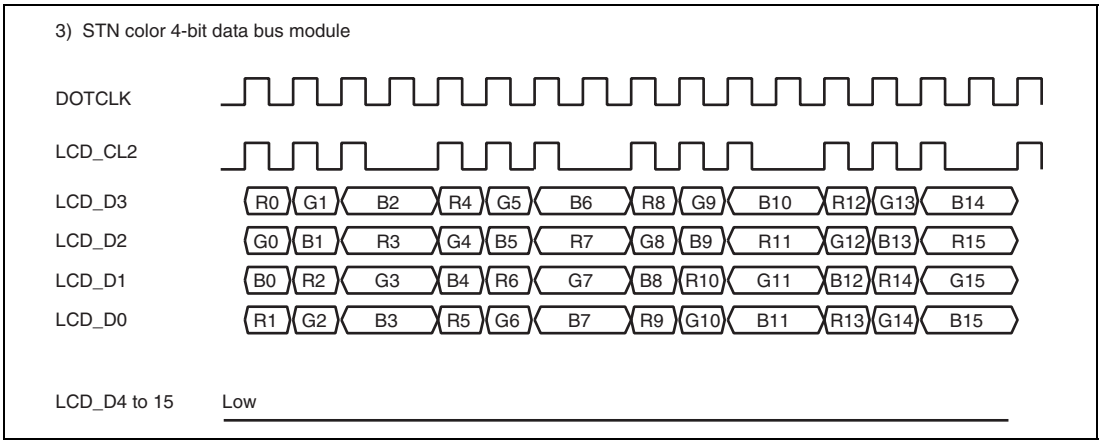


Figure 15.10 Clock and LCD Data Signal Example (STN Color 4-Bit Data Bus Module)

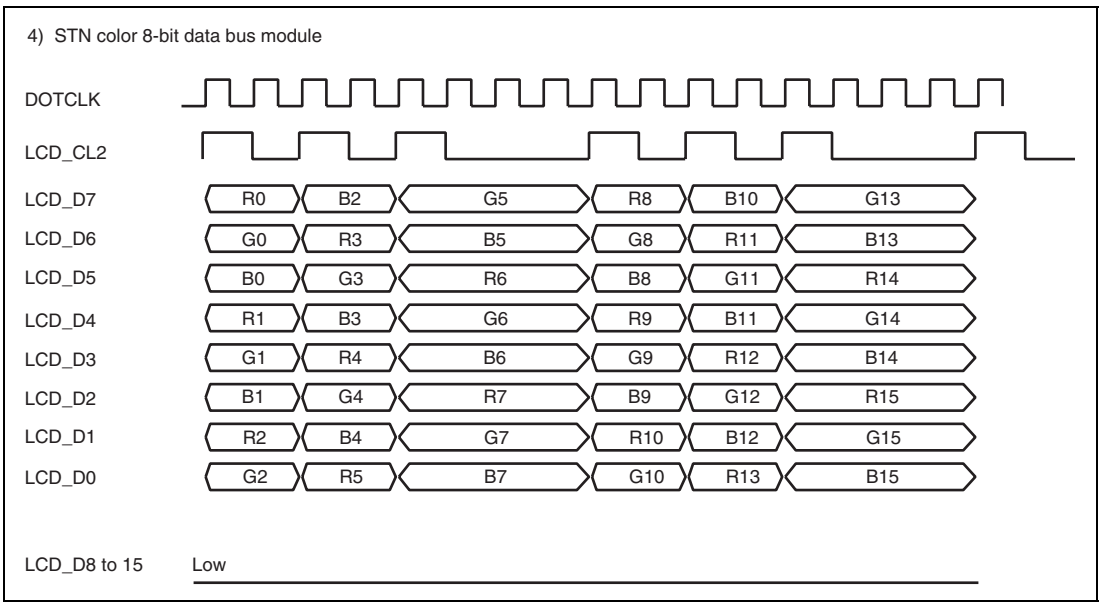


Figure 15.11 Clock and LCD Data Signal Example (STN Color 8-Bit Data Bus Module)

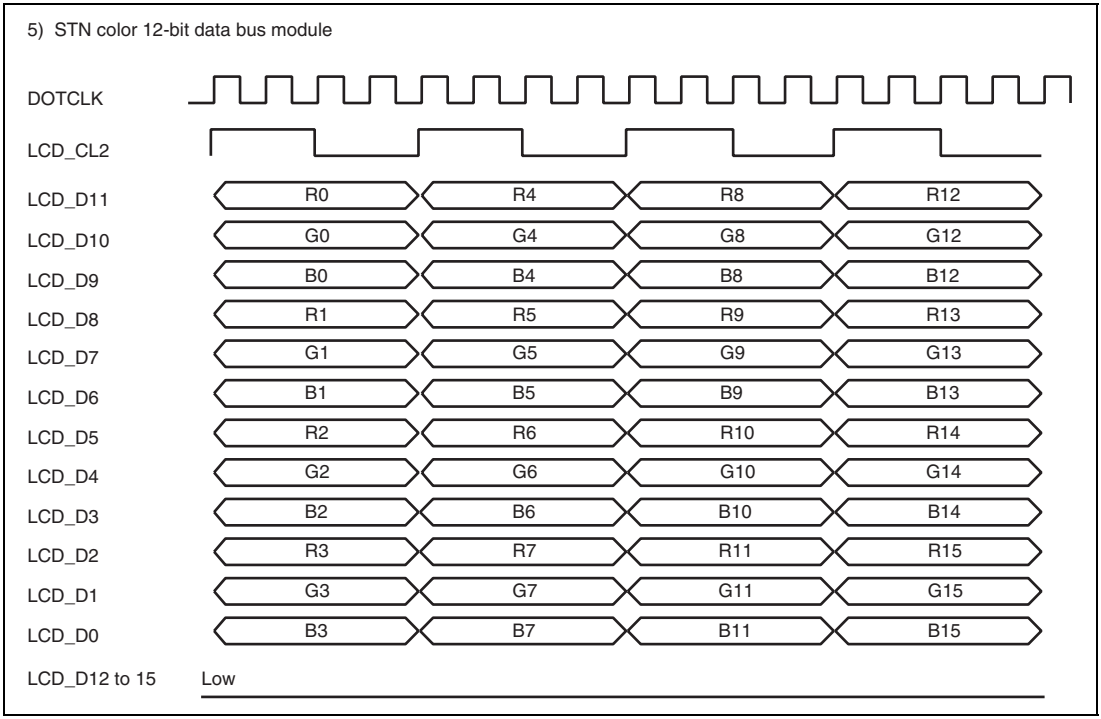


Figure 15.12 Clock and LCD Data Signal Example (STN Color 12-Bit Data Bus Module)

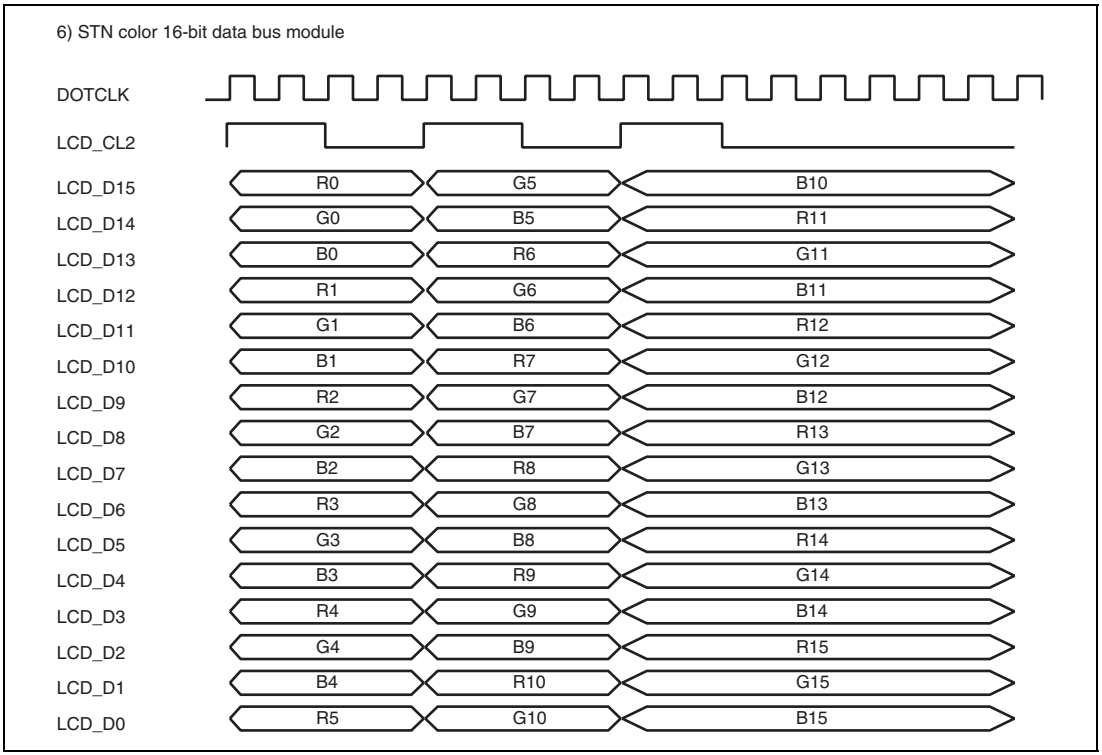


Figure 15.13 Clock and LCD Data Signal Example (STN Color 16-Bit Data Bus Module)

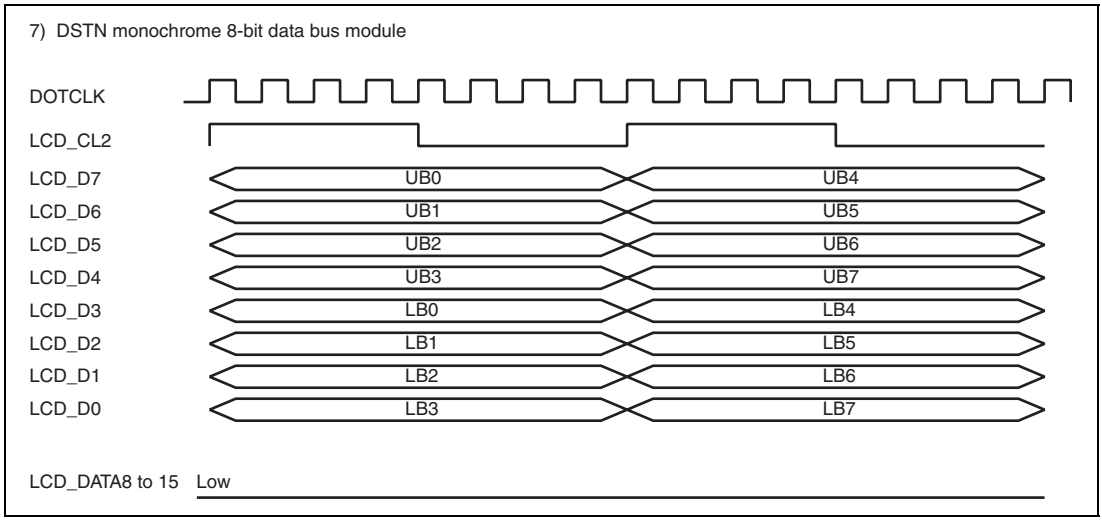


Figure 15.14 Clock and LCD Data Signal Example (DSTN Monochrome 8-Bit Data Bus Module)

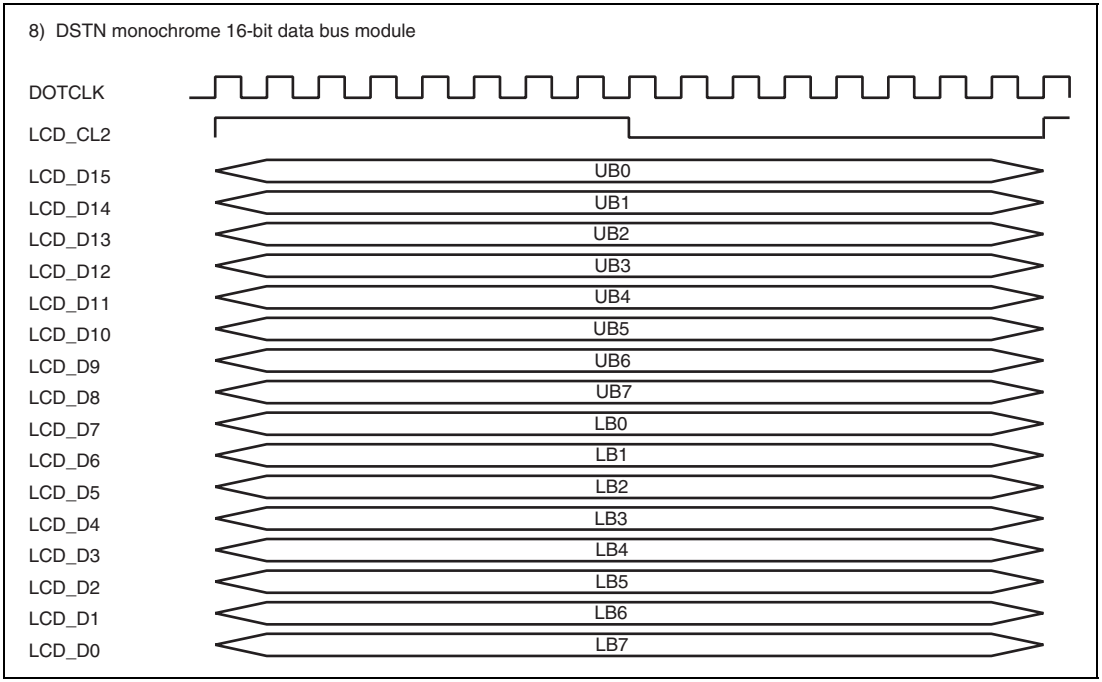


Figure 15.15 Clock and LCD Data Signal Example (DSTN Monochrome 16-Bit Data Bus Module)

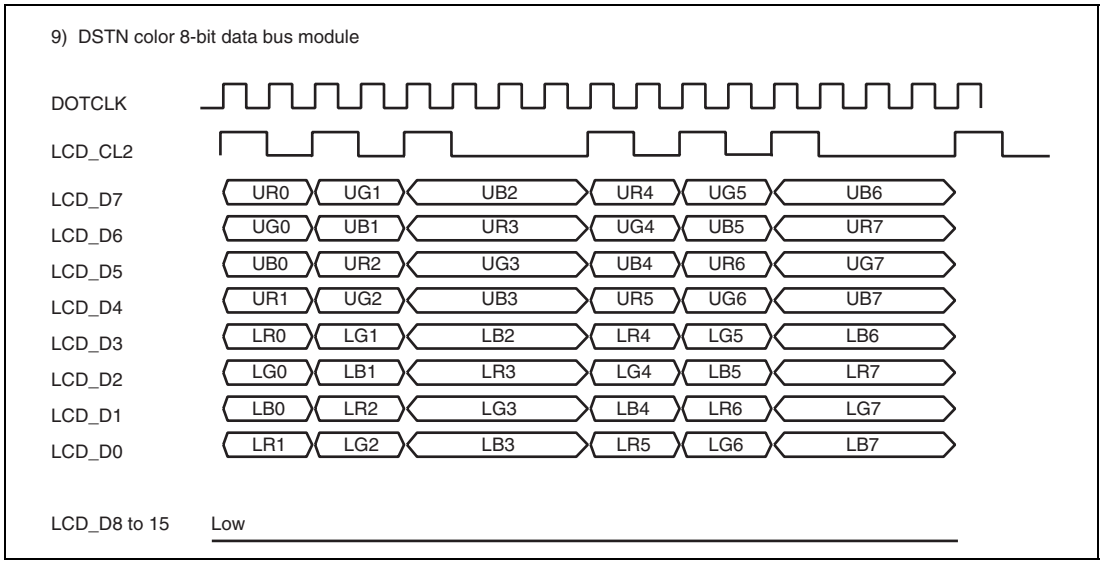


Figure 15.16 Clock and LCD Data Signal Example (DSTN Color 8-Bit Data Bus Module)

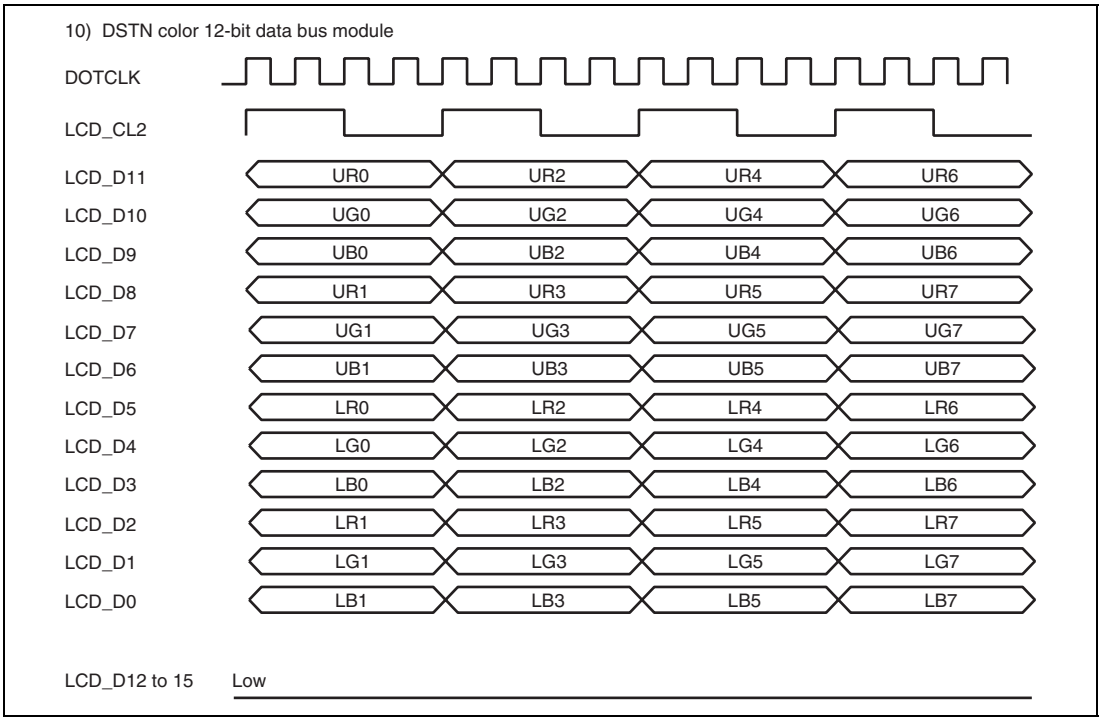


Figure 15.17 Clock and LCD Data Signal Example (DSTN Color 12-Bit Data Bus Module)

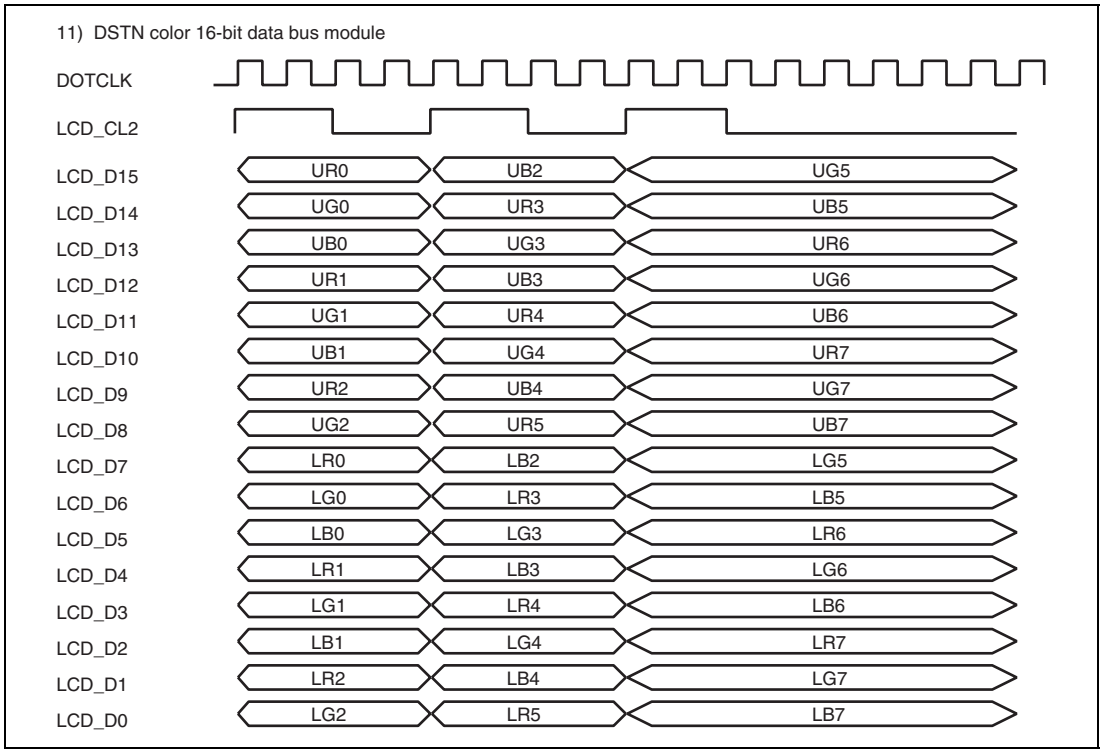


Figure 15.18 Clock and LCD Data Signal Example (DSTN Color 16-Bit Data Bus Module)

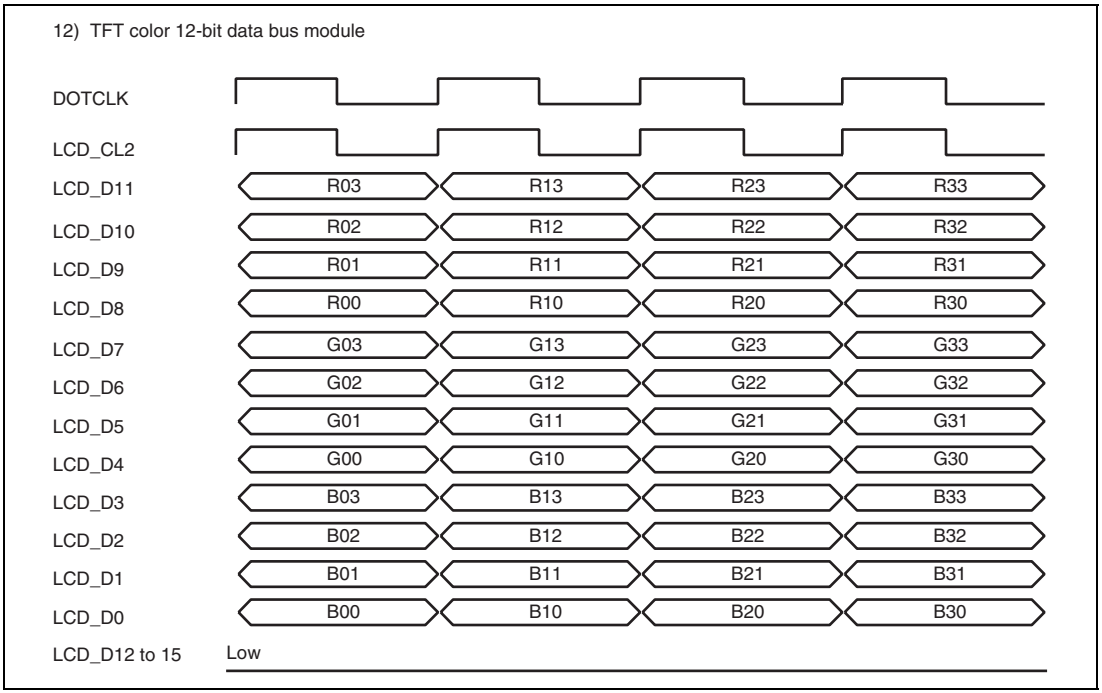


Figure 15.19 Clock and LCD Data Signal Example (TFT Color 12-Bit Data Bus Module)

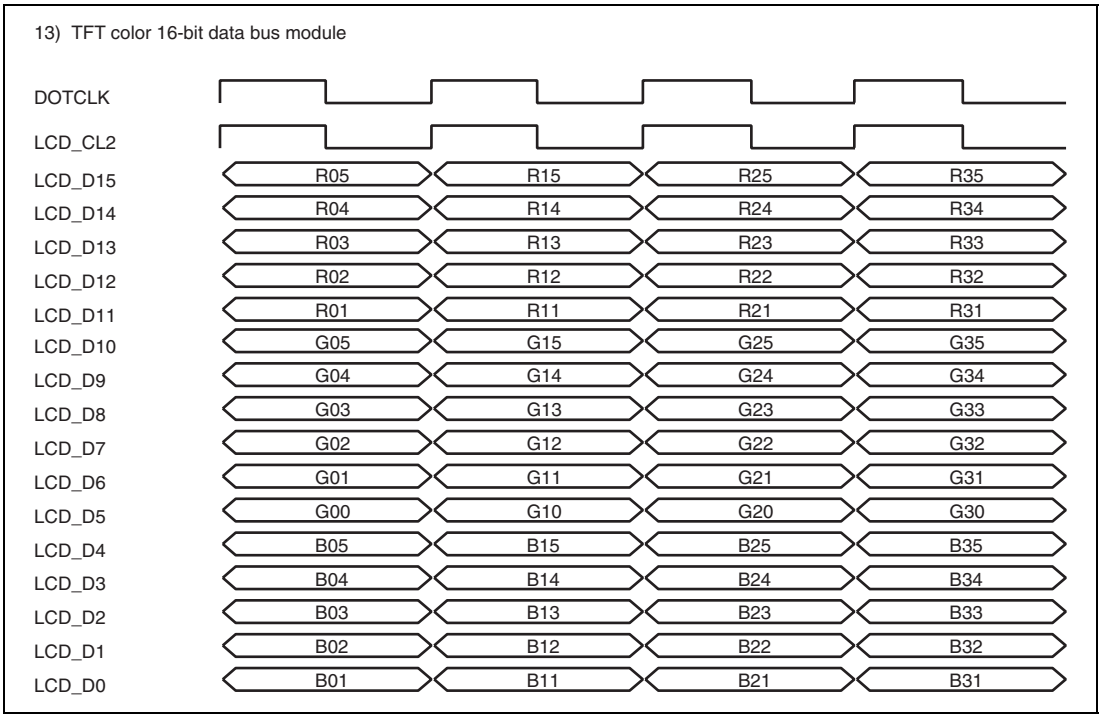


Figure 15.20 Clock and LCD Data Signal Example (TFT Color 16-Bit Data Bus Module)

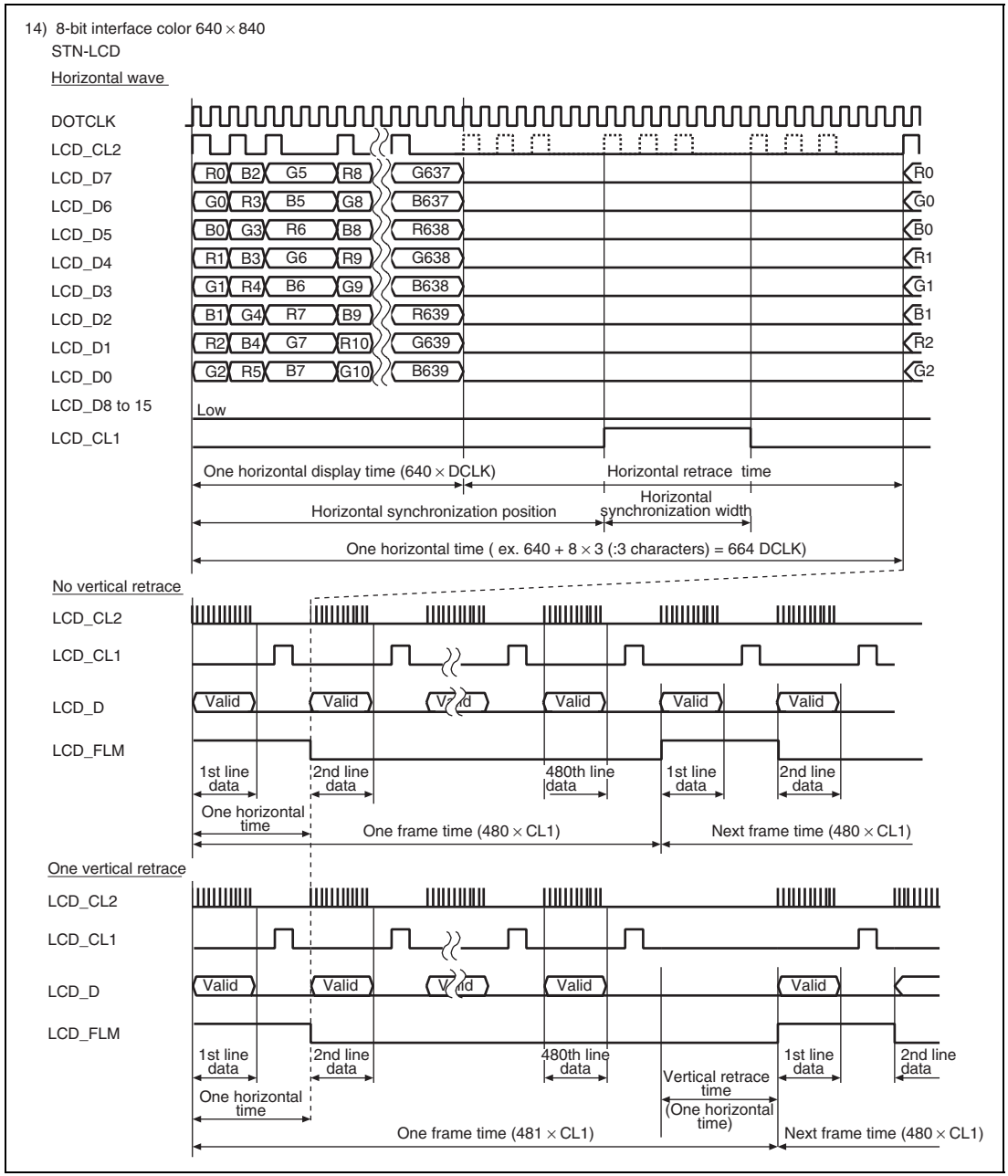


Figure 15.21 Clock and LCD Data Signal Example (8-Bit Interface Color 640 × 480)

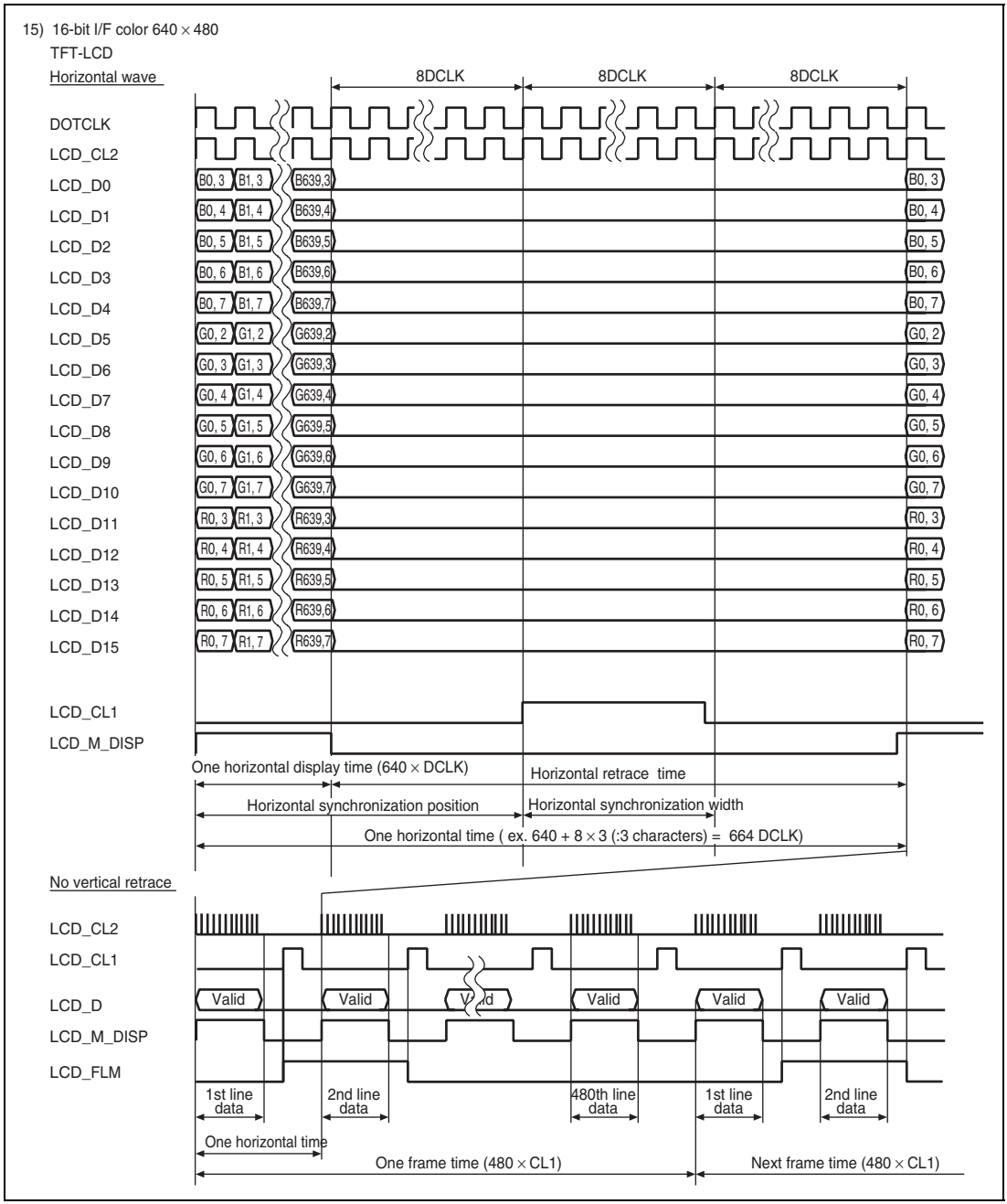


Figure 15.22 Clock and LCD Data Signal Example (16-Bit Interface Color 640 × 480)

15.6 Usage Notes

15.6.1 Procedure for Halting Access to Display Data Storage VRAM (Area 2 or 3)

Follow the procedure below to halt access to VRAM for storing display data (area 2 or 3).

Procedure for Halting Access to Display Data Storage VRAM:

1. Confirm that the LPS1 and LPS0 bits in LDPMMR are currently set to 1.
2. Clear the DON bit in LDCNTR to 0 (display-off mode).
3. Confirm that the LPS1 and LPS0 bits in LDPMMR have changed to 0.
4. Wait for the display time for a single frame to elapse.

This halting procedure is required before selecting self-refreshing for the display data storage VRAM (area 2 or 3) or making a transition to standby mode or module standby mode.

15.6.2 Notes on When Starting Display

At the start of display, confirm the state of the LCDC with the following procedure.

1. Write 11 to the DON2 and DON bits, dummy read the module standby control register 4 once, and then dummy read LDCNTR for n times to wait for the time it takes for the LPS[0] bit to change to 1.
2. If the LPS[0] bit does not change to 1, write to LDCNTR again.

Note: There are no changes in the procedure for halting display (setting the DON2 and DON bits to 00).

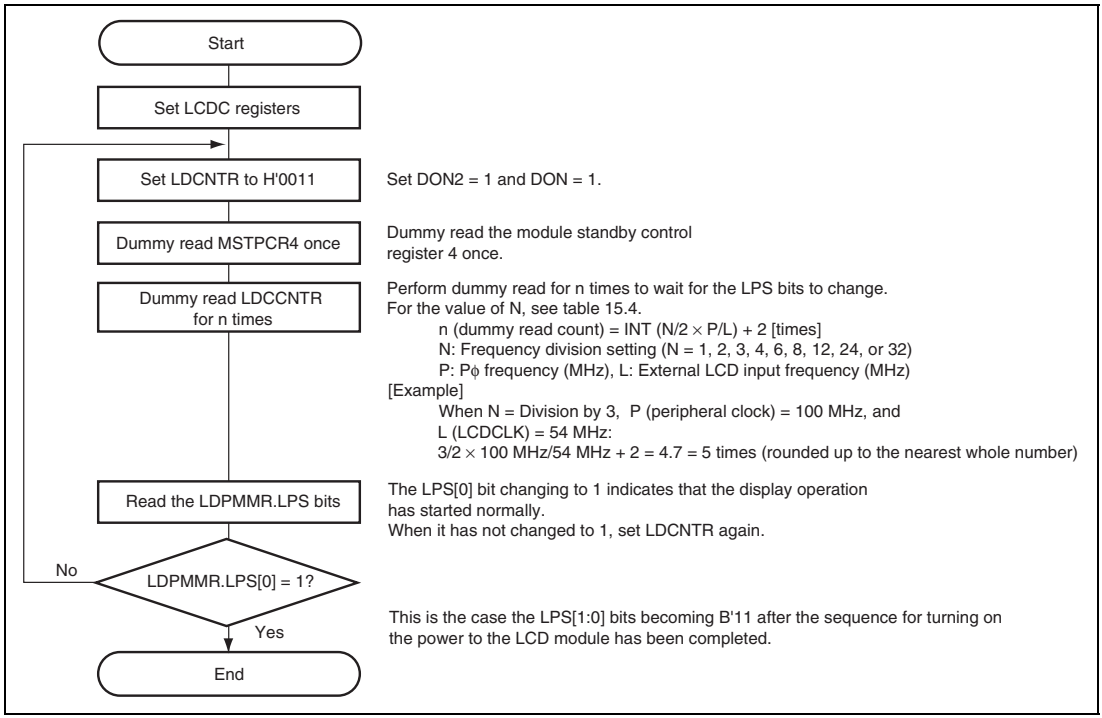


Figure 15.23 Procedure when Starting Display

Section 16 Serial Sound Interface and Clock Selector (SSS)

16.1 Features

Figure 16.1 shows a block diagram of the serial sound interface and clock selector (SSS). The SSS includes four serial sound-interface modules, and two baud rate generators for generating the AUDIO_CLK signals. Note that in this section clks and clkp refer to the SHwy clock and peripheral module clock, respectively.

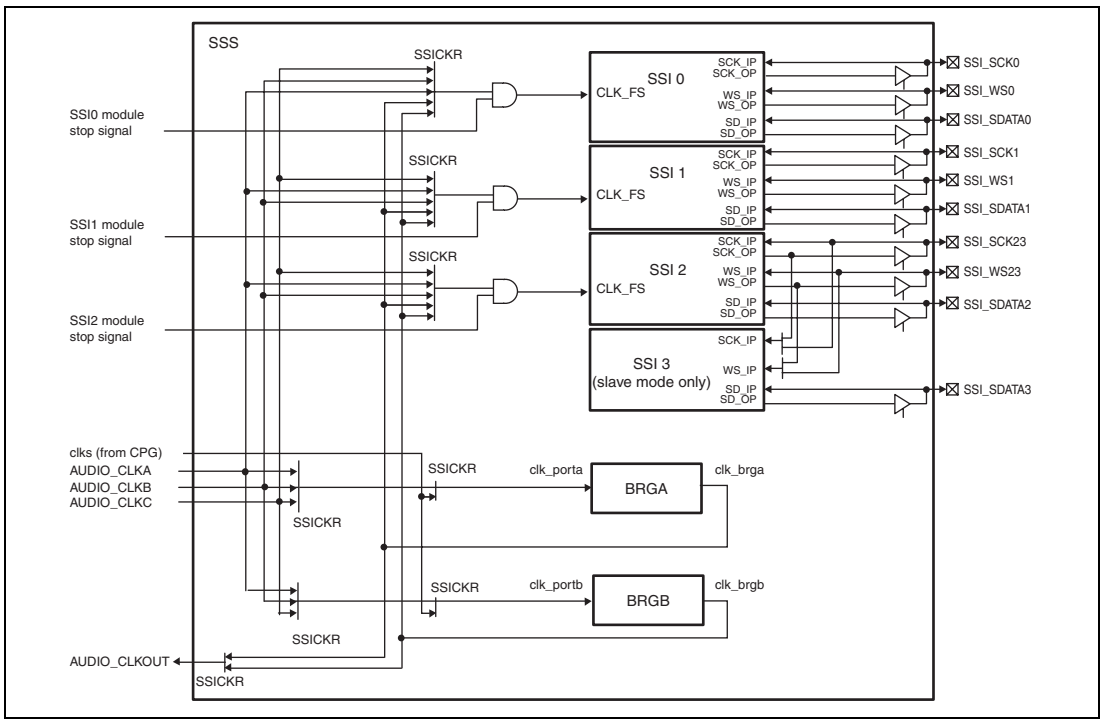


Figure 16.1 Block Diagram of SSS

16.2 Input/Output Pins

Table 16.1 shows the pin assignments relating to the SSS module.

Table 16.1 Pin Assignments

Pin Name	Number of Pins	I/O	Description
SSI_SCK0	1	I/O	Serial bit clock
SSI_WS0	1	I/O	Word selection (L/R clock)
SSI_SDATA0	1	I/O	Serial data input/output
SSI_SCK1	1	I/O	Serial bit clock
SSI_WS1	1	I/O	Word selection (L/R clock)
SSI_SDATA1	1	I/O	Serial data input/output
SSI_SCK23	1	I/O	Serial bit clock (shared by SSI2 and SSI3)
SSI_WS23	1	I/O	Word selection (L/R clock) (shared by SSI2 and SSI3)
SSI_SDATA2	1	I/O	Serial data input/output
SSI_SDATA3	1	I/O	Serial data input/output
AUDIO_CLKA	1	Input	Oversampling clock (Input 256/384/512 fs clock to the SSI modules.)
AUDIO_CLKB	1	Input	Oversampling clock (Input 256/384/512-fs clock to the SSI modules.)
AUDIO_CLKC	1	Input	Oversampling clock (Input 256/384/512-fs clock to the SSI modules.)
AUDIO_CLKOUT	1	Output	Frequency divided clock clk_brga or clk_brgb output

Notes: 1. SSI0 and SSI1 do not share any pins.
2. SSI2 and SSI3 share the word selection pin (SSI_WS23) and serial bit clock pin (SSI_SCK23), but have individual dedicated serial data I/O pins (SSI_SDATA2 and SSI_SDATA3).

16.3 Combinations of Connections in SSS

Table 16.2 shows combinations of connections in the SSS.

Table 16.2 Table of Combinations

Pin System	Pin Name	I/O	SSI0	SSI1	SSI2	SSI3	Access Size
SSI0	SSI_SCK0 SSI_WS0 SSI_SDATA0	out out out	master (transmitter)	—	—	—	Select one of the settings.
		out out in	master (receiver)	—	—	—	
		in in out	slave (transmitter)	—	—	—	
		in in in	slave (receiver)	—	—	—	
SSI1	SSI_SCK1 SSI_WS1 SSI_SDATA1	out out out	—	master (transmitter)	—	—	Select one of the settings.
		out out in	—	master (receiver)	—	—	
		in in out	—	slave (transmitter)	—	—	
		in in in	—	slave (receiver)	—	—	

Pin System	Pin Name	I/O	SSI0	SSI1	SSI2	SSI3	Access Size
SSI2, SSI3	SSI_SCK23 SSI_WS23 SSI_SDATA2 SSI_SDATA3	out out out out	—	—	master (transmitter)	slave (transmitter)	Select one of the settings. Settings in a bold line frame can be selected as one set.
		out out out in	—	—	master (transmitter)	slave (receiver)	
		out out in out	—	—	master (receiver)	slave (transmitter)	
		out out in in	—	—	master (receiver)	slave (receiver)	
		in in out out	—	—	slave (transmitter)	slave (transmitter)	
		in in out in	—	—	slave (transmitter)	slave (receiver)	

Pin System	Pin Name	I/O	SSI0	SSI1	SSI2	SSI3	Access Size
SSI2, SSI3	SSI_SCK23	in	—	—	slave (receiver)	slave (transmitter)	Select one of the settings. Settings in a bold line frame can be selected as one set.
	SSI_WS23	in					
	SSI_SDATA2	in					
	SSI_SDATA3	out					
		in	—	—	slave (receiver)	slave (receiver)	
		in					
		in					
		in					

[Legend]

—: Not connected

16.4 Configuration of Baud Rate Generator Registers

Table 16.3 shows the configuration of the baud rate generator registers.

Table 16.3 (1) Configuration of Baud Rate Generator Registers

Register Name	Abbreviation	Initial Value	Offset Address	Access Size
BRGA baud rate setting register	BRRA	H'0000_00FF	H'FFFE 0104	32 bits
BRGB baud rate setting register	BRRB	H'0000_00FF	H'FFFE 010C	32 bits
Clock select register	SSICKR	H'2300_0000	H'FFFE 0118	32 bits

Table 16.3 (2) Conditions for Initializing Baud Rate Generator Registers

Abbreviation	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
BRRA	Initialized	Initialized	Retained	Retained	Retained	Initialized
BRRB	Initialized	Initialized	Retained	Retained	Retained	Initialized
SSICKR	Initialized	Initialized	Retained	Retained	Retained	Initialized

16.4.1 BRGA Baud Rate Setting Register (BRRA: H'0104)

BRRA is a 32-bit readable/writable register that specifies a baud rate for the clock when clk_porta is selected as a baud rate generator.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CKS[1:0]		BRRA[7:0]							
Initial value:	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	CKS[1:0]	00	R/W	These bits specify the clock source for the internal baud rate generator. 00: clk_porta 01: clk_porta/4 10: clk_porta/16 11: clk_porta/64
7 to 0	BRRA[7:0]	H'FF	R/W	These bits specify the division ratio. For details, see table 16.4.

Table 16.4 shows the division ratio for BRGA.

Table 16.4 Division Ratio for BRGA

BRGA Operating Clock (CKS1, CKS0)	Division Ratio BRRA (N = 0 to 255)	Calculating Formula
clk_porta	1/2, 1/4, 1/6 to 1/512	1/(2(N + 1))
clk_porta/4	1/8, 1/16, 1/24 to 1/2048	1/(8(N + 1))
clk_porta/16	1/32, 1/64, 1/96 to 1/8192	1/(32(N + 1))
clk_porta/64	1/128, 1/256, 1/384 to 1/32768	1/(128(N + 1))

16.4.2 BRGB Baud Rate Setting Register (BRRB: H'010C)

BRRB is a 32-bit readable/writable register that specifies a baud rate for the clock when clk_portb is selected as a baud rate generator.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CKS[1:0]		BRRB[7:0]							
Initial value:	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	CKS[1:0]	00	R/W	These bits specify the clock source for the internal baud rate generator. 00: clk_portb 01: clk_portb/4 10: clk_portb/16 11: clk_portb/64
7 to 0	BRRB[7:0]	H'FF	R/W	These bits specify the division ratio. For details, see table 16.5.

Table 16.5 shows the division ratio for BRGB.

Table 16.5 Division Ratio for BRGB

BRGB Operating Clock (CKS1, CKS0)	Division Ratio BRRB (N = 0 to 255)	Calculating Formula
clk_portb	1/2, 1/4, 1/6 to 1/512	1/(2(N + 1))
clk_portb/4	1/8, 1/16, 1/24 to 1/2048	1/(8(N + 1))
clk_portb/16	1/32, 1/64, 1/96 to 1/8192	1/(32(N + 1))
clk_portb/64	1/128, 1/256, 1/384 to 1/32768	1/(128(N + 1))

16.4.3 Clock Select Register (SSICKR: H'0118)

SSICKR selects the source of the clock signal for input to or output from the SSS.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SSICKR[31]	SSICKR[30:28]			—	SSICKR[26:24]			—	SSICKR[22:20]			—	SSICKR[18:16]		
Initial value:	0	0	1	0	0	0	1	1	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	SSICKR[14:12]			—	SSICKR[10:8]			—	SSICKR[6:4]			—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	SSICKR[31]	0	R/W	This bit selects the source of the clock signal for output on the AUDIO_CLKOUT external pin. 1: BRGB output clock signal 0: BRGA output clock signal
30 to 28	SSICKR [30:28]	010	R/W	These bits select the source of an audio clock signal (clk_audio_ai) for input to the FSI. 11x: 1'b0 101: 1'b0 100: AUDIO_CLKC 011: BRGB output clock signal 010: BRGA output clock signal 001: AUDIO_CLKB 000: AUDIO_CLKA
27	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
26 to 24	SSICKR [26:24]	011	R/W	These bits select the source of an audio clock signal (clk_audio_bi) for input to the FSI. 11x: 1'b0 101: 1'b0 100: AUDIO_CLKC 011: BRGB output clock signal 010: BRGA output clock signal 001: AUDIO_CLKB 000: AUDIO_CLKA
23	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
22 to 20	SSICKR [22:20]	000	R/W	These bits select the input clock signal for BRGA. 11x: 1'b0 101: 1'b0 100: AUDIO_CLKC 01x: clks 001: AUDIO_CLKB 000: AUDIO_CLKA
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
18 to 16	SSICKR [18:16]	000	R/W	These bits select the input clock signal for BRGB. 11x: 1'b0 101: 1'b0 100: AUDIO_CLKC 01x: clks 001: AUDIO_CLKB 000: AUDIO_CLKA
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
14 to 12	SSICKR [14:12]	000	R/W	These bits select the input clock signal for SSI0. 11x: 1'b0 101: 1'b0 100: AUDIO_CLKC 011: AUDIO_CLKB 010: AUDIO_CLKA 001: BRGB output clock signal 000: BRGA output clock signal
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10 to 8	SSICKR [10:8]	000	R/W	These bits select the input clock signal for SSI1. 11x: 1'b0 101: 1'b0 100: AUDIO_CLKC 011: AUDIO_CLKB 010: AUDIO_CLKA 001: BRGB output clock signal 000: BRGA output clock signal
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6 to 4	SSICKR [6:4]	000	R/W	These bits select the input clock signal for SSI2. 11x: 1'b0 101: 1'b0 100: AUDIO_CLKC 011: AUDIO_CLKB 010: AUDIO_CLKA 001: BRGB output clock signal 000: BRGA output clock signal
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Section 16A Serial Sound Interface (SSI)

The serial sound-interface (hereinafter referred to as the "SSI") module is a transceiver module designed to send or receive audio data interfacing with a variety of devices offering Philips format. It also supports burst and multi-channel modes in addition to other common formats.

16A.1 Features

The SSI has the following features:

- Number of channels: Maximum of four channels
- Operating mode: Compressed mode and uncompressed mode
  The compressed mode is used to transfer continuous bit streams.
  The uncompressed mode supports serial audio streams divided by channels.
- The SSI module can serve as both a transmitter and a receiver. It can also use the serial bus format, irrespective of the compressed or uncompressed mode.
- Asynchronous transfer takes place between the data buffer and the shift register.
- It is possible to select a value as the dividing ratio for the clock used by the serial bus interface.
- It is possible to control data transmission or reception with DMAC or interrupt requests.
- TDM mode is supported.
- TDM mode operation is performed at a 44.1- or 48-kHz sampling rate.
- WS continue mode in which WS signal is not stopped is supported.

Figure 16A.1 shows a block diagram of a single SSI module.

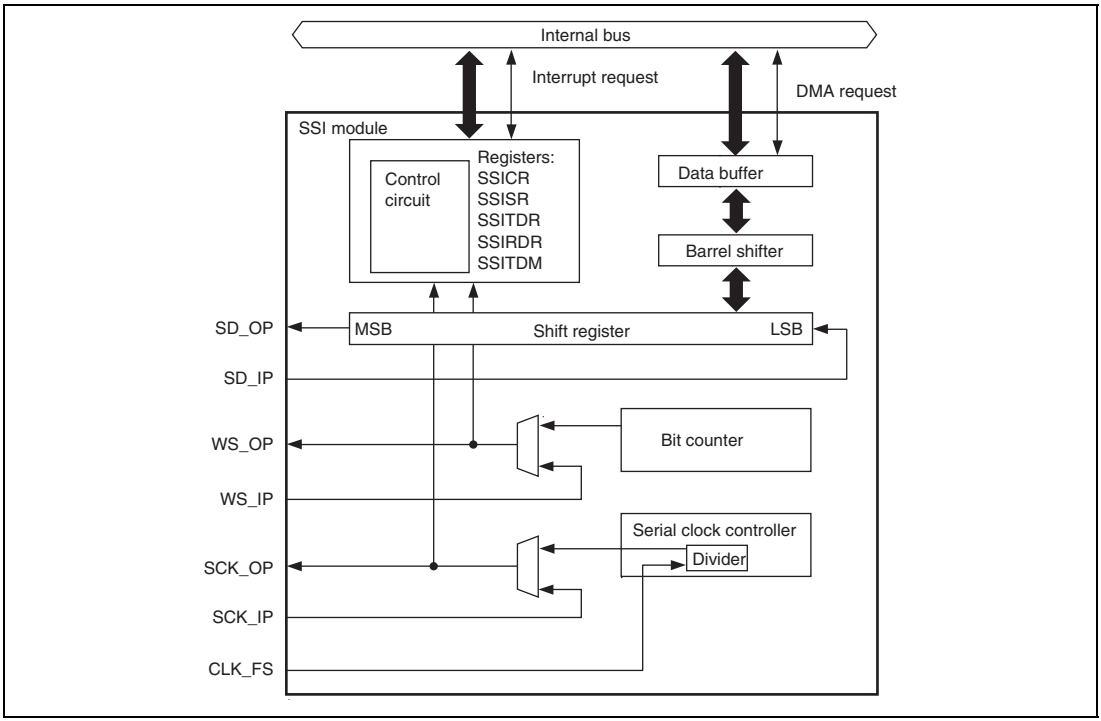


Figure 16A.1 Block Diagram of SSI

16A.2 Register Descriptions

The SSI has the following registers. Note that the module numbers are basically omitted from the register names in the text.

Table 16A.1 (1) Register Descriptions

Module	Register Name	Abbreviation	R/W	Address	Access Size
0	Control register 0	SSICR0	R/W	H'FFE0 0000	32
	Status register 0	SSISR0	R/W*	H'FFE0 0004	32
	Transmit data register 0	SSITDR0	R/W	H'FFE0 0008	32
	Receive data register 0	SSIRDR0	R	H'FFE0 000C	32
	TDM mode register 0	SSITDM0	R/W	H'FFE0 0020	32
1	Control register 1	SSICR1	R/W	H'FFE1 0000	32
	Status register 1	SSISR1	R/W*	H'FFE1 0004	32
	Transmit data register 1	SSITDR1	R/W	H'FFE1 0008	32
	Receive data register 1	SSIRDR1	R	H'FFE1 000C	32
	TDM mode register 1	SSITDM1	R/W	H'FFE1 0020	32
2	Control register 2	SSICR2	R/W	H'FFE2 0000	32
	Status register 2	SSISR2	R/W*	H'FFE2 0004	32
	Transmit data register 2	SSITDR2	R/W	H'FFE2 0008	32
	Receive data register 2	SSIRDR2	R	H'FFE2 000C	32
	TDM mode register 2	SSITDM2	R/W	H'FFE2 0020	32
3	Control register 3	SSICR3	R/W	H'FFE3 0000	32
	Status register 3	SSISR3	R/W*	H'FFE3 0004	32
	Transmit data register 3	SSITDR3	R/W	H'FFE3 0008	32
	Receive data register 3	SSIRDR3	R	H'FFE3 000C	32
	TDM mode register 3	SSITDM3	R/W	H'FFE3 0020	32

Note: Do not write to any addresses other than listed above. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than listed above are undefined.

* For this register, bits 26 and 27 are readable/writable bits, although the others are read-only bits. For details, refer to section 16A.2.2, Status Register (SSISRn) (n = 0 to 3).

Table 16A.1 (2) Register State in Each Operating Mode

Abbreviation (n)	Initial Value	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
SSICRn	H'00000000	Initialized	Initialized	Retained	Retained	Retained	Initialized
SSISRn	Bits 28/27/26/24/3/2 = 0, bit 1 = 1 (Undefined)	Initialized	Initialized	Retained	Retained	Retained	Initialized
SSITDRn	H'00000000	Initialized	Initialized	Retained	Retained	Retained	Initialized
SSIRDRn	H'00000000	Initialized	Initialized	Retained	Retained	Retained	Initialized
SSITDMn	H'00000000	Initialized	Initialized	Retained	Retained	Retained	Initialized

[Legend for Register Description]

Initial value: Register value after a reset.

—: Undefined value

R/W: Readable/writable. The written value can be read.

R/WC0: Readable/writable. Only writing 0 initializes the bit. Writing 1 is ignored.

R: Read-only. The write value should always be 0.

16A.2.1 Control Register (SSICRn) (n = 0 to 3)

SSICR is a readable/writable 32-bit register that controls the IRQ, selects the polarity status, and sets operating mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	DMEN	UIEN	OIEN	IIEN	DIEN	CHNL1	CHNL0	DWL2	DWL1	DWL0	SWL2	SWL1	SWL0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SCKD	SWSD	SCKP	SWSP	SPDP	SDTA	PDTA	DEL	BREN	CKDV2	CKDV1	CKDV0	MUEN	CPEN	TRMD	EN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	000	R	Reserved The read value is undefined. The write value should always be 0.
28	DMEN	0	R/W	DMA Enable Enables or disables the DMA request. 0: DMA request is disabled. 1: DMA request is enabled.
27	UIEN	0	R/W	Underflow Interrupt Enable 0: Underflow interrupt is disabled. 1: Underflow Interrupt is enabled.
26	OIEN	0	R/W	Overflow Interrupt Enable 0: Overflow interrupt is disabled. 1: Overflow interrupt is enabled.
25	IIEN	0	R/W	Idle Mode Interrupt Enable 0: Idle mode interrupt is disabled. 1: Idle mode interrupt is enabled.
24	DIEN	0	R/W	Data Interrupt Enable 0: Data interrupt is disabled. 1: Data interrupt is enabled.

Bit	Bit Name	Initial Value	R/W	Description
23, 22	CHNL[1:0]	00	R/W	Channels These bits indicate the number of channels in each system word. These bits are ignored if CPEN = 1. 00: A system word has one channel. 01: A system word has two channels. 10: A system word has three channels. 11: A system word has four channels. If TDM is set to 1 in SSITDM, these bits indicate as follows: 00: Setting prohibited 01: A TDM frame consists of four system words. 10: A TDM frame consists of six system words. 11: A TDM frame consists of eight system words.
21 to 19	DWL[2:0]	000	R/W	Data Word Length Indicate the number of bits in a data word. These bits are ignored if CPEN = 1. 000: 8 bits 001: 16 bits 010: 18 bits 011: 20 bits 100: 22 bits 101: 24 bits 110: 32 bits 111: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
18 to 16	SWL[2:0]	000	R/W	System Word Length Indicate the number of bits in a system word. These bits are ignored if CPEN = 1. 000: 8 bits 001: 16 bits 010: 24 bits 011: 32 bits 100: 48 bits 101: 64 bits 110: 128 bits 111: 256 bits
15	SCKD	0	R/W	Serial Bit Clock Direction 0: Serial bit clock is input, slave mode. 1: Serial bit clock is output, master mode. Note: In uncompressed mode (CPEN = 0), (SCKD, SWSD) = (0,0) or (1,1) can be set to SSI0 to SSI2. To SSI3, only (SCKD, SWSD) = (0,0) can be set. Other settings are prohibited.
14	SWSD	0	R/W	Serial WS Direction 0: Serial word select is input, slave mode. 1: Serial word select is output, master mode. Note: In uncompressed mode (CPEN = 0), (SCKD, SWSD) = (0,0) or (1,1) can be set to SSI0 to SSI2. To SSI3, only (SCKD, SWSD) = (0,0) can be set. Other settings are prohibited.

Bit	Bit Name	Initial Value	R/W	Description
13	SCKP	0	R/W	Serial Bit Clock Polarity 0: SSI_WS and SSI_SDATA change at the SSI_SCK falling edge (sampled at the SCK rising edge). 1: SSI_WS and SSI_SDATA change at the SSI_SCK rising edge (sampled at the SCK falling edge).

	SCKP = 0	SCKP = 1
SSI_SDATA input sampling timing at the time of reception (TRMD = 0)	SSI_SCK rising edge	SSI_SCK falling edge
SSI_SDATA output change timing at the time of transmission (TRMD = 1)	SSI_SCK falling edge	SSI_SCK rising edge
SSI_WS input sampling timing at the time of slave mode (SWSD = 0)	SSI_SCK rising edge	SSI_SCK falling edge
SSI_WS output change timing at the time of master mode (SWSD = 1)	SSI_SCK falling edge	SSI_SCK rising edge

12	SWSP	0	R/W	Serial WS Polarity The function of this bit depends on whether the module is in uncompressed mode or compressed mode. • CPEN = 0 (Uncompressed mode) 0: SSI_WS is low for 1st channel, high for 2nd channel. 1: SSI_WS is high for 1st channel, low for 2nd channel. • CPEN = 1 (Compressed mode) 0: SSI_WS is active high flow control. When WS is high, data is transferred. When low, data is not transferred. 1: SSI_WS is active low flow control. When WS is low, data is transferred. When high, data is not transferred. When TDM = 1 in SSITDM, this bit should be cleared to 0. The SYNC pulse is high during the system word 1, and low during the other periods. Note: Do not modify this bit when EN = 1.

Bit	Bit Name	Initial Value	R/W	Description
11	SPDP	0	R/W	Serial Padding Polarity This bit is ignored if CPEN = 1. 0: Padding bits are low. 1: Padding bits are high. Padding bits are low when MUEN = 1. (The MUTE function takes priority.)
10	SDTA	0	R/W	Serial Data Alignment This bit is ignored if CPEN = 1. 0: Transmitting and receiving in the order of serial data and padding bits 1: Transmitting and receiving in the order of padding bits and serial data
9	PDTA	0	R/W	Parallel Data Alignment This bit is ignored if CPEN = 1. When the data word length is 32, 16 or 8 bit, this configuration field has no meaning. This bit applies to SSIRDR in receive mode and SSITDR in transmit mode. 0: Parallel data (SSITDR, SSIRDR) is left-aligned 1: Parallel data (SSITDR, SSIRDR) is right-aligned. • DWL = 000 (with a data word length of 8 bits), the PDTA setting is ignored. All data bits in SSIRDR or SSITDR are used on the audio serial bus. Four data words are transmitted or received at each 32-bit access. The first data word is derived from bits 7 to 0, the second from bits 15 to 8, the third from bits 23 to 16 and the last data word is derived from bits 31 to 24.

Bit	Bit Name	Initial Value	R/W	Description
9	PDTA	0	R/W	• DWL = 001 (with a data word length of 16 bits), the PDTA setting is ignored. All data bits in SSIRDR or SSITDR are used on the audio serial bus. Two data words are transmitted or received at each 32-bit access. The first data word is derived from bits 15 to 0 and the second data word is derived from bits 31 to 16. • DWL = 010, 011, 100, 101 (with a data word length of 18, 20, 22 or 24 bits), PDTA = 0 (left-aligned) The data bits used in SSIRDR or SSITDR are the following: Bits 31 down to (32 minus the number of bits in the data word length specified by DWL). That is, If DWL = 011, the data word length is 20 bits; therefore, bits 31 to 12 in either SSIRDR or SSITDR are used. All other bits are ignored or reserved. • DWL = 010, 011, 100, 101 (with a data word length of 18, 20, 22 or 24 bits), PDTA = 1 (right-aligned) The data bits used in SSIRDR or SSITDR are the following: Bits (the number of bits in the data word length specified by DWL minus 1) to 0 i.e. if DWL = 011, then DWL = 20 and bits 19 to 0 are used in either SSIRDR or SSITDR. All other bits are ignored or reserved. • DWL = 110 (with a data word length of 32 bits), the PDTA setting is ignored. All data bits in SSIRDR or SSITDR are used on the audio serial bus.

Bit	Bit Name	Initial Value	R/W	Description
8	DEL	0	R/W	Serial Data Delay 0: 1 clock cycle delay between SSI_WS and SSI_SDATA 1: No delay between SSI_WS and SSI_SDATA Set this bit to 1 if CPEN = 1.
7	BREN	0	R/W	Burst Mode Enable 0: Burst mode is disabled. 1: Burst mode is enabled. This bit should be cleared to 0 when CPEN = 0. The burst mode can only be used in conjunction with the compressed mode (CPEN=1). When burst mode is enabled the SSI_SCK signal is gated. Clock pulses are only output when there is valid serial data being output on SSI_SDATA.

Bit	Bit Name	Initial Value	R/W	Description
6 to 4	CKDV[2:0]	000	R/W	Serial Oversampling Clock Division Ratio Sets the ratio between oversampling clock, CLK_FS, and the serial bit clock. This bit is ignored if SCKD = 0. The serial bit clock is used in the shift register and is provided on the SSI_SCK module pin. 000: Serial bit clock frequency = oversampling clock frequency/1 001: Serial bit clock frequency = oversampling clock frequency/2 010: Serial bit clock frequency = oversampling clock frequency/4 011: Serial bit clock frequency = oversampling clock frequency/8 100: Serial bit clock frequency = oversampling clock frequency/16 101: Serial bit clock frequency = oversampling clock frequency/6 110: Serial bit clock frequency = oversampling clock frequency/12 111: Setting prohibited CKDV = 000 is invalid when TDM = 1 or CONT = 1 in SSITDM.
3	MUEN	0	R/W	Serial Data Output Disable 0: Module is not muted. 1: Module is muted. Note: This bit can be used to stop output (low output) or to enable output. However, the operation is not synchronized with the change of SSI_WS.

Bit	Bit Name	Initial Value	R/W	Description
2	CPEN	0	R/W	Compressed Mode Enable 0: Compressed mode is disabled. 1: Compressed mode is enabled. Note: In compressed mode (CPEN=1), the operation must be other than with the slave transmitter (SWSD=0 and TRMD=1).
1	TRMD	0	R/W	Transmit/Receive Mode Select 0: Module is in receive mode. 1: Module is in transmit mode.
0	EN	0	R/W	SSI Module Enable 0: Module is disabled. 1: Module is enabled.

16A.2.2 Status Register (SSISRn) (n = 0 to 3)

SSISR consists of status flags indicating the operational status of the SSI module and bits indicating the current channel numbers and word numbers.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	DMRQ	UIRQ	OIRQ	IIRQ	DIRQ	—	—	—	—	—	—	—	—
Initial value:	—	—	—	0	0	0	1	0	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R/WC0	R/WC0	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	CHNO1	CHNO0	SWNO	IDST
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	0	0	1	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
28	DMRQ	0	R	DMA Request Status Flag This status flag allows the CPU to recognize the value of the DMA request pin on the SSI module. • TRMD = 0 (Receive mode) If DMRQ = 1, the SSIRDR has unread data. If SSIRDR is read, DMRQ = 0 until there is new unread data. • TRMD = 1 (Transmit mode) If DMRQ = 1, SSITDR requires data to be written to continue the transmission to the audio serial bus. Once data is written to SSITDR, DMRQ = 0 until it requires further transmit data.

Bit	Bit Name	Initial Value	R/W	Description
27	UIRQ	0	R/WC0	Underflow Error Interrupt Status Flag This status flag indicates that data was supplied at a lower rate than was required. In either case, this bit is set to 1 regardless of the value of the UIEN bit and can be cleared by writing 0 to this bit. If UIRQ = 1 and UIEN = 1, an interrupt occurs. • TRMD = 0 (Receive mode) If UIRQ = 1, SSIRDR was read before there was new unread data indicated by the DMRQ or DIRQ bit. This can lead to the same received data being stored twice by the host leading to potential corruption of multi-channel data. • TRMD = 1 (Transmit mode) If UIRQ = 1, SSITDR did not have data written to it before it was required for transmission. This will lead to the same data being transmitted once more and a potential corruption of multi-channel data. This is more serious error than a receive mode underflow as the output SSI data results in error. Note: When underflow error occurs, the current data in the data buffer of this module is transmitted until the next data is filled.

Bit	Bit Name	Initial Value	R/W	Description
26	OIRQ	0	R/WC0	Overflow Error Interrupt Status Flag This status flag indicates that data was supplied at a higher rate than was required. In either case this bit is set to 1 regardless of the value of the OIEN bit and can be cleared by writing 0 to this bit. If OIRQ = 1 and OIEN = 1, an interrupt occurs. • TRMD = 0 (Receive mode) If OIRQ = 1, SSIRDR was not read before there was new unread data written to it. This will lead to the loss of a data and a potential corruption of multi-channel data. Note: When overflow error occurs, the current data in the data buffer of this module is overwritten by the next incoming data from the SSI interface. • TRMD = 1 (Transmit mode) If OIRQ = 1, SSITDR had data written to it before it was transferred to the shift register. This will lead to the loss of a data and a potential corruption of multi-channel data.
25	IIRQ	Undefined	R	Idle Mode Interrupt Status Flag This interrupt status flag indicates whether the SSI module is in idle state. This bit is set regardless of the value of the IIEN bit to allow polling. "Idle state" refers to the state where the serial bus has been stopped after activation of the SSI. The interrupt can be masked by clearing IIEN, but cannot be cleared by writing to this bit. If IIRQ = 1 and IIEN = 1, an interrupt occurs. 0: The SSI module is not in idle state. 1: The SSI module is in idle state.

Bit	Bit Name	Initial Value	R/W	Description
24	DIRQ	0	R	Data Interrupt Status Flag This status flag indicates that the module has data to be read or requires data to be written. In either case this bit is set to 1 regardless of the value of the DIEN bit to allow polling. The interrupt can be masked by clearing DIEN, but cannot be cleared by writing to this bit. If DIRQ= 1 and DIEN = 1, an interrupt occurs. • TRMD = 0 (Receive mode) 0: No unread data in SSIRDR 1: Unread data in SSIRDR • TRMD = 1 (Transmit mode) 0: Transmit buffer is full. 1: Transmit buffer is empty and requires data to be written to SSITDR.
23 to 4	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
3, 2	CHNO[1:0]	00	R	Channel Number This value indicates the current channel number. However, if the length of data words is 8 or 16 bits, the value of this bit is meaningless. 00: 1 channel 01: 2 channels 10: 3 channels 11: 4 channels • TRMD = 0 (Receive mode) CHNO indicates which channel the data in SSIRDR currently represents. This value will change as the data in SSIRDR is updated from the shift register. • TRMD = 1 (Transmit mode) CHNO indicates which channel is required to be written to SSITDR. This value will change as the data is copied to the shift register, regardless of whether the data is written to SSITDR. These bits cannot be used when TDM = 1 or CONT = 1 in SSITDM.

Bit	Bit Name	Initial Value	R/W	Description
1	SWNO	1	R	System Word Number This status bit indicates the current word number. However, if the length of data words is 8 or 16 bits, the value of this bit is meaningless. • TRMD = 0 (Receive mode) SWNO indicates which system word the data in SSIRDR currently represents. This value will change as the data in SSIRDR is updated from the shift register, regardless of whether SSIRDR has been read. • TRMD = 1 (Transmit mode) SWNO indicates which system word is required to be written to SSITDR. This value will change as the data is copied to the shift register, regardless of whether the data is written to SSITDR. This bit cannot be used when TDM = 1 or CONT = 1 in SSITDM.

Bit	Bit Name	Initial Value	R/W	Description
0	IDST	Undefined	R	Idle Mode Status Flag This status flag indicates that the serial bus activity has stopped. This bit is cleared if EN = 1 and the serial bus are currently active. This bit is automatically set to 1 under the following conditions. • SSI = Master transmitter (SWSD=1 and TRMD=1) This bit is set to 1 when the EN bit is cleared and data written in SSITDR has been output from the serial data input/output pin (SSI_SDATA). • SSI = Master receiver (SWSD=1 and TRMD=0) This bit is set to 1 when the EN bit is cleared and transfer of the current system word is completed. • SSI = Slave transmitter/receiver (SWSD=0) This bit is set to 1 when the EN bit is cleared and transfer of the current system word is completed. Note: If an external device stops the serial bus clock before transfer of the current system word is completed, this bit is not set.

16A.2.3 Transmit Data Register (SSITDRn) (n = 0 to 3)

SSITDR is a 32-bit register that stores data to be transmitted.

Data written to this register is transferred to the shift register upon transmission request. If the data word length is less than 32 bits, the alignment is determined by the setting of the PDTA control bit in SSICR. The data in the buffer can be accessed by reading this register.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

16A.2.4 Receive Data Register (SSIRDRn) (n = 0 to 3)

SSIRDR is a 32-bit register that stores receive messages.

Data in this register is transferred from the shift register each time data word is received. If the data word length is less than 32 bits, the alignment is determined by the setting of the PDTA control bit in SSICR.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

16A.2.5 TDM Mode Register (SSITDM)

SSITDM is a 32-bit readable/writable register that sets TDM mode and WS continue mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	CONT	—	—	—	—	—	—	—	TDM
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved The read value is 0. The write value should always be 0.
8	CONT	0	R/W	WS Continue Mode 0: WS continue mode is disabled. 1: WS continue mode is enabled. Note: This bit can be set only when uncompressed mode (CPEN = 0 in SSICR) and master mode (SCKD = SWSD = 1 in SSICR) are set.
7 to 1	—	All 0	R	Reserved The read value is 0. The write value should always be 0.
0	TDM	0	R/W	TDM Mode 0: TDM mode is disabled. 1: TDM mode is enabled.

16A.3 Operation

16A.3.1 Bus Format

The SSI module can operate as a transmitter or a receiver and can be configured into many serial bus formats in either mode.

The bus format can be selected from one of the eleven major modes shown in table 16A.2.

Table 16A.2 Bus Format for SSI Module

	TRMD	CPEN	SCKD	SWSD	EN	MUEN	DIEN	IIEN	OIEN	UIEN	DEL	PDTA	SDTA	SPDP	SWSP	SCKP	SWL[2:0]	DWL[2:0]	CHNL[1:0]	TDM	CONT
Uncompressed slave receiver	0	0	0	0	Control bits						Configuration bits									0	0/1
Uncompressed slave transmitter	1	0	0	0																	
Uncompressed master receiver	0	0	1	1																	
Uncompressed master transmitter	1	0	1	1																	
Compressed slave receiver	0	1	0/1	0	Control bits						1	Ignored			Configu-ration bits		Ignored				0
Compressed master receiver	0	1	0/1	1																	
Compressed master transmitter	1	1	0/1	1																	
TDM slave receiver	0	0	0	0	Control bits						Configuration bits				0	Configuration bits				1	0/1
TDM slave transmitter	1	0	0	0																	
TDM master receiver	0	0	1	1																	
TDM master transmitter	1	0	1	1																	

16A.3.2 Uncompressed Modes

The uncompressed modes support all serial audio streams split into channels. It supports Philips, Sony and Panasonic modes as well as many more variants on these modes.

(1) **Slave Receiver**

This mode allows the module to receive serial data from another device. The clock and word select signal used for the serial data stream is also supplied from an external device. If these signals do not conform to the format specified in the configuration fields of the SSI module, operation is not guaranteed.

(2) **Slave Transmitter**

This mode allows the module to transmit serial data to another device. The clock and word select signal used for the serial data stream is also supplied from an external device. If these signals do not conform to the format specified in the configuration fields of the SSI module, operation is not guaranteed.

(3) **Master Receiver**

This mode allows the module to receive serial data from another device. The clock and word select signals are internally derived from the CLK_FS input clock. The format of these signals is defined in the configuration fields of the SSI module. If the incoming data does not follow the configured format, operation is not guaranteed.

(4) **Master Transmitter**

This mode allows the module to transmit serial data to another device. The clock and word select signals are internally derived from the CLK_FS input clock. The format of these signals is defined in the configuration fields of the SSI module.

(5) **Operating Setting (Related to Word Length)**

All bits related to the SSICR's word length are valid in uncompressed modes. There are many configurations the SSI module supports, but some of the combinations are shown below for the popular formats by Philips, Sony, and Panasonic.

1. Philips Format

   Figures 16A.2 and 16A.3 demonstrate the supported Philips format both with and without padding. Padding occurs when the data word length is smaller than the system word length.

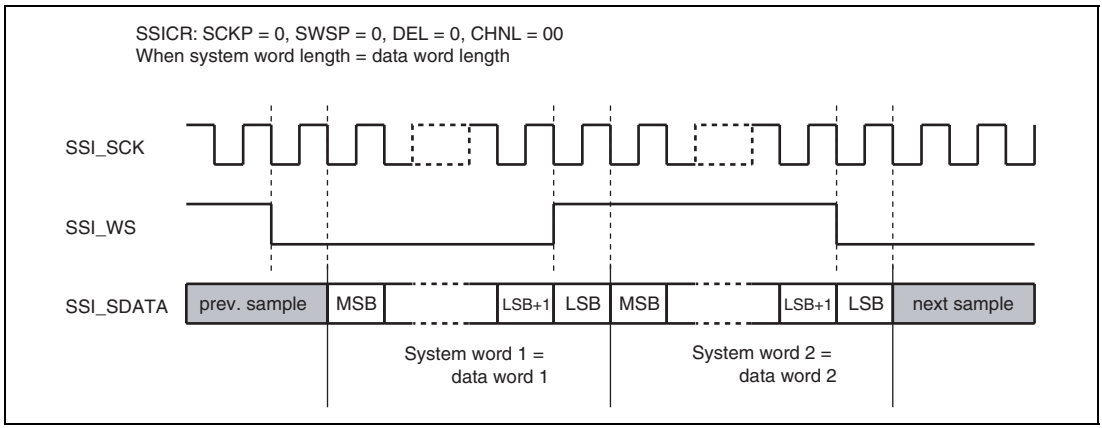


Figure 16A.2 Philips Format (without Padding)

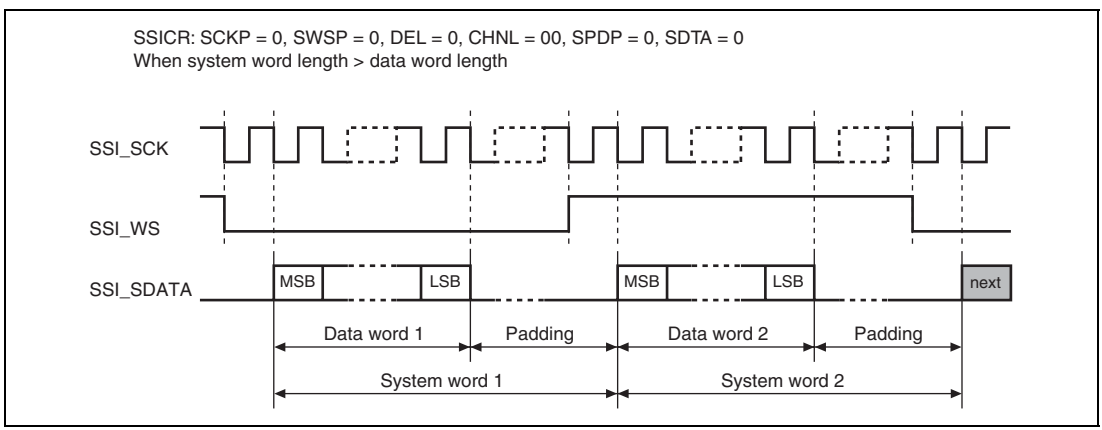


Figure 16A.3 Philips Format (with Padding)

Figure 16A.4 shows Sony format and figure 16A.5 shows Panasonic format. Padding is assumed in both cases, but may not be present in a final implementation if the system word length equals the data word length.

2. Sony Format

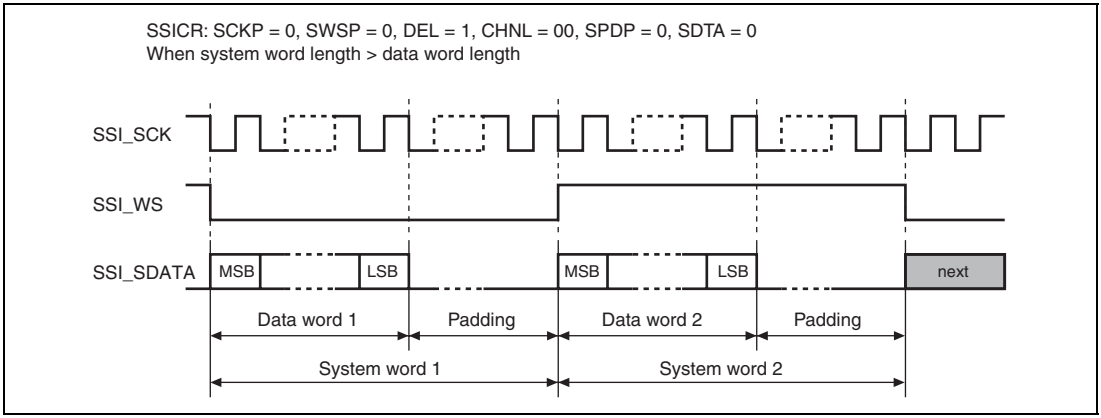


Figure 16A.4 Sony Format
(Transmitted and Received in the Order of Serial Data and Padding Bits)

3. Panasonic Format

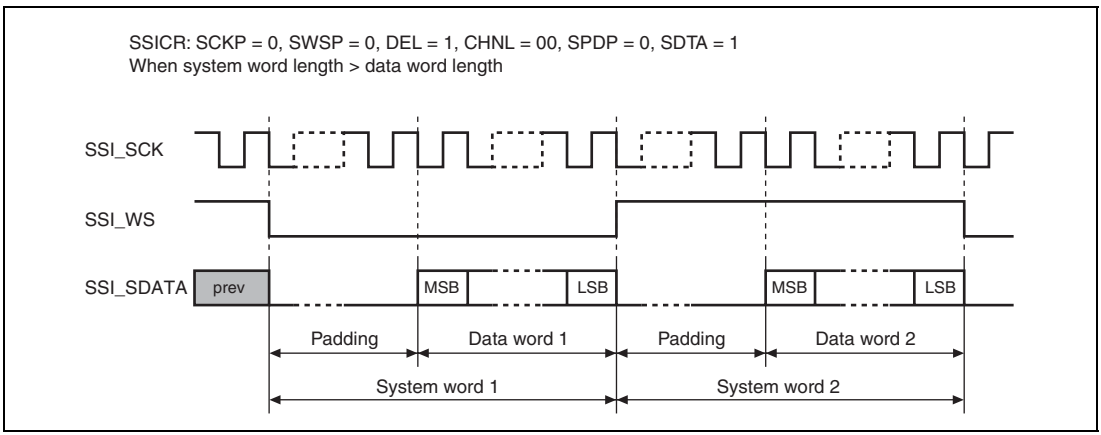


Figure 16A.5 Panasonic Format
(Transmitted and Received in the Order of Padding Bits and Serial Data)

(6) Multi-Channel Formats

Some devices extend the definition of the specification by Philips and allow more than two channels to be transferred within two system words.

The SSI module supports the transfer of two, three, and four channels by using the CHNL, SWL and DWL bits only when the system word length (SWL) is greater than or equal to the data word length (DWL) multiplied by channels (CHNL).

Table 16A.3 shows the number of padding bits for each of the valid setting. If setting is not valid, "—" is indicated instead of a number.

Table 16A.3 The Number of Padding Bits for Each Valid Setting

Padding Bits Per System Word		DWL[2:0]	000	001	010	011	100	101	110
CHNL [1:0]	Decoded Channels per System Word	SWL [2:0]	Decoded Word Length	8	16	18	20	22	24	32
00	1	000	8	0	—	—	—	—	—	—
		001	16	8	0	—	—	—	—	—
		010	24	16	8	6	4	2	0	—
		011	32	24	16	14	12	10	8	0
		100	48	40	32	30	28	26	24	16
		101	64	56	48	46	44	42	40	32
		110	128	120	112	110	108	106	104	96
		111	256	248	240	238	236	234	232	224
01	2	000	8	—	—	—	—	—	—	—
		001	16	0	—	—	—	—	—	—
		010	24	8	—	—	—	—	—	—
		011	32	16	0	—	—	—	—	—
		100	48	32	16	12	8	4	0	—
		101	64	48	32	28	24	20	16	0
		110	128	112	96	92	88	84	80	64
		111	256	240	224	220	216	212	208	192

CHNL [1:0]	Decoded Channels per System Word	SWL [2:0]	Decoded Word Length	Padding Bits Per System Word DWL[2:0] = 000	001	010	011	100	101	110
				8	16	18	20	22	24	32
10	3	000	8	—	—	—	—	—	—	—
		001	16	—	—	—	—	—	—	—
		010	24	0	—	—	—	—	—	—
		011	32	8	—	—	—	—	—	—
		100	48	24	0	—	—	—	—	—
		101	64	40	16	10	4	—	—	—
		110	128	104	80	74	68	62	56	32
		111	256	232	208	202	196	190	184	160
11	4	000	8	—	—	—	—	—	—	—
		001	16	—	—	—	—	—	—	—
		010	24	—	—	—	—	—	—	—
		011	32	0	—	—	—	—	—	—
		100	48	16	—	—	—	—	—	—
		101	64	32	0	—	—	—	—	—
		110	128	96	64	56	48	40	32	0
		111	256	224	192	184	176	168	160	128

When the SSI module acts as a transmitter, each word written to SSITDR is transmitted to the serial audio bus in the order they are written. When the SSI module acts as a receiver, each word received by the serial audio bus is read in the order received from SSIRDR.

Figures 16A.6 to 16A.8 show how two, three and four channels are transferred to the serial audio bus. Note that there are no padding bits in the first example, the second example is left-aligned and the third is right-aligned. This selection is arbitrary and is just for demonstration purposes only.

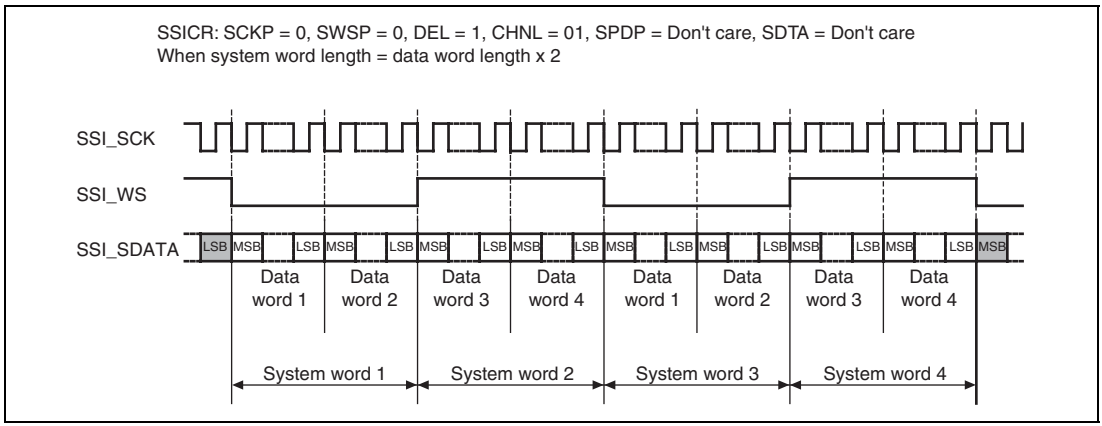


Figure 16A.6 Multi-channel Format (Two Channels without Padding)

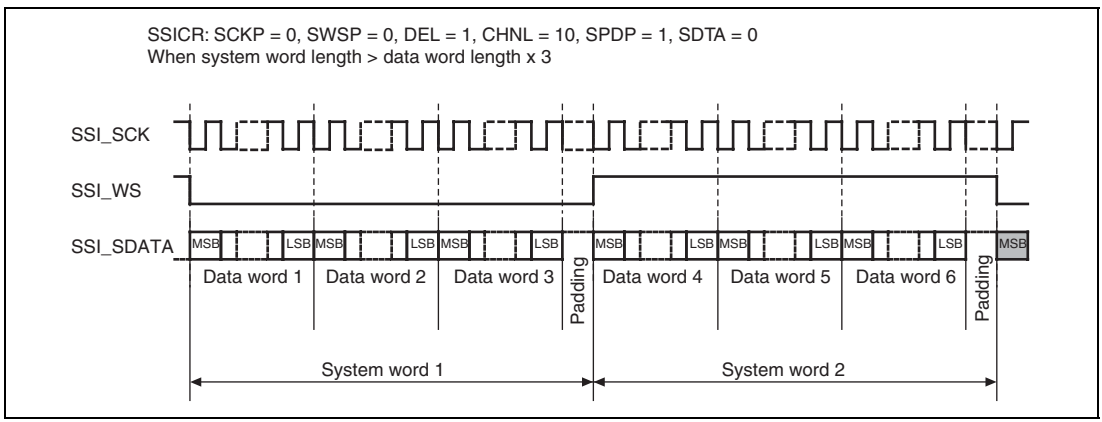


Figure 16A.7 Multi-channel Format (Three Channels with High Padding)

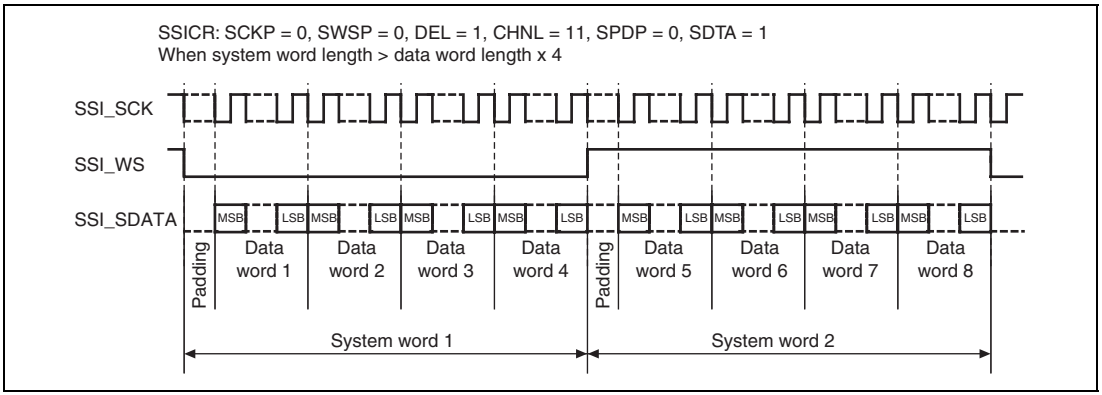


Figure 16A.8 Multi-channel Format (Four Channels; Transmitting and Receiving in the Order of Padding Bits and Serial Data; with Padding)

(7) Bit Setting Configuration Format

Several more configuration bits in uncompressed mode are shown below. These bits are not mutually exclusive, but some combinations may not be useful for any other device.

These configuration bits are described below with reference to figure 16A.9, Basic Sample Format.

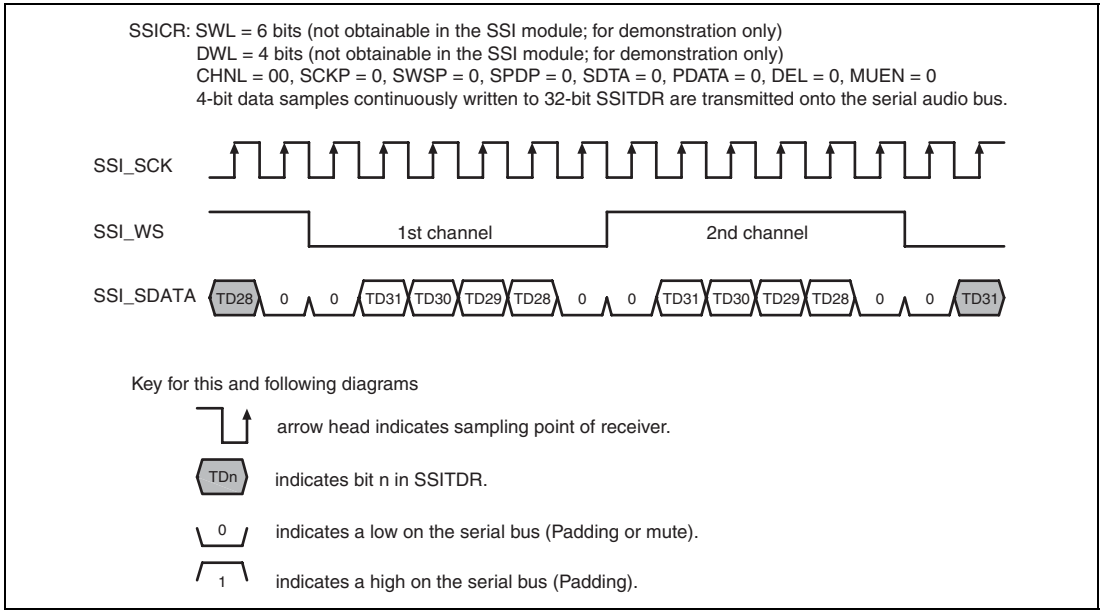


Figure 16A.9 Basic Sample Format
(Transmit Mode with Example System/Data Word Length)

Figure 16A.9 uses a system word length of 6 bits and a data word length of 4 bits. These settings are not possible with the SSI module but are used only for clarification of the other configuration bits.

1. Inverted Clock

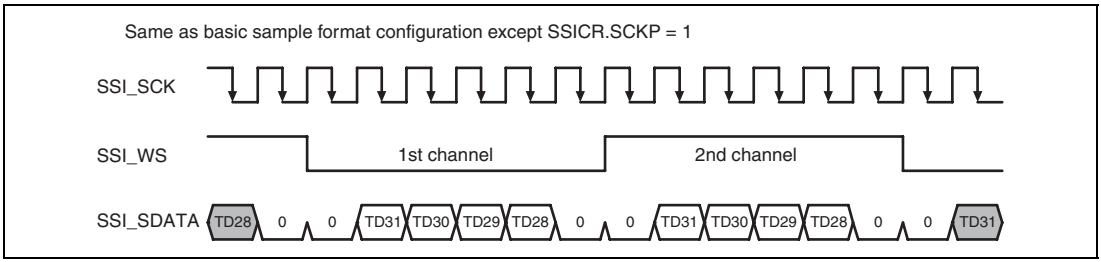


Figure 16A.10 Inverted Clock

2. Inverted Word Select

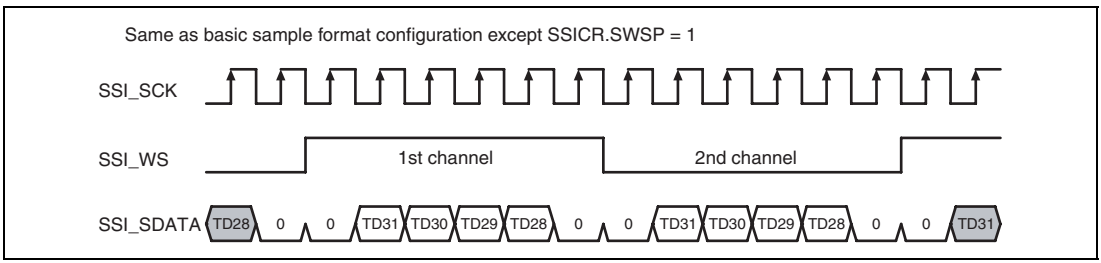


Figure 16A.11 Inverted Word Select

3. Inverted Padding Polarity

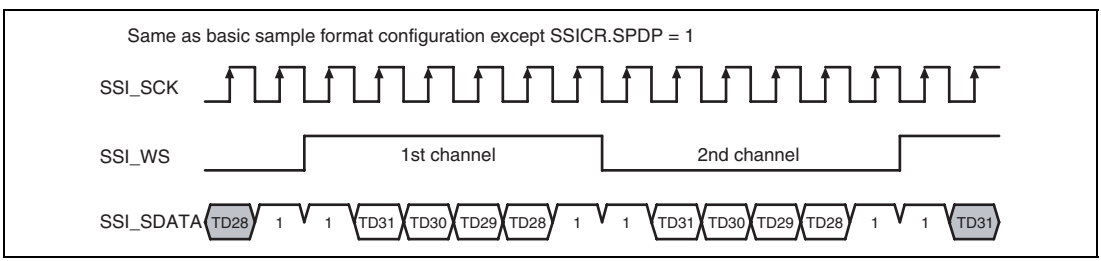


Figure 16A.12 Inverted Padding Polarity

4. Transmitting and Receiving in the Order of Padding Bits and Serial Data; with Delay

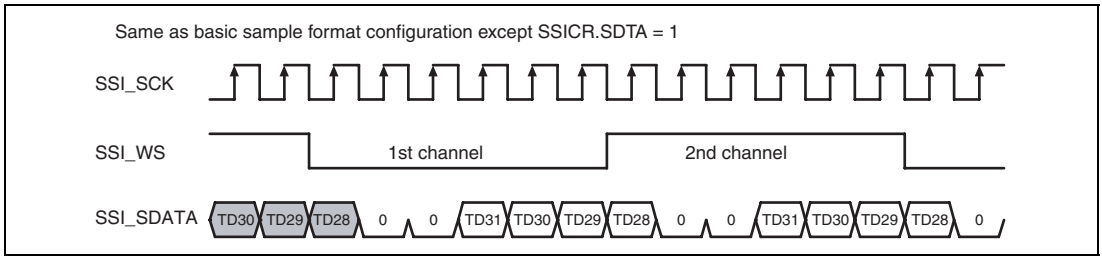


Figure 16A.13 Transmitting and Receiving in the Order of Padding Bits and Serial Data; with Delay

5. Transmitting and Receiving in the Order of Padding Bits and Serial Data; without Delay

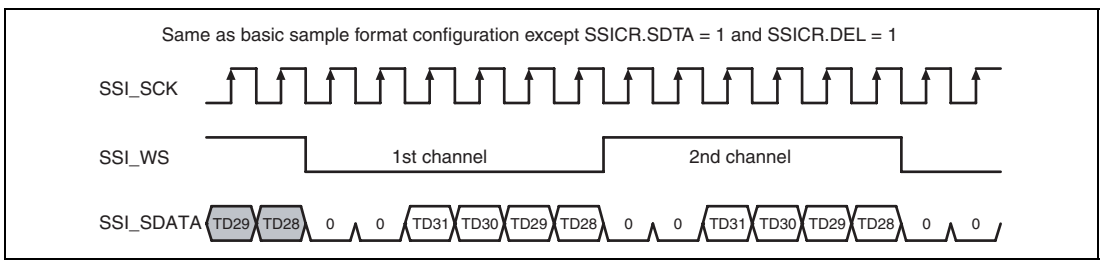


Figure 16A.14 Transmitting and Receiving in the Order of Padding Bits and Serial Data; without Delay

6. Transmitting and Receiving in the Order of Serial Data and Padding Bits; without Delay

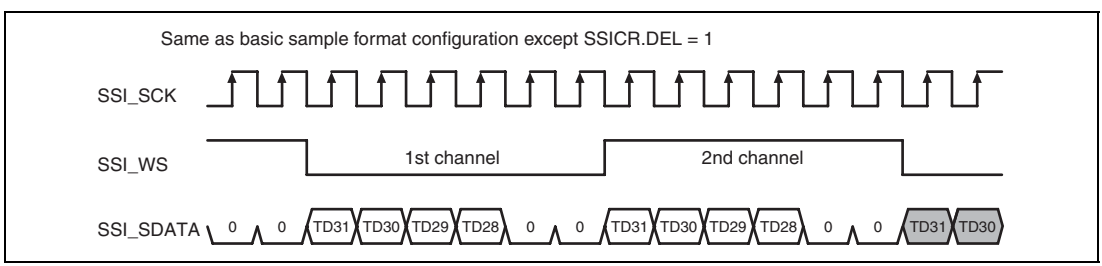


Figure 16A.15 Transmitting and Receiving in the Order of Serial Data and Padding Bits; without Delay

7. Parallel Right-Aligned with Delay

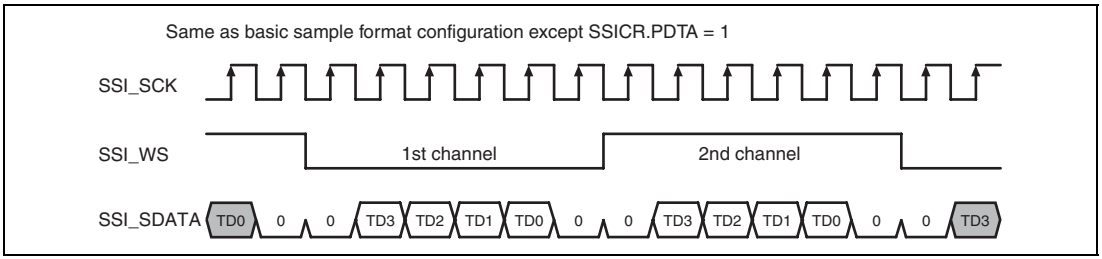


Figure 16A.16 Parallel Right-Aligned with Delay

8. Mute Enabled

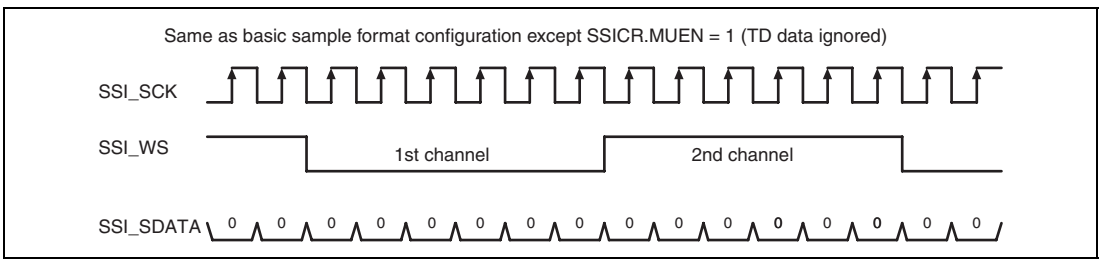


Figure 16A.17 Mute Enabled

16A.3.3 Compressed Modes

This mode is used to transfer a continuous bit stream. It requires a process that extracts data from a bit stream compressed on the receive side.

In streaming mode (the burst mode disabled) there is no concept of a data word. For transmission and reception, however, it is necessary to transfer between the serial bus and memory. The word boundary selection is arbitrary during receive/transmit and must be handled by another means. When burst mode is enabled, data bits being transmitted can be identified by the fact that the serial clock output is only activated when there is a word to output and only the required number of clock pulses necessary to clock out each 32-bit word are generated. The serial bit clock stops at the low level when SSICR.SCKP = 0 and stops at the high level when SSICR.SCKP = 1. Note that burst mode is only valid in the context of the module being a transmitter of data. The burst mode data cannot be received by this module.

Data is transmitted and received in blocks of 32 bits, and the first bit received/transmitted is bit 31 when stored in memory.

The word select pin in this mode does not act as a system word start signal unlike in uncompressed mode, but instead is used to indicate that the receiver can receive another data burst, or the transmitter can transmit another data burst.

Figures 16A.18 and 16A.19 show data transfer in compressed mode, with burst mode first disabled and then enabled respectively.

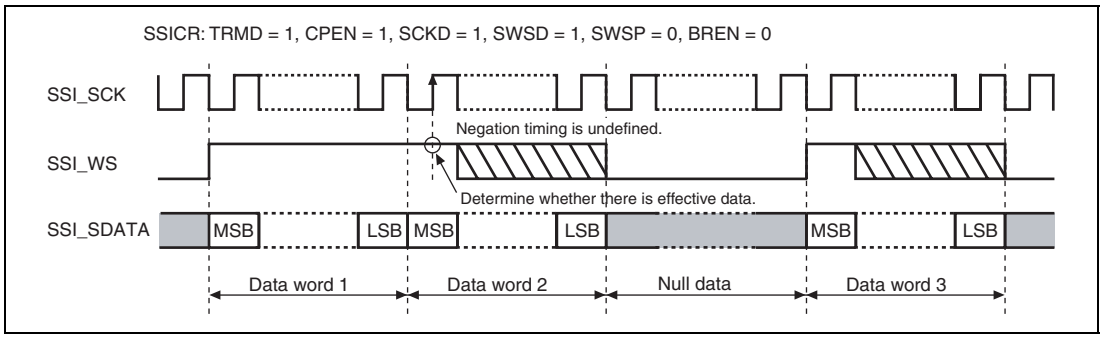


Figure 16A.18 Compressed Data Format, Master Transmitter, Burst Mode Disabled

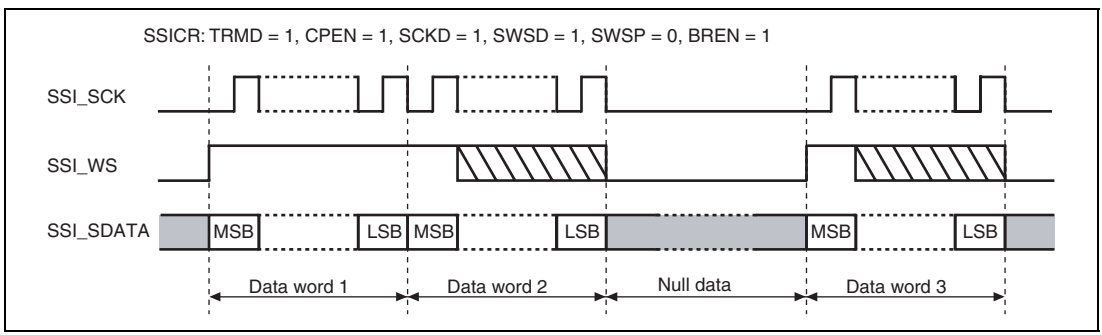


Figure 16A.19 Compressed Data Format, Master Transmitter, and Burst Mode Enabled

(1) **Slave Receiver**

This mode allows the module to receive a serial bit stream from another device and store it in memory. The shift register clock can be supplied from an external device or from an internal clock.

The word select pin is used as an input flow control. Assuming that SSICR.SWSP = 0 if SSI_WS is high, the module will receive the bit stream in blocks of 32 bits, one data bit per clock. If SSI_WS goes low, the module will complete the current 32-bit block and then stop any further reception, until SSI_WS goes high again.

(2) **Slave Transmitter**

This mode should not be used.

(3) **Master Receiver**

This mode allows the module to receive a serial bit stream from another device and store it in memory.

The shift register clock can be supplied from an external device or from an internal clock.

The word select pin is used as an output flow control. The module always asserts word select to indicate it can receive more data. It is the responsibility of the transmitting device to ensure it can transmit data to the SSI module in time to ensure no data is lost.

(4) **Master Transmitter**

This mode allows the module to transmit a serial bit stream from local memory to another device. The shift register clock can be supplied from an external device or from an internal clock.

The word select pin is used as an output flow control. The module always asserts word select to indicate it will transmit more data. However, word select is not asserted until the first word is ready to transmit. It is the responsibility of the receiving device to ensure it can receive the serial data in time to ensure no data is lost.

The SSI module requires minimum interaction from the CPU once it has been configured for a data transfer. The CPU is required to configure the SSI and DMAC modules and handle any overflow/underflow interrupts, if necessary.

16A.3.4 TDM Mode

TDM mode is used for connecting the SSI with a multi-channel device which supports TDM. TDM mode is set using the TDM bit in SSITDM. When this mode is set, SSI_WS is driven high during the period of system word 1, and pulled low in the other periods. The pulses generated on the SSI_WS signal are referred to as SYNC pulses. The SYNC pulses always have the positive polarity (driven high during system word 1).

Figures 16A.20 and 16A.21 show the TDM formats without padding and with padding.

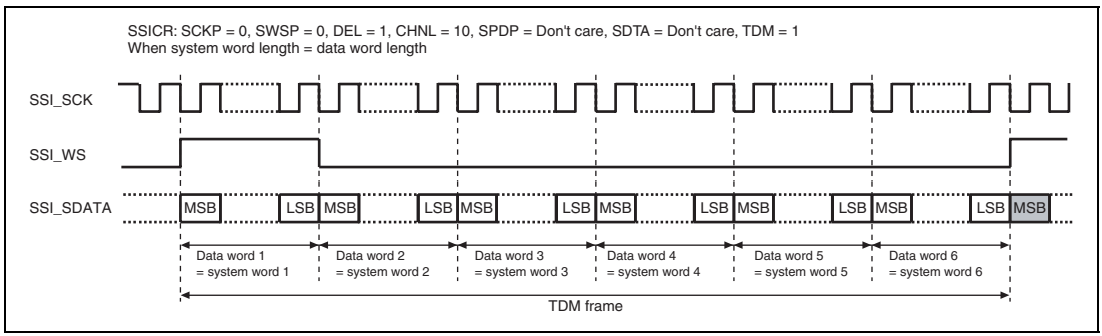


Figure 16A.20 TDM Format (Six System Words, without Padding)

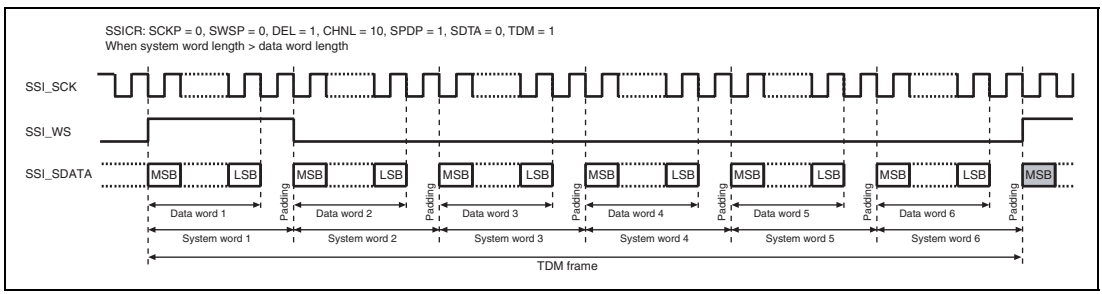


Figure 16A.21 TDM Format (Six System Words, with Padding)

(1) **Master Transmitter**

By a transfer start trigger (setting the EN bit in the control register to 1), a transfer of system word 1 begins synchronously with the SYNC pulses.

By a transfer stop trigger (setting the EN bit in the control register to 0), a transfer is stopped at the end of the currently-transferred system word and the SDATA signal is output according to the SPDP setting in SSICR (when SPDP = 0, a low-level signal is output).

If transmit data is not ready in the SSI module during transmission, an underflow will occur. When an underflow occurs, data to be output for the SYNC pulses cannot be determined. Therefore, stop and reconfigure the transfer.

(2) **Master Receiver**

By a transfer start trigger (setting the EN bit in the control register to 1), a reception of system word 1 data begins at the point where the SSI module recognizes a SYNC pulse.

By a transfer stop trigger (setting the EN bit in the control register to 0), a reception is stopped at the end of the currently-transferred system word.

The receive data register should not be read when it does not have data. If read, an underflow will occur. When an underflow occurs, stop and reconfigure the transfer.

The receive data should not be written to before read. If written to, an overflow will occur. When an overflow occurs, stop and reconfigure the transfer.

(3) **Slave Transmitter**

By a transfer start trigger (setting the EN bit in the control register to 1), a transfer of system word 1 begins synchronously with the SYNC pulses.

By a transfer stop trigger (setting the EN bit in the control register to 0), a transfer is stopped at the end of the currently-transferred system word and the SDATA signal is output according to the SPDP setting in SSICR (when SPDP = 0, a low-level signal is output).

If transmit data is not ready in the SSI module during transmission, an underflow will occur. When an underflow occurs, data to be output for SYNC pulses cannot be determined. Therefore, stop and re-set the transfer.

Transfers cannot be performed if the SCK signal and SYNC pulses are not provided to the SSI module during transfer.

(4) **Slave Receiver**

By a transfer start trigger (setting the EN bit in the control register to 1), a reception of system word 1 data begins at the point where the SSI module recognizes a SYNC pulse.

By a transfer stop trigger (setting the EN bit in the control register to 0), a reception is stopped at the end of the currently-transferred system word.

The receive data should not be read when it does not have data. If read, an underflow will occur. When an underflow occurs, stop and reconfigure the transfer.

The receive data register should not be written to before read. If written to, an overflow will occur. When an overflow occurs, stop and reconfigure the transfer.

Transfers cannot be performed if the SCK signal and SYNC pulses are not provided to the SSI module during transfer.

16A.3.5 WS Continue Modes

WS continue mode is used to output SSI_WS signal continuously regardless of whether data transfer is enabled or disabled. WS continue mode can be set using the CONT bit in SSITDM. This mode can be combined with uncompressed mode or TDM mode. When this mode is enabled, the SSI_WS signal continues to operate even if the EN bit in SSICR is cleared to 0 (stopping a transfer). When this mode is disabled, the SSI_WS signal operates synchronously with the EN bit.

Figures 16A.22 and 16A.23 show operations when WS continue mode is enabled and disabled, respectively.

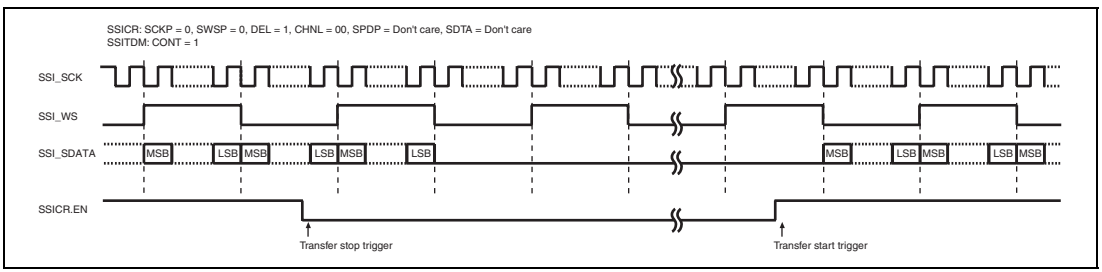


Figure 16A.22 WS Continue Mode Operation (When the Mode is Enabled)

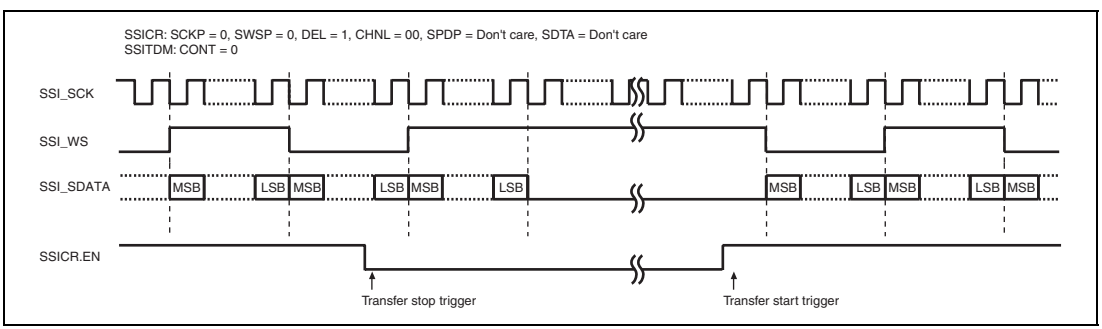


Figure 16A.23 WS Continue Mode Operation (When the Mode is Disabled)

16A.3.6 Operating Modes

There are three modes of operation: configuration, module enabled and module disabled. Figure 16A.24 shows how the module enters each of these modes.

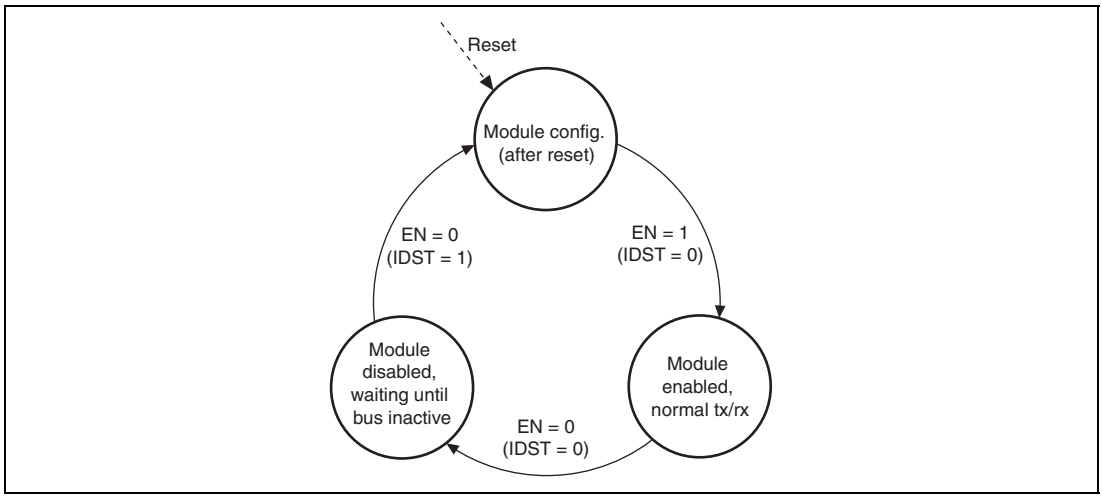


Figure 16A.24 Operating Modes

(1) **Configuration Mode**

This mode is entered after the module is released from reset. All required configuration fields in the control register should be defined in this mode, before the SSI module is enabled by setting the EN bit.

Setting the EN bit causes the module to enter the module enabled mode.

(2) **Module Enabled Mode**

Operation of the module in this mode is dependent on the operating mode selected. For details, refer to section 16A.3.7, Transmit Operation and section 16A.3.8, Receive Operation, below.

16A.3.7 Transmit Operation

Transmission can be controlled either by DMA or interrupt.

DMA control is preferred to reduce the processor load. In DMA control mode the processor will only receive interrupts if there is an underflow or overflow of data or the DMAC has finished its transfer.

The alternative method is using the interrupts that the SSI module generates to supply data as required. This mode has a higher interrupt load as the module is only double buffered and will require data to be written at least every system word period.

When disabling the module, the SSI clock* must remain present until the SSI module is in idle state, indicated by the SSISR.IIRQ bit.

Figure 16A.25 shows the transmit operation in DMA control mode, and figure 16A.26 shows the transmit operation in interrupt control mode

Note: * Input clock from the SSI_SCK pin when SCKD = 0.
Input clock from the CLK_FS pin when SCKD = 1.

(1) Transmission Using DMA Controller

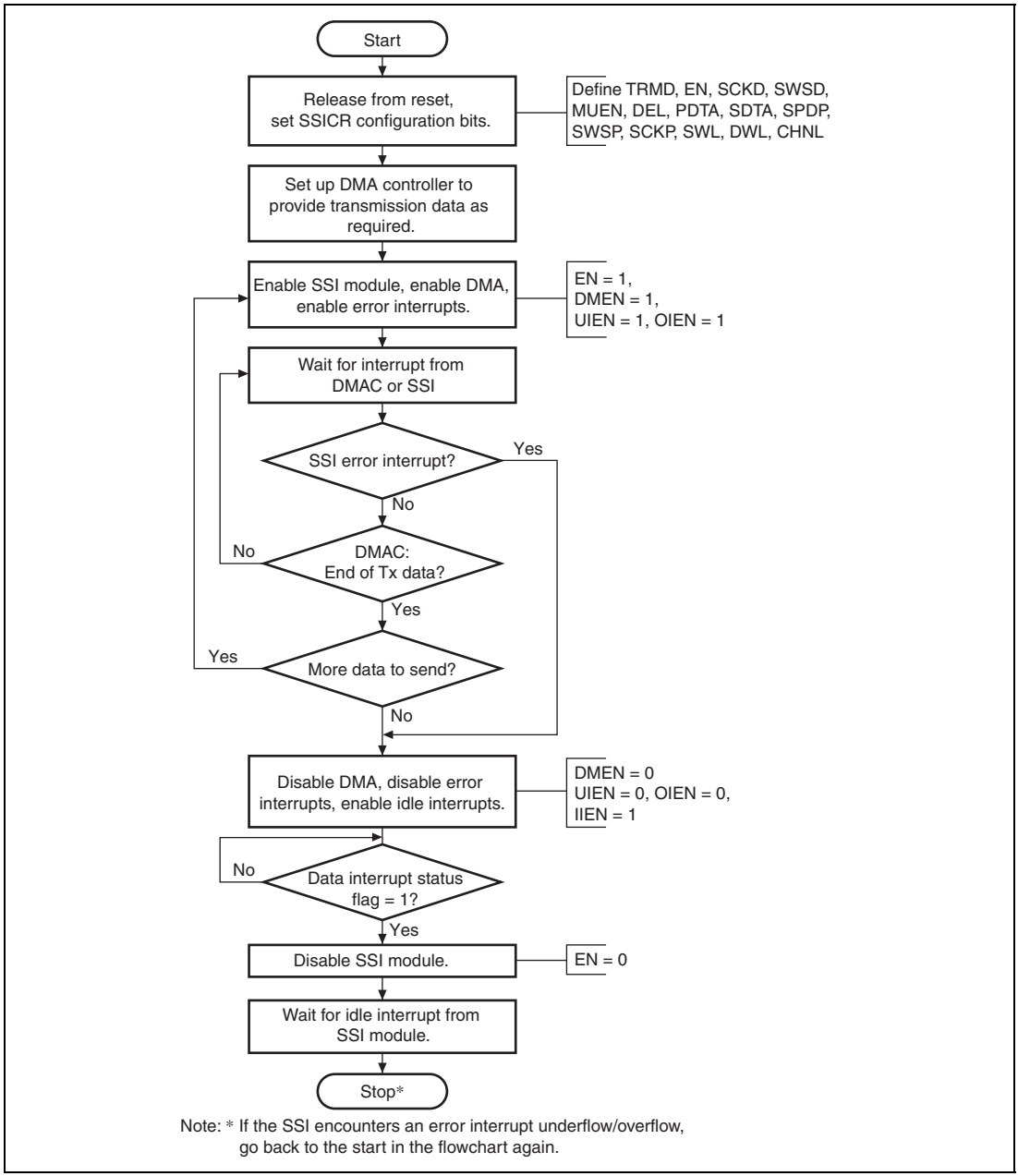


Figure 16A.25 Transmission Using DMA Controller

(2) Transmission using Interrupt Data Flow Control

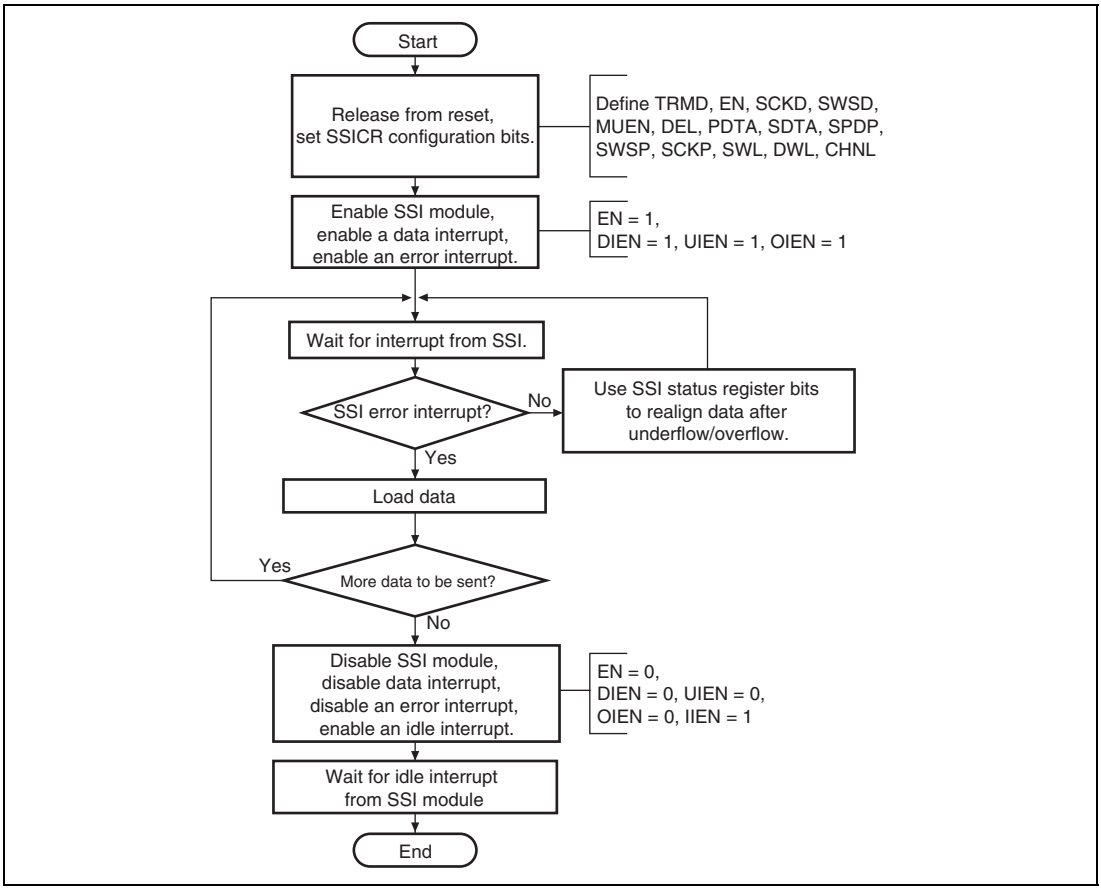


Figure 16A.26 Transmission Using Interrupt Data Flow Control

16A.3.8 Receive Operation

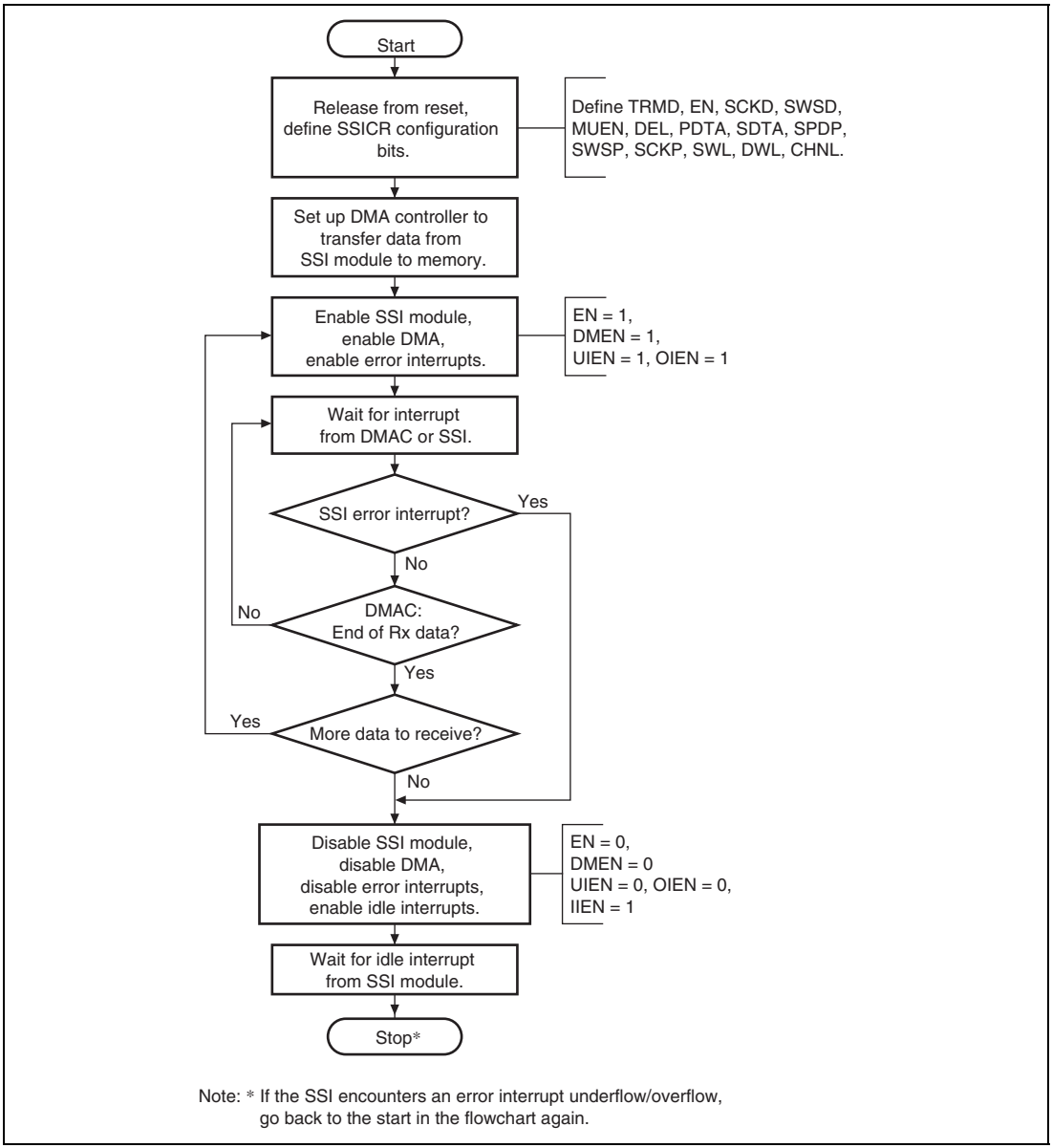
Like transmission, reception can be controlled either by DMA or interrupt.

Figures 16A.27 and 16A.28 show the flow of operation.

When disabling the SSI module, the SSI clock* must be kept supplied until the SSISR.IIRQ bit is in idle state.

Note: * Input clock from the SSI_SCK pin when SCKD = 0.
Input clock from the CLK_FS pin when SCKD = 1.

(1) Reception Using DMA Controller



Note: * If the SSI encounters an error interrupt underflow/overflow, go back to the start in the flowchart again.

Figure 16A.27 Reception Using DMA Controller

(2) Reception Using Interrupt Data Flow Control

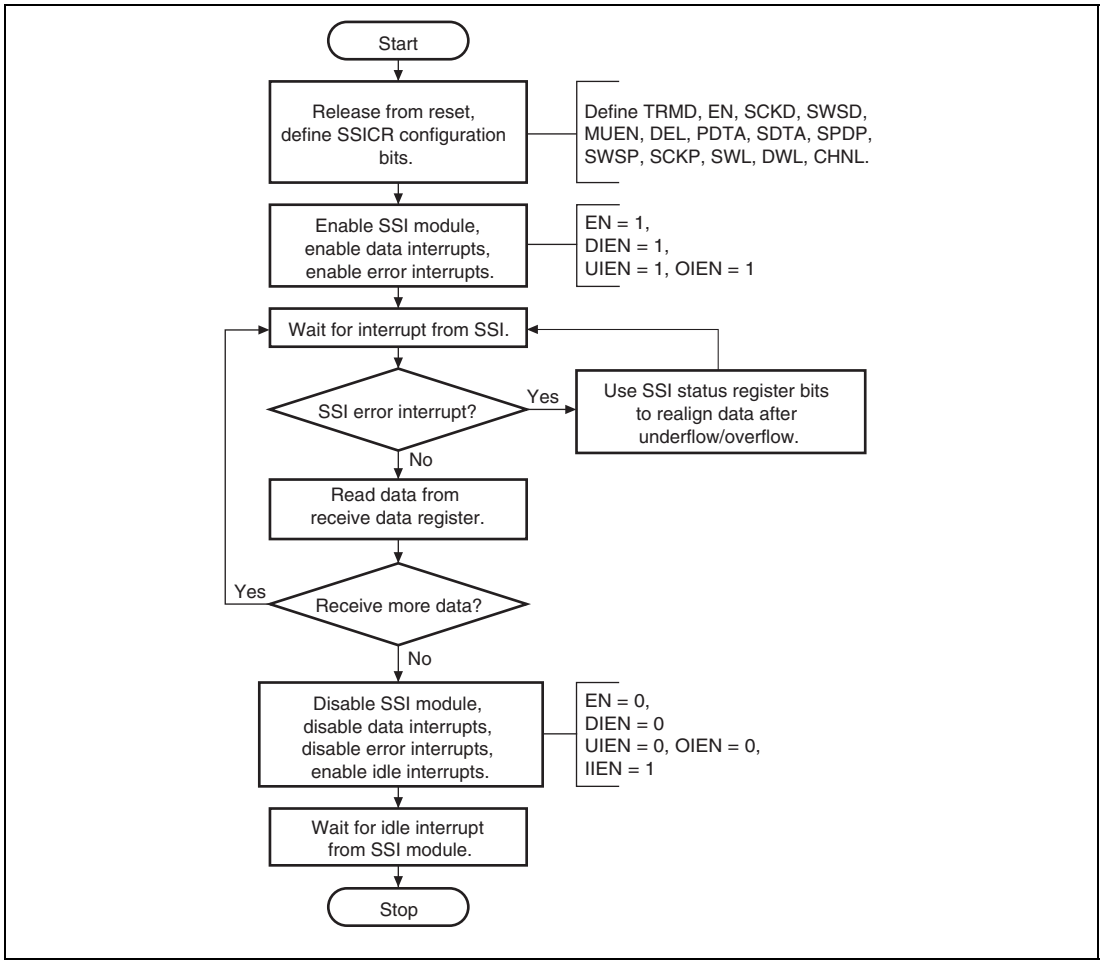


Figure 16A.28 Reception Using Interrupt Data Flow Control

When an underflow or overflow error condition has matched, the SSISR.CHNO[1:0] bits and the SSISR.SWNO bit can be used to recover the SSI module to a known status. When an underflow or overflow occurs, the host can read the channel number and system word number to determine what point the serial audio stream has reached. In the transmitter case, the host can skip forward through the data it wants to transmit until it finds the sample data that matches what the SSI module is expecting to transmit next, and so resynchronize with the audio data stream. In the receiver case the host CPU can store null data to make the number of receive data items consistent until it is ready to store the sample data that the SSI module is indicating will be received next, and so resynchronize with the audio data stream.

16A.3.9 Serial Bit Clock Control

This function is used to control and select which clock is used for the serial bus interface.

If the serial clock direction is set to input (SSICR.SCKD = 0), the SSI module is in clock slave mode and the shift register uses the bit clock that was input to the SSI_SCK pin.

If the serial clock direction is set to output (SSICR.SCKD = 1), the SSI module is in clock master mode, and the shift register uses the bit clock that is input from the CLK_FS pin or the bit clock that is obtained by dividing the input from the CLK_FS pin. In the latter case, the clock input from the CLK_FS pin is divided by the ratio in the serial oversampling clock divide ratio (CKDV) in SSICR and used as the bit clock in the shift register.

In either case, the SSI_SCK pin output is the same as the bit clock.

16A.4 Usage Notes

16A.4.1 Limitations from Overflow during Receive DMA Operation

If an overflow occurs while the receive DMA is in operation, the module should be restarted. The receive buffer in the SSI consists of 32-bit registers that share the L and R channels. Therefore, data to be received at the L channel may sometimes be received at the R channel if an overflow occurs, for example, under the following condition: the control register (SSICR) has a 32-bit setting for both data word length (DWL2 to DWL0) and system word length (SWL2 to SWL0).

If an overflow is confirmed with the overflow error interrupt or overflow error status flag (SSISR. OIRQ), write 0 to the EN and DMEN bits in SSICR to disable DMA in the SSI module, thus stopping the operation. (In this case, the controller setting should also be stopped.) After this, write 0 to the OIRQ bit in SSISR to clear the overflow status, set DMA again and restart the transfer.

16A.4.2 Limitations on Combinations of Modes Related to Common Pins to SSI2 and SSI3

Table 16A.4 shows the available combinations of operating modes for SSI2 and SSI3. Operating modes can be set with the control registers for the SSI2 and SSI3 (SSICR).

Table 16A.4 Usable Operating Mode Combinations for SSI2 and SSI3

	Operating Mode		Operating Mode of External Device		
No.	**SSI2**	**SSI3**	**SSI2 Side**	**SSI3 Side**	**Function**
1	Slave	Slave	Master	Slave	SSI2 and SSI3 operate synchronously with SSI_W23 and SSI_SCK23 input.
			Slave	Master	
			Slave	Slave	
2	Master	Slave	Slave	Slave	SSI3 and an external device operate synchronously with WS and SCK of SSI2.
3	Slave	Master	—	—	Setting prohibited
4	Master	Master	—	—	Setting prohibited

There are the following limitations on the operating mode combinations shown in table 16A.4.

For No. 1, SSI_WS23 and SSI_SCK23 should be input before either of SSI2 or SSI3 starts data transfer.

For No. 2, the SSI2 as the master should start transfer first thus allowing SSI_WS23 and SSI_SCK23 to be output, and then the SSI3 as the slave should be used. If the master-side transfer is stopped, the slave-side SSI3 should be stopped because SSI_WS23 is not output.

16A.4.3 Limitations on Slave Mode Operation

When this LSI is used in slave mode, in ending a data transfer process, data transfer on this LSI should be stopped (SSICR.EN = 0) before the input word selection signal (SSI_WS) is stopped.

In slave mode, data transfer is stopped by clearing the SSICR.EN bit (setting to stop transfer) and detecting the falling edge of the word selection signal (SSI_WS). If the input word selection signal is stopped first, the falling edge of the word selection signal cannot be detected and data transfer cannot be ended successfully.

16A.4.4 Limitations on Changes to Settings

The SSI_SCK and SSI_WS signals are not guaranteed immediately after changes to settings of the TDM mode register (SSITDM) and configuration bits in the control register (SSICR). Settings must not be changed dynamically if this would affect any connected device.

16A.4.5 Limitations on TDM Mode

The following limitations may arise in TDM Mode.

- When the settings are for transmission in slave mode, data transmitted after the start of transfer will sometimes not be synchronized with the sync pulse (SSI_WS).
- When the settings are for reception in slave mode, data received after the start of transfer will sometimes not be synchronized with the sync pulse (SSI_WS).
- When reception settings are made with WS continue mode disabled in master mode, the width of sync pulse after the transfer has started will sometimes be shorter than is stipulated.

The settings that lead to limitations are listed below:

- **Control Register (SSICR)**

Bit	23,22	15	14	12	7	2	1	Others
Bit Name	CHNL	SCKD	SWSD	SWSP	BREN	CPEN	TRMD	—
(1)	Other than 00	0	0	0	0	0	1	Don't care
(2)	Other than 00	0	0	0	0	0	0	Don't care
(3)	Other than 00	1	1	0	0	0	0	Don't care

- **TDM Mode Register (SSITDM)**

Bit	8	0	Others
Bit Name	CONT	TDM	—
(1)	Don't care	1	Don't care
(2)	Don't care	1	Don't care
(3)	0	1	Don't care

The procedure for avoiding these situations is as follows.

(1) Stop the SSI module (SSICR.EN = 0) and place it in the idle state (SSISR.IDST = 1).

(2) Wait for the fixed time corresponding to two frames.*

(3) Transfer can be restarted.

Note: * One frame corresponds to the period from one rising edge of SSI_WS to the next rising edge of SSI_WS.

Section 17 I²C Bus Interface 3

The I²C bus interface 3 supports and provides a subset of the Philips I²C (Inter-IC) bus interface functions. However, the configuration of the registers that control the I²C bus differs partly from the Philips register configuration.

The I²C bus interface 2 has three channels.

17.1 Features

- Selection of I²C format or clocked synchronous serial format
- Continuous transmission/reception

  Since the shift register, transmit data register, and receive data register are independent from each other, the continuous transmission/reception can be performed.

I²C bus format:

- Start and stop conditions generated automatically in master mode
- Selection of acknowledge output levels when receiving
- Automatic loading of acknowledge bit when transmitting
- Bit synchronization function

  In master mode, the state of SCL is monitored per bit, and the timing is synchronized automatically. If transmission/reception is not yet possible, set the SCL to low until preparations are completed.
- Six interrupt sources

  Transmit data empty (including slave-address match), transmit end, receive data full (including slave-address match), arbitration lost, NACK detection, and stop condition detection
- The direct memory access controller can be activated by a transmit-data-empty request or receive-data-full request to transfer data.
- Direct bus drive

  Two pins, SCL and SDA pins, function as NMOS open-drain outputs when the bus drive function is selected.

Clocked synchronous serial format:

- Four interrupt sources

  Transmit-data-empty, transmit-end, receive-data-full, and overrun error
- The direct memory access controller can be activated by a transmit-data-empty request or receive-data-full request to transfer data.

Figure 17.1 shows a block diagram.

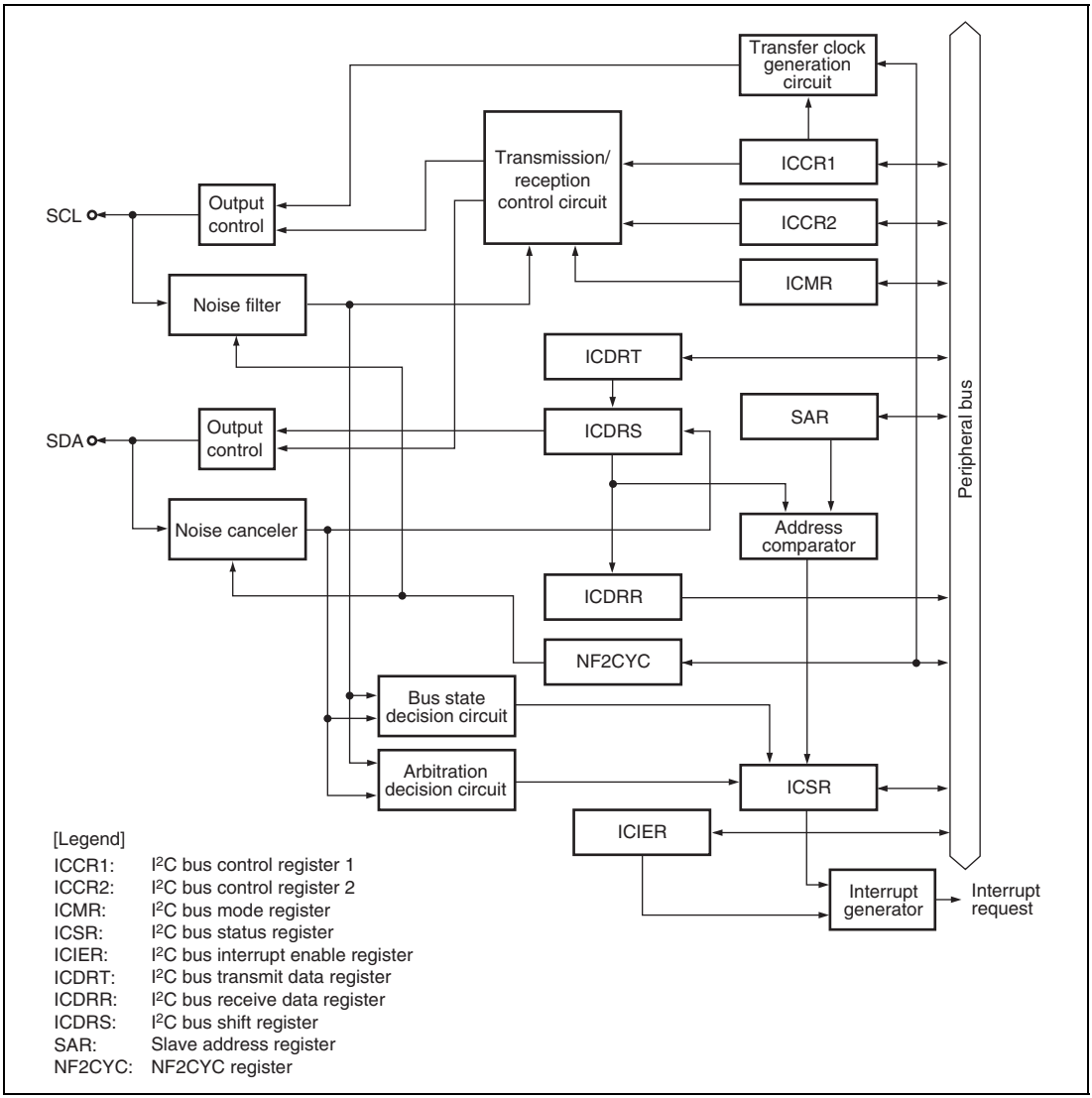


Figure 17.1 Block Diagram

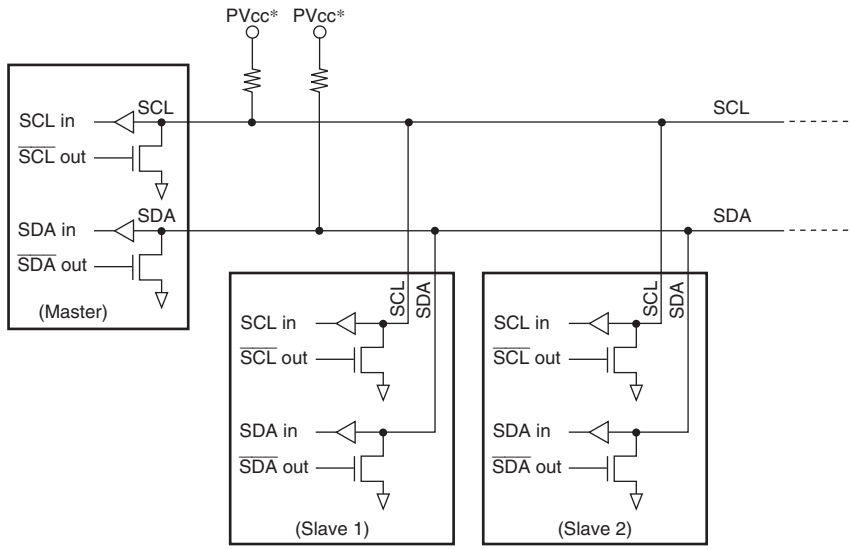
17.2 Input/Output Pins

Table 17.1 shows the pin configuration.

Table 17.1 Pin Configuration

Pin Name	Symbol	I/O	Function
Serial clock	SCL0, SCL1	I/O	I²C serial clock input/output
Serial data	SDA0, SDA1	I/O	I²C serial data input/output

Figure 17.2 shows an example of I/O pin connections to external circuits.



Note: * Turn on/off PVcc for the I²C bus power supply and for this LSI simultaneously.

Figure 17.2 External Circuit Connections of I/O Pins

17.3 Register Descriptions

Tables 17.2 (1) and 17.2 (2) show the register configuration and register state, respectively.

Table 17.2 (1) Register Configuration

Channel	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
0	I²C bus control register 1	ICCR1_0	R/W	H'00	H'FFC70000	8
	I²C bus control register 2	ICCR2_0	R/W	H'7D	H'FFC70001	8
	I²C bus mode register	ICMR_0	R/W	H'38	H'FFC70002	8
	I²C bus interrupt enable register	ICIER_0	R/W	H'00	H'FFC70003	8
	I²C bus status register	ICSR_0	R/W	H'00	H'FFC70004	8
	Slave address register	SAR_0	R/W	H'00	H'FFC70005	8
	I²C bus transmit data register	ICDRT_0	R/W	H'FF	H'FFC70006	8
	I²C bus receive data register	ICDRR_0	R/W	H'FF	H'FFC70007	8
	NF2CYC register	NF2CYC_0	R/W	H'00	H'FFC70008	8
1	I²C bus control register 1	ICCR1_1	R/W	H'00	H'FFC71000	8
	I²C bus control register 2	ICCR2_1	R/W	H'7D	H'FFC71001	8
	I²C bus mode register	ICMR_1	R/W	H'38	H'FFC71002	8
	I²C bus interrupt enable register	ICIER_1	R/W	H'00	H'FFC71003	8
	I²C bus status register	ICSR_1	R/W	H'00	H'FFC71004	8
	Slave address register	SAR_1	R/W	H'00	H'FFC71005	8
	I²C bus transmit data register	ICDRT_1	R/W	H'FF	H'FFC71006	8
	I²C bus receive data register	ICDRR_1	R/W	H'FF	H'FFC71007	8
	NF2CYC register	NF2CYC_1	R/W	H'00	H'FFC71008	8

Table 17.2 (2) Register State in Each Operating Mode

Abbreviation	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
ICCR1_0	Initialized	Initialized	Retained	Retained	Retained	Initialized
ICCR2_0	Initialized	Initialized	Retained	Retained	Retained	Initialized
ICMR_0	Initialized	Initialized	Retained	Retained	Retained*	Initialized
ICIER_0	Initialized	Initialized	Retained	Retained	Retained	Initialized
ICSR_0	Initialized	Initialized	Retained	Retained	Retained	Initialized
SAR_0	Initialized	Initialized	Retained	Retained	Retained	Initialized
ICDRT_0	Initialized	Initialized	Retained	Retained	Retained	Initialized
ICDRR_0	Initialized	Initialized	Retained	Retained	Retained	Initialized
NF2CYC_0	Initialized	Initialized	Retained	Retained	Retained	Initialized
ICCR1_1	Initialized	Initialized	Retained	Retained	Retained	Initialized
ICCR2_1	Initialized	Initialized	Retained	Retained	Retained	Initialized
ICMR_1	Initialized	Initialized	Retained	Retained	Retained*	Initialized
ICIER_1	Initialized	Initialized	Retained	Retained	Retained	Initialized
ICSR_1	Initialized	Initialized	Retained	Retained	Retained	Initialized
SAR_1	Initialized	Initialized	Retained	Retained	Retained	Initialized
ICDRT_1	Initialized	Initialized	Retained	Retained	Retained	Initialized
ICDRR_1	Initialized	Initialized	Retained	Retained	Retained	Initialized
NF2CYC_1	Initialized	Initialized	Retained	Retained	Retained	Initialized

Note: * Bits BC[2:0] in ICMR_0 and ICMR_1 are initialized.

17.3.1 I²C Bus Control Register 1 (ICCR1)

ICCR1 is an 8-bit readable/writable register that enables or disables the I²C bus interface 3, controls transmission or reception, and selects master or slave mode, transmission or reception, and transfer clock frequency in master mode.

Bit:	7	6	5	4	3	2	1	0
	ICE	RCVD	MST	TRS	CKS[3:0]			
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	ICE	0	R/W	I²C Bus Interface 3 Enable 0: SCL and SDA output is disabled. (Input to SCL and SDA is enabled.) 1: This bit is enabled for transfer operations. (SCL and SDA pins are bus drive state.)
6	RCVD	0	R/W	Reception Disable Enables or disables the next operation when TRS is 0 and ICDRR is read. 0: Enables next reception 1: Disables next reception

Bit	Bit Name	Initial Value	R/W	Description
5	MST	0	R/W	Master/Slave Select
4	TRS	0	R/W	Transmit/Receive Select In master mode with the I²C bus format, when arbitration is lost, MST and TRS are both reset by hardware, causing a transition to slave receive mode. Modification of the TRS bit should be made between transfer frames. When seven bits after the start condition is issued in slave receive mode match the slave address set to SAR and the 8th bit is set to 1, TRS is automatically set to 1. If an overrun error occurs in master receive mode with the clocked synchronous serial format, MST is cleared and the mode changes to slave receive mode. Operating modes are described below according to MST and TRS combination. When clocked synchronous serial format is selected and MST = 1, clock is output. 00: Slave receive mode 01: Slave transmit mode 10: Master receive mode 11: Master transmit mode
3 to 0	CKS[3:0]	0000	R/W	Transfer Clock Select These bits should be set according to the necessary transfer rate (table 17.3) in master mode.

Table 17.3 Transfer Rate

NF2CYC	ICCR1					Transfer Rate (kHz)				
Bit 4	Bit 3	Bit 2	Bit 1	Bit 0		Pϕ = 20.0 MHz	Pϕ = 24.0 MHz	Pϕ = 32.0 MHz	Pϕ = 36.0 MHz	Pϕ = 50.0 MHz
CKS4	CKS3	CKS2	CKS1	CKS0	Clock					
0	0	0	0	0	Pϕ/44	455	545	727	818	1136
				1	Pϕ/52	385	462	615	692	962
			1	0	Pϕ/64	313	375	500	563	781
				1	Pϕ/72	278	333	444	500	694
		1	0	0	Pϕ/84	238	286	381	429	595
				1	Pϕ/92	217	261	348	391	543
			1	0	Pϕ/100	200	240	320	360	500
				1	Pϕ/108	185	222	296	333	463
	1	0	0	0	Pϕ/176	114	136	182	205	284
				1	Pϕ/208	96.2	115	154	173	240
			1	0	Pϕ/256	78.1	93.8	125	141	195
				1	Pϕ/288	69.4	83.3	111	125	174
		1	0	0	Pϕ/336	59.5	71.4	95.2	107	149
				1	Pϕ/368	54.3	65.2	87.0	97.8	136
			1	0	Pϕ/400	50.0	60.0	80.0	90.0	125
				1	Pϕ/432	46.3	55.6	74.1	83.3	116
1	0	0	0	0	Pϕ/352	56.8	68.2	90.9	102	142
				1	Pϕ/416	48.1	57.7	76.9	86.5	120
			1	0	Pϕ/512	39.1	46.9	62.5	70.3	97.7
				1	Pϕ/576	34.7	41.7	55.6	62.5	86.8
		1	0	0	Pϕ/672	29.8	35.7	47.6	53.6	74.4
				1	Pϕ/736	27.2	32.6	43.5	48.9	67.9
			1	0	Pϕ/800	25.0	30.0	40.0	45.0	62.5
				1	Pϕ/864	23.1	27.8	37.0	41.7	57.9
	1	0	0	0	Pϕ/704	28.4	34.1	45.5	51.1	71.0
				1	Pϕ/832	24.0	28.8	38.5	43.3	60.1
			1	0	Pϕ/1024	19.5	23.4	31.3	35.2	48.8
				1	Pϕ/1152	17.4	20.8	27.8	31.3	43.4
		1	0	0	Pϕ/1344	14.9	17.9	23.8	26.8	37.2
				1	Pϕ/1472	13.6	16.3	21.7	24.5	34.0
			1	0	Pϕ/1600	12.5	15.0	20.0	22.5	31.3
				1	Pϕ/1728	11.6	13.9	18.5	20.8	28.9

Notes: 1. The settings should satisfy external specifications.
2. Pϕ refers to the HPB bus clock.

17.3.2 I²C Bus Control Register 2 (ICCR2)

ICCR2 is an 8-bit readable/writable register that issues start/stop conditions, manipulates the SDA pin, monitors the SCL pin, and controls reset in the control part of the I²C bus.

Bit:	7	6	5	4	3	2	1	0
	BBSY	SCP	SDAO	SDAOP	SCLO	-	IICRST	-
Initial value:	0	1	1	1	1	1	0	1
R/W:	R/W	R/W	R/W	R/W	R	R	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
7	BBSY	0	R/W	Bus Busy Enables to confirm whether the I²C bus is occupied or released and to issue start/stop conditions in master mode. With the clocked synchronous serial format, this bit is always read as 0. With the I²C bus format, this bit is set to 1 when the SDA level changes from high to low under the condition of SCL = high, assuming that the start condition has been issued. This bit is cleared to 0 when the SDA level changes from low to high under the condition of SCL = high, assuming that the stop condition has been issued. Write 1 to BBSY and 0 to SCP to issue a start condition. Follow this procedure when also re-transmitting a start condition. Write 0 in BBSY and 0 in SCP to issue a stop condition.
6	SCP	1	R/W	Start/Stop Issue Condition Disable Controls the issue of start/stop conditions in master mode. To issue a start condition, write 1 in BBSY and 0 in SCP. A retransmit start condition is issued in the same way. To issue a stop condition, write 0 in BBSY and 0 in SCP. This bit is always read as 1. Even if 1 is written to this bit, the data will not be stored.

Bit	Bit Name	Initial Value	R/W	Description
5	SDAO	1	R/W	SDA Output Value Control This bit is used with SDAOP when modifying output level of SDA. This bit should not be manipulated during transfer. 0: When reading, SDA pin outputs low. When writing, SDA pin is changed to output low. 1: When reading, SDA pin outputs high. When writing, SDA pin is changed to output Hi-Z (outputs high by external pull-up resistance).
4	SDAOP	1	R/W	SDAO Write Protect Controls change of output level of the SDA pin by modifying the SDAO bit. To change the output level, clear SDAO and SDAOP to 0 or set SDAO to 1 and clear SDAOP to 0. This bit is always read as 1.
3	SCLO	1	R	SCL Output Level Monitors SCL output level. When SCLO is 1, SCL pin outputs high. When SCLO is 0, SCL pin outputs low.
2	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
1	IICRST	0	R/W	Control Part Reset Resets bits BC[2:0] in ICMR and internal circuits. If this bit is set to 1 when hang-up occurs because of communication failure during I²C bus operation, bits BC[2:0] in ICMR and internal circuits can be reset.
0	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.

17.3.3 I²C Bus Mode Register (ICMR)

ICMR is an 8-bit readable/writable register that selects whether the MSB or LSB is transferred first, performs master mode wait control, and selects the transfer bit count.

Bits BC[2:0] are initialized to H'0 by the IICRST bit in ICCR2.

Bit:	7	6	5	4	3	2	1	0
	MLS	-	-	-	BCWP	BC[2:0]		
Initial value:	0	0	1	1	1	0	0	0
R/W:	R/W	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	MLS	0	R/W	MSB-First/LSB-First Select 0: MSB-first 1: LSB-first Set this bit to 0 when the I²C bus format is used.
6	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
5, 4	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
3	BCWP	1	R/W	BC Write Protect Controls the BC[2:0] modifications. When modifying the BC[2:0] bits, this bit should be cleared to 0. In clocked synchronous serial mode, the BC[2:0] bits should not be modified. 0: When writing, values of the BC[2:0] bits are set. 1: When reading, 1 is always read. When writing, settings of the BC[2:0] bits are invalid.

Bit	Bit Name	Initial Value	R/W	Description
2 to 0	BC[2:0]	000	R/W	Bit Counter These bits specify the number of bits to be transferred next. When read, the remaining number of transfer bits is indicated. With the I²C bus format, the data is transferred with one addition acknowledge bit. Should be made between transfer frames. If these bits are set to a value other than B'000, the setting should be made while the SCL pin is low. The bit value returns to B'000 automatically at the end of a data transfer including the acknowledge bit. And the value becomes B'111 automatically after the stop condition detection. These bits are cleared by a power-on reset and in module standby mode. These bits are also cleared by setting the IICRST bit of ICCR2 to 1. With the clocked synchronous serial format, these bits should not be modified.

I²C Bus Format	Clocked Synchronous Serial Format
000: 9 bits	000: 8 bits
001: 2 bits	001: 1 bit
010: 3 bits	010: 2 bits
011: 4 bits	011: 3 bits
100: 5 bits	100: 4 bits
101: 6 bits	101: 5 bits
110: 7 bits	110: 6 bits
111: 8 bits	111: 7 bits

17.3.4 I²C Bus Interrupt Enable Register (ICIER)

ICIER is an 8-bit readable/writable register that enables or disables interrupt sources and acknowledge bits, sets acknowledge bits to be transferred, and confirms acknowledge bits received.

Bit:	7	6	5	4	3	2	1	0
	TIE	TEIE	RIE	NAKIE	STIE	ACKE	ACKBR	ACKBT
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	TIE	0	R/W	Transmit Interrupt Enable When the TDRE bit in ICSR is set to 1 or 0, this bit enables or disables the transmit data empty interrupt (TXI). 0: Transmit data empty interrupt request (TXI) is disabled. 1: Transmit data empty interrupt request (TXI) is enabled.
6	TEIE	0	R/W	Transmit End Interrupt Enable Enables or disables the transmit end interrupt (TEI) at the rising of the ninth clock while the TDRE bit in ICSR is 1. TEI can be canceled by clearing the TEND bit or the TEIE bit to 0. 0: Transmit end interrupt request (TEI) is disabled. 1: Transmit end interrupt request (TEI) is enabled.
5	RIE	0	R/W	Receive Interrupt Enable Enables or disables the receive data full interrupt request (RXI) when receive data is transferred from ICDRS to ICDRR and the RDRF bit in ICSR is set to 1. RXI can be canceled by clearing the RDRF or RIE bit to 0. 0: Receive data full interrupt request (RXI) is disabled. 1: Receive data full interrupt request (RXI) is enabled.

Bit	Bit Name	Initial Value	R/W	Description
4	NAKIE	0	R/W	NACK Receive Interrupt Enable Enables or disables the NACK detection and arbitration lost/overrun error interrupt request (NAKI) when the NACKF or AL/OVE bit in ICSR is set. NAKI can be canceled by clearing the NACKF, AL/OVE, or NAKIE bit to 0. 0: NACK receive interrupt request (NAKI) is disabled. 1: NACK receive interrupt request (NAKI) is enabled.
3	STIE	0	R/W	Stop Condition Detection Interrupt Enable Enables or disables the stop condition detection interrupt request (STPI) when the STOP bit in ICSR is set. 0: Stop condition detection interrupt request (STPI) is disabled. 1: Stop condition detection interrupt request (STPI) is enabled.
2	ACKE	0	R/W	Acknowledge Bit Judgment Select 0: The value of the receive acknowledge bit is ignored, and continuous transfer is performed. 1: If the receive acknowledge bit is 1, continuous transfer is halted.
1	ACKBR	0	R	Receive Acknowledge In transmit mode, this bit stores the acknowledge data that are returned by the receive device. This bit cannot be modified. This bit can be canceled by setting the BBSY bit in ICCR2 to 1. 0: Receive acknowledge = 0 1: Receive acknowledge = 1
0	ACKBT	0	R/W	Transmit Acknowledge In receive mode, this bit specifies the bit to be sent at the acknowledge timing. 0: 0 is sent at the acknowledge timing. 1: 1 is sent at the acknowledge timing.

17.3.5 I²C Bus Status Register (ICSR)

ICSR is an 8-bit readable/writable register that confirms interrupt request flags and their status.

Bit:	7	6	5	4	3	2	1	0
	TDRE	TEND	RDRF	NACKF	STOP	AL/OVE	AAS	ADZ
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

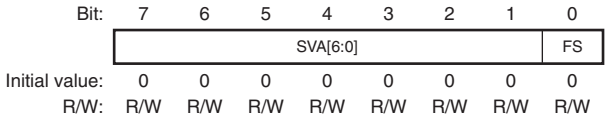
Bit	Bit Name	Initial Value	R/W	Description
7	TDRE	0	R/W	Transmit Data Register Empty [Clearing conditions] • When 0 is written in TDRE after reading TDRE = 1 • When data is written to ICDRT [Setting conditions] • When data is transferred from ICDRT to ICDRS and ICDRT becomes empty • When TRS is set • When the start condition (including retransmission) is issued • When slave mode is changed from receive mode to transmit mode
6	TEND	0	R/W	Transmit End [Clearing conditions] • When 0 is written in TEND after reading TEND = 1 • When data is written to ICDRT [Setting conditions] • When the ninth clock of SCL rises with the I²C bus format while the TDRE flag is 1 • When the final bit of transmit frame is sent with the clocked synchronous serial format

Bit	Bit Name	Initial Value	R/W	Description
5	RDRF	0	R/W	Receive Data Full [Clearing conditions] • When 0 is written in RDRF after reading RDRF = 1 • When ICDRR is read [Setting condition] • When a receive data is transferred from ICDRS to ICDRR
4	NACKF	0	R/W	No Acknowledge Detection Flag [Clearing condition] • When 0 is written in NACKF after reading NACKF = 1 [Setting condition] • When no acknowledge is detected from the receive device in transmission while the ACKE bit in ICIER is 1
3	STOP	0	R/W	Stop Condition Detection Flag [Clearing condition] • When 0 is written in STOP after reading STOP = 1 [Setting condition] • When a stop condition is detected after frame transfer is completed

Bit	Bit Name	Initial Value	R/W	Description
2	AL/OVE	0	R/W	Arbitration Lost Flag/Overrun Error Flag Indicates that arbitration was lost in master mode with the I²C bus format and that the final bit has been received while RDRF = 1 with the clocked synchronous format. When two or more master devices attempt to seize the bus at nearly the same time, if the I²C bus interface 3 detects data differing from the data it sent, it sets AL to 1 to indicate that the bus has been occupied by another master. [Clearing condition] • When 0 is written in AL/OVE after reading AL/OVE = 1 [Setting conditions] • If the internal SDA and SDA pin disagree at the rise of SCL in master transmit mode • When the SDA pin outputs high in master mode while a start condition is detected • When the final bit is received with the clocked synchronous format while RDRF = 1
1	AAS	0	R/W	Slave Address Recognition Flag In slave receive mode, this flag is set to 1 if the first frame following a start condition matches bits SVA[6:0] in SAR. [Clearing condition] • When 0 is written in AAS after reading AAS = 1 [Setting conditions] • When the slave address is detected in slave receive mode • When the general call address is detected in slave receive mode.
0	ADZ	0	R/W	General Call Address Recognition Flag This bit is valid in slave receive mode with the I²C bus format. [Clearing condition] • When 0 is written in ADZ after reading ADZ = 1 [Setting condition] • When the general call address is detected in slave receive mode

17.3.6 Slave Address Register (SAR)

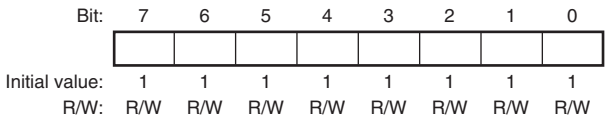
SAR is an 8-bit readable/writable register that selects the communications format and sets the slave address. In slave mode with the I²C bus format, if the upper seven bits of SAR match the upper seven bits of the first frame received after a start condition, this module operates as the slave device.



Bit	Bit Name	Initial Value	R/W	Description
7 to 1	SVA[6:0]	0000000	R/W	Slave Address These bits set a unique address in these bits, differing form the addresses of other slave devices connected to the I²C bus.
0	FS	0	R/W	Format Select 0: I²C bus format is selected 1: Clocked synchronous serial format is selected

17.3.7 I²C Bus Transmit Data Register (ICDRT)

ICDRT is an 8-bit readable/writable register that stores the transmit data. When ICDRT detects the space in the shift register (ICDRS), it transfers the transmit data which is written in ICDRT to ICDRS and starts transferring data. If the next transfer data is written to ICDRT while transferring data of ICDRS, continuous transfer is possible.



17.3.8 I²C Bus Receive Data Register (ICDRR)

ICDRR is an 8-bit register that stores the receive data. When data of one byte is received, ICDRR transfers the receive data from ICDRS to ICDRR and the next data can be received. ICDRR is a receive-only register, therefore the CPU cannot write to this register.

Bit:	7	6	5	4	3	2	1	0
Initial value:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

17.3.9 I²C Bus Shift Register (ICDRS)

ICDRS is a register that is used to transfer/receive data. In transmission, data is transferred from ICDRT to ICDRS and the data is sent from the SDA pin. In reception, data is transferred from ICDRS to ICDRR after data of one byte is received. This register cannot be read directly from the CPU.

Bit:	7	6	5	4	3	2	1	0
Initial value:	-	-	-	-	-	-	-	-
R/W:	-	-	-	-	-	-	-	-

17.3.10 NF2CYC Register (NF2CYC)

NF2CYC is an 8-bit readable/writable register that selects a transfer clock and the range of the noise filtering for the SCL and SDA pins. For details of the noise filter, see section 17.4.7, Noise Filter.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	CKS4	-	-	PRS	NF2CYC
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	CKS4	0	R/W	Transfer Clock Select This bit should be set according to the necessary transfer rate (table 17.3) in master mode. For 1-Mbyte version, this bit is reserved and always read as 0. The write value should always be 0.
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	PRS	0	R/W	Pulse Width Ratio Select Specifies the ratio of the high-level period to the low-level period for the SCL signal. 0: The ratio of high to low is 0.5 to 0.5. 1: The ratio of high to low is about 0.4 to 0.6.
0	NF2CYC	0	R/W	Noise Filtering Range Select 0: The noise less than one cycle of the peripheral clock can be filtered out 1: The noise less than two cycles of the peripheral clock can be filtered out

17.4 Operation

The I²C bus interface 3 can communicate either in I²C bus mode or clocked synchronous serial mode by setting FS in SAR.

17.4.1 I²C Bus Format

Figure 17.3 shows the I²C bus formats. Figure 17.4 shows the I²C bus timing. The first frame following a start condition always consists of eight bits.

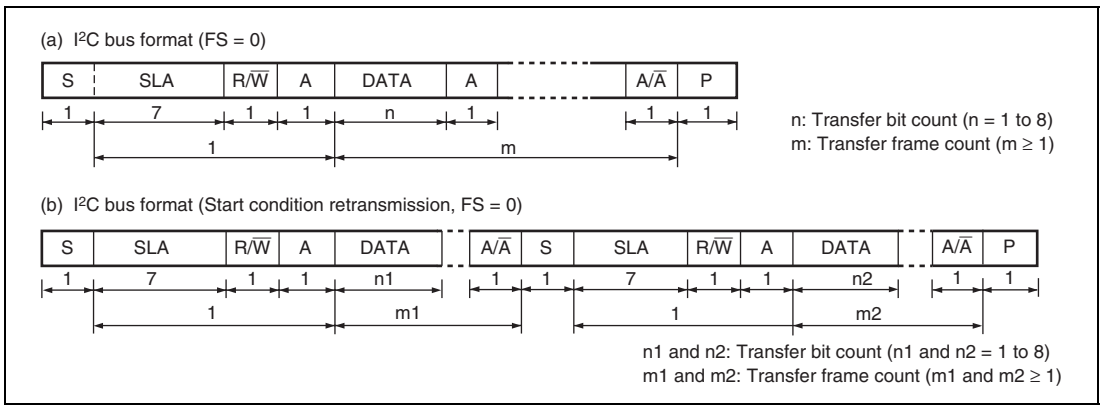


Figure 17.3 I²C Bus Formats

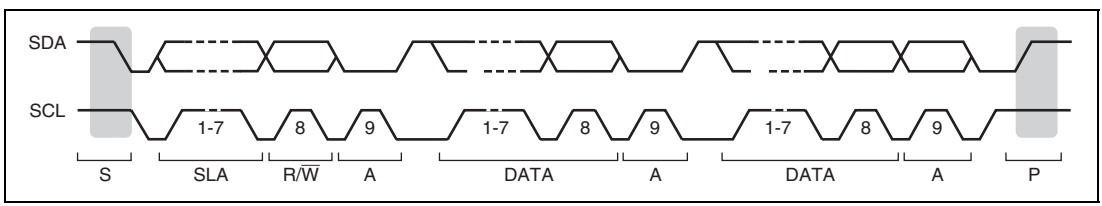


Figure 17.4 I²C Bus Timing

[Legend]

S: Start condition. The master device drives SDA from high to low while SCL is high.

SLA: Slave address

R/W: Indicates the direction of data transfer: from the slave device to the master device when R/W is 1, or from the master device to the slave device when R/W is 0.

A: Acknowledge. The receive device drives SDA to low.

DATA: Transfer data

P: Stop condition. The master device drives SDA from low to high while SCL is high.

17.4.2 Master Transmit Operation

In master transmit mode, the master device outputs the transmit clock and transmit data, and the slave device returns an acknowledge signal. For master transmit mode operation timing, refer to figures 17.5 and 17.6. The transmission procedure and operations in master transmit mode are described below.

1. Set the ICE bit in ICCR1 to 1. Also, set bits CKS[3:0] in ICCR1. (Initial setting)
2. Read the BBSY flag in ICCR2 to confirm that the bus is released. Set the MST and TRS bits in ICCR1 to select master transmit mode. Then, write 1 to BBSY and 0 to SCP. (Start condition issued) This generates the start condition.
3. After confirming that TDRE in ICSR has been set, write the transmit data (the first byte data show the slave address and R/$\overline{W}$) to ICDRT. At this time, TDRE is automatically cleared to 0, and data is transferred from ICDRT to ICDRS. TDRE is set again.
4. When transmission of one byte data is completed while TDRE is 1, TEND in ICSR is set to 1 at the rise of the 9th transmit clock pulse. Read the ACKBR bit in ICIER, and confirm that the slave device has been selected. Then, write second byte data to ICDRT. When ACKBR is 1, the slave device has not been acknowledged, so issue the stop condition. To issue the stop condition, write 0 to BBSY and SCP. SCL is fixed low until the transmit data is prepared or the stop condition is issued.
5. The transmit data after the second byte is written to ICDRT every time TDRE is set.
6. Write the number of bytes to be transmitted to ICDRT. Wait until TEND is set (the end of last byte data transmission) while TDRE is 1, or wait for NACK (NACKF in ICSR = 1) from the receive device while ACKE in ICIER is 1. Then, issue the stop condition to clear TEND or NACKF.
7. When the STOP bit in ICSR is set to 1, the operation returns to the slave receive mode.

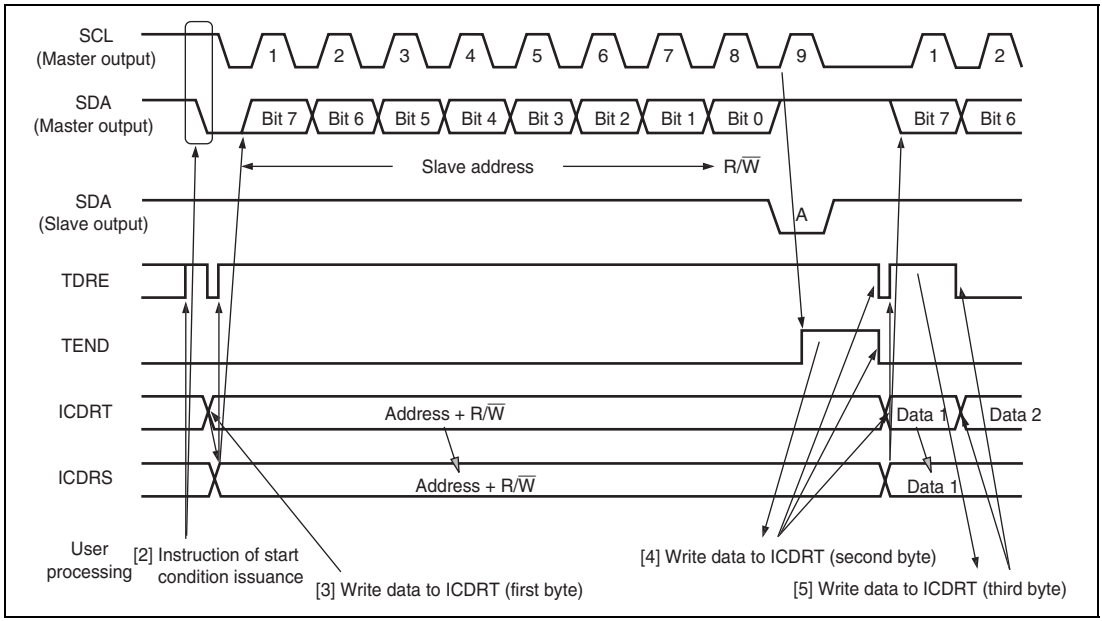


Figure 17.5 Master Transmit Mode Operation Timing (1)

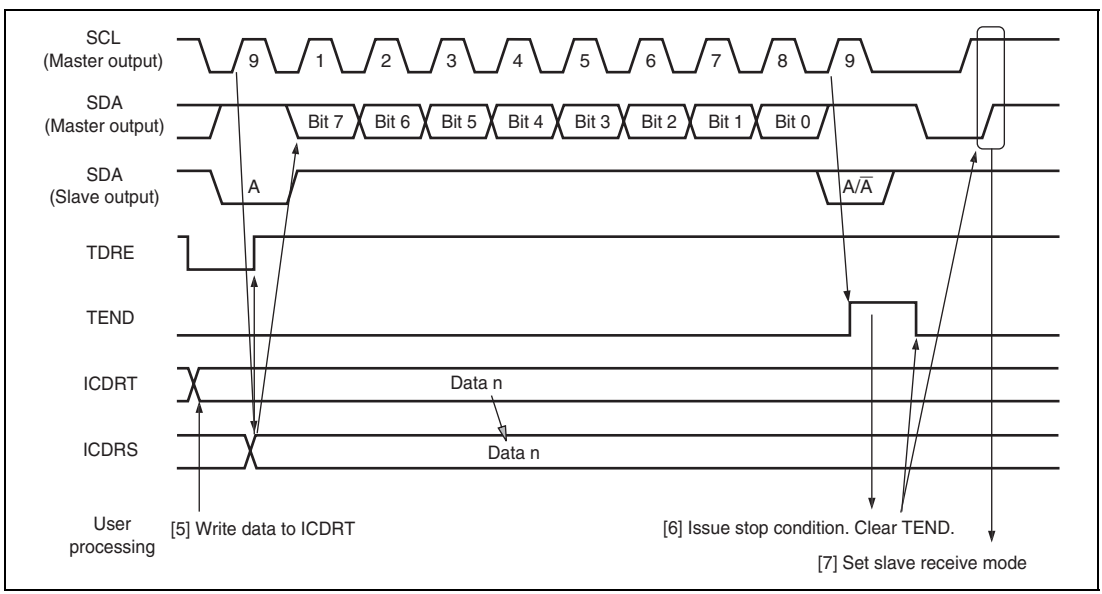


Figure 17.6 Master Transmit Mode Operation Timing (2)

17.4.3 Master Receive Operation

In master receive mode, the master device outputs the receive clock, receives data from the slave device, and returns an acknowledge signal. For master receive mode operation timing, refer to figures 17.7 and 17.8. The reception procedure and operations in master receive mode are shown below.

1. Clear the TEND bit in ICSR to 0, then clear the TRS bit in ICCR1 to 0 to switch from master transmit mode to master receive mode. Then, clear the TDRE bit to 0.
2. When ICDRR is read (dummy data read), reception is started, and the receive clock is output, and data received, in synchronization with the internal clock. The master device outputs the level specified by ACKBT in ICIER to SDA, at the 9th receive clock pulse.
3. After the reception of first frame data is completed, the RDRF bit in ICSR is set to 1 at the rise of 9th receive clock pulse. At this time, the receive data is read by reading ICDRR, and RDRF is cleared to 0.
4. The continuous reception is performed by reading ICDRR every time RDRF is set. If 8th receive clock pulse falls after reading ICDRR by the other processing while RDRF is 1, SCL is fixed low until ICDRR is read.
5. If next frame is the last receive data, set the RCVD bit in ICCR1 to 1 before reading ICDRR. This enables the issuance of the stop condition after the next reception.
6. When the RDRF bit is set to 1 at rise of the 9th receive clock pulse, issue the stage condition.
7. When the STOP bit in ICSR is set to 1, read ICDRR. Then clear the RCVD bit to 0.
8. The operation returns to the slave receive mode.

Note: If only one byte is received, read ICDRR (dummy-read) after the RCVD bit in ICCR1 is set.

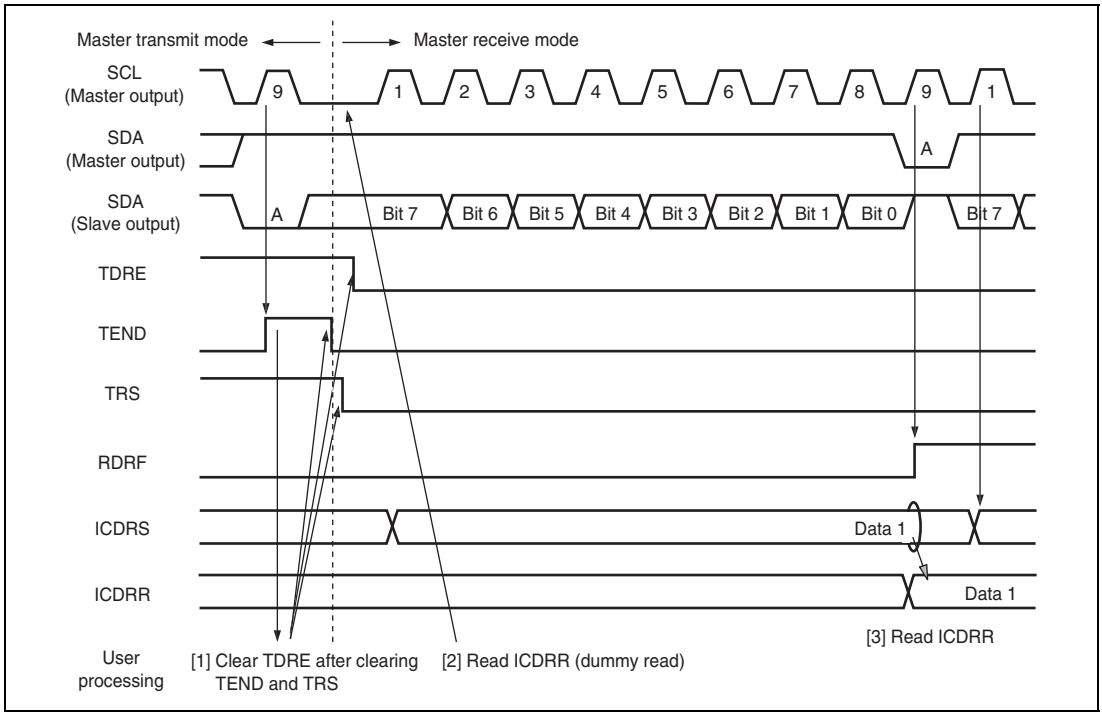


Figure 17.7 Master Receive Mode Operation Timing (1)

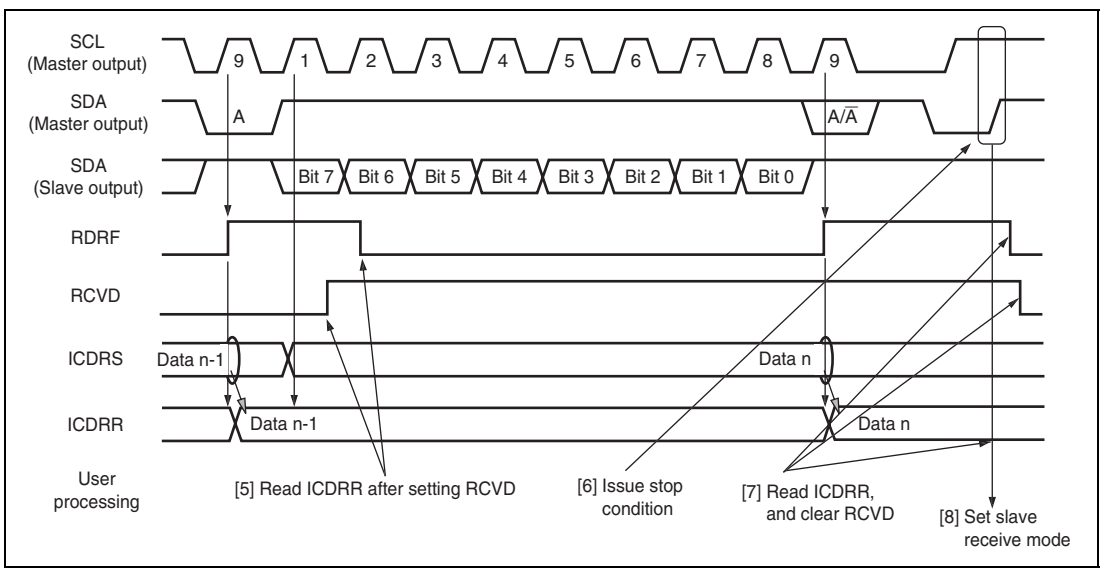


Figure 17.8 Master Receive Mode Operation Timing (2)

17.4.4 Slave Transmit Operation

In slave transmit mode, the slave device outputs the transmit data, while the master device outputs the receive clock and returns an acknowledge signal. For slave transmit mode operation timing, refer to figures 17.9 and 17.10.

The transmission procedure and operations in slave transmit mode are described below.

1. Set the ICE bit in ICCR1 to 1. Set bits CKS[3:0] in ICCR1. (Initial setting) Set the MST and TRS bits in ICCR1 to select slave receive mode, and wait until the slave address matches.
2. When the slave address matches in the first frame following detection of the start condition, the slave device outputs the level specified by ACKBT in ICIER to SDA, at the rise of the 9th clock pulse. At this time, if the 8th bit data (R/W) is 1, the TRS bit in ICCR1 and the TDRE bit in ICSR are set to 1, and the mode changes to slave transmit mode automatically. The continuous transmission is performed by writing transmit data to ICDRT every time TDRE is set.
3. If TDRE is set after writing last transmit data to ICDRT, wait until TEND in ICSR is set to 1, with TDRE = 1. When TEND is set, clear TEND.
4. Clear TRS for the end processing, and read ICDRR (dummy read). SCL is opened.
5. Clear TDRE.

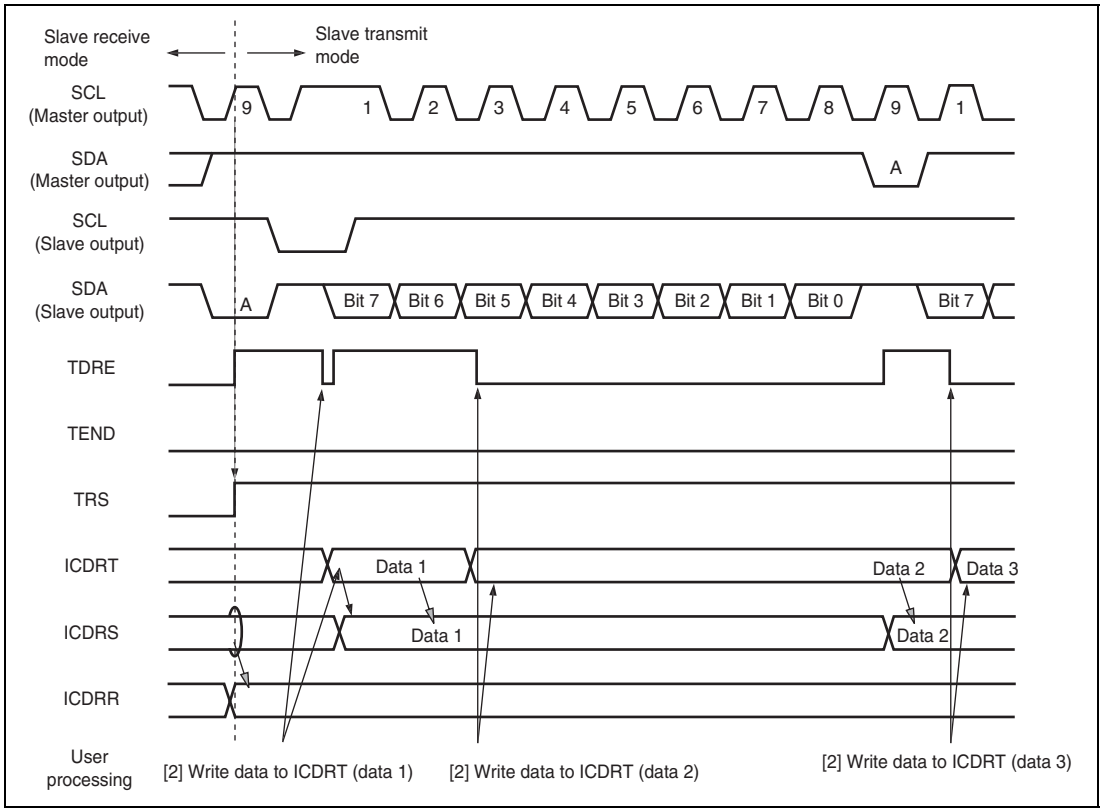


Figure 17.9 Slave Transmit Mode Operation Timing (1)

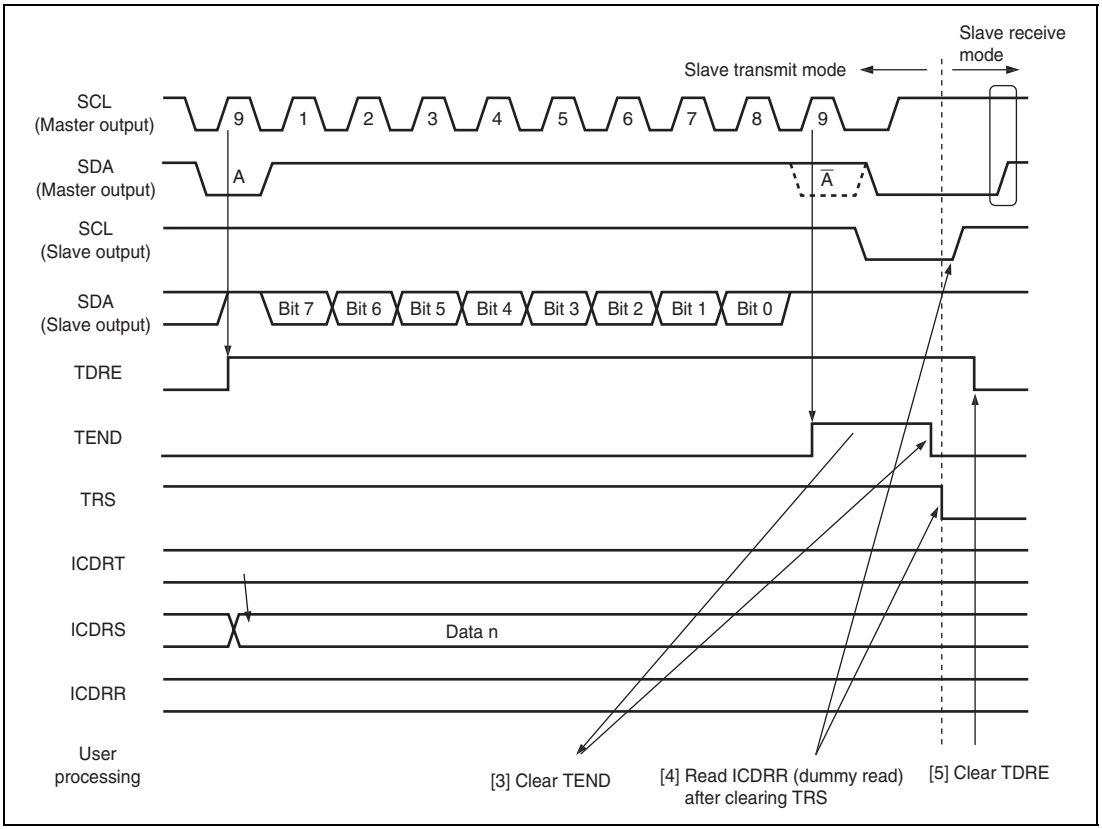


Figure 17.10 Slave Transmit Mode Operation Timing (2)

17.4.5 Slave Receive Operation

In slave receive mode, the master device outputs the transmit clock and transmit data, and the slave device returns an acknowledge signal. For slave receive mode operation timing, refer to figures 17.11 and 17.12. The reception procedure and operations in slave receive mode are described below.

1. Set the ICE bit in ICCR1 to 1. Set bits CKS[3:0] in ICCR1. (Initial setting) Set the MST and TRS bits in ICCR1 to select slave receive mode, and wait until the slave address matches.
2. When the slave address matches in the first frame following detection of the start condition, the slave device outputs the level specified by ACKBT in ICIER to SDA, at the rise of the 9th clock pulse. At the same time, RDRF in ICSR is set to read ICDRR (dummy read). (Since the read data show the slave address and R/W, it is not used.)
3. Read ICDRR every time RDRF is set. If 8th receive clock pulse falls while RDRF is 1, SCL is fixed low until ICDRR is read. The change of the acknowledge before reading ICDRR, to be returned to the master device, is reflected to the next transmit frame.
4. The last byte data is read by reading ICDRR.

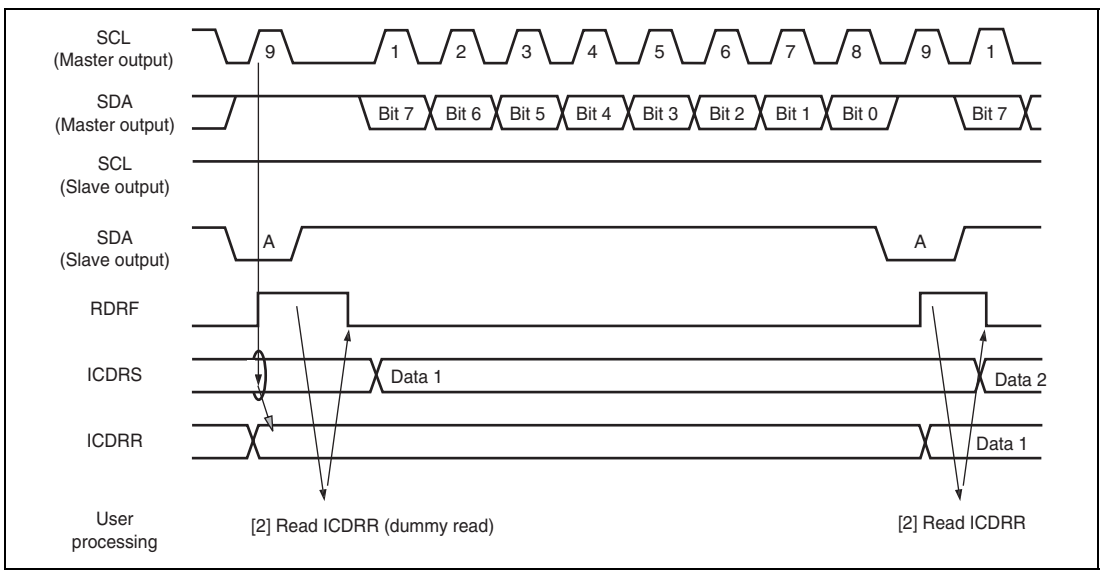


Figure 17.11 Slave Receive Mode Operation Timing (1)

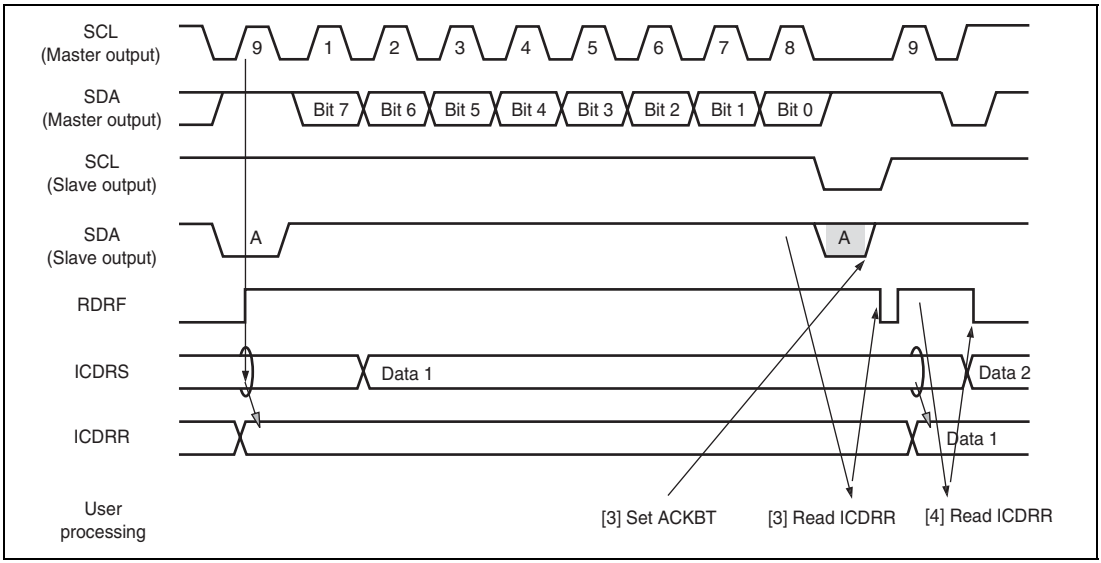


Figure 17.12 Slave Receive Mode Operation Timing (2)

17.4.6 Clocked Synchronous Serial Format

This module can be operated with the clocked synchronous serial format, by setting the FS bit in SAR to 1. When the MST bit in ICCR1 is 1, the transfer clock output from SCL is selected. When MST is 0, the external clock input is selected.

(1) Data Transfer Format

Figure 17.13 shows the clocked synchronous serial transfer format.

The transfer data is output from the fall to the fall of the SCL clock, and the data at the rising edge of the SCL clock is guaranteed. The MLS bit in ICMR sets the order of data transfer, in either the MSB first or LSB first. The output level of SDA can be changed during the transfer wait, by the SDAO bit in ICCR2.

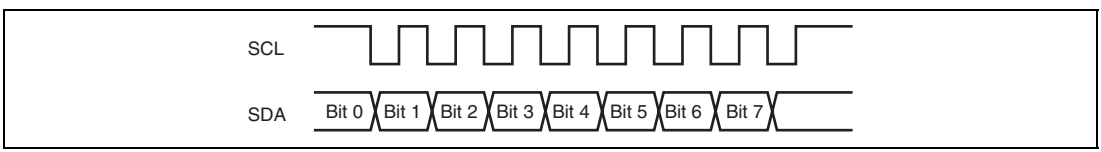


Figure 17.13 Clocked Synchronous Serial Transfer Format

(2) Transmit Operation

In transmit mode, transmit data is output from SDA, in synchronization with the fall of the transfer clock. The transfer clock is output when MST in ICCR1 is 1, and is input when MST is 0. For transmit mode operation timing, refer to figure 17.14. The transmission procedure and operations in transmit mode are described below.

1. Set the ICE bit in ICCR1 to 1. Set the MST and CKS[3:0] bits in ICCR1. (Initial setting)
2. Set the TRS bit in ICCR1 to select the transmit mode. Then, TDRE in ICSR is set.
3. Confirm that TDRE has been set. Then, write the transmit data to ICDRT. The data is transferred from ICDRT to ICDRS, and TDRE is set automatically. The continuous transmission is performed by writing data to ICDRT every time TDRE is set. When changing from transmit mode to receive mode, clear TRS while TDRE is 1.

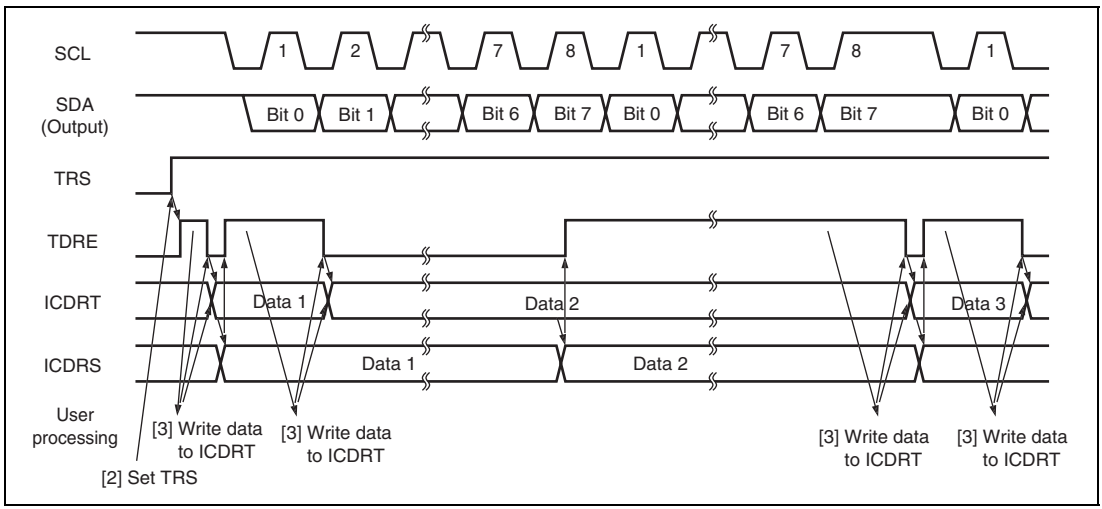


Figure 17.14 Transmit Mode Operation Timing

(3) Receive Operation

In receive mode, data is latched at the rise of the transfer clock. The transfer clock is output when MST in ICCR1 is 1, and is input when MST is 0. For receive mode operation timing, refer to figure 17.15. The reception procedure and operations in receive mode are described below.

1. Set the ICE bit in ICCR1 to 1. Set bits CKS[3:0] in ICCR1. (Initial setting)
2. When the transfer clock is output, set MST to 1 to start outputting the receive clock.
3. When the receive operation is completed, data is transferred from ICDRS to ICDRR and RDRF in ICSR is set. When MST = 1, the next byte can be received, so the clock is continually output. The continuous reception is performed by reading ICDRR every time RDRF is set. When the 8th clock is risen while RDRF is 1, the overrun is detected and AL/OVE in ICSR is set. At this time, the previous reception data is retained in ICDRR.
4. To stop receiving when MST = 1, set RCVD in ICCR1 to 1, then read ICDRR. Then, SCL is fixed high after receiving the next byte data.

Notes: Follow the steps below to receive only one byte with MST = 1 specified. See figure 17.16 for the operation timing.

1. Set the ICE bit in ICCR1 to 1. Set bits CKS[3:0] in ICCR1. (Initial setting)
2. Set MST = 1 while the RCVD bit in ICCR1 is 0. This causes the receive clock to be output.
3. Check if the BC2 bit in ICMR is set to 1 and then set the RCVD bit in ICCR1 to 1. This causes the SCL to be fixed to the high level after outputting one byte of the receive clock.

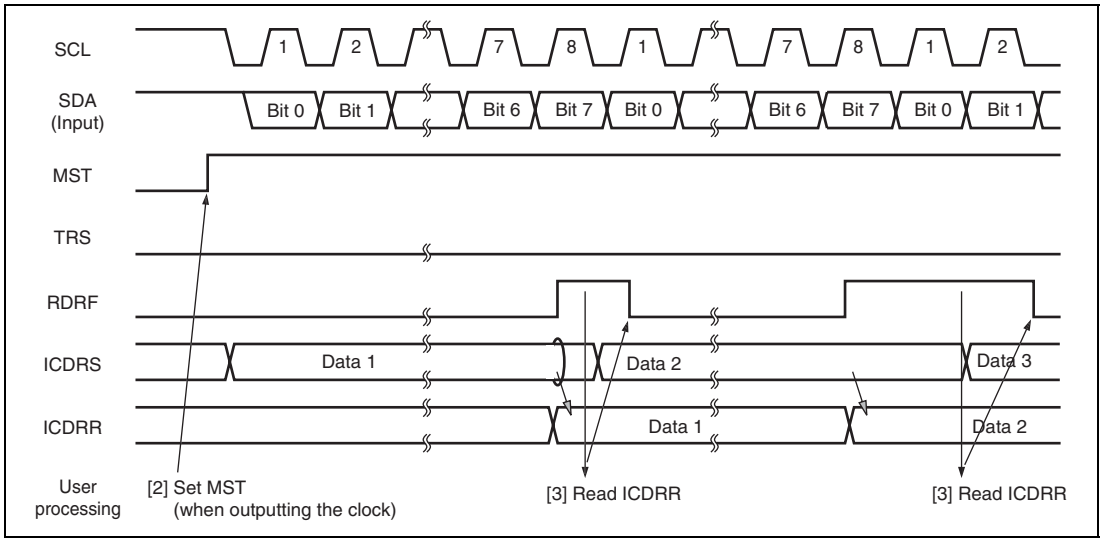


Figure 17.15 Receive Mode Operation Timing

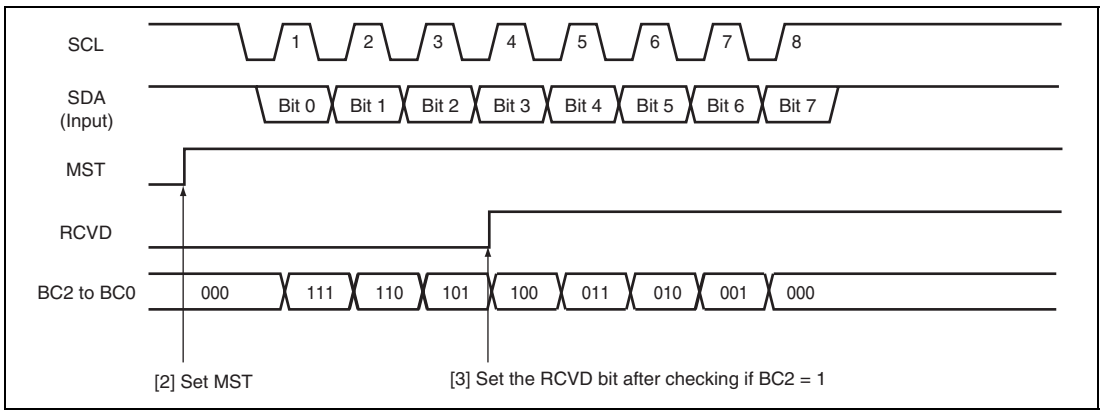


Figure 17.16 Operation Timing For Receiving One Byte (MST = 1)

17.4.7 Noise Filter

The logic levels at the SCL and SDA pins are routed through noise filters before being latched internally. Figure 17.17 shows a block diagram of the noise filter circuit.

The noise filter consists of three cascaded latches and a match detector. The SCL (or SDA) input signal is sampled on the peripheral clock. When NF2CYC is set to 0, this signal is not passed forward to the next circuit unless the outputs of both latches agree. When NF2CYC is set to 1, this signal is not passed forward to the next circuit unless the outputs of three latches agree. If they do not agree, the previous value is held.

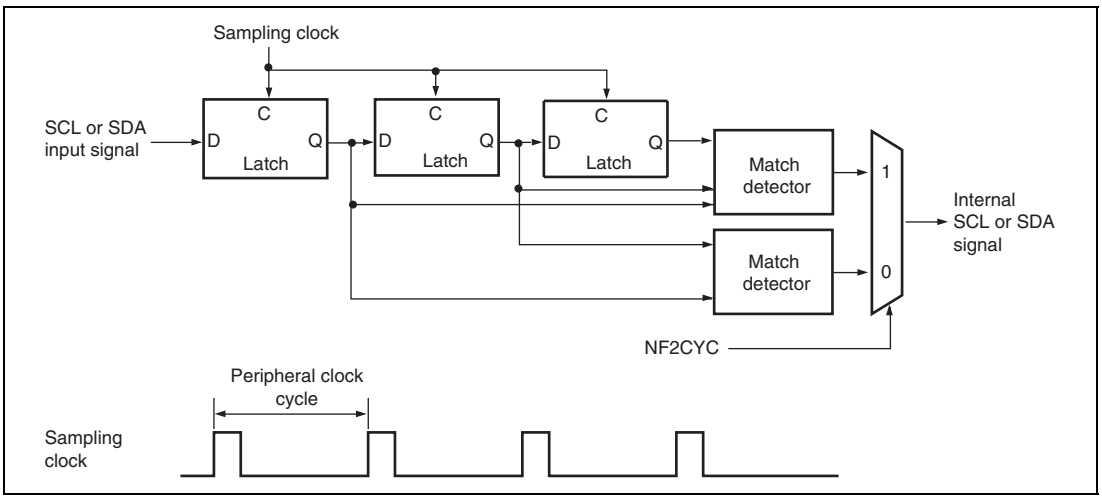


Figure 17.17 Block Diagram of Noise Filter

17.4.8 Example of Use

Flowcharts in respective modes that use the I²C bus interface 3 are shown in figures 17.18 to 17.21.

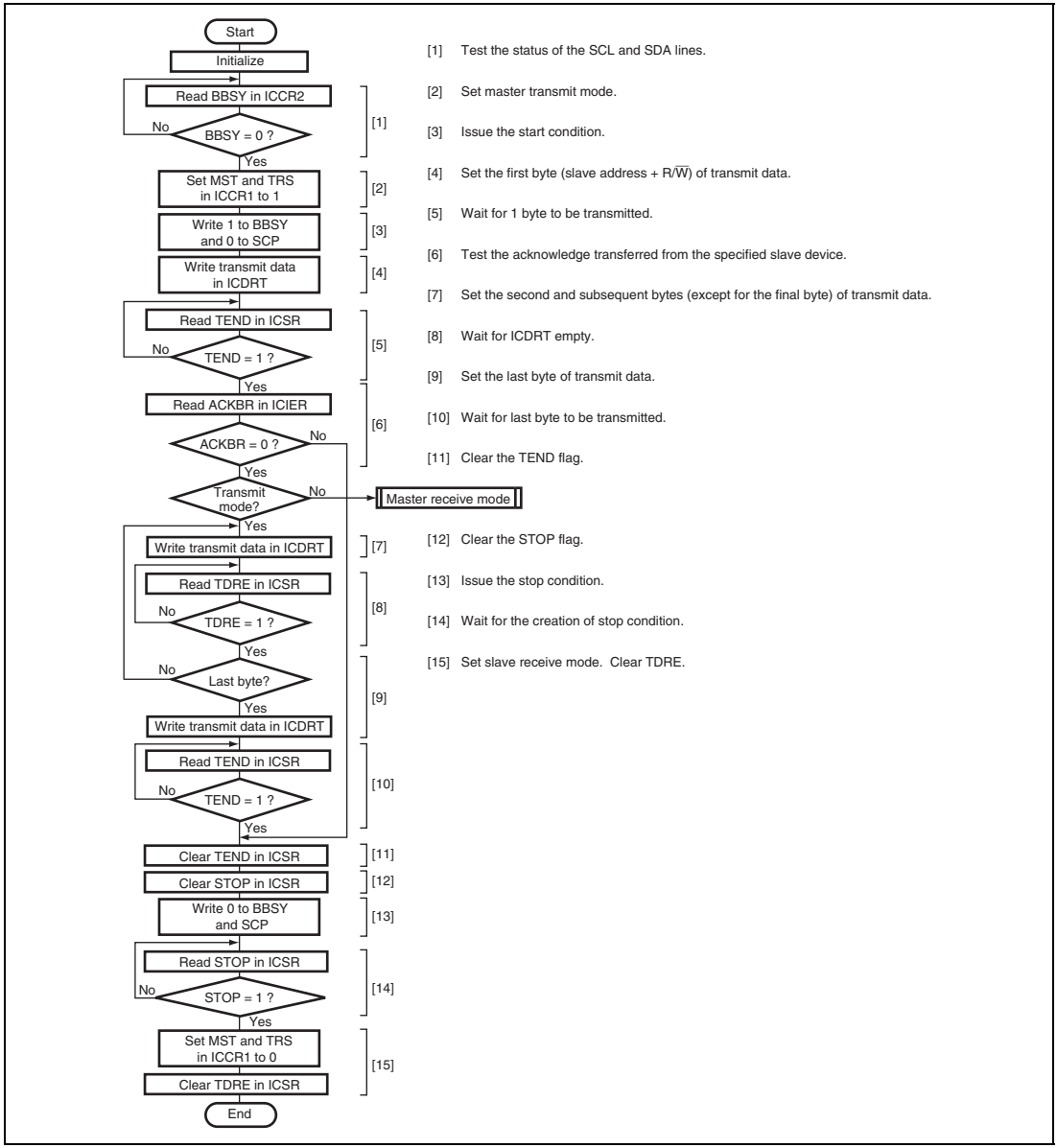
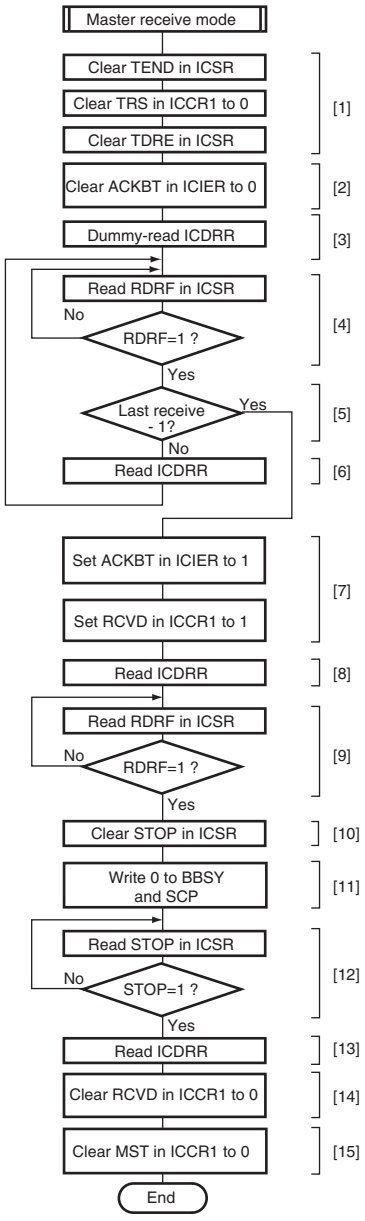


Figure 17.18 Sample Flowchart for Master Transmit Mode



[1] Clear TEND, select master receive mode, and then clear TDRE. *1

[2] Set acknowledge to the transmit device. *1

[3] Dummy-read ICDDR. *1

[4] Wait for 1 byte to be received*2

[5] Check whether it is the (last receive - 1).*2

[6] Read the receive data.

[7] Set acknowledge of the final byte. Disable continuous reception (RCVD = 1).*2

[8] Read the (final byte - 1) of received data.

[9] Wait for the last byte to be receive.

[10] Clear the STOP flag.

[11] Issue the stop condition.

[12] Wait for the creation of stop condition.

[13] Read the last byte of receive data.

[14] Clear RCVD.

[15] Set slave receive mode.

Notes: 1. Make sure that no interrupt will be generated during steps [1] to [3].
2. When the (last receive -1) is checked (when step [5] is approved), make sure that no interrupt will be generated during steps [4], [5], and [7].

[Complement] When the size of receive data is only one byte in reception, steps [2] to [6] are skipped after step [1], before jumping to step [7]. The step [8] is dummy-read in ICDRR.

Figure 17.19 Sample Flowchart for Master Receive Mode

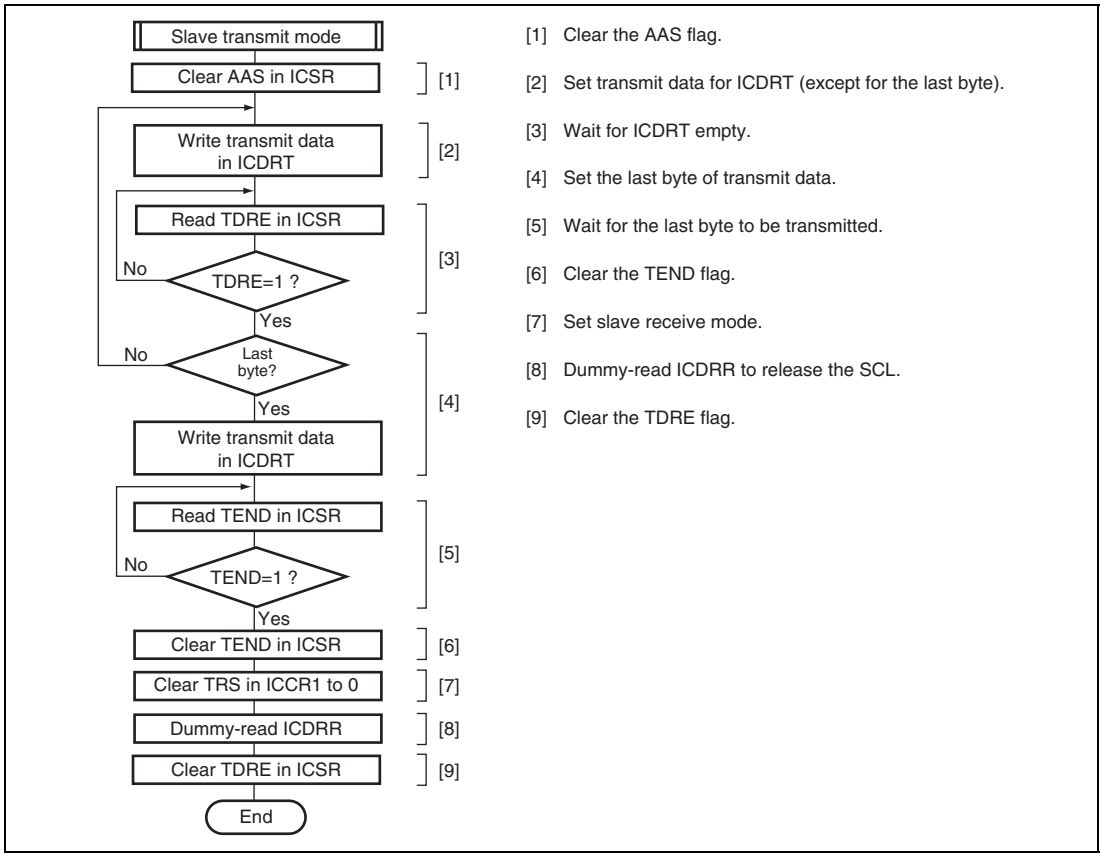


Figure 17.20 Sample Flowchart for Slave Transmit Mode

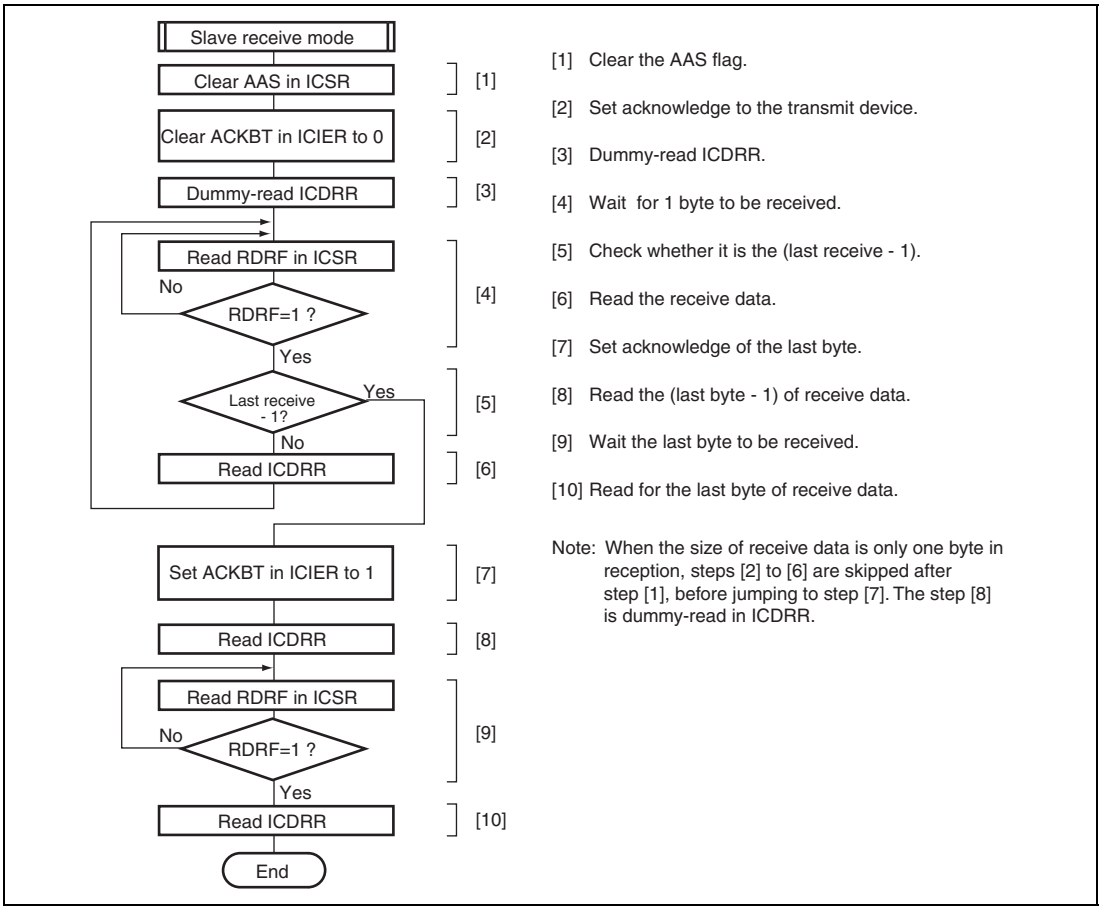


Figure 17.21 Sample Flowchart for Slave Receive Mode

17.5 Interrupt Requests

There are six interrupt requests in this module; transmit data empty, transmit end, receive data full, NACK detection, STOP recognition, and arbitration lost/overrun error. Table 17.4 shows the contents of each interrupt request.

Table 17.4 Interrupt Requests

Interrupt Request	Abbreviation	Interrupt Condition	I²C Bus Format	Clocked Synchronous Serial Format
Transmit data Empty	TXI	(TDRE = 1) • (TIE = 1)	√	√
Transmit end	TEI	(TEND = 1) • (TEIE = 1)	√	√
Receive data full	RXI	(RDRF = 1) • (RIE = 1)	√	√
STOP recognition	STPI	(STOP = 1) • (STIE = 1)	√	—
NACK detection	NAKI	{(NACKF = 1) + (AL = 1)} • (NAKIE = 1)	√	—
Arbitration lost/ overrun error			√	√

When the interrupt condition described in table 17.4 is 1, the CPU executes an interrupt exception handling. Note that a TXI or RXI interrupt can activate the direct memory access controller if the setting for direct memory access controller activation has been made. In such a case, an interrupt request is not sent to the CPU. Interrupt sources should be cleared in the exception handling. The TDRE and TEND bits are automatically cleared to 0 by writing the transmit data to ICDRT. The RDRF bit is automatically cleared to 0 by reading ICDRR. The TDRE bit is set to 1 again at the same time when the transmit data is written to ICDRT. Therefore, when the TDRE bit is cleared to 0, then an excessive data of one byte may be transmitted.

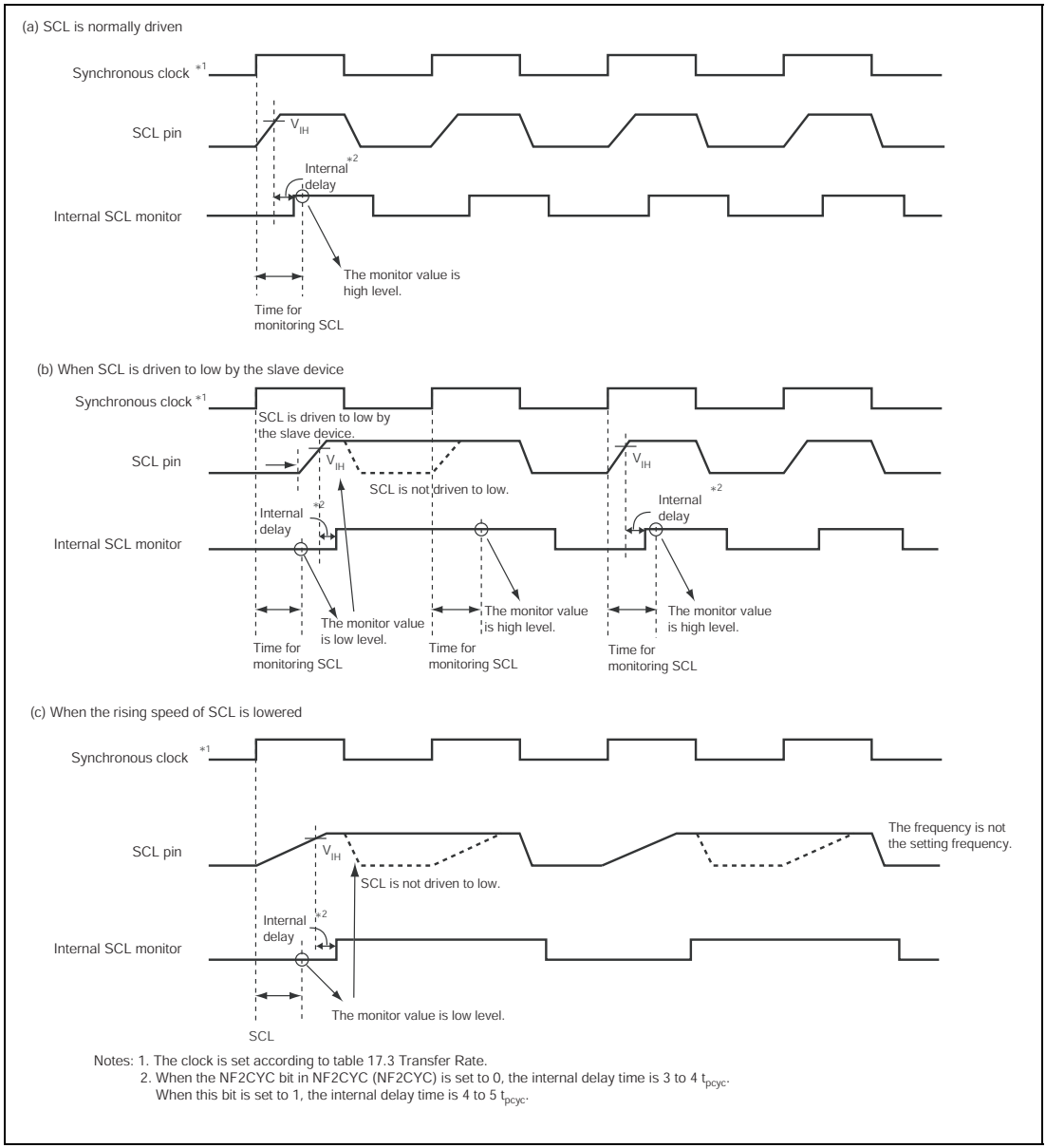
17.6 Bit Synchronous Circuit

In master mode, this module has a possibility that high level period may be short in the two states described below.

- When SCL is driven to low by the slave device
- When the rising speed of SCL is lowered by the load of the SCL line (load capacitance or pull-up resistance)

Therefore, it monitors SCL and communicates by bit with synchronization.

Figure 17.22 shows the timing of the bit synchronous circuit and table 17.5 shows the time when the SCL output changes from low to Hi-Z then SCL is monitored.



Notes: 1. The clock is set according to table 17.3 Transfer Rate.
2. When the NF2CYC bit in NF2CYC (NF2CYC) is set to 0, the internal delay time is 3 to 4 t_{pcyc}.
When this bit is set to 1, the internal delay time is 4 to 5 t_{pcyc}.

Figure 17.22 Bit Synchronous Circuit Timing

Table 17.5 Time for Monitoring SCL

CKS4	CKS3	CKS2	Time for Monitoring SCL
0	0	0	9 tpcyc*
		1	21 tpcyc*
	1	0	39 tpcyc*
		1	87 tpcyc*
1	0	0	79 tpcyc*
		1	175 tpcyc*
	1	0	159 tpcyc*
		1	351 tpcyc*

Note: * tpcyc indicates the frequency of the HPB bus clock (Pϕ).

17.7 Usage Notes

17.7.1 Note on Setting for Multi-Master Operation

In multi-master operation, when the transfer rate setting for this module (ICCR1.CKS[3:0]) makes this LSI slower than the other masters, pulse cycles with an unexpected length will infrequently be output on SCL.

Be sure to specify a transfer rate that is at least 1/1.8 of the fastest transfer rate among the other masters.

17.7.2 Note on Master Receive Mode

Reading ICDRR around the falling edge of the 8th clock might fail to fetch the receive data.

In addition, when RCVD is set to 1 around the falling edge of the 8th clock and the receive buffer full, a stop condition may not be issued.

Use either 1 or 2 below as a measure against the situations above.

1. In master receive mode, read ICDRR before the rising edge of the 8th clock.
2. In master receive mode, set the RCVD bit to 1 so that transfer proceeds in byte units.

17.7.3 Note on Setting ACKBT in Master Receive Mode

In master receive mode operation, set ACKBT before the falling edge of the 8th SCL cycle of the last data being continuously transferred. Not doing so can lead to an overrun for the slave transmission device.

17.7.4 Note on the States of Bits MST and TRN when Arbitration is Lost

When sequential bit-manipulation instructions are used to set the MST and TRS bits to select master transmission in multi-master operation, a conflicting situation where AL in ICSR = 1 but the mode is master transmit mode (MST = 1 and TRS = 1) may arise; this depends on the timing of the loss of arbitration when the bit manipulation instruction for TRS is executed.

This can be avoided in either of the following ways.

- In multi-master operation, use the MOV instruction to set the MST and TRS bits.
- When arbitration is lost, check whether the MST and TRS bits are 0. If the MST and TRS bits have been set to a value other than 0, clear the bits to 0.

17.7.5 Notes on Master Reception Mode of I²C Bus Interface Mode

After the completion of a master reception, confirm the falling of the ninth clock on SCL signal, and then issue a stop condition or re-issue a start condition.

17.7.6 Notes on Bits IICRST and BBSY

Writing 1 to the IICRST bit in ICCR2 causes the MCU to release the SCL and SDA pins. At this point, if the SDA pin state changes from low to high level while the SCL pin is held high, a stop condition is recognized and the BBSY bit in ICCR2 is cleared to 0.

17.7.7 Notes on the Time of Stop Condition Generation in Master Transmit Mode

(1) Phenomenon

When a stop condition is issued in master transmit mode while the ACKE bit in the I²C bus interrupt enable register (ICIER) is 1, the stop condition may not be normally output depending on the issued timing.

(2) Countermeasure

In master transmit mode while the ACKE bit in the I²C bus interrupt enable register (ICIER) is 1, recognize the falling edge of the ninth clock before issuance of the stop condition. The falling of the ninth clock can be confirmed by checking the SCLO bit in the ICCR2 register (ICCR2).

Section 18 Serial Peripheral Interface (HSPI)

18.1 Overview

This LSI incorporates one channel of the Serial Peripheral Interface (HSPI).

18.1.1 Features

The HSPI has the following features.

- Operating mode: Master mode or Slave mode.
- The transmission and receive sections within the module are double buffered to allow duplex communication.
- A flexible internal bus clock division strategy allows a wide range of bit rates to be supported.
- The programmable clock control logic allows setting for two different transmit protocols and accommodates transmit and receive functions on either edge of the serial bit clock.
- Error detection logic is provided for warning of the reception buffer overflow.
- The HSPI has a facility to generate the chip select signal to slave modules when configured as a master mode either automatically as part of the data transfer process, or under the manual control of the host processor.
- Independent DMA transfer of data for transmission and received data is possible through two DMA channels.

Figure 18.1 is a block diagram of the HSPI.

Note that in this section "internal bus clock" and "peripheral clock" refer to the peripheral module clock (clkp).

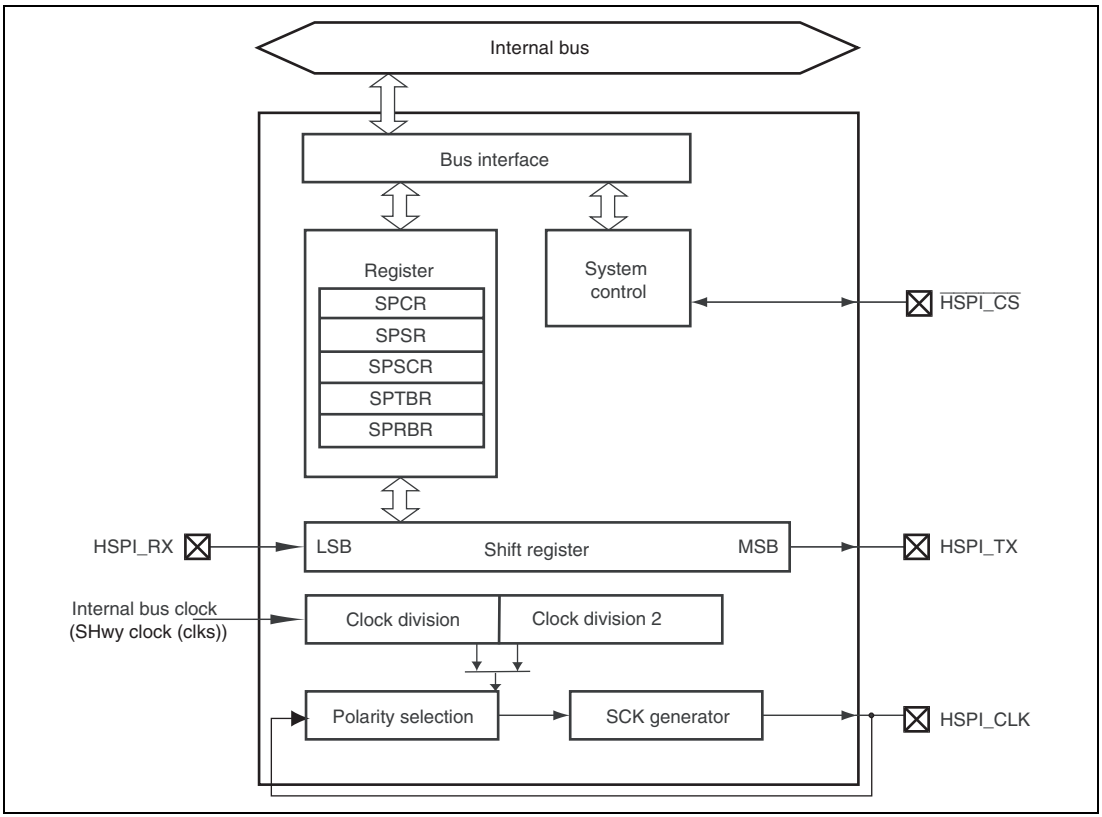


Figure 18.1 Block Diagram of HSPI

18.1.2 Input/Output Pins

The input/output pins of the HSPI are shown in table 18.1.

Table 18.1 Pin Configuration

Function	Pin Name	I/O	Description
Serial bit clock pin	HSPI_CLK	Input/Output	Clock input/output
Transmit data pin	HSPI_TX	Output	Transmit data output
Receive data pin	HSPI_RX	Input	Receive data input
Chip select pin	$\overline{\text{HSPI\_CS}}$	Input/Output	Chip select

18.1.3 Register Configuration

Table 18.2 shows the HSPI register configuration.

Base address:
H'FFFC 7000

Table 18.2 (1) Register Configuration

Register Name	Abbreviation	R/W	Offset Address from Base Address	Access Size
Control register	SPCR	R/W	H'00	32
Status register	SPSR	R*	H'04	32
System control register	SPSCR	R/W	H'08	32
Transmit buffer register	SPTBR	R/W	H'0C	32
Receive buffer register	SPRBR	R	H'10	32

Notes: Do not write to addresses other than those listed above, otherwise normal operation cannot be guaranteed. Values read from other addresses are undefined.

* To clear the flag, only 0s are written to bits 4 and 3.

Table 18.2 (2) Register State in Each Operating Mode

Abbreviation	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
SPCR	Initialized	Initialized	Retained	Retained	Retained	Initialized
SPSR	Initialized	Initialized	Retained	Retained	Retained	Initialized
SPSCR	Initialized	Initialized	Retained	Retained	Retained	Initialized
SPTBR	Initialized	Initialized	Retained	Retained	Retained	Initialized
SPRBR	Initialized	Initialized	Retained	Retained	Retained	Initialized

18.2 Register Descriptions

[Legend for Register Description]

Initial value:	Register value after a reset
—:	Undefined value
R/W:	Readable/writable. The written value can be read.
R/WC0:	Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored.
R:	Read-only. The write value should always be 0.
—/W:	Write-only. The read value is undefined.

18.2.1 Control Register (SPCR)

SPCR is a 32-bit readable/writable register that controls the transfer data of shift timing and specifies the clock polarity and frequency.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	FBS	CLKP	IDIV	CLKC4	CLKC3	CLKC2	CLKC1	CLKC0
Initial value:	—	—	—	—	—	—	—	—	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	—	R	Reserved These bits are always read as undefined value. The write value should always be 0.
7	FBS	0	R/W	First Bit Start This bit can be read or written to. The write value can be read. Controls the timing relationship between each bit of transferred data and the serial bit clock. 0: The first bit transmitted from the HSPI module is set up such that it can be sampled by the receiving device on the first edge of HSPI_CLK after the $\overline{\text{HSPI\_CS}}$ pin goes low. Similarly the first received bit is sampled on the first edge of HSPI_CLK after the $\overline{\text{HSPI\_CS}}$ pin goes low. 1: The first bit transmitted from the HSPI module is set up such that it can be sampled by the receiving device on the second edge of HSPI_CLK after the $\overline{\text{HSPI\_CS}}$ pin goes low. Similarly the first received bit is sampled on the second edge of HSPI_CLK after the $\overline{\text{HSPI\_CS}}$ pin goes low.

Bit	Bit Name	Initial Value	R/W	Description
6	CLKP	0	R/W	Serial Clock Polarity 0: HSPI_CLK signal is not inverted and so is low when inactive. 1: HSPI_CLK signal is inverted and so is high when inactive.
5	IDIV	0	R/W	Initial Clock Division Ratio 0: The internal bus clock (clks) is divided by a factor of 16 initially to create an intermediate frequency, which is further divided to create the serial bit clock when master mode. 1: The internal bus clock (clks) is divided by a factor of 128 initially to create an intermediate frequency, which is further divided to create the serial bit clock when master mode.
4	CLKC4	0	R/W	Clock Division Count These bits determine the number of intermediate frequency cycles long both the high and low periods of the serial bit clock. 00000: 1 intermediate frequency cycle. Serial bit clock frequency = Intermediate frequency / 2. 00001: 2 Intermediate frequency cycles. Serial bit clock frequency = Intermediate frequency / 4. 00010: 3 intermediate frequency cycles. Serial bit clock frequency = Intermediate frequency / 6. : : 11111: 32 intermediate frequency cycles. Serial bit clock frequency = Intermediate frequency / 64.
3	CLKC3	0	R/W	
2	CLKC2	0	R/W	
1	CLKC1	0	R/W	
0	CLKC0	0	R/W	

The serial bit clock frequency can be computed using the following formula:

$$\text{Serial bit clock frequency} = \frac{\text{Peripheral clock frequency}}{(\text{Initial division ratio} \times (\text{Clock division count} + 1) \times 2)}$$

When the HSPI is configured as a slave, the IDIV and CLKC bits are ignored and the HSPI synchronizes to the externally supplied serial bit clock. The maximum value of the external serial bit clock that the module can operate with is internal bus clock frequency (clks) / 32.

If any of the FBS, CLKP, IDIV or CLKC bit values are changed, then the HSPI will undergo the HSPI software reset.

18.2.2 Status Register (SPSR)

SPSR is a 32-bit readable/writable register. The status flag in SPSR can confirm whether the system correctly operates or not. If the ROIE bit in SPSCR is set to 1, an interrupt request is generated due to the occurrence of the reception buffer overrun error or the warning of the receive buffer overrun error. When the TFIE bit in SPSCR is set to 1, an interrupt request is generated by the transmit completion status flag. If the appropriate enable bit in SPSCR is set to 1, an interrupt request is generated due to the receive FIFO halfway, receive FIFO full, transmit FIFO empty, or transmit FIFO halfway flag. If the RNIE bit in SPSCR is set to 1, an interrupt request is generated when the receive FIFO is not empty.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	TXFU	TXHA	TXEM	RXFU	RXHA	RXEM	RXOO	RXOW	RXFL	TXFN	TXFL
Initial value:	—	—	—	—	—	0	0	1	0	0	1	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/WC0	R/WC0	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	—	R	Reserved These bits are always read as undefined value. The write value should always be 0.
10	TXFU	0	R	Transmit FIFO Full Flag This status flag is enabled only to operation in FIFO mode. The flag is set to 1 when the transmit FIFO is full of bytes for transmission and cannot accept any more. It is cleared to 0 when data is transmitted from the transmit FIFO.
9	TXHA	0	R	Transmit FIFO Halfway Flag This status flag is enabled only to operation in FIFO mode. The flag is set to 1 when the transmit FIFO reaches the halfway point, that is, it has 4 bytes of data and 4 spaces for more data. It is cleared to 0 when more data is written to the transmit FIFO. It remains set to 1 until cleared to 0 even if the subsequent FIFO level becomes under the halfway point (4 bytes). If TXHA = 1 and THIE = 1 then irq = 1

Bit	Bit Name	Initial Value	R/W	Description
8	TXEM	1	R	Transmit FIFO Empty Flag This status flag is enabled only to operation in FIFO mode. The flag is set to 1 when the transmit FIFO is empty of data to transmit. It is cleared to 0 when more data is written to the transmit FIFO. If TXEM = 1 and TEIE = 1 then irq = 1
7	RXFU	0	R	Receive FIFO Full Flag This status flag is enabled only to operation in FIFO mode. The flag is set to 1 when the receive FIFO is full of received bytes and cannot accept any more. It is cleared to 0 when data is read out of the receive FIFO. If RXFU = 1 and RFIE = 1 then irq = 1
6	RXHA	0	R	Receive FIFO Halfway Flag This status flag is enabled only to operation in FIFO mode. The flag is set to 1 when the receive FIFO reaches the halfway point, that is, it has 4 bytes of data and 4 spaces for more data. This flag is cleared to 0 when the receive data is read from receive FIFO and the FIFO level becomes under 4 bytes (halfway point). If RXHA = 1 and RHIE = 1 then irq = 1
5	RXEM	1	R	Receive FIFO Empty Flag This status flag is enabled only to operation in FIFO mode. The flag is set to 1 when the receive FIFO is empty of received data. It is cleared to 0 when more data is received into to the receive FIFO. If RXEM = 0 and RNIE = 1 then irq = 1
4	RXOO	0	R/WC0	Receive Buffer Overrun Occurred Flag This status flag is set to 1 when new data has been received but the previous received data has not been read from SPRBR. The previously received data will not be overwritten by the newly received data. The RXOO flag remain set to 1 until writing 0 to its bit position. If RXOO = 1 and ROIE = 1 then irq = 1
3	RXOW	0	R/WC0	Receive Buffer Overrun Warning Flag This status flag is set to 1 when a new serial data transfer starts and the previous received data has not been read from SPRBR. The RXOW remain set to 1 until writing a 0 to its bit position. If RXOW= 1 and ROIE = 1 then irq = 1

Bit	Bit Name	Initial Value	R/W	Description
2	RXFL	0	R	Receive Buffer Full Status Flag This status flag indicates that new data is available in the SPRBR and has not yet been read. It is set to 1 at the completion of a serial bus transfer at the point the shift register contents are loaded into the SPRBR. This bit is cleared to 0 by reading SPRBR.
1	TXFN	0	R	Transmit Complete Status Flag This status flag indicates that the last transmission has completed. It is set to 1 when SPTBR is able to write more data from the internal bus. This bit is cleared to 0 by writing more data SPTBR. If TXFN = 1 and TFIE = 1 then irq = 1
0	TXFL	0	R	Transmit Buffer Full Status Flag This status flag indicates SPTBR has transmitted data. It is set to 1 when SPTBR is written with data from the internal bus. This bit is cleared to 0 when SPTBR is able to accept more data from the internal bus.

18.2.3 System Control Register (SPSCR)

SPSCR is a 32-bit readable/writable register that enables or disables interrupts or FIFO mode, selects either LSB first or MSB first in transmitting/receiving data, and generates a software reset.

If any of the FFEN, LMSB, CSA or MASL bit values are changed, then the module will undergo the HSPI software reset.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	TEIE	THIE	RNIE	RHIE	RFIE	FFEN	LMSB	CSV	CSA	TFIE	ROIE	RXDE	TXDE	MASL
Initial value:	—	—	0	0	0	0	0	0	0	1	0	0	0	0	0	0
R/W:	R	R	W	W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved These bits are always read as an undefined value. The write value should always be 0.
13	TEIE	0	W	Transmit FIFO Empty Interrupt Enable 0: Transmit FIFO empty interrupt disabled 1: Transmit FIFO empty interrupt enabled TEIE is a write-only bit that always read as undefined.
12	THIE	0	W	Transmit FIFO Halfway Interrupt Enable 0: Transmit FIFO halfway interrupt disabled 1: Transmit FIFO halfway interrupt enabled THIE is a write-only bit that always read as undefined.
11	RNIE	0	R/W	Receive FIFO Not Empty Interrupt Enable 0: Receive FIFO not empty interrupt disabled 1: Receive FIFO not empty interrupt enabled
10	RHIE	0	R/W	Receive FIFO Halfway Interrupt Enable 0: Receive FIFO halfway interrupt disabled 1: Receive FIFO halfway interrupt enabled

Bit	Bit Name	Initial Value	R/W	Description
9	RFIE	0	R/W	Receive FIFO Full Interrupt Enable 0: Receive FIFO full interrupt disabled 1: Receive FIFO full interrupt enabled
8	FFEN	0	R/W	FIFO Mode Enable Enables or disables the FIFO mode. When FIFO mode is enabled two 8-entry FIFOs are made available, one for transmit data and one for receive data. These FIFOs are read and written via SPTBR and SPRBR. When FIFO mode is disabled the SPTBR and SPRBR are used directly so new data must be written to SPTBR and read from SPRBR for each and every transfer. The FIFO mode must be disabled if DMA requests are used for SPTBR and SPRBR. 0: FIFO mode disabled 1: FIFO mode enabled
7	LMSB	0	R/W	LSB/MSB First Control 0: Data is transmitted and received most significant bit (MSB) first. 1: Data is transmitted and received least significant bit (LSB) first.
6	CSV	1	R/W	Chip Select Value Controls the value output from the chip select when the HSPI is a master and the chip select generation has been selected manually. 0: Chip select output is low. 1: Chip select output is high.
5	CSA	0	R/W	Automatic/Manual Chip Select 0: Chip select output is automatically generated during data transfer. 1: Chip select output is manually controlled, with its value being determined by the CSV bit.
4	TFIE	0	R/W	Transmit Complete Interrupt Enable 0: Transmit complete interrupt disabled 1: Transmit complete interrupt enabled

Bit	Bit Name	Initial Value	R/W	Description
3	ROIE	0	R/W	Receive Overrun Occurred / Warning Interrupt Enable 0: Receive overrun occurred / warning interrupt disabled 1: Receive overrun occurred / warning interrupt enabled
2	RXDE	0	R/W	Receive DMA Enable 0: Receive DMA transfer request disabled 1: Receive DMA transfer enabled
1	TXDE	0	R/W	Transmit DMA Enable 0: Transmit DMA transfer disabled 1: Transmit DMA transfer enabled
0	MASL	0	R/W	Master/Slave Select Bit 0: HSPI module configured as a slave 1: HSPI module configured as a master

18.2.4 Transmit Buffer Register (SPTBR)

SPTBR is a 32-bit readable/writable register that stores data to be transmitted.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TD							
Initial value:	—	—	—	—	—	—	—	—	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	—	R	Reserved These bits are always read as an undefined value. The write value should always be 0.
7 to 0	TD	H'00	R/W	Transmit Data Data written to this register is transferred to the shift register for transmission. When reading these bits, always read as data stored in the transmit buffer.

18.2.5 Receive Buffer Register (SPRBR)

SPRBR is a 32-bit read-only register that stores the number of received data.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RD							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as an undefined value. The write value should always be 0.
7 to 0	RD	H'00	R	Receive Data Data from the shift register, which is stored as each byte, is received, if the previously received data has been read.

18.3 Operation

18.3.1 Operation without DMA (FIFO Mode Disabled)

Figure 18.2 shows the flow of a transmit/receive operation procedure.

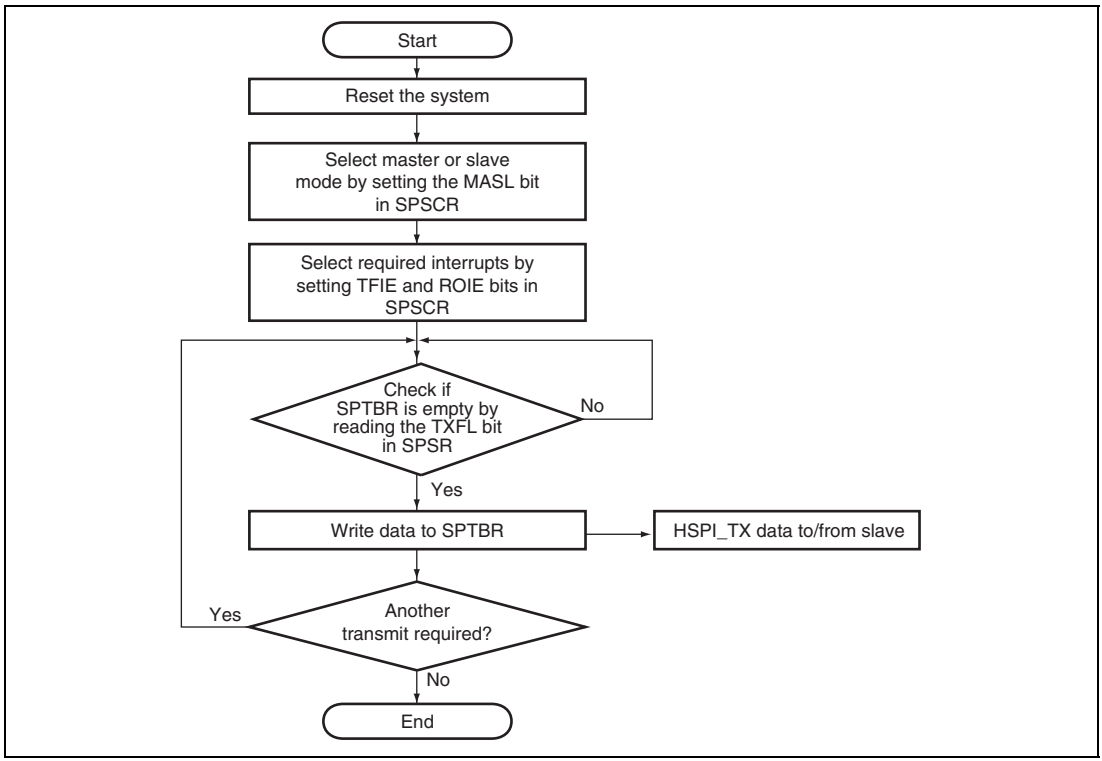


Figure 18.2 Operational Flowchart

Depending on the settings of SPCR, the master transmits data to the slave on either the falling or rising edge of HSPI_CLK and samples data from the slave on the opposite edge. The data transfer between the master and slave completes when the transmit completion status flag (TXFN) in SPSR is set to 1. This flag should be used to identify when an HSPI transfer event (byte transmitted and byte received) has occurred, even in the case where the HSPI module is being used to receive data only (null data being transmitted). By default data is transmitted MSB first, but LSB first is also possible depending on how the LMSB bit in SPSCR is set.

During the transmit function the slave responds by sending data to the master synchronized with the HSPI_CLK from the master transmitted. Data from the slave is sampled and transferred to the shift register in the module and on completion of the transmit function, is transferred to SPRBR.

The $\overline{\text{HSPI\_CS}}$ pin is used to select the HSPI module when the HSPI is configured as a slave, and prepare it to receive data from an external master. When the FBS bit in SPCR is 0, the $\overline{\text{HSPI\_CS}}$ pin must be driven high between successive bytes. When the FBS = 1, the $\overline{\text{HSPI\_CS}}$ pin can stay low for several byte transmissions. In this case, if the system is configured such that the FBS is always 1, then the $\overline{\text{HSPI\_CS}}$ line can be fixed at ground (if the HSPI will only be used as a slave).

18.3.2 Operation Using DMA

Operation of the HSPI is simpler with data transfer by DMA than with non-DMA transfer. HSPI settings are the same as for non-DMA transfer. The FIFO mode should be disabled. After that, set up the DMA controller (DMAC) to transfer the required data. DMA requests by the HSPI are enabled, so transfer is executed without further processing.

After the DMAC indicates the completion of transfer, remaining DMA requests must be cleared by disabling the DMA request signal from the HSPI. If this is not done, the HSPI will continue to request data for transmission.

18.3.3 Operation with FIFO Mode Enabled

In order to reduce the interrupt overhead on the processor for the operation without using DMA mode, FIFO mode has been provided. When FIFO mode is enabled, up to 8 bytes can be written in advance for transmission and up to 8 bytes can be received before the receive FIFO needs to be read. To transfer the specified amount of data between the HSPI module and an external device, follow the following procedure:

1. Set up the module for the required HSPI transfer characteristics (master/slave, clock polarity etc.) and enable FIFO mode.
2. Write bytes into the transmit FIFO via SPTBR. If more than 8 bytes are to be transmitted then enable the transmit FIFO halfway interrupt to keep track of the FIFO level as data is transmitted.
3. Respond to the transmit FIFO halfway interrupt when it occurs by writing more data to the transmit FIFO and reading data from the receive FIFO via SPRBR.
4. When all of the transmit data has been written into the transmit FIFO, disable the transmit FIFO halfway interrupt and read the contents of the receive FIFO until it is empty. Enable the receive FIFO not empty interrupt to keep track of when the final bytes of the transfer are received.
5. Respond to the receive FIFO not empty interrupt until all the expected data has been received.
6. Disable the module until it is required again.

In some applications, an undefined amount of data will be received from an external HSPI device. If this is the case, follow the following procedure:

1. Set up the module for the required HSPI transfer characteristics (master/slave, clock polarity etc.) and enable FIFO mode.
2. Fill the transmit FIFO with the data to transmit. Enable the receive FIFO not empty interrupt.
3. Respond to the receive FIFO not empty interrupt and read data from the receive FIFO until it is empty. Write more data to the transmit FIFO if required.
4. Disable the module when the transfer is to stop.

18.3.4 Timing Diagrams

The following diagrams explain the timing relationship of all shift and sample processes in the HSPI. Figure 18.3 shows the conditions when FBS = 0, while figure 18.4 shows the conditions when FBS = 1. It can be seen that if CLKP in SPCR is 0 then transmit data is shifted on the falling edge of HSPI_CLK and receive data is sampled on the rising edge. The opposite is true when CLKP = 1.

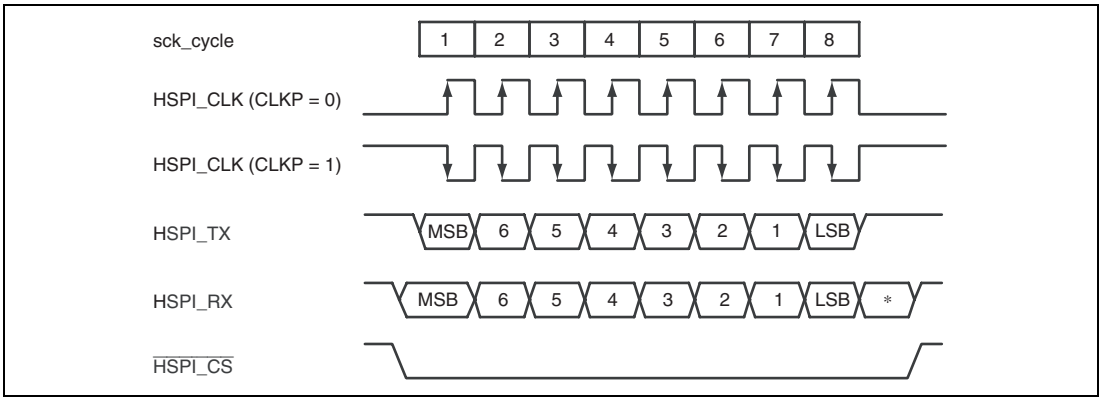


Figure 18.3 Timing Conditions when FBS = 0

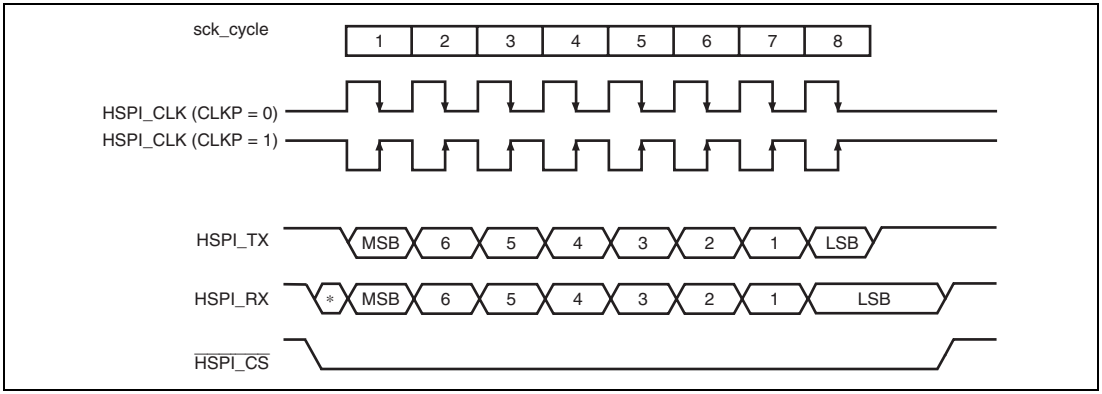


Figure 18.4 Timing Conditions when FBS = 1

18.3.5 HSPI Software Reset

The module is restored to the specified state, and the receive and transmit FIFO pointers can be initialized by the HSPI software reset. The HSPI software reset is generated when the control bits except SPCR, the interrupt/DMA enable bits and the chip select value bit (CSV) of SPSCR are modified.

Reset CSV and CSA after inserting a software reset when the HSPI drives $\overline{\text{HSPI\_CS}}$ to low level in slave mode but not during data being transferred by the master device. This setting prevents receiving the data wrongly.

18.3.6 Clock Polarity and Transmit Control

SPCR also allows the user to define the shift timing for transmit data and polarity. The FBS bit in SPCR allows selection between two different transfer formats. The MSB or LSB is valid on the falling edge of $\overline{\text{HSPI\_CS}}$. The CLKP bit in SPCR allows for control of the polarity select block, which controls which edges of HSPI_CLK shift and sample data in the master and slave.

18.3.7 Transmit and Receive Routines

The master and slave can be considered linked together as a circular shift register synchronized with HSPI_CLK. The transmit byte from the master is replaced with the receive byte from the slave in eight HSPI_CLK cycles. Both the transmit and receive functions are double buffered to allow for continuous reads and writes. When FIFO mode is enabled, eight entry FIFOs are available for both transmit and receive data.

Section 19 Communication Interface with FIFO (SCIF)

19.1 Overview

This LSI has a serial communication interface with built-in 8-channel FIFO buffers (Serial Communication Interface with FIFO: SCIF) that handles asynchronous communication and clock synchronous serial communication. The SCIF has two 16-stage FIFO buffers separately for transmission and reception, which enables fast, efficient, and uninterrupted communication. The modem control functions ($\overline{\text{RTS}}$, $\overline{\text{CTS}}$) are enabled at the channel 1 and 0. The following table lists the functions of the SCIF for each channel.

Channel	Name	Function	Pin	Base Address	Remarks
0	SCIF-0	Asynchronous mode (modem control is enabled)	RX0, TX0, SCK0, $\overline{\text{RTS0}}$, and $\overline{\text{CTS0}}$	H'FFE40000	Transmission/reception clock can be supplied externally from SCK0 pin.
1	SCIF-1	Clock synchronous mode	RX1, TX1, SCK1, $\overline{\text{RTS1}}$, and $\overline{\text{CTS1}}$	H'FFE41000	Transmission/reception clock can be supplied externally from SCK1 pin.
2	SCIF-2	Asynchronous mode Clock synchronous mode	RX2, TX2 and SCK2	H'FFE42000	Transmission/reception clock can be supplied externally from SCK2 pin.
3	SCIF-3	Asynchronous mode	RX3 and TX3	H'FFE43000	—
4	SCIF-4	Asynchronous mode	RX4 and TX4	H'FFE44000	—
5	SCIF-5	Asynchronous mode	RX5 and TX5	H'FFE45000	—

19.1.1 Features

The SCIF has the following features.

- Asynchronous serial communication mode

  The SCIF performs serial data communication based on a character-by-character asynchronous system. This feature enables serial data communication with standard asynchronous communication chips that support Universal Asynchronous Receiver/Transmitter (UART) or Asynchronous Communication Interface Adapter (ACIA). There is a choice of eight serial data transfer formats.

  — Data length: 7 or 8 bits

  — Stop bit length: 1 or 2 bits

  — Parity: Even/odd/none

  — Receive error detection: Parity, framing, and overrun errors

  — Break detection:

  A break is detected when a framing error lasts for more than 1 frame length at Space 0 (low level).

  When a framing error occurs, a break can also be detected by reading the RX pin level directly from the serial port register (SCSPTR).

- Clock synchronous serial communication mode

  The SCIF performs serial data communication synchronized with a clock. This feature enables serial data communication with other LSIs that support synchronous communication. There is a single serial data communication format for clock synchronous serial communication.

  — Data length: 8 bits

  — Receive error detection: Overrun errors

- Full-duplex communication capability

  The SCIF has an independent transmitter and receiver that enable simultaneous transmission and reception. The transmitter and receiver both have a 16-stage FIFO buffer structure, enabling continuous serial data transmission and reception.

- On-chip baud rate generator, enabling any bit rate to be selected

  The SCIF enables choice of a clock source for transmission/reception: a clock from the on-chip baud rate generator based on the internal clock or an external clock.

- Eight interrupt sources

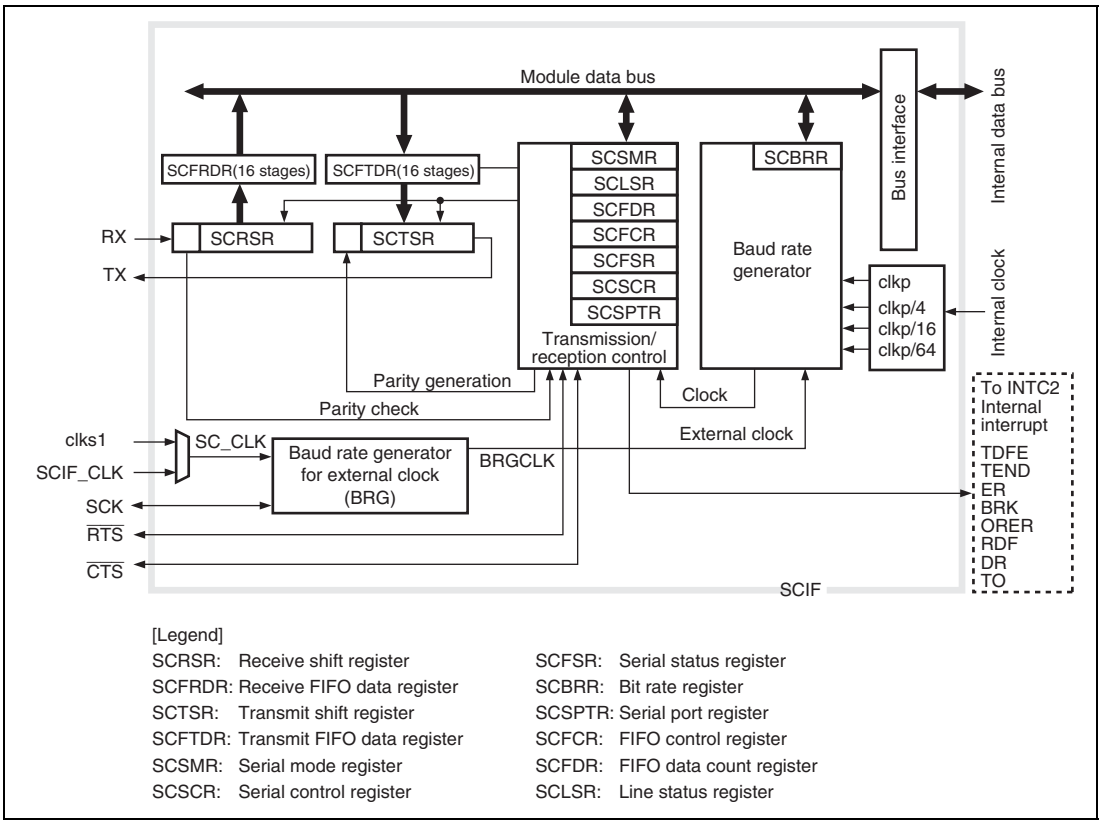
  The SCIF has eight types of interrupt sources – receive-data-ready, receive-FIFO-data-full, break, transmit-FIFO-data-empty, transmit-end, receive-error, overrun-error and time-out and enables any of them to be requested independently.

- DMA data transfer

  When the transmit FIFO register is empty or the receive FIFO register has received data, issuing a DMA transfer request activates the DMA controller (DMAC) to execute a data transfer.
- In asynchronous mode using channels 0 and 1, modem control functions ($\overline{\text{RTS}}$ and $\overline{\text{CTS}}$) are stored.
- The amount of data in the transmit/receive FIFO registers and the number of receive errors in receive data in the receive FIFO register are available.
- In asynchronous mode, a receive data ready (DR) or a timeout error (TO) can be detected during reception.

19.1.2 Block Diagram

Figure 19.1 shows the SCIF block diagram.



[Legend]

SCRSR:	Receive shift register	SCFSR:	Serial status register
SCFRDR:	Receive FIFO data register	SCBRR:	Bit rate register
SCTSR:	Transmit shift register	SCSPTR:	Serial port register
SCFTDR:	Transmit FIFO data register	SCFCR:	FIFO control register
SCSMR:	Serial mode register	SCFDR:	FIFO data count register
SCSCR:	Serial control register	SCLSR:	Line status register

Figure 19.1 SCIF Block Diagram

19.1.3 Pin Configuration

Table 19.1 shows the SCIF pin configuration.

Pin functions can differ with the interface number. Other functions are also multiplexed on the same pins, so the multiplexed pin settings may restrict usage of the pins.

Table 19.1 Pin Configuration

Function	Name	I/O	Descriptions
Serial clock pin	SCK	I/O	Synchronous clock I/O
Receive data pin	RX	Input	Receive data input
Transmit data pin	TX	Output	Transmit data output
Modem control pin	$\overline{\text{CTS}}$	I/O	Transmission enabled
Modem control pin	$\overline{\text{RTS}}$	I/O	Transmission request
Baud rate generation clock pin	SCIF_CLK	Input	Clock for input to the baud rate generator for the external clock

Note: These pins are made to function as serial pins by setting up SCIF operation using bit C/$\overline{\text{A}}$ in SCSMR, bits TE, RE, CKE1, and CKE0 in SCSCR, and bit MCE in SCFCR.
SCSPTR of the SCIF can be used to handle the transmission and detection of break states.

19.1.4 Register Configuration

Table 19.2 shows the registers in the SCIF.

Table 19.2 Register Configuration

Register Name	Abbreviation	R/W	Offset Address from Base Address	Access Size
Serial mode register	SCSMR	R/W	H'00	16
Bit rate register	SCBRR	R/W	H'04	8
Serial control register	SCSCR	R/W	H'08	16
Transmit FIFO data register	SCFTDR	W	H'0C	8
Serial status register	SCFSR	R/W*1	H'10	16
Receive FIFO data register	SCFRDR	R	H'14	8
FIFO control register	SCFCR	R/W	H'18	16
FIFO data count register	SCFDR	R	H'1C	16
Serial port register	SCSPTR	R/W	H'20	16
Line status register	SCLSR	R/W*2	H'24	16

Notes: 1. Only 0 can be written to bits 7 to 4, 1, and 0 to clear the flags. Bits 15 to 8, 3, and 2 are read-only bits and cannot be modified.

2. Only 0 can be written to bits 2 and 1 to clear the flags. Bits 15 to 3, and 1 are read-only bits and cannot be modified.

The table below lists the registers used in asynchronous mode, asynchronous mode with modem control, and clock synchronous mode. When setting up the registers, set the bits related to unsupported modes in the SCIF channels to their initial values. Otherwise, the SCIF may malfunction. Do not write to any registers other than listed below. Operations cannot be guaranteed if writing is attempted. Values read from registers other than listed below are undefined.

Table 19.3 Register Settings in Each Mode

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SCFRDR	—	—	—	—	—	—	—	—	*	*	*	*	*	*	*	*
SCFTDR	—	—	—	—	—	—	—	—	*	*	*	*	*	*	*	*
SCSMR	0	0	0	0	0	0	0	0	C	A	A	A	A	0	*	*
SCSCR	0	0	0	0	*	0	0	0	*	*	*	*	*	A	*	*
SCFSR	A	A	A	A	A	A	A	A	A	*	*	A	A	A	*	A
SCBRR	—	—	—	—	—	—	—	—	*	*	*	*	*	*	*	*
SCFCR	0	0	0	0	0	M	M	M	*	*	*	*	M	*	*	M
SCFDR	0	0	0	*	*	*	*	*	0	0	0	*	*	*	*	*
SCSPTR	0	0	0	0	0	0	0	0	M	M	M	M	C	C	*	*
SCLSR	0	0	0	0	0	0	0	0	0	0	0	0	0	A	0	*
DL	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
CKS	*	*	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[Legend]

* : Used in any mode.

A : Used in asynchronous mode.

M : Used in asynchronous mode (modem control is enabled) (in addition to A).

C : Used in clock synchronous mode.

0 : Reserved (the write value should always be 0.)

— : Undefined

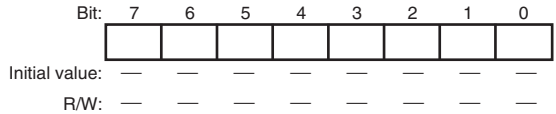
19.2 Register Descriptions

19.2.1 Receive Shift Resister (SCRSR)

SCRSR is a register that receives serial data.

The SCIF sets serial data input to the SCRSR from the RX pin in the order received, starting with the LSB (bit 0), and converts it to parallel data. When one byte of data has been received, it is transferred to the receive FIFO register SCFRDR, automatically.

SCRSR cannot be read from and written to by the CPU.



Bit:	7	6	5	4	3	2	1	0
Initial value:	—	—	—	—	—	—	—	—
R/W:	—	—	—	—	—	—	—	—

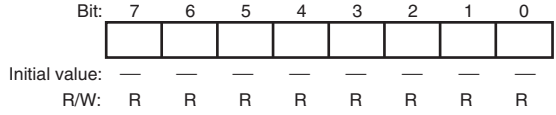
19.2.2 Receive FIFO Data Register (SCFRDR)

SCFRDR is a 16-stage FIFO register that stores received serial data.

When the SCIF has received one byte of serial data, it transfers the received data from the receive shift resister (SCRSR) to SCFRDR for storage, and reception is thus completed. SCRSR is then ready for reception, and is capable of receiving up to 16 consecutive bytes of data before SCFRDR is full.

SCFRDR is a read-only register and cannot be modified by the CPU. Note that the read value will be undefined while there is no receive data in SCFRDR. When SCFRDR is full of receive data, subsequent serial data is lost.

SCFRDR is read as an undefined value after a power-on reset or manual reset and in deep standby mode.



Bit:	7	6	5	4	3	2	1	0
Initial value:	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R

19.2.3 Transmit Shift Register (SCTSR)

SCTSR is a register that transmits serial data.

To perform serial data transmission, the SCIF first transfers transmit data from the transmit FIFO data register (SCFTDR) to SCTSR, then sends the data to the TX pin starting with the LSB (bit 0). When transmission of one byte is completed, the SCIF transfers the next transmit data from SCFTDR to SCTSR automatically, then starts transmission.

SCTSR cannot be read from and written to directly by the CPU.

Bit:	7	6	5	4	3	2	1	0
Initial value:	—	—	—	—	—	—	—	—
R/W:	—	—	—	—	—	—	—	—

19.2.4 Transmit FIFO Data Register (SCFTDR)

SCFTDR is an 8-bit FIFO register of 16 stages that stores data for serial transmission.

If SCTSR is empty after transmit data has been written to SCFTDR, the SCIF transfers the data from SCFTDR to SCTSR and starts serial transmission.

SCFTDR is a write-only register and cannot be read from by the CPU. Writing further data to SCFTDR is no longer possible when it is full (contains 16 bytes of transmit data). Attempts at writing data to the register in this situation are ignored.

SCFTDR is read as an undefined value on a power-on reset or manual reset and in deep standby mode.

Bit:	7	6	5	4	3	2	1	0
Initial value:	—	—	—	—	—	—	—	—
R/W:	W	W	W	W	W	W	W	W

19.2.5 Serial Mode Register (SCSMR)

SCSMR is a 16-bit register that sets the SCIF's serial transfer format and selects the baud rate generator clock source.

SCSMR can always be read from and written to by the CPU.

SCSMR is initialized to H'0000 by a power-on reset or a manual reset and in deep standby mode.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	C/$\overline{A}$	CHR	PE	O/$\overline{E}$	STOP	—	CKS1	CKS0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	C/$\overline{A}$	0	R/W	Communication Mode Selects asynchronous mode or clock synchronous mode for the SCIF operation. 0: Asynchronous mode 1: Clock synchronous mode Note: Whether clock synchronous mode can be selected or not depends on the channel.
6	CHR	0	R/W	Character Length Selects 7 or 8 bits for asynchronous mode data length. In clock synchronous mode, the data length is fixed at 8 bits regardless of the CHR bit setting. 0: 8 bits 1: 7 bits* Note: * When the 7-bit data is selected, the MSB (bit 7) in the transmit FIFO data register (SCFTDR) is not transmitted.

Bit	Bit Name	Initial Value	R/W	Description
5	PE	0	R/W	Parity Enable Determines whether parity bit is added in transmission or not, and parity bit is checked in reception in asynchronous mode or not. In clock synchronous mode, the parity bit is not added and checked regardless of the PE bit setting. When bit PE is set to 1, the parity (even or odd) specified by bit O/$\overline{E}$ is added to transmit data. In reception, the parity bit is checked for the parity (even or odd) specified by bit O/$\overline{E}$. 0: Disables parity bit addition and check. 1: Enables parity bit addition and check.
4	O/$\overline{E}$	0	R/W	Parity Mode Selects either even or odd parity to use in parity addition and check. In asynchronous mode, the O/$\overline{E}$ bit setting is valid only when bit PE is set to 1, enabling parity bit addition and check. In clock synchronous mode or when parity addition or check is disabled in asynchronous mode, the O/$\overline{E}$ bit setting is invalid. 0: Even parity*[1] 1: Odd parity*[2] Notes: 1. When even parity is set, the parity bit is added in transmission so that the total number of 1-bits in the transmit character plus the parity bit is even. In reception, a check is performed to see if the total number of 1-bits in the receive character plus the parity bit is even. 2. When odd parity is set, the parity bit is added in transmission so that the total number of 1-bits in the transmit character plus the parity bit is odd. In reception, a check is performed to see if the total number of 1-bits in the receive character plus the parity bit is odd.

Bit	Bit Name	Initial Value	R/W	Description
3	STOP	0	R/W	Stop Bit Length Selects 1 bit or 2 bits as the stop bit length in asynchronous mode. The stop bit setting is valid only in asynchronous mode. Since the stop bit is not added in clock synchronous mode, the STOP bit setting is invalid in this mode. In reception, only the first stop bit is checked, regardless of the STOP bit setting. If the second stop bit is 1, it is treated as a stop bit. If it is 0, it is treated as the start bit of the next transmit character. 0: 1 stop bit*1 1: 2 stop bits*2 Notes: 1. In transmission, a single 1-bit (stop bit) is added to the end of a transmit character before it is sent. 2. In transmission, two 1-bits (stop bits) are added to the end of a transmit character before it is sent.
2	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
1, 0	CKS[1:0]	00	R/W	Clock Select 1 and 0 These bits select the clock source for the on-chip baud rate generator. The clock source can be selected from clkp, clkp/4, clkp/16, and clkp/64, according to the setting of bits CKS1 and CKS0. 00: clkp 01: clkp/4 10: clkp/16 11: clkp/64 Note: clkp: Peripheral clock

19.2.6 Serial Control Register (SCSCR)

SCSCR is a register that enables or disables transmission/reception by the SCIF, enables or disables interrupt requests, and selects transmission/reception clock source for the SCIF.

SCSCR can always be read from and written to by the CPU.

SCSCR is initialized to H'0000 by a power-on reset or manual reset and in deep standby mode.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	TEIE	—	—	—	TIE	RIE	TE	RE	REIE	TOIE	CKE1	CKE0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11	TEIE	0	R/W	Transmit End Interrupt Enable When a transmit-end request is enabled by the TIE bit, the TEIE bit selects the source of the transmit end interrupt request from the following: • Setting the TDFE flag in SCFSR • Setting the TEND flag in SCFSR 0: The transmit FIFO data empty (TDFE) interrupt request is used. 1: The transmit end (TEND) interrupt request is used.
10 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
7	TIE	0	R/W	Transmit Interrupt Enable Enables or disables a transmit-FIFO-data-empty interrupt (TDFE) request when the TEIE bit in SCSCR is pulled 0, if the following conditions are satisfied: • Serial transmit data has been transferred from SCFTDR to SCTSR, • The number of data bytes in SCFTDR is equal to or less than the transmit trigger count, and • The TDFE flag in SCFSR is set to 1. Enables or disables a transmit-end interrupt (TEND) request when the TEIE bit of SCSCR is set to 1, if the following conditions are satisfied: • Transmission was ended because there is no valid data in SCFTDR when the last bit of the transmit character in SCTSR was transmitted, and • The TEND flag of SCFSR is set to 1. 0: When the TEIE bit is 0, disables transmit-FIFO-data-empty interrupt (TDFE) request.* When the TEIE bit is 1, disables transmit-end interrupt (TEND) request.* 1: When the TEIE bit is 0, enables transmit-FIFO-data-empty interrupt (TDFE) request. When the TEIE bit is 1, enables transmit-end interrupt (TEND) request.

Bit	Bit Name	Initial Value	R/W	Description
6	RIE	0	R/W	Receive Interrupt Enable Enables or disables a receive-FIFO-data-full interrupt request when the RDF flag in SCFSR is set to 1, a receive-data-ready interrupt request when the DR flag in SCFSR is set to 1, a receive-error interrupt request when the ER flag in SCFSR is set to 1, a break interrupt request when the BRK flag in SCFSR is set to 1, and an overrun error interrupt request when the ORER flag in SCLSR is set to 1. 0: Disables receive-FIFO-data-full interrupt (RDF) requests, receive-data-ready interrupt (DR) requests, receive-error interrupt (ER) requests, break interrupt (BRK) requests, and overrun error interrupt (ORER) requests. 1: Enables receive-FIFO-data-full interrupt (RDF) requests, receive-data-ready interrupt (DR) requests, receive-error interrupt (ER) requests, break interrupt (BRK) requests, and overrun error interrupt (ORER) requests.
5	TE	0	R/W	Transmit Enable Enables or disables the start of SCIF serial transmission. The SCIF starts serial transmission when transmit data is written to the SCFTDR while TE is 1. Before setting TE to 1, set SCSMR and SCFCR to specify the transmission format and reset the transmit FIFO. 0: Disables transmission. 1: Enables transmission.
4	RE	0	R/W	Receive Enable Enables or disables the start of SCIF serial reception. When RE is 1, the SCIF starts serial reception by detecting a start bit in asynchronous mode or by detecting a synchronization clock input in clock synchronous mode. Before setting RE to 1, set SCSMR and SCFCR to specify the reception format and reset the receive FIFO. 0: Disables reception.* 1: Enables reception. Note: * Even if RE is cleared to 0, the DR, ER, BRK, RDF, FER, PER, TO and ORER flags are not affected, and retain their states.

Bit	Bit Name	Initial Value	R/W	Description
3	REIE	0	R/W	Receive Error Interrupt Enable Enables or Disables generation of receive-error interrupt (ER) requests, break interrupt (BRK) requests and overrun-error interrupt (ORER) requests. 0: Disables receive-error interrupt (ER) requests, break interrupt (BRK) requests and overrun-error interrupt (ORER) requests.* 1: Enables receive-error interrupt (ER) requests, break interrupt (BRK) requests and overrun-error interrupt (ORER) requests. Note: * When REIE is 1, ER, BRK or ORER interrupt requests will occur even if RIE is cleared to 0. This setting is used to notify the interrupt controller of ER, BRK, and ORER interrupt requests during DMAC transfer.
2	TOIE	0	R/W	Timeout Interrupt Enable Enables or disables generation of timeout interrupt (TO) requests when the TO flag in SCLSR is set to 1. 0: Disables timeout interrupts (TO). 1: Enables timeout interrupts (TO).
1, 0	CKE[1:0]	00	R/W	Clock Enable 1 and 0 These bits select the SCIF clock source and enables or disables the clock output from the SCK pin. Whether the SCK pin functions as a serial clock output pin or a serial clock input pin is determined by combination of the CKE1 and CKE0 bit settings. To specify synchronization clock output in clock synchronous mode, set C/$\overline{A}$ to 1 in SCSMR then set CKE1 and CKE0. See table 19.4 for the bit settings.

Table 19.4 Clock Selection

CKE1	CKE0	Mode	Clock Source	SCK Pin
0	0	Asynchronous mode	Internal clock (clkp, clkp/4, clkp/16, clkp/64)	The SCK pin is not used. The SCK pin functions as an input pin (Input signals are ignored). (Initial value)
		Clock synchronous mode		The SCK pin outputs the synchronization clock. (Initial value)
0	1	Asynchronous mode		The SCK pin outputs the clock (with a frequency 16 times the bit rate).
		Clock synchronous mode		The SCK pin outputs the synchronization clock.
1	0	Asynchronous mode	Baud rate generator output for external clock or SCK	When SC_CLK is selected: The SCK pin is an input pin (Input signals are ignored). Set the SC_CLK frequency so that the frequency of BRGCLK is 16 times the bit rate.
				When SCK is selected: The SCK pin inputs the clock (with a frequency 16 times the bit rate).
		Clock synchronous mode	SCK*	The SCK pin inputs the synchronization clock.
1	1	Prohibited		

Note: * It is not allowed to set synchronous communication using SC_CLK for input.

19.2.7 Serial Status Register (SCFSR)

SCFSR is a 16-bit register. The lower 8 bits are a status flag that indicates the operating status of the SCIF, and the upper 8 bits indicate the number of reception errors of data in the receive FIFO register.

SCFSR can always be read from and written to by the CPU. However, the flags ER, TEND, TDFE, BRK, RDF, and DR cannot be written by 1. The FER and PER flags are read-only flags and cannot be modified.

SCFSR is initialized to H'0060 by a power-on reset or a manual reset and in deep standby mode.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PER3	PER2	PER1	PER0	FER3	FER2	FER1	FER0	ER	TEND	TDFE	BRK	FER	PER	RDF	DR
Initial value:	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W*	R/W*	R/W*	R/W*	R	R	R/W*	R/W*

Note: * Only 0 can be written to clear the flag.

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	PER[3:0]	0000	R	Parity Error Count These bits indicate the number of parity errors of receive data stored in the receive FIFO data register (SCFRDR). After the ER is set in SCFSR, the value represented by bits 15 to 12 indicates the number of parity errors. If a parity error occurs in all 16-byte receive data in SCFRDR, PER3 to PER0 are cleared to 0.
11 to 8	FER[3:0]	0000	R	Framing Error Count These bits indicate the number of framing errors of receive data stored in the receive FIFO data register (SCFRDR). After the ER bit is set in SCFSR, the value represented by bits 11 to 8 indicates the number of framing errors. If a framing error occurs in all 16-byte receive data in SCFRDR, FER3 to FER0 are cleared to 0.

Bit	Bit Name	Initial Value	R/W	Description
7	ER	0	R/W*	Receive Error Indicates that a framing error or a parity error has occurred in reception. The ER flag is not affected by an error and retains its previous state when the RE bit is 0 in SCSCR. If a receive error occurs, receive data will be transferred to SCFRDR and reception operation will be continued. Whether there is a receive error in data read from SCFRDR can be determined by the FER and PER bits in SCFSR. 0: Indicates that no framing or parity error has occurred in reception. [Clearing conditions] • Execution of a power-on reset or manual reset or time in deep standby mode. • 0 is written to ER. 1: Indicates that a framing error or a parity error has occurred in reception. [Setting conditions] • The SCIF checks whether the stop bit at the end of receive data is 1, but the stop bit is 0.* • The number of 1-bits in receive data plus the parity bit does not match the parity setting (even or odd) specified by the O/$\overline{E}$ bit in SCSMR during reception. Note: * In the 2-stop-bit mode, only the first stop bit is checked that the value is 1; the second stop bit is not checked.

Bit	Bit Name	Initial Value	R/W	Description
6	TEND	1	R/W*	Indicates that transmission has been ended * because there was no valid data in SCFTDR when the last bit of the transmit character was transmitted. 0: Indicates that transmission is in progress. [Clearing conditions] • Transmit data is written to SCFTDR, and 0 is written to TEND. • Data is written to SCFTDR by the DMAC. 1: Indicates that transmission has been ended. [Setting conditions] • Execution of a power-on reset or manual reset or time in deep standby mode. • The TE bit in SCSCR is 0. • There is no transmit data in SCFTDR when the last bit of a 1-byte serial transmission character is transmitted. Note: When transmit data is written to SCFTDR by the DMAC In clock synchronous mode, the TEND flag may not be cleared. Therefore, if the DMAC is used for transmission in clock synchronous mode, the TEND flag should be read using the following procedure. 1. Check that data transfer is completed in the DMAC. 2. Read the TEND flag. 3. If TEND = 1, clear the TEND flag to 0. 4. Read the TEND flag again. 5. Use the second read TEND flag.

Bit	Bit Name	Initial Value	R/W	Description
5	TDFE	1	R/W*	Transmit FIFO Data Empty Indicates that the SCIF has transferred data from SCFTDR to SCTSR, the number of data bytes in SCFTDR becomes equal to or less than the transmit trigger count specified by the TTRG1 and TTRG0 bits in SCFCR, and SCFTDR is ready to be written by new transmit data. SCFTDR is a 16-byte FIFO register. The maximum number of bytes that can be written to when TDFE = 1 is "16 – [the transmit trigger count]". If data exceeding this value is attempted to be written, the data will be ignored. The number of data bytes in SCFTDR is indicated by the upper bits of SCFDR. If the number of data written in SCFTDR is equal to or less than the transmit trigger count, this bit will be set to 1 even if it is cleared to 0 after it is read as 1. 0: Indicates that the number of transmit data written to SCFTDR exceeds the transmit trigger count. [Clearing conditions] • Transmit data exceeding the specified transmit trigger count have been written to SCFTDR, and 0 is written to TDFE. • Transmit data exceeding the specified transmit trigger count have been written to SCFTDR by the DMAC. 1: Indicates that the number of transmit data in SCFTDR is equal to or less than the transmit trigger count. [Setting conditions] • Execution of a power-on reset or manual reset or time in deep standby mode. • The number of transmit data in SCFTDR is equal to or less than the transmit trigger count after transmission.

Bit	Bit Name	Initial Value	R/W	Description
4	BRK	0	R/W*	Break Detect Indicates that a receive data break signal has been detected. If a break signal is detected, receive data (H'00) transfer to SCFRDR is stopped. After the break is canceled and the receive signal returns to 1, the receive data transfer resumes. 0: Indicates that no break signal has been received. [Clearing conditions] • Execution of a power-on reset or manual reset or time in deep standby mode. • 0 is written to BRK. 1: Indicates that a break signal has been received. [Setting condition] • Data with a framing error is received, followed by the space "0" (low level) for at least one frame length.
3	FER	0	R	Framing Error Indicates that a framing error has been found in the data that is to be read next from SCFRDR in asynchronous mode. 0: Indicates that there is no framing error in the receive data that is to be read from SCFRDR. [Clearing conditions] • Execution of a power-on reset or manual reset or time in deep standby mode. • There is no framing error in the data that is to be read next from SCFRDR. 1: Indicates that there is a framing error in the receive data that is to be read from SCFRDR. [Setting condition] • There is a framing error in the data that is to be read next from SCFRDR.

Bit	Bit Name	Initial Value	R/W	Description
2	PER	0	R	Parity Error This bit indicates that a parity error has been found in the data that is to be read next from SCFRDR in asynchronous mode. 0: Indicates that there is no parity error in the receive data that is to be read from SCFRDR. [Clearing conditions] • Execution of a power-on reset or manual reset or time in deep standby mode. • There is no parity error in the data that is to be read next from SCFRDR. 1: Indicates that there is a parity error in the receive data that is to be read from SCFRDR. [Setting condition] • There is a parity error in the data that is to be read next from SCFRDR.

Bit	Bit Name	Initial Value	R/W	Description
1	RDF	0	R/W*	Receive FIFO Data Full Indicates that the received data has been transferred from SCRSR to SCFRDR, and the number of receive data bytes in SCFRDR becomes equal to or more than the receive trigger count specified by the RTRG1 and RTRG0 bits in SCFCR. SCFRDR is a 16-byte FIFO register. When RDF = 1, data equal to or more than the number of receive trigger data bytes can be read. When SCFRDR is empty, SCFRDR is read as an undefined value. The number of receive data bytes in SCFRDR is indicated by the lower bits of SCFDR. If the number of data in SCFRDR is equal to or more than the trigger count, this bit will be set to 1 even if it is cleared to 0. At this time, read receive data until the number of data in SCFRDR is less than the trigger count, read RDF as 1, and then clear RDF. 0: Indicates that the number of receive data bytes in SCFRDR is less than the specified receive trigger count. [Clearing conditions] • Execution of a power-on reset or manual reset or time in deep standby mode. • SCFRDR is read until the number of receive data bytes in SCFRDR is less than the receive trigger count, and 0 is written to RDF. • SCFRDR is read by the DMAC until the number of receive data bytes in SCFRDR is less than the receive trigger count. 1: Indicates that the number of receive data bytes in SCFRDR is equal to or more than the specified receive trigger count. [Setting condition] • Receive data more than the receive trigger count have been stored in SDFRDR.

Bit	Bit Name	Initial Value	R/W	Description
0	DR	0	R/W*	Receive Data Ready Indicates that the number of data bytes in SCFRDR is less than the receive trigger count and that no further data has been received for at least 15 etu after the stop bit of the data received last in asynchronous mode. This bit is not set in clock synchronous mode. 0: Indicates that data is being received or has been successfully received, and there is no receive data in SCFRDR. [Clearing conditions] • Execution of a power-on reset or manual reset or time in deep standby mode. • All the receive data in SCFRDR has been read, and 0 is written to DR. • All the receive data in SCFRDR has been read by the DMAC. 1: Indicates that no further receive data has been received. [Setting condition] • The receive FIFO data register (SCFRDR) has the number of data byte that is below the receive trigger number of data bytes and no further data has arrived for at least 15 etu after the stop bit of the data received last.* Note: * etu: Elementary Time Unit (time for transfer of 1 bit) Equivalent to 1.5 frames with an 8-bit, 1-stop-bit format.

Note: * Writing 0 is only available to clear the flag.

19.2.8 Bit Rate Register (SCBRR)

SCBRR is an 8-bit register that sets the serial transmission/reception bit rate in accordance with the baud rate generator operating clock selected by the CKS1 and CKS0 bits in SCSMR. This baud rate generator is intended for clkp, clkp/4, clkp/16, and clkp/64. For details on the baud rate generator for external clock, see section 19.6, Baud Rate Generator for External Clock (BRG).

SCBRR can always be read from and written to by the CPU.

SCBRR is initialized to H'FF by a power-on reset or a manual reset and in deep standby mode.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—
Initial value:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

The SCBRR setting is determined by the following equation:

[Asynchronous mode]

$$N = \frac{P1\phi}{64 \times 2^{2n-1} \times B} \times 10^6 - 1$$

[Clock synchronous mode]

$$N = \frac{P1\phi}{8 \times 2^{2n-1} \times B} \times 10^6 - 1$$

B: Bit rate (bit/s)

N: SCBRR setting for the baud rate generator ($0 \leq N \leq 255$) (which satisfies the electrical characteristics)

P1ϕ: Peripheral module operating frequency (MHz)

n: 0, 1, 2, 3
(See table 19.5 for the relation between n and the baud rate generator input clock.)

Table 19.5 SCSMR Settings

n	Baud Rate Generator Input Clock	SCSMR Setting	
		CKS1	CKS0
0	clkp	0	0
1	clkp/4	0	1
2	clkp/16	1	0
3	clkp/64	1	1

The bit rate error in asynchronous mode is determined by the following equation:

$$\text{error (\%)} = \left\{ \frac{P1\phi \times 10^6}{(N+1) \times B \times 64 \times 2^{2n-1}} - 1 \right\} \times 100$$

19.2.9 FIFO Control Register (SCFCR)

SCFCR is a register that resets data counts and sets the number of trigger data bytes for the transmit and receive FIFO registers. It also has a loopback test enable bit.

SCFCR can always be read from and written to by the CPU.

SCFCR is initialized to H'0000 by a power-on reset or a manual reset and in deep standby mode.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	RSTRG2	RSTRG1	RSTRG0	RTRG1	RTRG0	TTRG1	TTRG0	MCE	TFRST	RFRST	LOOP
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
10 to 8	RSTRG [2:0]	000	R/W	RTS Output Active Trigger The $\overline{\text{RTS}}$ signal is high when the number of receive data bytes stored in SCFRDR is equal to or more than the specified trigger number shown below: 000: 15 001: 1 010: 4 011: 6 100: 8 101: 10 110: 12 111: 14
7, 6	RTRG[1:0]	00	R/W	Receive FIFO Data Count Trigger These bits specify the number of receive data bytes that makes the RDF (receive data full) flag to be set in SCFSR. The RDF flag is set when the number of receive data bytes in SCFRDR is equal to or more than the trigger count shown below:

RTRG1	RTRG0	Asynchronous Mode	Clock Synchronous Mode
0	0	1	1
0	1	4	2
1	0	8	8
1	1	14	14

Bit	Bit Name	Initial Value	R/W	Description
5, 4	TTRG[1:0]	00	R/W	Transmit FIFO Data Count Trigger These bits specify the number of remaining transmit data bytes that makes the transmit FIFO data register empty (TDFE) flag to be set in SCFSR. The TDFE flag is set when the number of transmit data bytes in SCFTDR is equal to or less than the specified trigger count shown below, after transmission: 00: 8 (8) 01: 4 (12) 10: 2 (14) 11: 0 (16) Note: Values in parentheses indicate the empty space in SCFTDR in bytes.
3	MCE	0	R/W	Modem Control Enable Enables or disables modem control signals $\overline{\text{CTS}}$ and $\overline{\text{RTS}}$. This bit should always be set to 0 in clock synchronous mode. Whether modem control can be selected or not depends on the channel. 0: Disables modem signals.* 1: Enables modem signals. Note: * $\overline{\text{CTS}}$ and $\overline{\text{RTS}}$ control ports.
2	TFRST	0	R/W	Transmit FIFO Data Register Reset Enables or disables a transmit FIFO data register reset that empties the register. 0: Disables the reset.* 1: Enables the reset. Note: * The register is reset by a power-on reset or a manual reset and in deep standby mode.
1	RFRST	0	R/W	Receive FIFO Data Register Reset Enables or disables a receive FIFO data register reset that empties the register. 0: Disables the reset.* 1: Enables the reset. Note: * The register is reset by a power-on reset or a manual reset and in deep standby mode.

Bit	Bit Name	Initial Value	R/W	Description
0	LOOP	0	R/W	Loopback Test Enables or disables the loopback test by internally connecting the transmit output pin (TX) and receive input pin (RX), and the $\overline{\text{RTS}}$ pin and $\overline{\text{CTS}}$ pin. 0: Disables the loopback test. 1: Enables the loopback test.

19.2.10 FIFO Data Count Register (SCFDR)

SCFDR is a 16-bit register that indicates the number of data bytes stored in SCFTDR and that in SCFRDR.

The upper 8 bits indicate the number of transmit data bytes in SCFTDR, and the lower 8 bits indicates the number of receive data bytes in SCFRDR.

SCFDR can always be read by the CPU.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	T[4:0]					—	—	—	R[4:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 8	T[4:0]	00000	R	These bits show the number of data bytes not transmitted and still stored in SCFTDR. H'00 indicates that there is no transmit data in SCFTDR, and H'10 indicates that SCFTDR is full of transmit data.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
4 to 0	R[4:0]	00000	R	These bits show the number of receive data stored in SCFRDR in bytes. H'00 indicates that there is no receive data in SCFRDR, and H'10 indicates that SCFRDR is full of receive data.

19.2.11 Serial Port Register (SCSPTR)

SCSPTR controls multiplexed input/output and data on the serial communication interface (SCIF) ports. Bits 1 and 0 control breaks in serial transmission/reception by reading input data from the RX pin and writing output data to the TX pin. Bits 3 and 2 read input data from and write output data to the SCK pin. Bits 5 and 4 read input data from and write output data to the $\overline{\text{CTS}}$ pin. Bits 7 and 6 read input data from and write output data to the $\overline{\text{RTS}}$ pin.

SCSPTR is a 16-bit register that can always be read from and written to by the CPU.

All SCSPTR bits except bits 6, 4, 2, and 0 are initialized to 0 by a power-on reset or a manual reset and in deep standby mode. The values of bits 6, 4, 2, and 0 are undefined.

Note: Whether modem control can be selected or not depends on the channel.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RTSIO	RTSDT	CTSIO	CTSDT	SCKIO	SCKDT	SPB2IO	SPB2DT
Initial value:	0	0	0	0	0	0	0	0	0	—	0	—	0	—	0	—
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	RTSIO	0	R/W	Serial Port - RTS Port Input/Output Specifies input or output for the serial port $\overline{\text{RTS}}$ pin. To actually set the $\overline{\text{RTS}}$ pin as a port output pin to output the value set by the RTSDT bit, the MCE bit in SCFCR should be cleared to 0. 0: Indicates that this bit does not output the value of the RTSDT bit to the $\overline{\text{RTS}}$ pin. 1: Indicates that this bit outputs the value of the RTSDT bit to the $\overline{\text{RTS}}$ pin.

Bit	Bit Name	Initial Value	R/W	Description
6	RTSDT	—	R/W	Serial Port - $\overline{\text{RTS}}$ Port Data Specifies the input/output data level of the serial port $\overline{\text{RTS}}$ pin. Whether the pin is set for input or output is determined by the RTSIO bit. When the pin is set for output, the value of the RTSDT bit is output to the $\overline{\text{RTS}}$ pin. Regardless of the value of the RTSIO bit, the value of the $\overline{\text{RTS}}$ pin is read from the RTSDT bit. The initial value of this bit is undefined after a power-on reset or a manual reset and in deep standby mode. 0: Indicates that the input/output data is low level. 1: Indicates that the input/output data is high level.
5	CTSIO	0	R/W	Serial Port - $\overline{\text{CTS}}$ Port Input/Output Specifies input or output for the serial port $\overline{\text{CTS}}$ pin. To actually set the $\overline{\text{CTS}}$ pin as a port output pin to output the value set by the CTSDT bit, the MCE bit in SCFCR should be cleared to 0. 0: Indicates that the CTSDT bit value is not output to the $\overline{\text{CTS}}$ pin. 1: Indicates that the CTSDT bit value is output to the $\overline{\text{CTS}}$ pin.
4	CTSDT	—	R/W	Serial Port - $\overline{\text{CTS}}$ Port Data Specifies the input/output data level of the serial port $\overline{\text{CTS}}$ pin. Whether the pin is set for input or output is determined by the CTSIO bit. When the pin is set for output, the value of the CTSDT bit is output to the $\overline{\text{CTS}}$ pin. Regardless of the value of the CTSIO bit, the value of the $\overline{\text{CTS}}$ pin is read from the CTSDT bit. The initial value of this bit is undefined after a power-on reset or a manual reset and in deep standby mode. 0: Indicates that the input/output data is low level. 1: Indicates that the input/output data is high level.

Bit	Bit Name	Initial Value	R/W	Description
3	SCKIO	0	R/W	Serial Port – Clock Port Input/Output Specifies input or output for the serial port SCK pin. To actually set the SCK pin as a port output pin to output the value set by the SCKDT bit, the CKE1 and CKE0 bits in SCSCR should be cleared to 0. 0: Indicates that the SCKDT bit value is not output to the SCK pin. 1: Indicates that the SCKDT bit value is output to the SCK pin.
2	SCKDT	—	R/W	Serial Port – Clock Port Data Specifies the input/output data level of the serial port SCK pin. Whether the pin is set for input or output is determined by the SCKIO bit. When the pin is set for output, the value of the SCKDT bit is output to the SCK pin. Regardless of the value of the SCKIO bit, the value of the SCK pin is read from the SCKDT bit. The initial value of this bit is undefined after a power-on reset or a manual reset and in deep standby mode. 0: Indicates that the input/output data is low level. 1: Indicates that the input/output data is high level.
1	SPB2IO	0	R/W	Serial Port – Break Input/Output Specifies the output condition of the serial port TX pin. To actually set the TX pin as a port output pin to output the value set by the SPB2DT bit, the TE bit in SCSCR should be cleared to 0. 0: Indicates that the SPB2DT bit value is not output to the TX pin. 1: Indicates that the SPB2DT bit value is output to the TX pin.

Bit	Bit Name	Initial Value	R/W	Description
0	SPB2DT	—	R/W	Serial Port – Break Data Specifies the input level of the serial port RX pin and the output level of the TX pin. The TX pin output conditions are determined by the SPB2IO bit. When the TX pin is set for output, the value of the SPB2DT bit is output to the TX pin. Regardless of the value of the SPB2IO bit, the value of the RX pin is read from the SPB2DT bit. The initial value of this bit is undefined after a power-on reset or a manual reset and in deep standby mode. 0: Indicates that the input/output data is low level. 1: Indicates that the input/output data is high level.

Figures 19.2 to 19.6 show the block diagrams of the SCIF I/O ports.

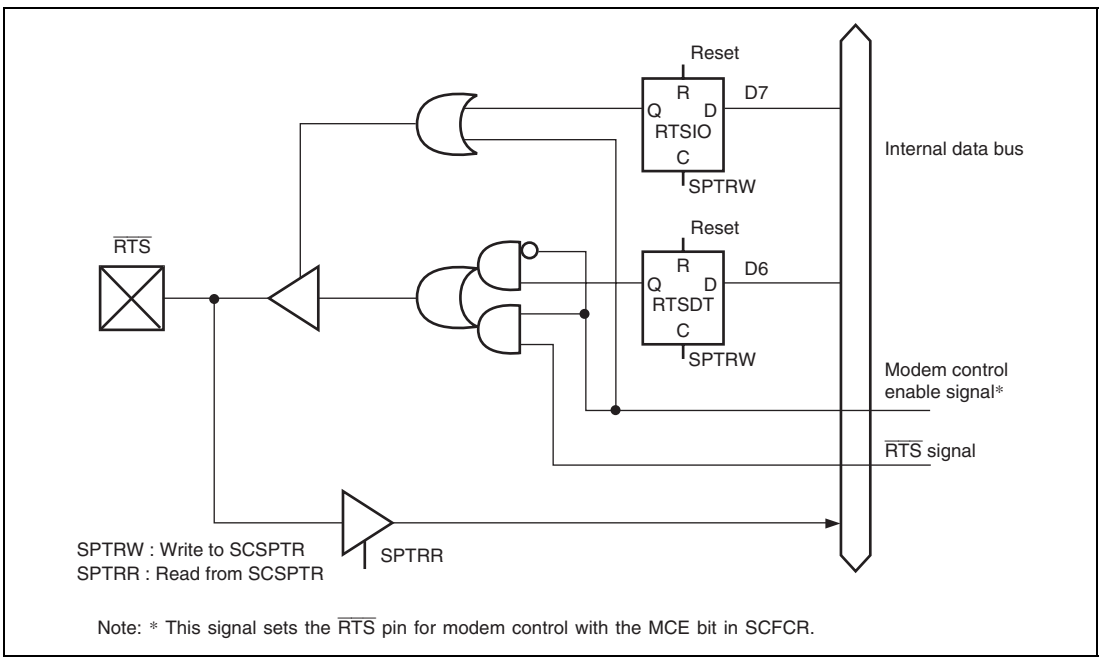


Figure 19.2 RTS Pin

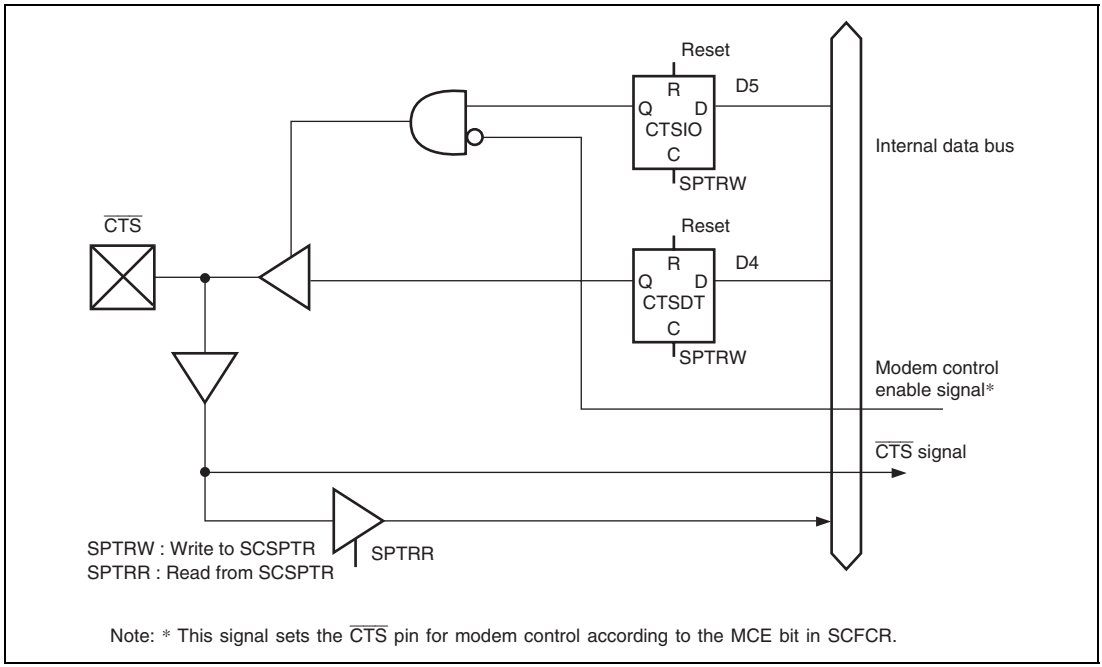
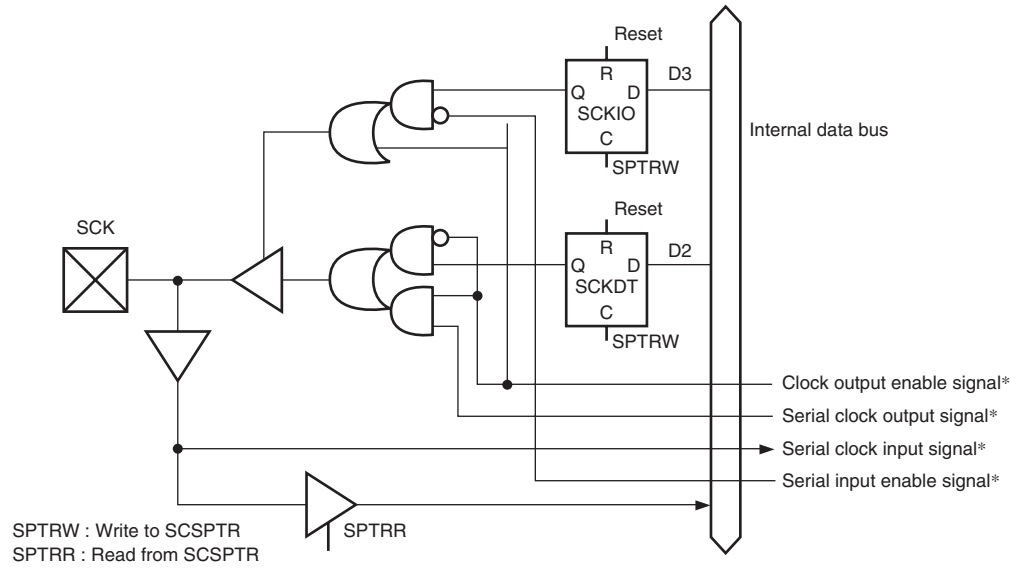


Figure 19.3 $\overline{\text{CTS}}$ Pin



Note: * These signals set the SCK pin for internal clock output or external clock input with the C/$\overline{A}$ bit in SCSMR and the CKE0 and CKE1 bits in SCSCR.

Figure 19.4 SCK Pin

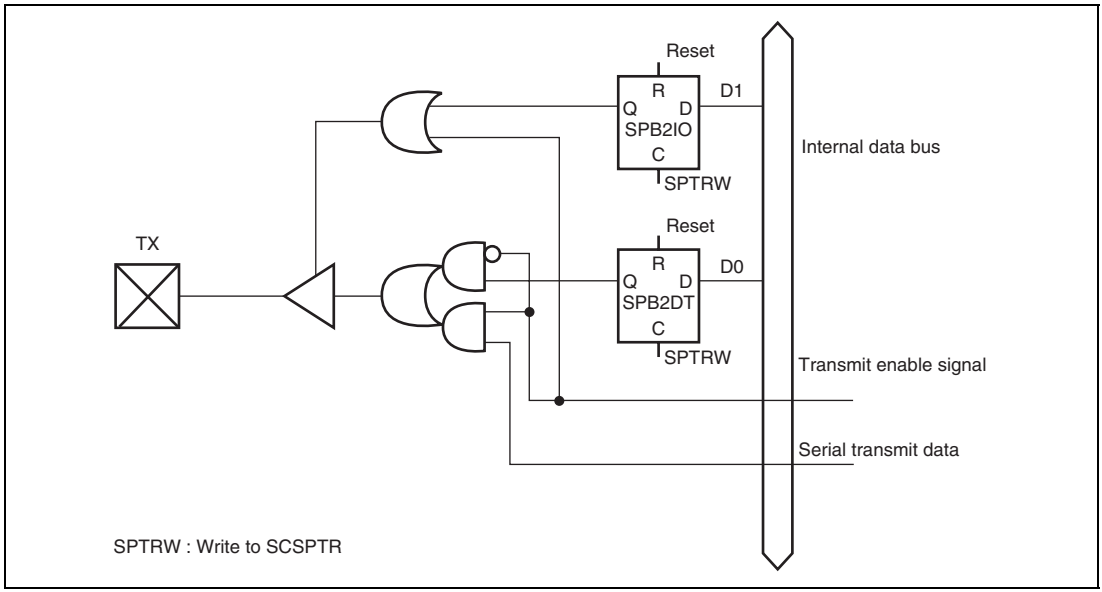


Figure 19.5 TX Pin

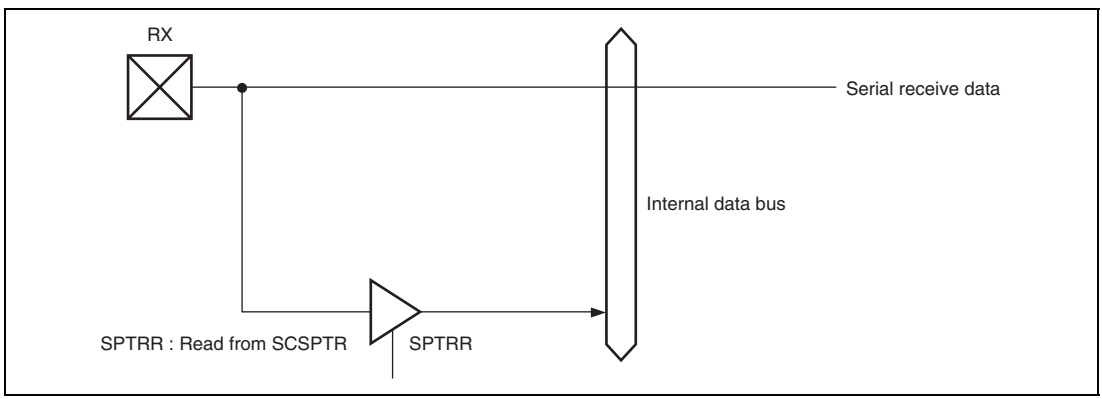


Figure 19.6 RX Pin

19.2.12 Line Status Register (SCLSR)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	TO	—	ORER
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W*	R	R/W*

Note: Only 0 can be written to clear the flag.

Bit	Bit Name	Initial Value	R/W	Description
15 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	TO	0	R/W*	Timeout Indicates that the number of data bytes in SCFRDR is less than the receive trigger count, and that no further data has been received for at least 15 etu after the stop bit of the last receive data in asynchronous mode. This bit is not set in clock synchronous mode. 0: Indicates that data is being received or has been successfully received and that there is no receive data in SCFRDR. [Clearing conditions] • Execution of a power-on reset or manual reset or time in deep standby mode. • All the receive data in the SCFRDR has been read, and 0 is written to TO. 1: Indicates that no further receive data has been received (receive timeout). [Setting condition] • The number of data bytes in SCFRDR is less than the receive trigger count and no further data has been received for at least 15 etu after the stop bit of the last receive data.* Note: * etu: Elementary Time Unit (time for transfer of 1 bit) Equivalent to 1.5 frames with an 8-bit, 1-stop-bit format.

Bit	Bit Name	Initial Value	R/W	Description
1	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
0	ORER	0	R/W*	Overrun Error Indicates that an overrun error has occurred in reception and abnormal termination is caused. If an overrun error occurs, the receive data prior to the overrun error is retained in SCFRDR and the data received subsequently is discarded. Any subsequent serial reception is disabled while the ORER flag is 1. To resume data reception after clearing the ORER flag, be sure to first read (or clear) data in the receive FIFO and handle the error, then clear the ORER flag. 0: Indicates that data is being received or has been successfully received. [Clearing conditions] • Execution of a power-on reset or manual reset or time in deep standby mode. • 0 is written to ORER. 1: Indicates that an overrun error has occurred in reception. [Setting condition] • The next serial reception has been completed while SCFRDR is full of 16-byte data. Note: when bit RE in SCSCR is cleared to 0, the ORER flag is not affected and its previous state is retained.

Note: Writing 0 is only available to clear the flag.

19.3 Operation

19.3.1 Operation in Asynchronous Mode

In asynchronous mode, the SCIF performs serial communication, in which data is transmitted/received in character units using the attached start bit indicating the start of communication and stop bit indicating the end of communication.

Figure 19.7 shows the general format for asynchronous serial communication.

In asynchronous serial communication, the transmission line is usually kept marked (high level). The SCIF monitors the transmission line, and when it finds a space (low level), it regards the space as a start bit and starts serial communication.

One character in serial communication consists of a start bit (low level), followed by transmit/receive data (LSB-first; from the lowest bit), a parity bit (high or low level), and a stop bit (high level).

During reception in asynchronous mode, the SCIF performs synchronization at the falling edge of the start bit. Communication data is acquired at the center of each bit because the SCIF samples data at the eighth pulse of the clock which has a frequency of 16 times the bit rate.

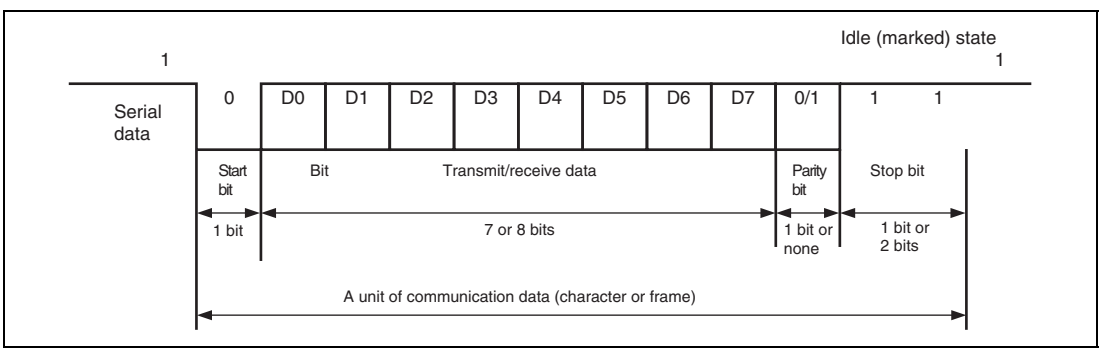


Figure 19.7 Data Format in Asynchronous Mode (Example of 8-Bit Data with Parity and Two Stop Bits)

(1) Transmission/Reception Format and Clock

Table 19.6 shows available data transfer formats. The SCIF supports 8 transfer formats, which can be specified by SCSMR.

The transfer clock can be selected from the following two clocks using the CKE1 and CKE0 bits in SCSCR:

- Internal clock generated by the on-chip baud rate generator
- External clock generated by an external clock baud rate generator

Table 19.6 Serial Transmission/Reception Formats (Asynchronous Mode)

SCSMR settings			Serial transmission/reception format and frame length											
CHR	PE	STOP	1	2	3	4	5	6	7	8	9	10	11	12
0	0	0	S	8-bit data								STOP		
0	0	1	S	8-bit data								STOP	STOP	
0	1	0	S	8-bit data								P	STOP	
0	1	1	S	8-bit data								P	STOP	STOP
1	0	0	S	7-bit data							STOP			
1	0	1	S	7-bit data							STOP	STOP		
1	1	0	S	7-bit data							P	STOP		
1	1	1	S	7-bit data							P	STOP	STOP	

[Legend]
S : Start bit
STOP : Stop bit
P : Parity bit

(2) Data Transmission/Reception

(a) Initialization of SCIF (Asynchronous Mode)

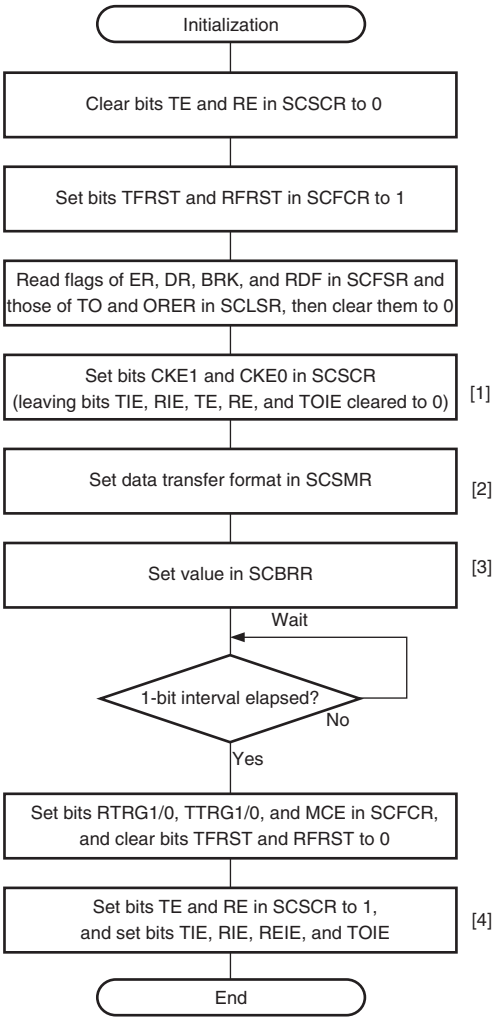
Before transmitting/receiving data or changing the operating mode or communication format, the SCIF should be initialized using the sample flowchart for SCIF initialization shown in Figure 19.8.

[Notes]

Clearing the TE bit to 0 initializes SCTSR. However, SCFSR, SCFTDR, and SCFRDR contents are retained even if the TE and RE bits are cleared to 0.

The TE bit should be cleared to 0 after all transmit data has been sent and the TEND flag has been set in SCFSR. The TE bit can be cleared to 0 during transmission, but the data being transmitted will enter the marked state after clearing. In addition, before setting the TE bit to 1 to restart the transmission, set the TFRST bit to 1 in SCFCR to reset SCFTDR.

When an external clock is used, do not stop the clock during operation or initialization. If stopped, the operation will be unreliable. Furthermore, when the baud rate generator for external clock is also to be used, be sure to make settings for it before starting initialization of the SCIF.



[1] Set the clock selection in SCSCR. Be sure to clear bits TIE, RIE, TE, RE, and TOIE to 0.

[2] Set the data transfer format in SCSMR.

[3] Write a value for the bit rate to SCBRR. (Not necessary if an external clock is used.)

[4] Wait at least one bit interval, then set bit TE or RE in SCSCR to 1. Also, set bits TIE, RIE, REIE, and TOIE. Setting bit TE enables the TX and RX pins to use.
In transmission, the SCIF will go to the mark state, and in reception, it will go to the idle state, waiting for a start bit.

Figure 19.8 Sample Flowchart for Initializing the SCIF

(b) Serial Data Transmission (Asynchronous Mode)

Figure 19.9 shows a sample flowchart for serial transmission.

After the SCIF transmission operation is enabled, serial data transmission can be performed using the following procedure:

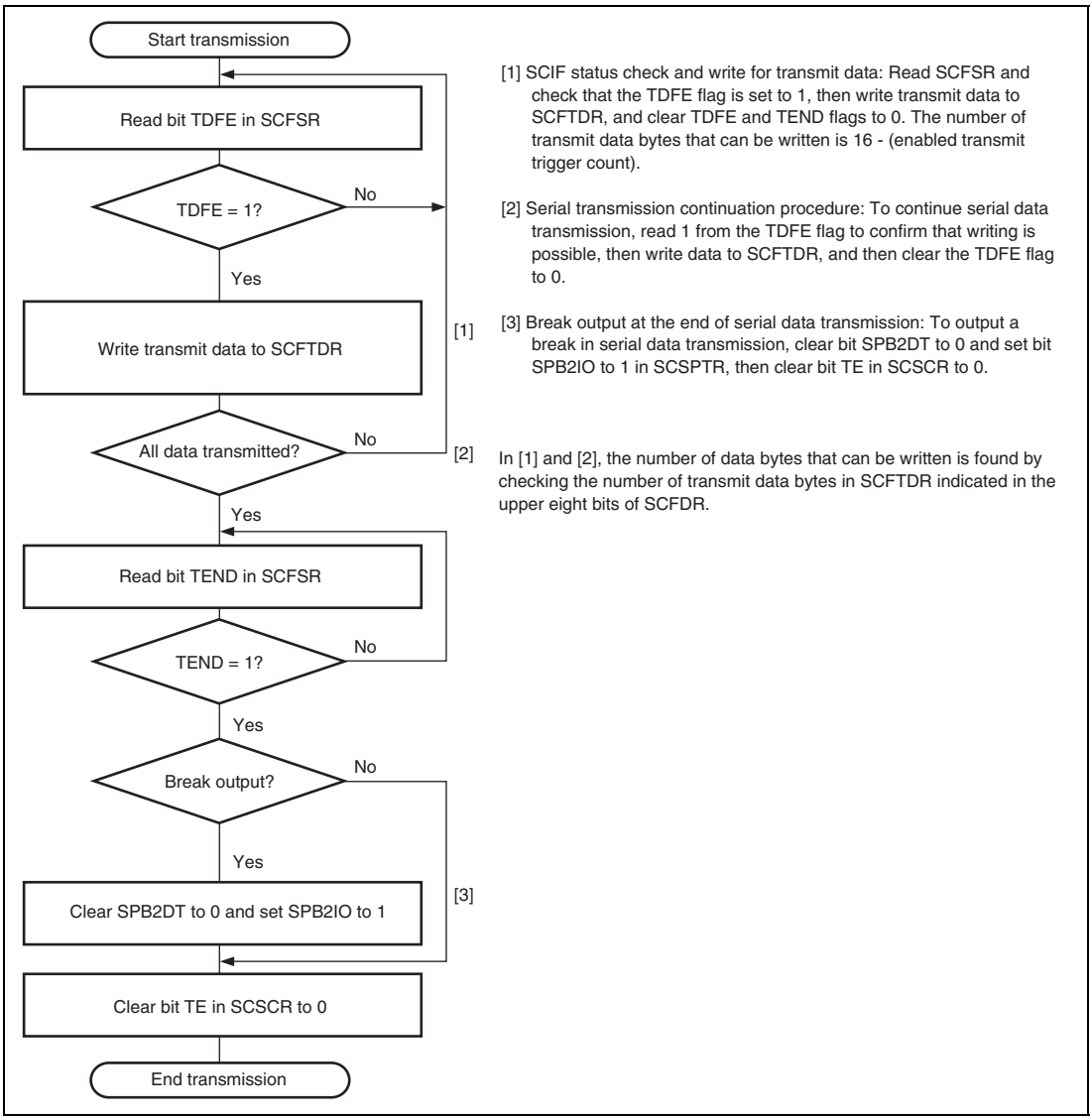


Figure 19.9 Sample Flowchart for Serial Transmission

In serial transmission, the SCIF operates as follows:

1. When data is written to SCFTDR, the SCIF transfers the data from SCFTDR to SCTSR and starts transmitting. Confirm that the TDFE flag in SCFSR is set to 1 before writing transmit data to SCFTDR. The number of data bytes that can be written is at least (16- (transmit trigger count)).
2. When data is transferred from SCFTDR to SCTSR and the SCIF starts transmission, consecutive transmission is performed until there is no transmit data left in SCFTDR. When the number of transmit data bytes in SCFTDR is equal to or less than the transmit trigger count specified in SCFCR, the TDFE flag is set. If the TIE and TEIE bits in SCSCR are set to 1 and 0, respectively at this time, a transmit-FIFO-data-empty interrupt (TDFE) request occurs. The serial transmit data is sent from the TX pin in the following order:
   A. Start bit: One 0-bit is output.
   B. Transmit data: 8- or 7-bit data is output in LSB-first order.
   C. Parity bit: One parity bit (even or odd parity) is output.
      A format that does not output a parity bit can also be selected.
   D. Stop bit(s): One or two 1-bits (stop bits) are output.
   E. Marked state: 1 is output continuously until the start bit that starts the next transmission is sent.
3. The SCIF checks transmit data in SCFTDR when sending the stop bit. If there is data in it, the SCIF transfers the data from SCFTDR to SCTSR, sends the stop bit, and then starts serial transmission of the next frame. If there is no transmit data, the SCIF sets the TEND flag to 1 in SCFSR and sends out the stop bit, and then the marked state is entered to output 1 continuously. At this time, if the TIE and TEIE bits in SCSCR are set to 1, a transmit-end interrupt (TEND) request occurs.

Figure 19.10 shows an example of transmission in asynchronous mode.

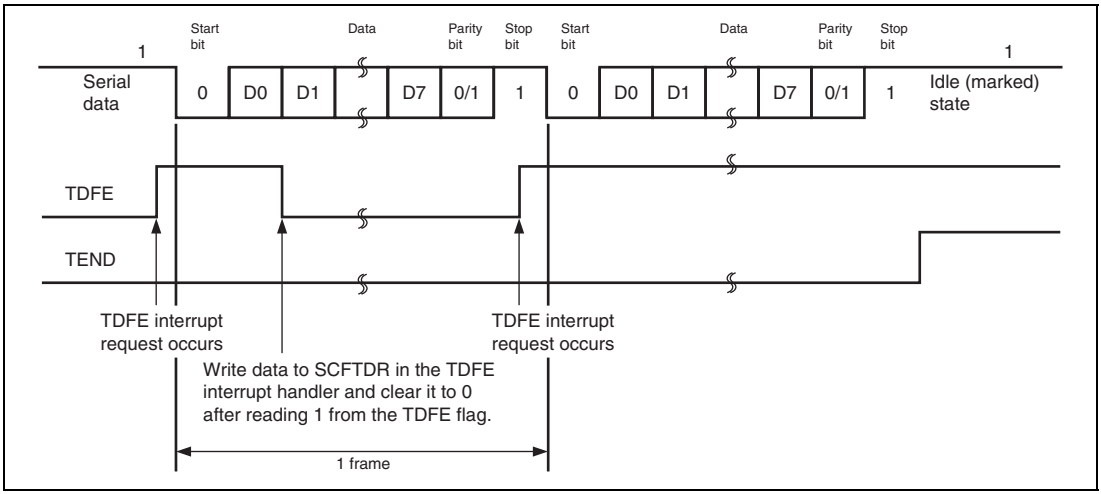


Figure 19.10 Sample SCIF Transmission Operation (Example of 8-Bit Data with Parity and One Stop Bit)

4. When modem control is enabled, transmission can be stopped or resumed in accordance with the $\overline{\text{CTS}}$ input value. When $\overline{\text{CTS}}$ is set to 1 during transmission, the marked state is entered after one frame of data transmission is ended. Setting $\overline{\text{CTS}}$ to 0 restarts outputting the next transmit data from the start bit. Figure 19.11 shows an example of the operation with modem control enabled.

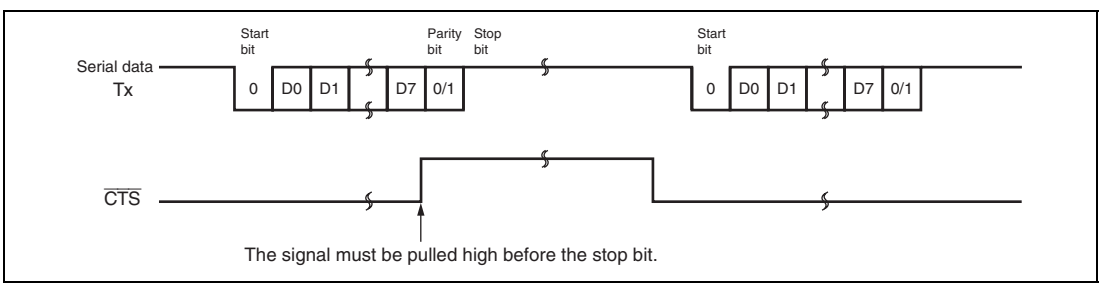


Figure 19.11 Sample Operation with Modem Control Enabled ($\overline{\text{CTS}}$)

(c) Serial Data Reception (Asynchronous Mode)

Figures 19.12 and 19.13 show sample flowcharts for serial reception.

After the SCIF reception operation is enabled, serial data reception can be performed using the following procedure:

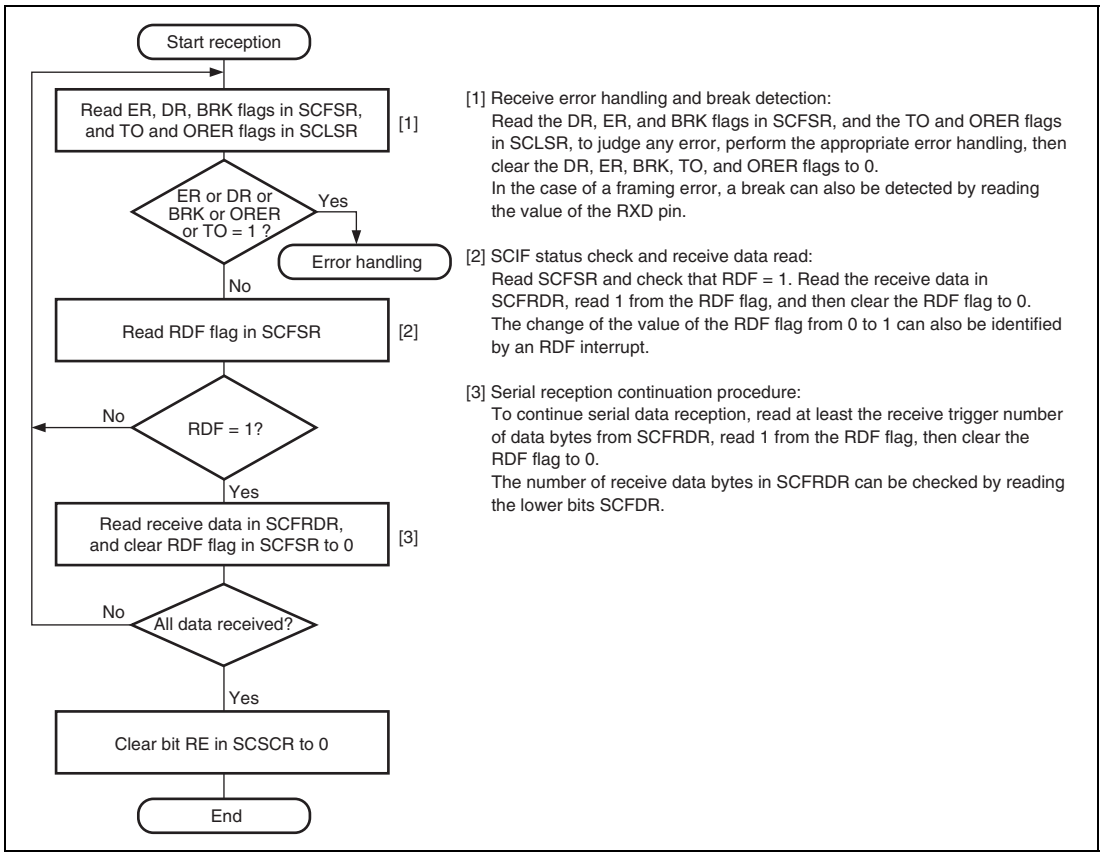
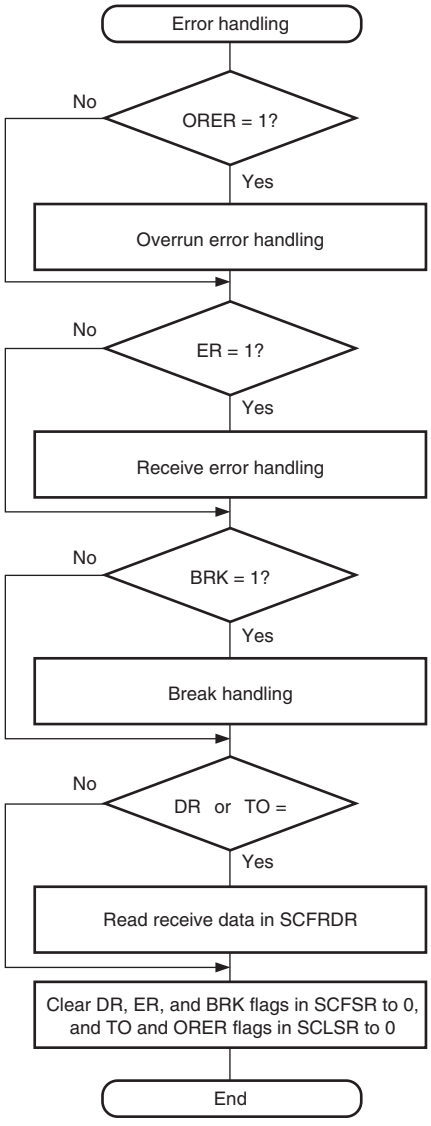


Figure 19.12 Sample Flowchart for Serial Reception (1)



Whether a framing error or parity error has occurred in the receive data that is to be read from SCFRDR can be determined by checking bits FER and PER in SCFSR.

Figure 19.13 Sample Flowchart for Serial Reception (2)

In serial reception, the SCIF operates as follows:

1. The SCIF monitors the transmission line, and when detecting the start bit 0, it performs internal synchronization and starts reception.
2. The SCIF stores the received data in SCRSR in LSB-to-MSB order.
3. The SCIF receives the parity bit and stop bit.

   After receiving these bits, the SCIF performs the following checks. If the SCIF can confirm the conditions of B, C, and D, it stores the receive data in SCFRDR.

Note: The SCIF continues to receive data even when a parity error or a framing error occurs.

   A. Stop bit: The SCIF checks whether the stop bit is 1.

      If there are two stop bits, it checks only the first stop bit.

   B. Receive data: The SCIF checks that receive data can be transferred from the receive shift register (SCRSR) to SCFRDR.

   C. Overrun error: The SCIF checks that the ORER flag is 0, indicating that no overrun error has occurred.

   D. Break state: The SCIF checks that the BRK flag is 0, indicating that the break state is not set.

4. If the RDF flag changes to 1 while the RIE bit in SCSCR is 1, a receive-FIFO-data-full interrupt (RDF) request occurs.

   If the DR flag changes to 1 while the RIE bit in SCSCR is 1, a receive-data-ready interrupt (DR) request occurs.

   If the TO flag changes to 1 while the TOIE bit in SCSCR is 1, a timeout interrupt (TO) request occurs.

   If the ER flag changes to 1 while the RIE or REIE bit in SCSCR is 1, a receive-error interrupt (ER) request occurs.

   If the BRK flag changes to 1 while the RIE or REIE bit in SCSCR is 1, a break interrupt (BRK) request occurs.

   If the ORER flag changes to 1 while the RIE or REIE bit in SCSCR is 1, an overrun-error interrupt (ORER) request occurs.

Figure 19.14 shows an example of reception in asynchronous mode.

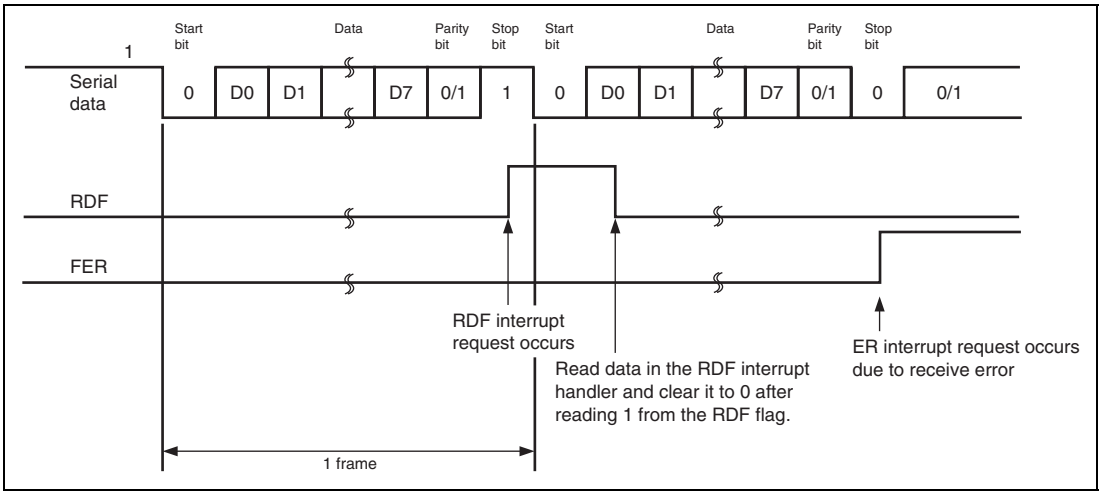


Figure 19.14 Sample SCIF Receive Operation (Example of 8-Bit Data with Parity and One Stop Bit)

5. When modem control is enabled, the $\overline{\text{RTS}}$ signal is output when SCFRDR is empty. When $\overline{\text{RTS}}$ is 0, data can be received. When $\overline{\text{RTS}}$ is set to 1, the number of data bytes in SCFRDR is equal to or more than the $\overline{\text{RTS}}$ output active trigger count.

Figure 19.15 shows an example of the operation with modem control enabled.

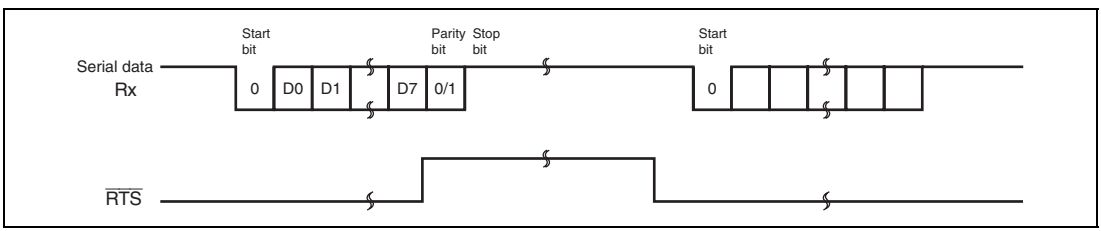


Figure 19.15 Example of the Operation with Modem Control Enabled ($\overline{\text{RTS}}$)

19.3.2 Operation in Clock Synchronous Mode

Clock synchronous mode, in which data is transmitted and received in synchronization with clock pulses, is suitable for fast serial communication.

Figure 19.16 shows the general format for clock synchronous serial communication. In the clock synchronous serial communication, data on the communication line is output between one falling edge of the synchronization clock and the next falling edge. The data decision is done at the rising edge of the clock.

In serial communication, each character is output starting with the LSB and ending with the MSB. After the MSB is output, the communication line retains the state of the last data.

In clock synchronous mode, the SCIF receives data in synchronization with a rising edge of the synchronization clock.

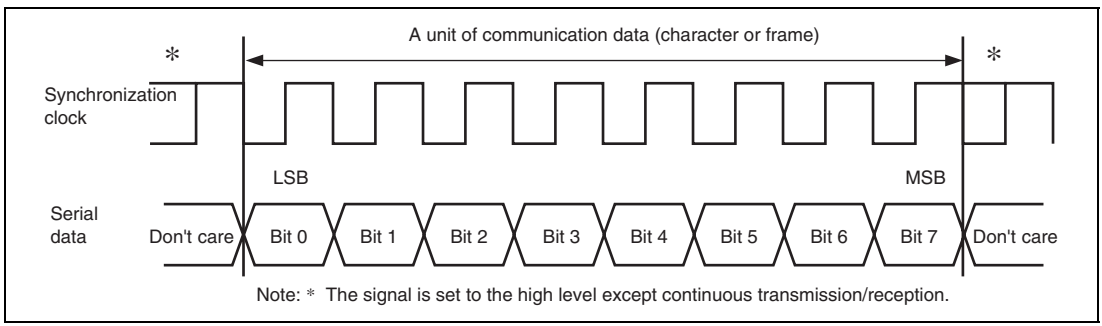


Figure 19.16 Data Format for Clock Synchronous Communication

(1) Data Transfer Format and Clock

The data transfer format is fixed to 8 bits. No parity bit can be added.

The clock can be selected from the following two clocks using the C/$\overline{A}$ bit in SCSMR and the CKE1 and CKE0 bits in SCSCR:

- Internal clock generated by the on-chip baud rate generator
- External synchronization clock input at the SCK pin

When the SCIF is operated on an internal clock, the synchronization clock is output from the SCK pin. Eight synchronization clock pulses are output in one-character transfer, and when no transfer is performed the clock is fixed high. When only reception operation is performed, selecting an internal clock outputs clock pulses until the number of data bytes in the receive FIFO reaches the specified receive trigger count, while the RE bit in SCSCR is 1.

(2) Data Transmission/Reception

(a) Initialization of SCIF (Synchronous Mode)

Before transmitting/receiving data or changing the operating mode or communication format, the TE and RE bits in SCSCR to 0 and then the SCIF should be initialized using the sample flowchart for SCIF initialization shown in figure 19.17.

[Note]

Clearing the TE bit to 0 initializes SCTSR. However, clearing the RE bit to 0 does not affect the RDF, PER, FER, and ORER flags and SCFRDR contents.

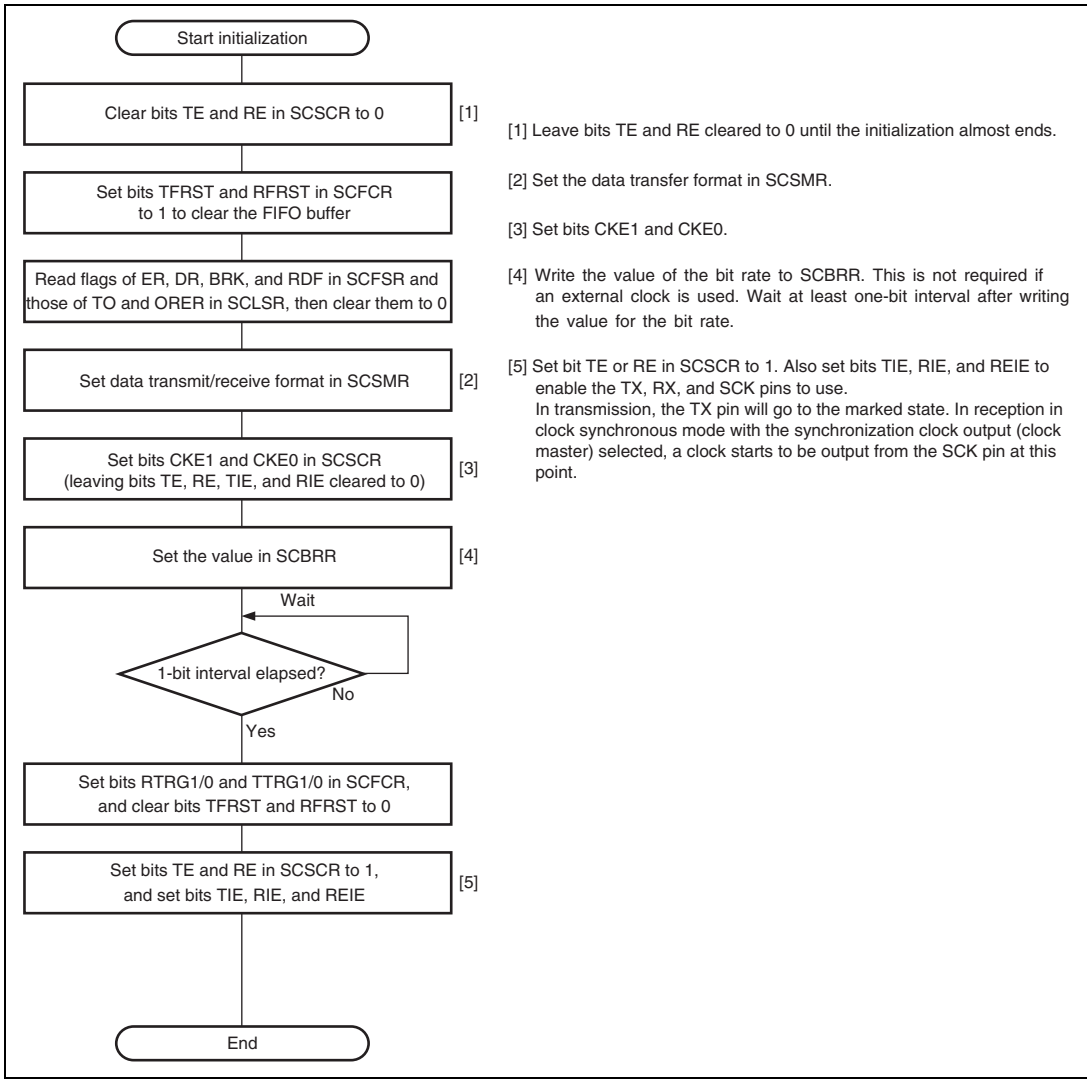


Figure 19.17 Sample Flowchart for SCIF Initialization

(b) Serial Data Transmission (Clock Synchronous mode)

Figure 19.18 shows a sample flowchart for serial transmission.

After the SCIF transmission operation is enabled, serial data transmission can be performed using the following procedure:

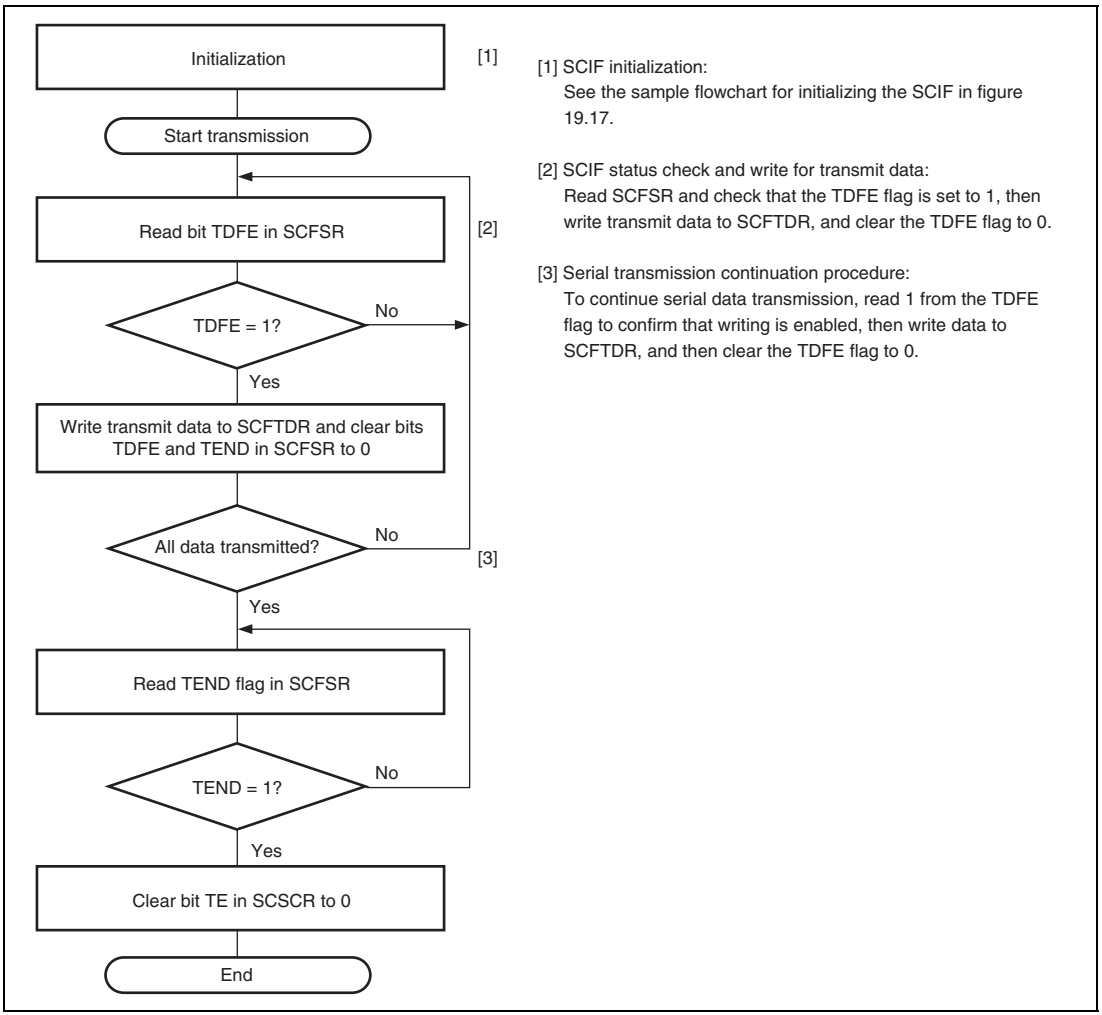


Figure 19.18 Sample Flowchart for Serial Transmission

In serial data transmission, the SCIF operates as described below:

1. When data is written to SCFTDR, the SCIF transfers the data from SCFTDR to SCTSR to start transmission. Confirm that the TDFE flag in SCFSR is set to 1 before writing transmit data to SCFTDR. The number of data bytes that can be written is at least (16- (the transmit trigger count)).
2. After the SCIF transfers data from SCFTDR to SCTSR and starts data serial transmission, the SCIF consecutively transmits it until there is no transmit data left in SCFTDR. When the number of transmit data bytes in SCFTDR is equal to or less than the transmit trigger count set in SCFCR, the TDFE flag is set. If the TIE and TEIE bits in SCSCR are set to 1 and 0, respectively at this time, a transmit-FIFO-data-empty interrupt (TDFE) request occurs. When the external clock is selected, data is output in synchronization with the input clock. The serial transmit data is output at the TX pin in the LSB-first order.
3. The SCIF checks transmit data in SCFTDR when sending the last bit. If there is transmit data, the SCIF transfers the data to SCTSR, and starts serial transmission of the next frame. If there is no transmit data left, the TEND flag in SCFSR is set to 1, and the transmit data pin (TX pin) retains its state after the last bit is sent. At this time, if the TIE and TEIE bits in SCSCR are set to 1, a transmit-end interrupt (TEND) request occurs.
4. After serial transmission ends, the SCK pin is fixed high.

Note: In clock synchronous mode, when transmit data is written to SCFTDR by the DMAC, the TEND flag may no be cleared. Therefore, if the DMAC is used for transmission in clock synchronous mode, read the TEND flag in the following method.

1. Confirm that data transfer is completed in the DMAC.
2. Read the TEND flag.
3. Clear the TEND flag to 0 if TEND = 1.
4. Read the TEND flag again.
5. Use the second-read TEND flag.

Figure 19.19 shows an example of SCIF serial transmission in clock synchronous mode.

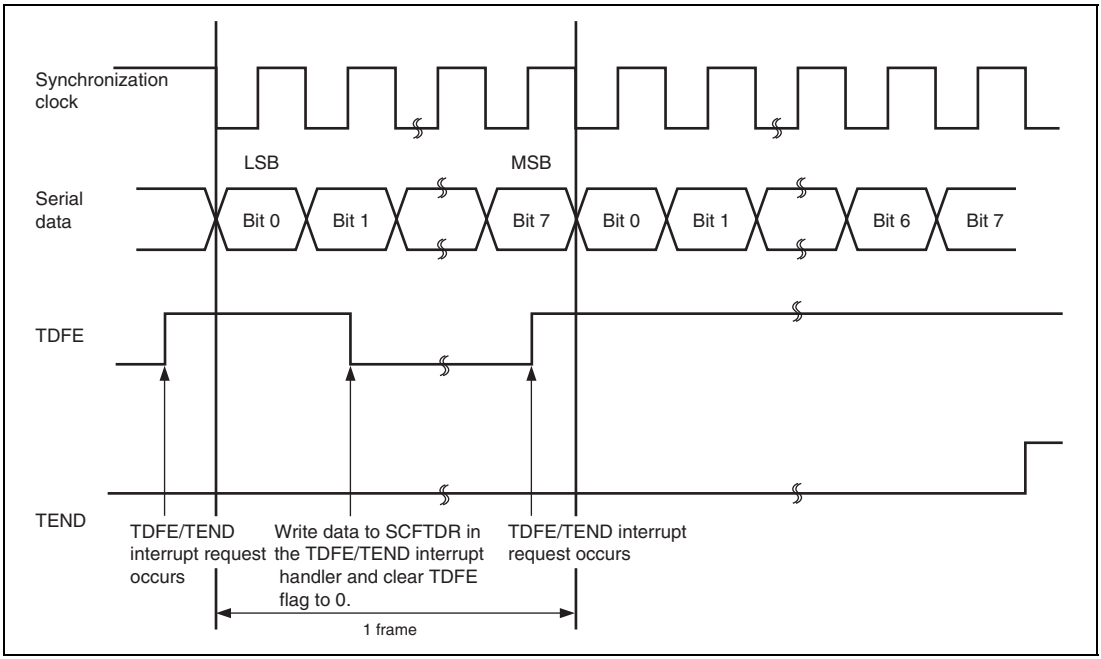


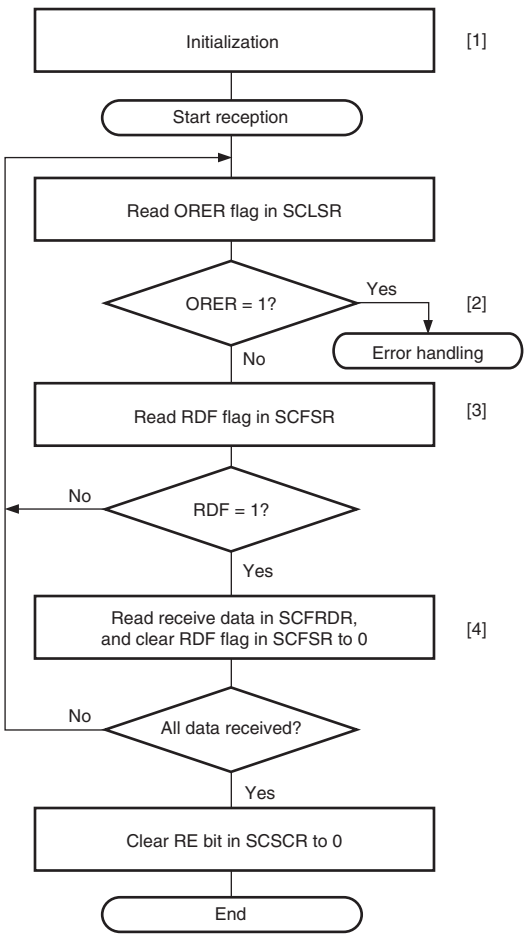
Figure 19.19 Example of SCIF Serial Transmission in Clock Synchronous Mode

(c) Serial Data Reception (Clock Synchronous Mode)

Figures 19.20 and 19.21 show sample flowcharts for serial reception.

The SCIF reception should be enabled before taking the following steps for serial data transmission.

When switching operating mode from asynchronous mode to clock synchronous mode without initializing the SCIF, check that the ORER, PER3 to PER0, and FER3 to FER0 flags are cleared to 0.



[1] SCIF initialization:
See figure 19.17.

[2] Receive error handling:
If a receive error occurs, read the ORER flag in SCLSR to identify any error, perform the appropriate error handling, then clear the ORER flag to 0. The SCIF cannot resume reception while the ORER flag is set to 1.

[3] SCIF status check and receive data read:
Check the SCIF status, read receive data, read SCFSR and check that RDF =1, then read the receive data in SCFRDR, and clear the RDF flag to 0. The change of the value of the RDF flag from 0 to 1 can also be identified by an RDF interrupt.

[4] Serial reception continuation procedure:
To continue serial data reception, read at least the receive trigger count of receive data bytes from SCFRDR, read 1 from the RDF flag, then write 0. The number of receive data bytes in SCFRDR can be found by reading the lower bits of SCFDR. However, as starting up the DMAC using RDF interrupt and reading the value of SCFRDR automatically clears bit RDF, cleaning bit RDF is not required.

Figure 19.20 Sample Flowchart for Serial Data Reception

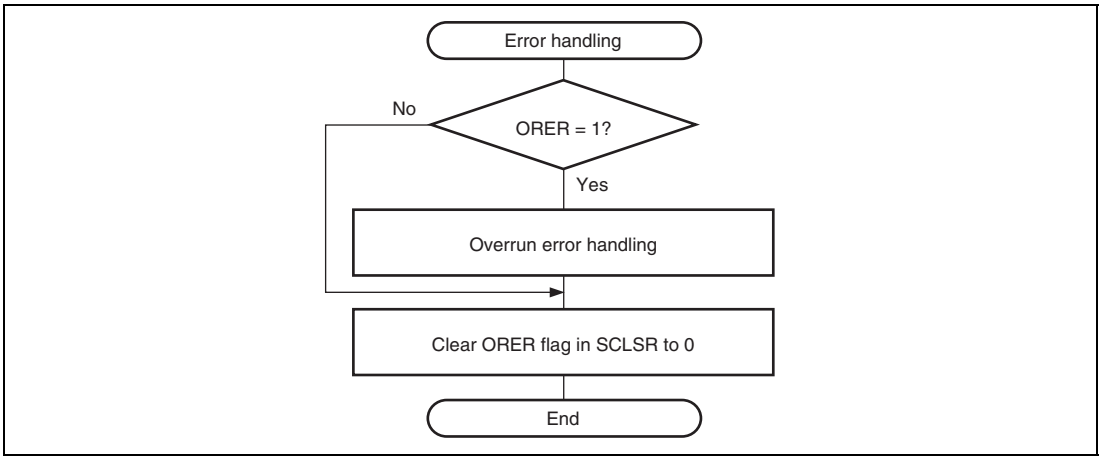


Figure 19.21 Sample Flowchart for Serial Data Reception

In serial data reception, the SCIF operates as described below:

1. The SCIF starts receiving data in synchronization with synchronization clock input or output.
2. The SCIF stores receive data in SCRSR in LSB-to-MSB order.

   After receiving the data, the SCIF first checks if the receive data can be transferred from SCRSR to SCFRDR, then starts storing the receive data in SCFRDR.

   If the SCIF detects an overrun error, the SCIF cannot receive subsequent data.
3. If the RDF flag changes to 1 while the RIE bit in SCSCR is 1, a receive-FIFO-data-full interrupt (RDF) request occurs. If the ORER flag changes to 1 while the RIE or REIE bit in SCSCR is 1, a break interrupt (BRI) request occurs.

Figure 19.22 shows an example of SCIF reception in clock synchronous mode.

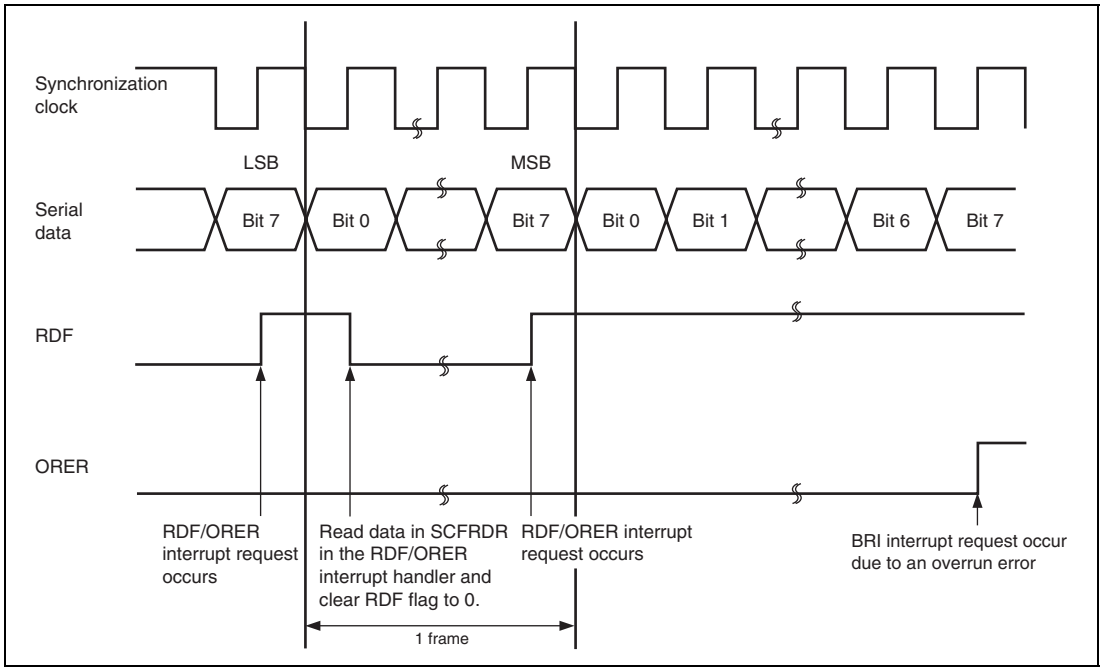
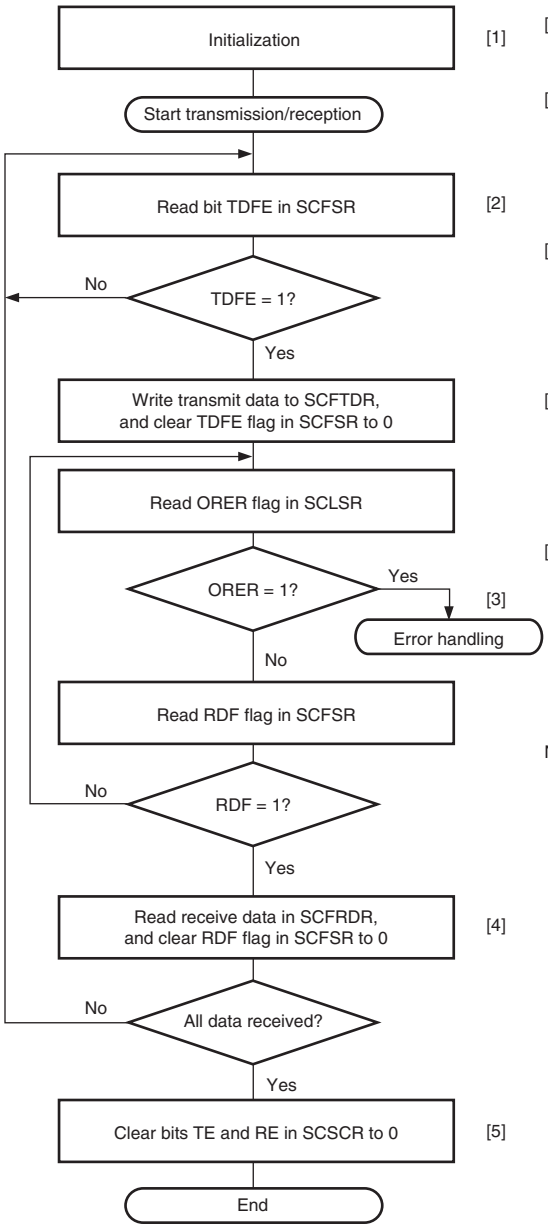


Figure 19.22 Example of SCIF Reception in Clock Synchronous Mode

(d) Simultaneous Serial Data Transmission/Reception (Clock Synchronous Mode)

Figure 19.23 shows a sample flowchart for simultaneous serial data transmission/reception.

The SCIF transmission and reception should be enabled before taking the following steps for simultaneous serial data transmission/reception.



[1] SCIF initialization:
See figure 19.17.

[2] SCIF status check and write for transmit data:
Read SCFSR and check that the TDFE flag is set to 1, then write transmit data to SCFTDR, and clear the TDFE flag to 0. The change of the value of the TDFE flag from 0 to 1 can also be identified by a TDFE/TEND interrupt.

[3] Receive error handling:
If a receive error occurs, read the ORER flag in SCLSR, perform the appropriate error handling, then clear the ORER flag to 0. The SCIF cannot resume reception while the ORER flag is set to 1.

[4] SCIF status check and read for receive data:
Read SCFSR and check that RDF =1, then read the receive data in SCFRDR, and clear the RDF flag to 0. The change of the value of the RDF flag from 0 to 1 can also be identified by an RDF interrupt.

[5] Serial transmission and reception continuation procedure:
To continue serial data transmission and reception, read RDF flag and SCFRDR, and finish clearing the RDF flag to 0 before receiving the MSB in the current frame.
Similarly, read 1 from the TDFE flag to confirm that writing is enabled before transmitting the MSB in the current frame. Then write data to SCFTDR and clear the TDFE flag to 0.

Note : When switching from transmission or reception to simultaneous transmission and reception, clear bits TE and RE to 0, and then set them to 1 simultaneously.

Figure 19.23 Sample Flowchart for Simultaneous Serial Data Transmission/Reception

19.4 SCIF Interrupt Sources and the DMAC

If the DMAC is used for transmission/reception, set and enable the DMAC before setting the SCIF.

Transmission Interrupts and DMA Transfer:

If the TDFE/TEND flag in SCFSR is set to 1 when the TDFE/TEND interrupt is enabled by the TIE bit, a TDFE/TEND interrupt request and a transmit-FIFO-data-empty DMA transfer request will occur. If the TDFE/TEND flag is set to 1 when TDFE/TEND interrupt is disabled by the TIE bit, only the transmit-FIFO-data-empty DMA transfer request will occur. (A transmit-FIFO-data-empty DMA transfer request is generated when the TDFE flag is set while TEIE is 0, or when the TEND flag is set while TEIE is 1. DMA transfer requests are not affected by the TEIE bit.)

When TDFE/TEND interrupt requests are enabled, the interrupt requests are cleared by the DMAC regardless of the interrupt handling program.

Reception Interrupts and DMA Transfer:

If the RDF/DR flag in SCFSR is set to 1 when RDF/DR interrupt is enabled by the RIE bit, an RDF/DR interrupt request occurs. If the RDF/DR flag is set to 1, a receive-FIFO-data-full DMA transfer request occurs. If the RDF/DR flag is set to 1 when RDF/DR interrupt is disabled by the RIE bit, and only a receive-FIFO-data-full DMA transfer request occurs and DMAC can be activated to perform data transfer.

Setting the RIE bit in SCSCR to 0 and the REIE it to 1 generates the ER/BRK/ORER interrupt requests without generating RDF/DR interrupt requests. When the BRK flag in SCFSR or the ORER flag in SCLSR is set to 1, BRK/ORER interrupt requests occur.

If the TO flag is set to 1 in SCLSR when TO interrupts are enabled by the TOIE bit, TO interrupt requests occur.

DR/TO interrupt requests generated by setting the DR or TO flag to 1 or receive-FIFO-data-full DMA transfer requests occur only in asynchronous mode.

When DR/TO interrupt requests are enabled to be issued, interrupt requests generated by the DR flag are cleared by the DMAC regardless of the interrupt handling program, however, those generated by the TO flag are not cleared by the DMAC. Therefore, the TO flag interrupt requests need to be cleared with the interrupt handling program. (The DR and TO flags are set at the same time, but cleared separately.)

Table 19.7 SCIF Interrupt Sources

Interrupt Source	DMAC Activation	Priority on Reset
Interrupts generated by receive error flag (ER)	Not possible	High
Interrupts generated by receive-FIFO-data-full (RDF), receive-data-ready (DR) or timeout (TO) *	Possible	
Interrupts generated by break (BRK) or overrun error (ORER)	Not possible	
Interrupts generated by transmit FIFO data empty (TDFE)	Possible	Low

Note: * Interrupts by means of the DR or TO flag are available only in asynchronous mode.

19.5 Usage Notes

Note the following on use of the SCIF.

(1) Break Detection and Operation

Break signals can also be detected by reading the RX pin value directly when a framing error (FER) is detected. In the break state, the input values from the RX pin are all 0s. So, the parity error flag (PER) may be set after the FER flag is set to 1.

Although the SCIF stops receive data transfer to SCFRDR after detecting a break, it continues data reception.

(2) Sending a Break Signal

The input/output condition and level of the TX pin are determined by the SPB2IO and SPB2DT bits in SCSPTR. This enables to send a break signal.

The pin does not function as the TX pin from the initialization of the serial transmitter to setting of the TE bit (enabling transmission). In this period, the marked state is substituted by the value of the SPB2DT bit. Therefore, the SPB2IO and SPB2DT bits should be set to 1 (designating output and high level) beforehand.

To send a break signal during serial transmission, clear the SPB2DT bit to 0 (low level), and then clear the TE bit to 0 (halting transmission). When the TE bit is cleared, the transmitter is initialized regardless of the current transmission state, and the TX pin outputs 0.

(3) Data Sampling Timing and Reception Margin in Asynchronous Mode

In asynchronous mode, the SCIF operates on the base clock with a frequency 16 times the bit rate.

In reception, the SCIF performs the internal synchronization by sampling the fall edge of the start bit using the base clock. In addition, the SCIF takes receive data at the rising edge of the eighth pulse on the base clock.

Figure 19.24 shows the timing of this operation.

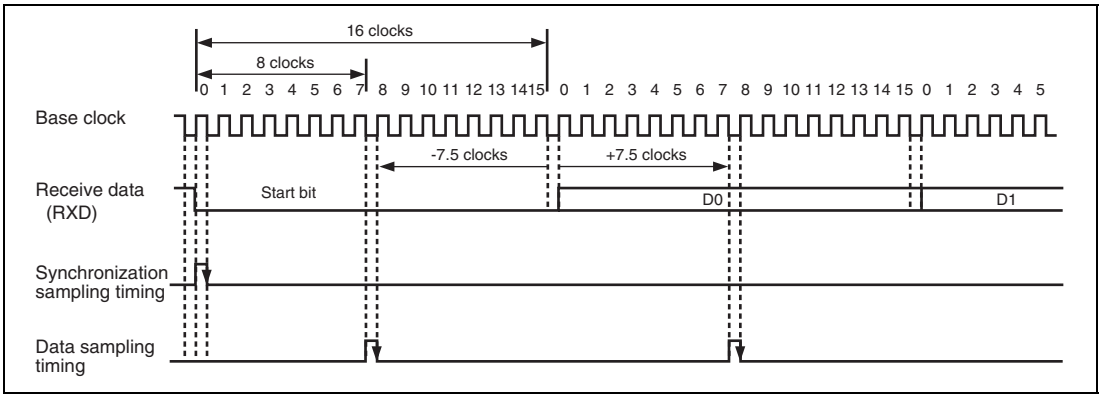


Figure 19.24 Timing Chart of Receive Data Sampling

The reception margin in asynchronous mode is given by formula (1).

$$M = \left| (0.5 - \frac{1}{2N}) - (L - 0.5)F - \frac{|D - 0.5|}{N}(1 + F) \right| \times 100\% \quad \cdots\cdots \text{Expression (1)}$$

M: Receive margin (%)

N: Ratio of bit rate to clock (N = 16)

D: Clock duty (D = 0 to 1.0)

L: Frame length (L = 9 to 12)

F: Absolute value of clock frequency deviation

Assuming F = 0 and D = 0.5 for expression (1), the reception margin obtained with expression (2) is 46.875 % as shown below:

Assuming D = 0.5 and F = 0

$M = (0.5 - 1/(2 \times 16)) \times 100\ \%$

$= 46.875\ \%$ ……expression (2)

This is a theoretical value. A reasonable margin allowed in system designs is 20 % to 30 %.

(4) Reception Margin and Baud Rate Error

The value of 46.875 % obtained by the above expression (2) indicates the reception margin when the baud rate error is 0 (F = 0). If there is no error in the reception and transmission baud rates, reception is possible even with misalignment of approx. 1/2 bit. If there is an error in the reception and transmission baud rates, the errors are accumulated up to the stop bit reception, which reduces the reception margin. The allowable baud rate error can be obtained by modifying the F in expression (1). When D = 0.5:

$F = \{(15/32 - M)/(L - 0.5)\} \times 100\ (\%)$……expression (3)

By using expression (3), the relationship between the allowable error and reception margin with the frame length = 12 can be summarized as follows:

Allowed Error (%)	Reception Margin (%)
4.07	0
3.64	5
3.20	10
2.33	20
1.46	30

19.6 Baud Rate Generator for External Clock (BRG)

19.6.1 Overview

The SCIF incorporates a baud rate generator for external clock (abbreviated as BRG, hereafter). The BRG supplies a sampling clock (BRGCLK) to the SCIF core by dividing the external clock SC_CLK (selectable between SCIF_CLK and clks1) by 1 to $2^{16} - 1$. In addition, the BRG switches the output between the external clock SCK and divided clock.

19.6.2 Description of Blocks

Figure 19.25 shows a block diagram of the BRG.

(1) Reset Controller:

The reset controller handles resetting the control register, base counter, and trigger generator.

(2) Control Register:

The control register has the frequency division register and clock select register.

(3) Base Counter:

The base counter is a 16-bit CLK synchronization counter that is used to determine the timing for generating a frequency divided clock.

(4) Trigger Generator:

The trigger generator generates rising-edge/falling-edge triggers for a frequency divided clock with the timing according to values of the frequency division register and base counter. The triggers are used to generate the frequency divided clock. In addition, the trigger generator switches the output between the SCK (external clock input) and frequency divided clock.

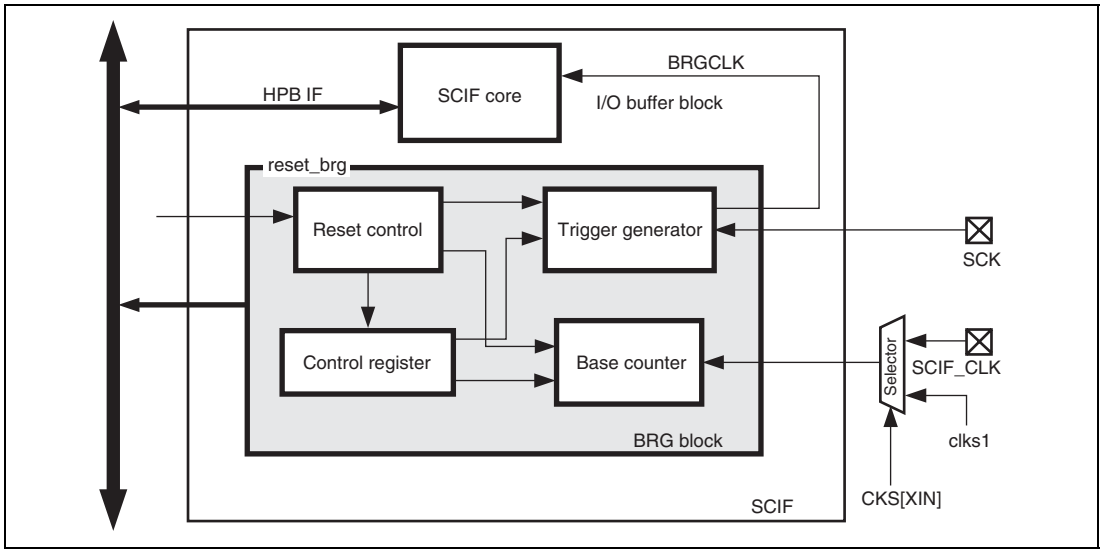


Figure 19.25 BRG Block Diagram

19.6.3 Register Configuration

Table 19.8 shows the registers in the BRG block.

Table 19.8 List of Registers

Name	Abbreviation	R/W	Initial Value	Relative Address	Access Size
Frequency division register	DL	R/W	H'00	H'30	16
Clock select register	CKS	R/W	H'00	H'34	16

(1) Frequency Division Register (DL)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DL15	DL14	DL13	DL12	DL11	DL10	DL9	DL8	DL7	DL6	DL5	DL4	DL3	DL2	DL1	DL0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	DL[15:0]	H'0000	R/W	Specifies the value of frequency division for the frequency divided clock generated by the BRG. This register supports a 16-bit binary format that allows specifying a value in the rage of 1 to 65535. Setting all 0s in these bits makes the BRG output the frequency divided clock at the low level. The value of frequency division is given by the following formula: The value of frequency division = (clock input frequency)/(required baud rate x 16) Table 19.9 shows how to use the baud rate generator with a 3.6864-MHz crystal resonator.

(2) Clock Select Register (CKS)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CKS	XIN	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	CKS	0	R/W	This bit switches the output between the frequency divided clock (SC_CLK) and external clock (SCK). 0: Selects the frequency divided clock. 1: Selects the external clock.
14	XIN	0	R/W	Selects the baud rate generator clock source for the external clock between SCIF_CLK and clks1. 0: Selects the external clock (SCIF_CLK). 1: Selects the internal clock (clks1).
13 to 0	—	All 0	R	Reserved These bits are always read as 0. Only 0 should be written to these bits.

Table 19.9 Baud Rate (3.6864-MHz clock)

Baud Rate	Value of Frequency Division	Error Rate (*)
50	4608	—
75	3072	—
110	2095	–0.022
134.5	1713	0.001
150	1536	—
300	768	—
600	384	—
1200	192	—
1800	128	—
2000	115	0.174
2400	96	—
3600	64	—
4800	48	—
7200	32	—
9600	24	—
14400	16	—
19200	12	—
38400	6	—
76800	3	—
115200	2	—

Note: —: Indicates that the error rate is 0.

19.6.4 Notes on Setting Frequency Division Register

1. For the initial setting of the register after a reset, at least one bit of waiting period is required to secure the clock stabilization time.

   (Example) One bit period when DL = 2

   3.68 (MHz) × 1/2 × 1/16 = 0.115 (MHz) → 8695 (ns)

2. For modifying the register value after the setting stated in <1> above, at least one bit of waiting period at the maximum bit rate (DL = '65535') is required.

   The SCIF registers and BRG registers should be set as the following table:

- Asynchronous mode (SC_CLK external input)

SCIF		BRG	
Register and Bit Names	**Setting Value**	**Register Name**	**Setting Value**
SCSMR.C/$\overline{A}$	0	CKS	H'0000
SCSCR.CKE1, CKE0	10	DL	H'1 to H'FFFF

- Asynchronous mode (SCK external input)

SCIF		BRG	
Register and Bit Names	**Setting Value**	**Register Name**	**Setting Value**
SCSMR.C/$\overline{A}$	0	CKS	H'8000
SCSCR.CKE1, CKE0	10	DL	Don't care

- Clock synchronous mode (external input)

SCIF		BRG	
Register and Bit Names	**Setting Value**	**Register Name**	**Setting Value**
SCSMR.C/$\overline{A}$	1	CKS	H'8000
SCSCR CKE1, CKE0	10	DL	Don't care

3. The register settings for the baud rate generator for external clock should be made before starting initialization of the SCIF.

Section 19A IrDA

19A.1 Overview

The IrDA module is used with channel 3 of the SCIF, and has functions to modulate or demodulate the data format for the serial communication interface to that for the IrDA infrared communication.

19A.2 Register Configuration

The base address of the IrDA register is as follows;

H'FFE4 3000

Table 19A.1 Register List

Register Name	Abbreviation	R/W	Initial Value	Offset Address	Access Size
Serial Mode Register	SCSMRIR	R/W	H'0000	H'0040	16 bits

19A.2.1 Serial Mode Register (SCSMRIR)

SCSMRIR is initialized to H'0000 by a power-on reset or a manual reset and in deep standby mode.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	EDGEN	LOOP	IRMOD	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W*	R/W*	R/W*	R/W*	R/W*	R	R

Note: * When writing to bits 6 to 2, write 0. Do not write 1 to these bits.

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved
9	EDGEN	0	R/W	Rx3 Sampling Mode 0: The Rx3 pin is sampled by an edge. 1: The Rx3 pin is sampled by both an edge and level.

Bit	Bit Name	Initial Value	R/W	Description
8	LOOP	0	R/W	IrDA Loop Back Test 0: Normal operation 1: Loop back operation from Tx3 to Rx3.
7	IRMOD	0	R/W	IrDA Mode 0: Channel 3 of the SCIF is used as the SCIF. 1: Channel 3 of the SCIF is used as the IrDA.
6 to 2	—	All 0	R/W*	Reserved
1, 0	—	All 0	R	Reserved

19A.3 Interface

19A.3.1 Data Transmission Reception Format

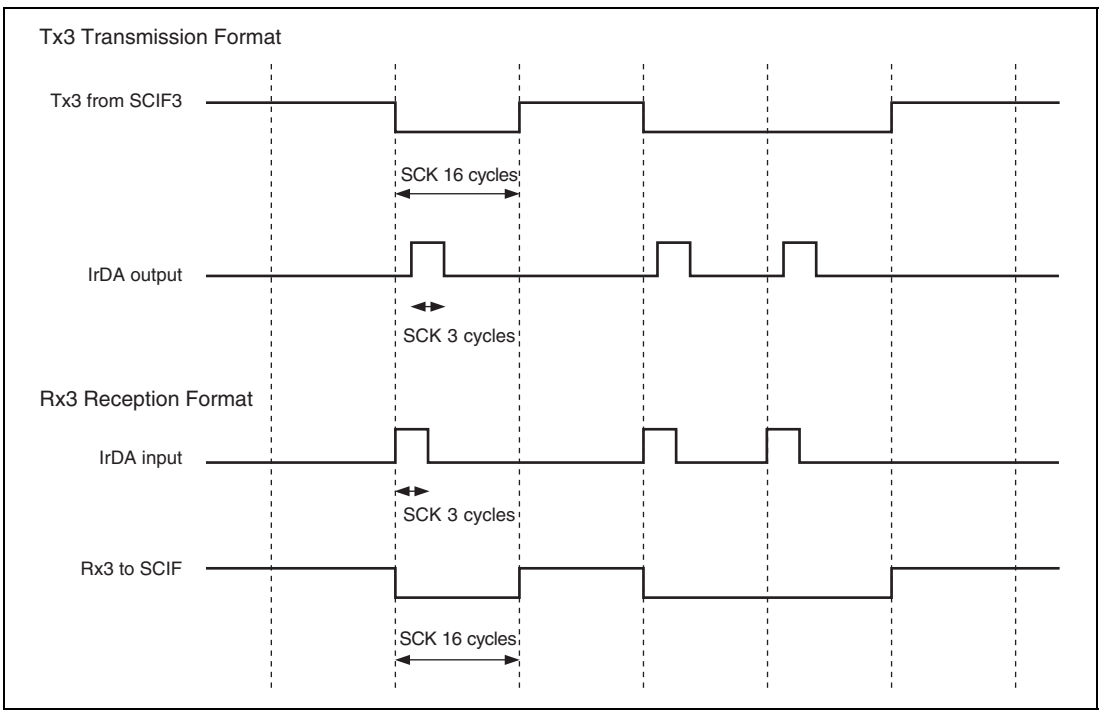


Figure 19A.1 Data Transmission Reception Format

19A.3.2 Data-Selector Block Operation

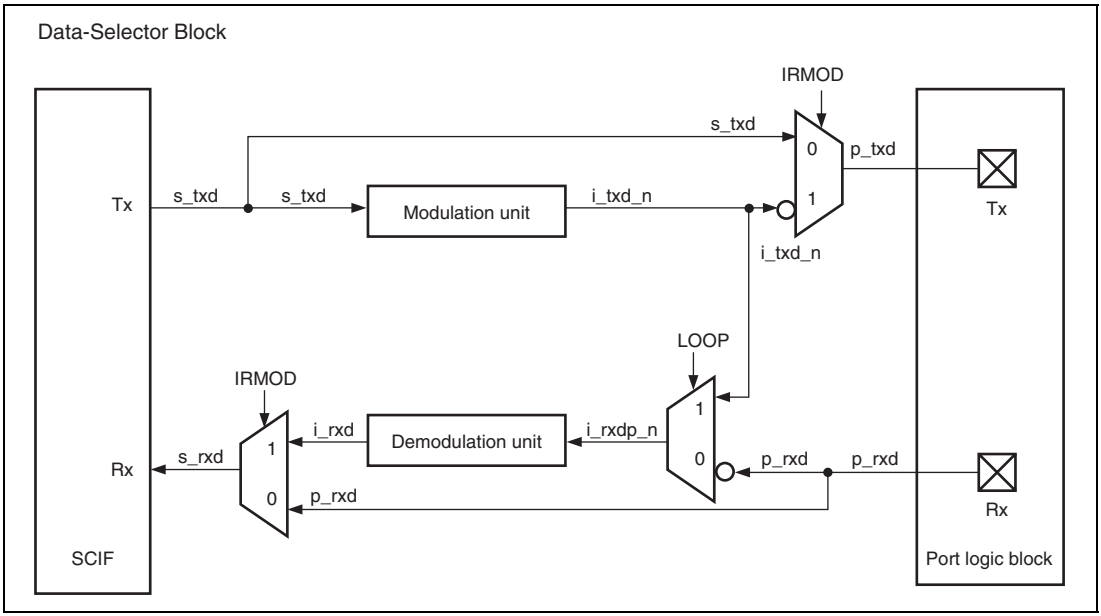


Figure 19A.2 Block Diagram of Data-Selector Part

The data-selector block selects received data for input to the SCIF (s_rxd in figure 19A.2) and data for transmission to be output to the port logic block (p_txd in figure 19A.2).

When the IRMOD bit in SCSMRIR is 1, the modulated/demodulated IrDA data is input/output.

When the IRMOD bit in SCSMRIR is 0, data from the SCIF bypass the port logic block, that is, the data are not modulated or demodulated.

When the LOOP bit in SCSMRIR has the "active" setting, data for transmission from the SCIF are modulated (i_txd_n in figure 19A.2) and then input to the demodulation circuit in the data-selector block (for loopback testing).

The selector does not use clocks for its logic. Therefore, the status of the selector after a reset or in standby depends on only the value of the SCSMRIR register.

Note: This module can detect short pulses such as chattering in reception, be sure to secure the shortest pulse width specified by the IrDA standard.

Section 20 Renesas Serial Peripheral Interface

This LSI includes two-channel Renesas serial peripheral interfaces.

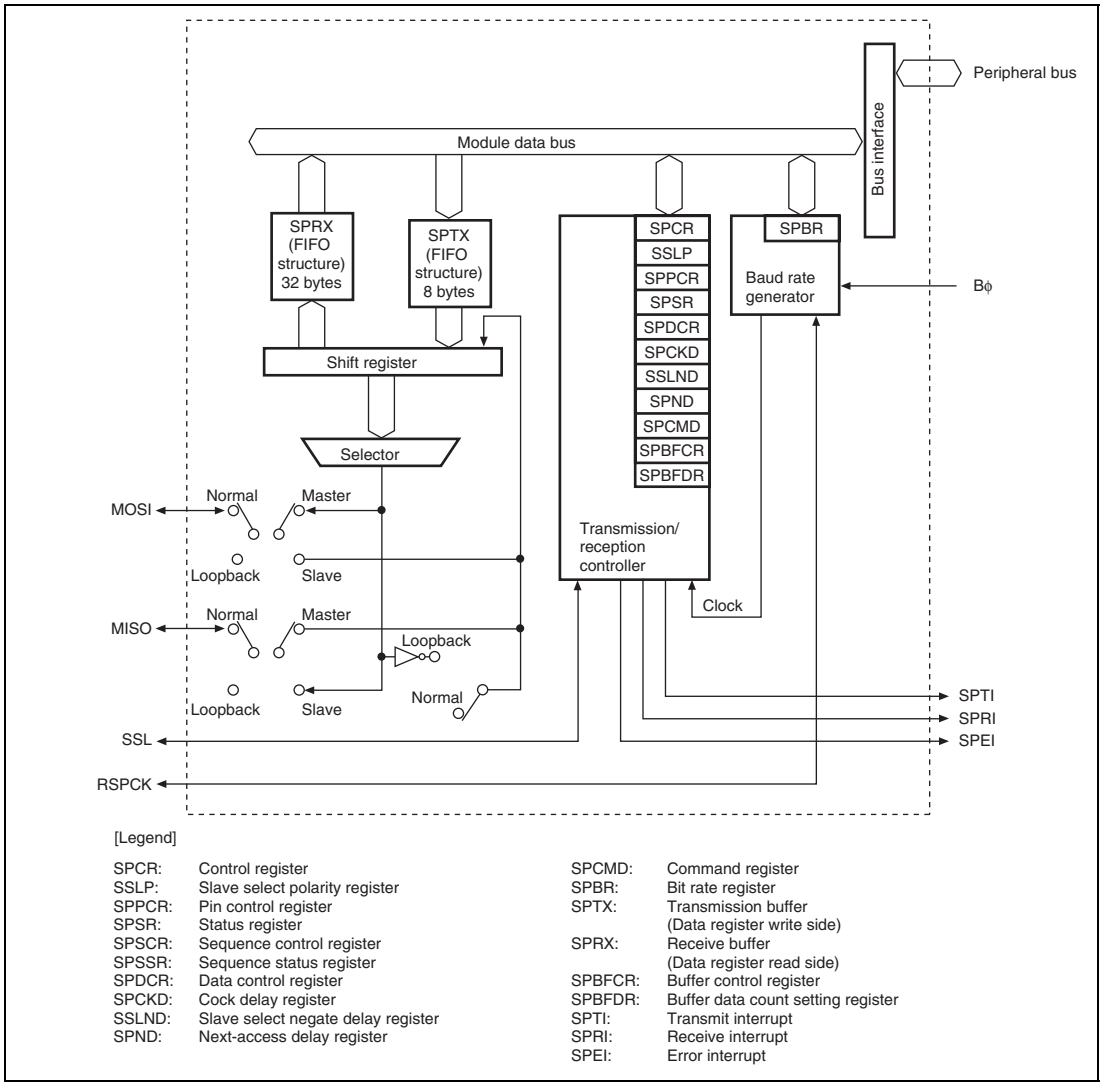
This module is capable of full-duplex serial communication.

20.1 Features

This module has the following features.

- SPI transfer functions
  Use of MOSI (master out/slave in), MISO (master in/slave out), SSL (slave select), and RSPCK (SPI clock) signals allow for serial communications through SPI operation (four-wire method).
  Capable of serial communications in master/slave mode
  Supports mode fault error detection (only in SPI slave mode)
  Supports overrun error detection (only in SPI slave mode)
  Switching of the polarity of the serial transfer clock
  Switching of the clock phase of serial transfer
- Data format
  MSB-first/LSB-first selectable
  Transfer bit-length is selectable as 8, 16, or 32 bits.
- Bit rate
  RSPCK can be divided by a maximum of 4096 in master mode
  RSPCK can be generated by dividing Bϕ by the on-chip baud rate generator.
  An externally input clock can be used as a serial clock.
- Buffer configuration
  8 bytes for transmission and 32 bytes for reception.

- SSL control function
  One SSL signal for each channel
  In master mode, outputs SSL signal.
  In slave mode, inputs SSL signal.
  Controllable delay from SSL output assertion to RSPCK operation (RSPCK delay)
  Range: 1 to 8 RSPCK cycles (set in RSPCK-cycle units)
  Controllable delay from RSPCK stoppage to SSL output negation (SSL negation delay)
  Range: 1 to 8 RSPCK cycles (set in RSPCK-cycle units)
  Controllable wait for next-access SSL output assertion (next-access delay)
  Range: 1 to 8 RSPCK cycles (set in RSPCK-cycle units)
  Function for changing SSL polarity
- Control in master transfer
  A transfer of up to four commands can be executed sequentially in looped execution.
  For each command, the following can be set:
  SSL signal value, bit rate, RSPCK polarity/phase, transfer data length, LSB/MSB first, burst, RSPCK delay, SSL negation delay, and next-access delay.
  A transfer can be initiated by writing to the transmit buffer.
  A transfer can be initiated by clearing the SPTEF bit.
  MOSI signal value specifiable in SSL negation
- Interrupt sources
  Maskable interrupt sources:
  Receive interrupt (receive buffer full)
  Transmit interrupt (transmit buffer empty)
  Error interrupt (mode fault, overrun)
- Others
  Provides loop back mode
  Provides a function for disabling (initializing) this module



[Legend]

SPCR:	Control register	SPCMD:	Command register
SSLP:	Slave select polarity register	SPBR:	Bit rate register
SPPCR:	Pin control register	SPTX:	Transmission buffer (Data register write side)
SPSR:	Status register	SPRX:	Receive buffer (Data register read side)
SPSCR:	Sequence control register	SPBFCR:	Buffer control register
SPSSR:	Sequence status register	SPBFDR:	Buffer data count setting register
SPDCR:	Data control register	SPTI:	Transmit interrupt
SPCKD:	Cock delay register	SPRI:	Receive interrupt
SSLND:	Slave select negate delay register	SPEI:	Error interrupt
SPND:	Next-access delay register		

Figure 20.1 Block Diagram

20.2 Input/Output Pins

Table 20.1 shows the pin configuration. This module automatically switches the input/output direction of the SSL pin. SSL is set as an output in master mode and as an input in slave mode. Pins RSPCK, MOSI, and MISO are automatically set as inputs or outputs according to the setting of master or slave and the level input on SSL (see section 20.4.2, Pin Control).

Table 20.1 Pin Configuration

Pin Name	Abbreviation	I/O	Function
Clock pin	D6 (RSPCK)	I/O	Clock input/output
Master transmit data pin	D10 (MOSI)	I/O	Master transmit data
Slave transmit data pin	D11 (MISO)	I/O	Slave transmit data
Slave select pin	D7 (SSL)	I/O	Slave selection

20.3 Register Descriptions

Table 20.2 (1) shows the register configuration. These registers enable this module to perform the following controls: specifying master/slave modes, specifying a transfer format, and controlling the transmitter and receiver.

Do not write to addresses other than those listed in the table below, otherwise normal operation cannot be guaranteed. Values read from other addresses are undefined.

Table 20.2 (1) Register Configuration

Register Name	Abbreviation	R/W	Address	Access Size
Control register	SPCR	R/W	H'FFFC2000	8, 16
Slave select polarity register	SSLP	R/W	H'FFFC2001	8, 16
Pin control register	SPPCR	R/W	H'FFFC2002	8, 16
Status register	SPSR	R/(W)*	H'FFFC2003	8, 16
Data register	SPDR	R/W	H'FFFC2004	8, 16, 32
Sequence control register	SPSCR	R/W	H'FFFC2008	8, 16
Sequence status register	SPSSR	R	H'FFFC2009	8, 16
Bit rate register	SPBR	R/W	H'FFFC200A	8, 16
Data control register	SPDCR	R/W	H'FFFC200B	8, 16
Clock delay register	SPCKD	R/W	H'FFFC200C	8, 16
Slave select negation delay register	SSLND	R/W	H'FFFC200D	8, 16
Next-access delay register	SPND	R/W	H'FFFC200E	8
Command register 0	SPCMD0	R/W	H'FFFC2010	16
Command register 1	SPCMD1	R/W	H'FFFC2012	16
Command register 2	SPCMD2	R/W	H'FFFC2014	16
Command register 3	SPCMD3	R/W	H'FFFC2016	16
Buffer control register	SPBFCR	R/W	H'FFFC2020	8, 16
Buffer data count setting register	SPBFDR	R/W	H'FFFC2022	16

Note: * Only 0 can be written to clear the flag.

Table 20.2 (2) Register State in Each Operating Mode

Abbreviation	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
SPCR	H'00	H'00	Retained	Retained	Retained	Initialized
SSLP	H'00	H'00	Retained	Retained	Retained	Initialized
SPPCR	H'00	H'00	Retained	Retained	Retained	Initialized
SPSR	H'60	H'60	Retained	Retained	Retained	Initialized
SPDR	Undefined	Undefined	Retained	Retained	Retained	Initialized
SPSCR	H'00	H'00	Retained	Retained	Retained	Initialized
SPSSR	H'00	H'00	Retained	Retained	Retained	Initialized
SPBR	H'FF	H'FF	Retained	Retained	Retained	Initialized
SPDCR	H'20	H'20	Retained	Retained	Retained	Initialized
SPCKD	H'00	H'00	Retained	Retained	Retained	Initialized
SSLND	H'00	H'00	Retained	Retained	Retained	Initialized
SPND	H'00	H'00	Retained	Retained	Retained	Initialized
SPCMD0	H'070D	H'070D	Retained	Retained	Retained	Initialized
SPCMD1	H'070D	H'070D	Retained	Retained	Retained	Initialized
SPCMD2	H'070D	H'070D	Retained	Retained	Retained	Initialized
SPCMD3	H'070D	H'070D	Retained	Retained	Retained	Initialized
SPBFCR	H'00	H'00	Retained	Retained	Retained	Initialized
SPBFDR	H'0000	H'0000	Retained	Retained	Retained	Initialized

Note: Initialized: The value is the one written in "Power-on Reset".

20.3.1 Control Register (SPCR)

SPCR sets the operating mode. If the MSTR and MODFEN bits are changed while the function of this module is enabled by setting the SPE bit to 1, subsequent operations cannot be guaranteed.

Bit:	7	6	5	4	3	2	1	0
	SPRIE	SPE	SPTIE	SPEIE	MSTR	MOD FEN	—	—
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	SPRIE	0	R/W	Receive Interrupt Enable Enables or disables generation of receive interrupt requests (SPRI) when the number of receive data units in the receive buffer (SPRX) is equal to or greater than the specified receive buffer data triggering number and the SPRF flag in SPSR is set to 1. 0: Disables the generation of receive interrupt requests. 1: Enables the generation of receive interrupt requests.
6	SPE	0	R/W	Function Enable Setting this bit to 1 enables the module function. When the MODF bit in the status register (SPSR) is 1, the SPE bit cannot be set to 1 (see section 20.4.6, Error Detection). Setting the SPE bit to 0 disables the module function, and initializes a part of the module function (see section 20.4.7, Initialization). 0: Disables the module function 1: Enables the module function

Bit	Bit Name	Initial Value	R/W	Description
5	SPTIE	0	R/W	Transmit Interrupt Enable Enables or disables generation of transmit interrupt requests (SPTI) when the number of transmit data units in the transmit buffer (SPTX) is equal to or less than the specified transmit buffer data triggering number and the SPTEF flag in SPSR is set to 1. 0: Disables the generation of transmit interrupt requests. 1: Enables the generation of transmit interrupt requests.
4	SPEIE	0	R/W	Error Interrupt Enable Enables or disables the generation of error interrupt requests when this module detects a mode fault error and sets the MODF bit in the status register (SPSR) to 1, or when this module detects an overrun error and sets the OVRF bit in SPSR to 1 (see section 20.4.6, Error Detection). 0: Disables the generation of error interrupt requests. 1: Enables the generation of error interrupt requests. Note: This bit is valid only in SPI slave mode.
3	MSTR	0	R/W	Master/Slave Mode Select Selects master/slave mode. According to MSTR bit settings, this module determines the direction of pins RSPCK, MOSI, MISO, and SSL pins. 0: Slave mode 1: Master mode
2	MODFEN	0	R/W	Mode Fault Error Detection Enable Enables or disables the detection of mode fault errors (see section 20.4.6, Error Detection). 0: Disables the detection of mode fault errors 1: Enables the detection of mode fault errors Note: This bit is valid only in SPI slave mode. When master mode is specified with the MSTR bit, this bit should always be cleared to 0.
1, 0	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.

20.3.2 Slave Select Polarity Register (SSLP)

SSLP sets the polarity of the SSL signal. If the contents of SSL0P are changed while the function of this module is enabled by setting the SPE bit in the control register (SPCR) to 1, subsequent operations cannot be guaranteed.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SSL0P
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
0	SSL0P	0	R/W	SSL Signal Polarity Setting Sets the polarity of the SSL signal. The value of SSL0P indicates the active polarity of the SSL signal. 0: SSL signal 0-active 1: SSL signal 1-active

20.3.3 Pin Control Register (SPPCR)

SPPCR sets the modes of the pins. If the contents of this register are changed while the function of this module is enabled by setting the SPE bit in the control register (SPCR) to 1, subsequent operations cannot be guaranteed.

Bit:	7	6	5	4	3	2	1	0
	—	—	MOIFE	MOIFV	—	—	—	SPLP
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
5	MOIFE	0	R/W	MOSI Idle Value Fixing Enable Fixes the MOSI output value when this module in master mode is in an SSL negation period (including the SSL retention period during a burst transfer). When MOIFE is 0, this module outputs the last data from the previous serial transfer during the SSL negation period. When MOIFE is 1, this module outputs the fixed value set in the MOIFV bit to the MOSI bit. 0: MOSI output value equals final data from previous transfer 1: MOSI output value equals the value set in the MOIFV bit
4	MOIFV	0	R/W	MOSI Idle Fixed Value If the MOIFE bit is 1 in master mode, this module, according to MOIFV bit settings, determines the MOSI signal value during the SSL negation period (including the SSL retention period during a burst transfer). 0: MOSI Idle fixed value equals 0 1: MOSI Idle fixed value equals 1
3 to 1	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.

Bit	Bit Name	Initial Value	R/W	Description
0	SPLP	0	R/W	Loopback When the SPLP bit is set to 1, this module shuts off the path between the MISO pin and the shift register, and between the MOSI pin and the shift register, and connects (reverses) the input path and the output path for the shift register. 0: Normal mode 1: Loopback mode

20.3.4 Status Register (SPSR)

SPSR indicates the operating status.

Bit:	7	6	5	4	3	2	1	0
	SPRF	TEND	SPTEF	—	—	MODF	—	OVRF
Initial value:	0	1	1	0	0	0	0	0
R/W:	R	R	R	R	R	R/(W)*	R	R/(W)*

Note: * Only 0 can be written to clear the flag after reading 1.

Bit	Bit Name	Initial Value	R/W	Description
7	SPRF	0	R	Receive Buffer Full Flag Indicates that the number of receive data units in the receive buffer (SPRX) is equal to or greater than the receive buffer data triggering number specified in the buffer control register (SPBFCR). 0: The number of receive data units in the receive buffer is less than the receive buffer data triggering number. 1: The number of receive data units in the receive buffer is equal to or greater than the receive buffer data triggering number. [Clearing conditions] • The receive buffer data is read until the number of data units in the receive buffer becomes less than the specified receive buffer data triggering number. • Receive buffer data reset is enabled. • Power-on reset [Setting condition] • The number of data units in the receive buffer is equal to or greater than the specified receive buffer data triggering number.

Bit	Bit Name	Initial Value	R/W	Description
6	TEND	1	R	Transmit End This bit is set to 1 when transmission is completed, and this bit is 0 when transmission is not completed. [Clearing condition] • When transmit data are moved from the transmit register to the shift register. [Setting condition] • When the number of data units in the transmit buffer (SPTX) is zero when a serial transfer is completed.
5	SPTEF	1	R	Transmit Buffer Empty Flag Indicates that the number of transmit data units in the transmit buffer (SPTX) is equal to or less than the transmit buffer data triggering number specified in the buffer control register (SPBFCR). 0: The number of transmit data units in the transmit buffer is equal to or greater than the specified transmit buffer data triggering number. 1: The number of transmit data units in the transmit buffer is less than the specified transmit buffer data triggering number. [Clearing condition] • When data is written to the transmit buffer until the number of transmit data units in the transmit buffer exceeds the specified transmit buffer data triggering number. [Setting conditions] • When the number of transmit data units in the transmit buffer is less than the specified transmit buffer data triggering number. • When transmit buffer data reset is enabled. • Power-on reset
4, 3	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.

Bit	Bit Name	Initial Value	R/W	Description
2	MODF	0	R/(W)*	Mode Fault Error Flag Indicates the occurrence of a mode fault error. If the MODFEN bit is set to 1 when this module is in slave mode and the SSL pin is negated before the RSPCK cycle necessary for data transfer ends, this module detects a mode fault error. The active level of the SSL signal is determined by the SSL0P bit in the slave select polarity register (SSLP). [Clearing conditions] • SPSR is read when the MODF bit is 1, and then 0 is written to the MODF bit. • Power-on reset 0: No mode fault error occurred 1: A mode fault error occurred Note: This bit is valid only in SPI slave mode.
1	—	0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
0	OVRF	0	R/(W)*	Overrun Error Flag Indicates the occurrence of an overrun error. If a serial transfer ends when there is not enough space for receiving the specified length of data in the receive buffer (SPRX), this module detects an overrun error, and sets the OVRF bit to 1. [Clearing conditions] • SPSR is read when the OVRF bit is 1, and then 0 is written to the OVRF bit. • Power-on reset 0: No overrun error occurred 1: An overrun error occurred Note: This bit is valid only in SPI slave mode.

Note: * Only 0 can be written to clear the flag after reading 1.

20.3.5 Data Register (SPDR)

SPDR is a buffer that holds data for transmission and reception.

The transmit buffer (SPTX) and receive buffer (SPRX) are independent and are mapped to SPDR.

SPDR should be read or written to in byte, word, or longword units according to the access width specification bit (SPLW) in the data control register (SPDCR).

The bit length to be used is determined by the data length specification bits (SPB3 to SPB0) in the command register (SPCMD).

When data is written to SPDR, the data will be written to the transmit buffer from SPDR if the transmit buffer has a space equal to or more than the SPDR access width. If there is not enough space, data will not be written to the transmit buffer. Even if an attempt is made to write data to the buffer, the data is ignored.

When data is read from SPDR, receive data in the receive buffer will be read. If SPDR is read when there is no receive data in the receive buffer, the read value is undefined.

When SPDR is written to with the longword-, word-, or byte-access width, the transmit data should be written to the following bits. If data is written to the other bits, the data is not guaranteed.

- Longword: Bits 31 to 0
- Word: Bits 31 to 16
- Byte: Bits 31 to 24

When SPDR is read with the longword-, word-, or byte-access width, the receive data should be read from the following bits. If data is read from the other bits, the data is not guaranteed.

- Longword: Bits 31 to 0
- Word: Bits 31 to 16
- Byte: Bits 31 to 24

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SPD31	SPD30	SPD29	SPD28	SPD27	SPD26	SPD25	SPD24	SPD23	SPD22	SPD21	SPD20	SPD19	SPD18	SPD17	SPD16
Initial value:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPD15	SPD14	SPD13	SPD12	SPD11	SPD10	SPD9	SPD8	SPD7	SPD6	SPD5	SPD4	SPD3	SPD2	SPD1	SPD0
Initial value:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

20.3.6 Sequence Control Register (SPSCR)

SPSCR sets the sequence controlled method when this module operates in master mode. If the contents of SPSCR are changed while the MSTR and SPE bits in the control register (SPCR) are 1 with the function of this module enabled in master mode, the subsequent operation cannot be guaranteed.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SPSLN1	SPSLN0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 2	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.

Bit	Bit Name	Initial Value	R/W	Description
1	SPSLN1	0	R/W	Sequence Length Specification
0	SPSLN0	0	R/W	These bits specify a sequence length when this module in master mode performs sequential operations. This module in master mode changes command registers 0 to 3 (SPCMD0 to SPCMD3) to be referenced and the order in which they are referenced according to the sequence length that is set in the SPSLN1 and SPSLN0 bits. The relationship among the setting of bits SPSLN1 and SPSLN0, sequence length, and SPCMD0 to SPCMD3 referenced by this module is shown below. In slave mode, SPCMD0 is always referenced. 00: Sequence Length 1 — Referenced SPCMD # 0 → 0 → … 01: Sequence Length 2 — Referenced SPCMD # 0 → 1 → 0 → … 10: Sequence Length 3 — Referenced SPCMD # 0 → 1 → 2 → 0 → … 11: Sequence Length 4 — Referenced SPCMD # 0 → 1 → 2 → 3 → 0 → …

20.3.7 Sequence Status Register (SPSSR)

SPSSR indicates the sequence control status when this module operates in master mode.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SPCP1	SPCP0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 2	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
1	SPCP1	0	R	Command Pointer
0	SPCP0	0	R	During sequence control, these bits indicate one of the command registers 0 to 3 (SPCMD0 to SPCMD3) that is currently pointed to by the pointer. The relationship between the setting of SPCP1 and SPCP0 and SPCMD0 to SPCMD3 is shown below. For the sequence control, see section 20.4.8 (1) (c), Sequence Control. 00: SPCMD0 01: SPCMD1 10: SPCMD2 11: SPCMD3

20.3.8 Bit Rate Register (SPBR)

SPBR sets the bit rate in master mode. If the contents of SPBR are changed while the MSTR and SPE bits in the control register (SPCR) are 1 with the function of this module enabled in master mode, the subsequent operation cannot be guaranteed.

Bit:	7	6	5	4	3	2	1	0
	SPR7	SPR6	SPR5	SPR4	SPR3	SPR2	SPR1	SPR0
Initial value:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

When this module is used in slave mode, the bit rate depends on the bit rate of the input clock regardless of the settings of SPBR and BRDV.

The bit rate is determined by combinations of SPBR settings and the bit settings in the BRDV1 and BRDV0 bits in the command registers (SPCMD0 to SPCMD3). The equation for calculating the bit rate is given below. In the equation, n denotes an SPBR setting (0, 1, 2, …, 255), and N denotes bit settings in the bits BRDV1 and BRDV0 (0, 1, 2, 3).

$$\text{Bit rate} = \frac{f(B\phi)}{2 \times (n + 1) \times 2^{N}}$$

Table 20.3 shows examples of the relationship between the SPBR register and BRDV1 and BRDV0 bit settings.

Table 20.3 Relationship between SPBR and BRDV1 and BRDV0 Settings

			Bit Rate
SPBR (n)	**BRDV[1:0] (N)**	**Division Ratio**	**Bϕ = 100 MHz**
0	0	2	50.0 Mbps
1	0	4	25.0 Mbps
2	0	6	16.67 Mbps
3	0	8	12.50 Mbps
4	0	10	10.00 Mbps
5	0	12	8.33 Mbps
5	1	24	4.17 Mbps
5	2	48	2.08 Mbps
5	3	96	1042 kbps
255	3	4096	24.41 kbps

20.3.9 Data Control Register (SPDCR)

SPDCR selects the width to access SPDR from longword-, word-, and byte-width, and enables or disables dummy data transmission for the master mode operation.

If the contents of bits other than the TXDMY of SPDCR are changed while bit TEND in the status register (SPSR) indicates that transmission is not completed, the subsequent operation cannot be guaranteed.

Bit:	7	6	5	4	3	2	1	0
	TXDMY	SPLW1	SPLW0	—	—	—	—	—
Initial value:	0	0	1	0	0	0	0	0
R/W:	R/W	R/W	R/W	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	TXDMY	0	R/W	Dummy Data Transmission Enable Enables or disables dummy data transmission. When communication is performed with this bit set to 1, dummy data is transmitted from the MOSI pin and a serial communication can be performed even if there is no transmit data in the transmit buffer. Specifically, if there is no transmit data in the transmit buffer and this bit is set to 1, dummy data is transferred to the shift register. Data previously transmitted from the pin is used as dummy data. If this bit is set to 1 after the initialization and a transfer is performed, the transmitted dummy data is undefined. 0: Disables dummy data transmission. 1: Enables dummy data transmission. Note: This bit is valid only in the master mode.

Bit	Bit Name	Initial Value	R/W	Description
6	SPLW1	0	R/W	Access Width Specification
5	SPLW0	1	R/W	Specifies the width for accessing the data register (SPDR). If the length of data transferred to SPDR does not agree with these bit settings, operation is not guaranteed. 00: Setting prohibited 01: SPDR is accessed in bytes. 10: SPDR is accessed in words. 11: SPDR is accessed in longwords. Note: When burst transfer by the HPB-DMAC is used to write to the data register (SPDR), these bits should not be set to 10 or 11.
4 to 0	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.

20.3.10 Clock Delay Register (SPCKD)

SPCKD sets a period from the beginning of SSL signal assertion to RSPCK oscillation (RSPCK delay) when the SCKDEN bit in the command register (SPCMD) is 1. If the contents of SPCKD are changed while the MSTR and SPE bits in the control register (SPCR) are 1 with the function of this module enabled in master mode, the subsequent operation cannot be guaranteed.

When using this module in slave mode, set B'000 to SCKDL2 to SCKDL0.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	SCKDL2	SCKDL1	SCKDL0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
2	SCKDL2	0	R/W	RSPCK Delay Setting
1	SCKDL1	0	R/W	These bits set an RSPCK delay value when the SCKDEN bit in SPCMD is 1.
0	SCKDL0	0	R/W	The relationship between the setting of SCKDL2 to SCKDL0 and the RSPCK delay value is shown below. 000: 1 RSPCK 001: 2 RSPCK 010: 3 RSPCK 011: 4 RSPCK 100: 5 RSPCK 101: 6 RSPCK 110: 7 RSPCK 111: 8 RSPCK

20.3.11 Slave Select Negation Delay Register (SSLND)

SSLND sets a period (SSL negation delay) from the transmission of a final RSPCK edge to the negation of the SSL signal during a serial transfer by this module in master mode. If the contents of SSLND are changed while the MSTR and SPE bits in the control register (SPCR) are 1 with the function of this module enabled in master mode, the subsequent operation cannot be guaranteed.

When using this module in slave mode, set B'000 to SLNDL2 to SLNDL0.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	SLNDL2	SLNDL1	SLNDL0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
2	SLNDL2	0	R/W	SSL Negation Delay Setting
1	SLNDL1	0	R/W	These bits set an SSL negation delay when the SLNDEN bit in SPCMD is 1.
0	SLNDL0	0	R/W	The relationship between the setting of SLNDL2 to SLNDL0 and the SSL negation delay value is shown below. 000: 1 RSPCK 001: 2 RSPCK 010: 3 RSPCK 011: 4 RSPCK 100: 5 RSPCK 101: 6 RSPCK 110: 7 RSPCK 111: 8 RSPCK

20.3.12 Next-Access Delay Register (SPND)

SPND sets a non-active period (next-access delay) after termination of a serial transfer when the SPNDEN bit in the command register (SPCMD) is 1. If the contents of SPND are changed while the MSTR and SPE bits in the control register (SPCR) are 1 with the function of this module enabled in master mode, the subsequent operation cannot be guaranteed.

When using this module in slave mode, set B'000 to SPNDL2 to SPNDL0.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	SPNDL2	SPNDL1	SPNDL0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
2	SPNDL2	0	R/W	Next-Access Delay Setting
1	SPNDL1	0	R/W	These bits set a next-access delay when the SPNDEN bit in SPCMD is 1.
0	SPNDL0	0	R/W	The relationship between the setting of SPNDL2 to SPNDL0 and the next-access delay value is shown below. 000: 1 RSPCK + 2 Bϕ 001: 2 RSPCK + 2 Bϕ 010: 3 RSPCK + 2 Bϕ 011: 4 RSPCK + 2 Bϕ 100: 5 RSPCK + 2 Bϕ 101: 6 RSPCK + 2 Bϕ 110: 7 RSPCK + 2 Bϕ 111: 8 RSPCK + 2 Bϕ

20.3.13 Command Register (SPCMD)

Each channel has four command registers (SPCMD0 to SPCMD3). SPCMD0 to SPCMD3 are used to set a transfer format for master mode operation. Some of the bits in SPCMD0 are used to set a transfer mode for slave mode operation. In master mode, this module sequentially references SPCMD0 to SPCMD3 according to the settings in bits SPSLN1 and SPSLN0 in the sequence control register (SPSCR), and executes the serial transfer that is set in the referenced SPCMD.

While bit TEND in the status register (SPSR) indicates that transmission is not completed, correct operation of this module cannot be guaranteed if SPCMD is changed that is referred by this module. SPCMD referenced by this module in master mode can be checked by means of bits SPCP1 and SPCP0 in the sequence status register (SPSSR). When the function of this module in slave mode is enabled, operation cannot be guaranteed if the value set in SPCMD0 is changed.

Bit:	15	14	13	12	11	10	9	8
	SCK DEN	SLN DEN	SPN DEN	LSBF	SPB3	SPB2	SPB1	SPB0
Initial value:	0	0	0	0	0	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	7	6	5	4	3	2	1	0
	SSLKP	—	—	—	BRDV1	BRDV0	CPOL	CPHA
Initial value:	0	0	0	0	1	1	0	1
R/W:	R/W	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	SCKDEN	0	R/W	RSPCK Delay Setting Enable Sets the period from the point this module in master mode activates the SSL signal until the RSPCK starts oscillation (RSPCK delay). If the SCKDEN bit is 0, this module sets the RSPCK delay to 1 RSPCK. If the SCKDEN bit is 1, this module starts the oscillation of RSPCK at an RSPCK delay in compliance with the clock delay register (SPCKD) settings. To use this module in slave mode, the SCKDEN bit should be set to 0. 0: An RSPCK delay of 1 RSPCK 1: An RSPCK delay equal to SPCKD settings.

Bit	Bit Name	Initial Value	R/W	Description
14	SLNDEN	0	R/W	SSL Negation Delay Setting Enable Sets the period from the point this module in master mode stops RSPCK oscillation until this module sets the SSL signal inactive (SSL negation delay). If the SLNDEN bit is 0, this module sets the SSL negation delay to 1 RSPCK. If the SLNDEN bit is 1, this module negates the SSL signal at an SSL negation delay in compliance with the slave select negation delay register (SSLND) settings. To use this module in slave mode, the SLNDEN bit should be set to 0. 0: An SSL negation delay of 1 RSPCK 1: An SSL negation delay equal to SSLND settings.
13	SPNDEN	0	R/W	Next-Access Delay Enable Sets the period from the point this module in master mode terminates a serial transfer and sets the SSL signal inactive until this module enables the SSL signal assertion for the next access (next-access delay). If the SPNDEN bit is 0, this module sets the next-access delay to 1 RSPCK + 2Bϕ. If the SPNDEN bit is 1, this module inserts a next-access delay in compliance with the next-access delay register (SPND) settings. To use this module in slave mode, the SPNDEN bit should be set to 0. 0: A next-access delay of 1 RSPCK + 2 Bϕ 1: A next-access delay equal to SPND settings.

Bit	Bit Name	Initial Value	R/W	Description
12	LSBF	0	R/W	LSB First Sets the data format in master mode or slave mode to MSB first or LSB first. 0: MSB first 1: LSB first
11	SPB3	0	R/W	Data Length Setting These bits set a transfer data length in master mode or slave mode. 0100 to 0111: 8 bits 1111: 16 bits 0010, 0011: 32 bits Others: Setting prohibited
10	SPB2	1	R/W	
9	SPB1	1	R/W	
8	SPB0	1	R/W	
7	SSLKP	0	R/W	SSL Signal Level Keeping When this module in master mode performs a serial transfer, this bit specifies whether the SSL signal level for the current command is to be kept or negated between the SSL negation timing associated with the current command and the SSL assertion timing associated with the next command. To use this module in slave mode, the SSLKP bit should be set to 0. 0: Negates all SSL signals upon completion of transfer. 1: Keeps the SSL signal level from the end of the transfer until the beginning of the next access.

Bit	Bit Name	Initial Value	R/W	Description
6 to 4	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
3	BRDV1	1	R/W	Bit Rate Division Setting
2	BRDV0	1	R/W	These bits are used to determine the bit rate. A bit rate is determined by combinations of bits BRDV1 and BRDV 0 and the settings in the bit rate register (SPBR) (see section 20.3.8, Bit Rate Register (SPBR)). The settings in SPBR determine the base bit rate. The settings in bits BRDV1 and BRDV0 are used to select a bit rate which is obtained by dividing the base bit rate by 1, 2, 4, or 8. In the bits SPCMD0 to SPCMD3, different BRDV1 and BRDV0 settings can be specified. This permits the execution of serial transfers at a different bit rate for each command. 00: Select the base bit rate 01: Select the base bit rate divided by 2 10: Select the base bit rate divided by 4 11: Select the base bit rate divided by 8

Bit	Bit Name	Initial Value	R/W	Description
1	CPOL	0	R/W	RSPCK Polarity Setting Sets an RSPCK polarity in master or slave mode. When data communication is performed between the Renesas serial peripheral interface module and the other modules, the same RSPCK polarity should be set for both modules. 0: RSPCK = 0 when idle 1: RSPCK = 1 when idle
0	CPHA	1	R/W	RSPCK Phase Setting Sets an RSPCK phase in master or slave mode. When data communication is performed between the Renesas serial peripheral interface module and the other modules, the same RSPCK phase should be set for both modules. 0: Data sampling on odd edge, data variation on even edge 1: Data variation on odd edge, data sampling on even edge

20.3.14 Buffer Control Register (SPBFCR)

SPBFCR resets the number of data units in the transmit buffer (SPTX) or receive buffer (SPRX) and sets the number of triggering data units.

Bit:	7	6	5	4	3	2	1	0
	TXRST	RXRST	TXTRG[1:0]		—	RXTRG[2:0]		
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	TXRST	0	R/W	Transmit Buffer Data Reset Resets the transmit buffer to an empty state. Transmit data in the transmit buffer becomes invalid when this bit is set to 1. 0: Disables the reset operation*. 1: Enables the reset operation Note: The reset operation is performed after a power-on reset.
6	RXRST	0	R/W	Receive Buffer Data Reset Resets the receive buffer to an empty state. Receive data in the receive buffer becomes invalid when this bit is set to 1. 0: Disables the reset operation*. 1: Enables the reset operation Note: The reset operation is performed after a power-on reset.

Bit	Bit Name	Initial Value	R/W	Description
5, 4	TXTRG[1:0]	00	R/W	Transmit Buffer Data Triggering Number Specifies the timing at which the transmit buffer empty state is determined, that is when the SPTEF flag in the status register is set. When the number of bytes of data in the transmit buffer (SPTX) is equal to or less than the specified triggering number, the SPTEF flag is set to 1. 00: 7 bytes (1)* 01: 6 bytes (2)* 10: 4 bytes (4)* 11: 0 bytes (8)* Note: The value in the parenthesis shows the number of available bytes in the transmit buffer (SPTX).
3	—	0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
2 to 0	RXTRG[2:0]	000	R/W	Receive Buffer Data Triggering Number Specifies the timing at which the receive buffer full state is determined, that is when the SPRF flag in the status register is set. When the number of bytes of data in the receive buffer (SPRX) is equal to or greater than the specified triggering number, the SPRF flag is set to 1. 000: 1 byte (31)* 001: 2 bytes (30)* 010: 4 bytes (28)* 011: 8 bytes (24)* 100: 16 bytes (16)* 101: 24 bytes (8)* 110: 32 bytes (0)* 111: 5 bytes (27)* Note: * The value in the parenthesis shows the number of available bytes in the receive buffer (SPRX).

20.3.15 Buffer Data Count Setting Register (SPBFDR)

SPBFDR indicates the number of data units stored in the transmit buffer (SPTX) and receive buffer (SPRX). The upper eight bits indicate the number of transmit data units in SPTX and the lower eight bits indicate the number of receive data units in SPRX.

Bit:	15	14	13	12	11	10	9	8
	—	—	—	—	T[3:0]			
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
Bit:	7	6	5	4	3	2	1	0
	—	—	R[5:0]					
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
11 to 8	T[3:0]	0000	R	Indicates the number of bytes of data to be transmitted in SPTX. B'0000 indicates that SPTX is empty. B'1000 indicates that SPTX is full.
7, 6	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
5 to 0	R[5:0]	000000	R	Shows the number of bytes of received data in SPRX. B'000000 indicates that SPRX is empty. B'100000 indicates that SPRX is full.

20.4 Operation

In this section, the serial transfer period means a period from the beginning of driving valid data to the fetching of the final valid data.

20.4.1 Overview of Operations

This module is capable of serial transfers in slave mode and master mode. A particular mode of this module can be selected by using the MSTR bit in the control register (SPCR). Table 20.4 gives the relationship between the modes and SPCR settings, and a description of each mode.

Table 20.4 Relationship between Modes and SPCR and Description of Each Mode

Mode	Slave (SPI Operation)	Master (SPI Operation)
MSTR bit setting	0	1
MODFEN bit setting	0 or 1	0
RSPCK signal	Input	Output
MOSI signal	Input	Output
MISO signal	Output/Hi-Z	Input
SSL signal	Input	Output
SSL polarity modification function	Supported	Supported
Transfer rate	Up to Bϕ/8	Up to Bϕ/2
Clock source	RSPCK input	On-chip baud rate generator
Clock polarity	Two	Two
Clock phase	Two	Two
First transfer bit	MSB/LSB	MSB/LSB
Transfer data length	8 to 32 bits	8 to 32 bits
Burst transfer	Possible (CPHA = 1)	Possible (CPHA = 0,1)
RSPCK delay control	Not supported	Supported
SSL negation delay control	Not supported	Supported
Next-access delay control	Not supported	Supported
Transfer activation method	SSL input active or RSPCK oscillation	Transmit buffer is written when SPE = 1
Sequence control	Not supported	Supported

Mode	Slave (SPI Operation)	Master (SPI Operation)
Transmit buffer empty detection	Supported	Supported
Receive buffer full detection	Supported	Supported
Overrun error detection	Supported	Not Supported
Mode fault error detection	Supported (MODFEN = 1)	Not supported

20.4.2 Pin Control

According to the MSTR bit in the control register (SPCR), this module can automatically switch pin directions and output modes. Table 20.5 shows the relationship between pin states and bit settings.

Table 20.5 Relationship between Pin States and Bit Settings

Mode	Pin	Pin State
Master mode (SPI operation) (MSTR = 1)	RSPCK	CMOS output
	SSL	CMOS output
	MOSI	CMOS output
	MISO	Input
Slave mode (SPI operation) (MSTR = 0)	RSPCK	Input
	SSL	Input
	MOSI	Input
	MISO*	CMOS output/Hi-Z

Note: When SSL is at the non-active level or the SPE bit in SPCR is cleared to 0, the pin state is Hi-Z.

This module in master mode (SPI operation) determines MOSI signal values during the SSL negation period (including the SSL retention period during a burst transfer) according to MOIFE and MOIFV bit settings in SPPCR, as shown in table 20.6.

Table 20.6 MOSI Signal Value Determination during SSL Negation Period

MOIFE	MOIFV	MOSI Signal Value during SSL Negation Period
0	0, 1	Final data from previous transfer
1	0	Always 0
1	1	Always 1

20.4.3 System Configuration Example

(1) Master/Slave (with This LSI Acting as Master)

Figure 20.2 shows a master/slave system configuration example when this LSI is used as a master. In master/slave configuration, the SSL output of this LSI (master) is not used. The SSL input of the slave is fixed to the low level, and the slave is always maintained in a selected state. In the transfer format corresponding to the case where the CPHA bit in the control register (SPCR) is 0, there are slave devices for which the SSL signal cannot be fixed to the active level. In situations where the SSL signal cannot be fixed, the SSL output of this LSI should be connected to the SSL input of the slave device.

This LSI (master) always drives the RSPCK and MOSI. The slave always drives the MISO.

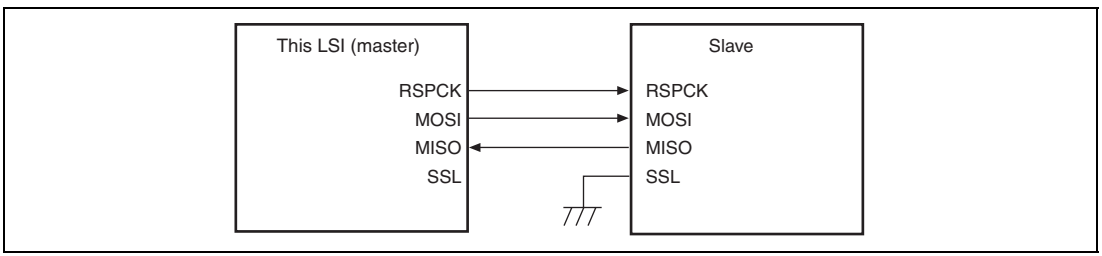


Figure 20.2 Master/Slave Configuration Example (This LSI = Master)

(2) Master/Slave (with This LSI Acting as Slave)

Figure 20.3 shows a master/slave system configuration example when this LSI is used as a slave. When this LSI is to operate as a slave, the SSL pin is used as SSL input. The master always drives the RSPCK and MOSI. This LSI (slave) always drives the MISO. When SSL is at the non-active level, the pin state is Hi-Z.

In the slave configuration in which the CPHA bit in the command register (SPCMD) is set to 1, the SSL input of this LSI (slave) is fixed to the 0 level, this LSI (slave) is always maintained in a selected state, and in this manner it is possible to execute serial transfer (figure 20.4).

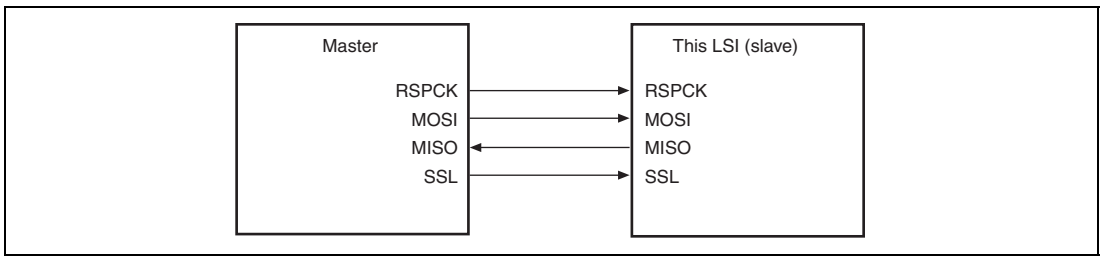


Figure 20.3 Master/Slave Configuration Example (This LSI = Slave)

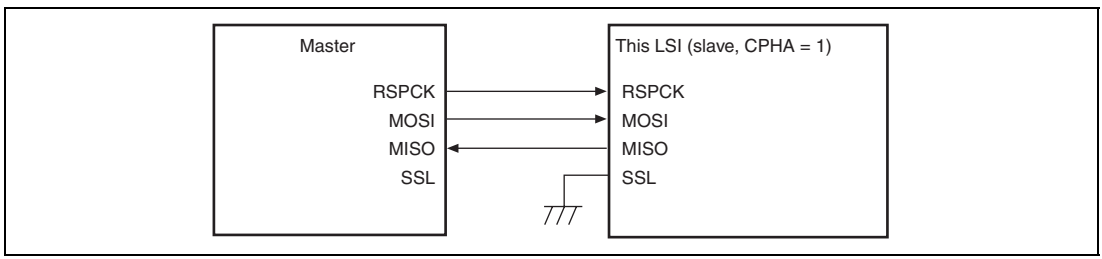


Figure 20.4 Master/Slave Configuration Example (This LSI = Slave, CPHA = 1)

(3) Master/Multi-Slave (with This LSI Acting as Slave)

Figure 20.5 shows a master/multi-slave system configuration example when this LSI is used as a slave. In the example of figure 20.5, the system is comprised of an master and two LSIs (slave X and slave Y).

The RSPCK and MOSI outputs of the master are connected to the RSPCK and MOSI inputs of the LSIs (slave X and slave Y). The MISO outputs of the LSIs (slave X and slave Y) are all connected to the MISO input of the master. SSLX and SSLY outputs of the master are connected to the SSL inputs of the LSIs (slave X and slave Y), respectively.

The master always drives RSPCK, MOSI, SSLX, and SSLY. Of the LSIs (slave X and slave Y), the slave that receives low level input into the SSL0 input drives MISO.

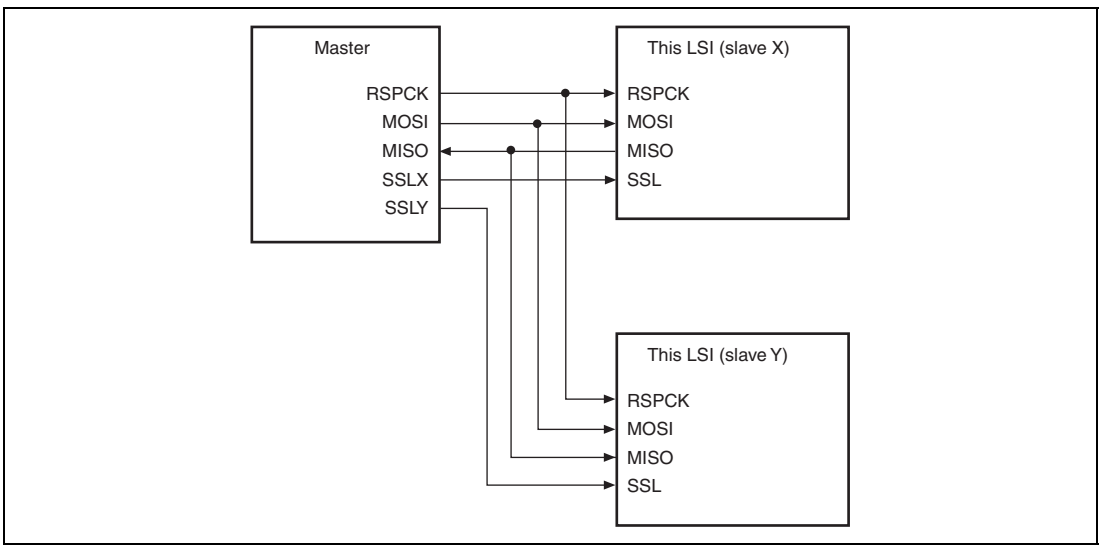


Figure 20.5 Master/Multi-Slave Configuration Example (This LSI = Slave)

20.4.4 Transfer Format

(1) CPHA = 0

Figure 20.6 shows a sample transfer format for the serial transfer of 8-bit data when the CPHA bit in the command register (SPCMD) is 0. In figure 20.6, RSPCK (CPOL = 0) indicates the RSPCK signal waveform when the CPOL bit in SPCMD is 0; RSPCK (CPOL = 1) indicates the RSPCK signal waveform when the CPOL bit is 1. The sampling timing represents the timing at which this module fetches serial transfer data into the shift register. The input/output directions of the signals depend on the settings of this module. For details, see section 20.4.2, Pin Control.

When the CPHA bit is 0, the driving of valid data to the MOSI and MISO signals commences at an SSL signal assertion timing. The first RSPCK signal change timing that occurs after the SSL signal assertion becomes the first transfer data fetching timing. After this timing, data is sampled at every 1 RSPCK cycle. The change timing for MOSI and MISO signals is always 1/2 RSPCK cycle after the transfer data fetch timing. The settings in the CPOL bit do not affect the RSPCK signal operation timing; they only affect the signal polarity.

t1 denotes a period from an SSL signal assertion to RSPCK oscillation (RSPCK delay). t2 denotes a period from the cessation of RSPCK oscillation to an SSL signal negation (SSL negation delay). t3 denotes a period in which SSL signal assertion is suppressed for the next transfer after the end of serial transfer (next-access delay). t1, t2, and t3 are controlled by a master device running on the system. For a description of t1, t2, and t3 when this module is in master mode, see section 20.4.3 (1), Master/Slave (with This LSI Acting as Master).

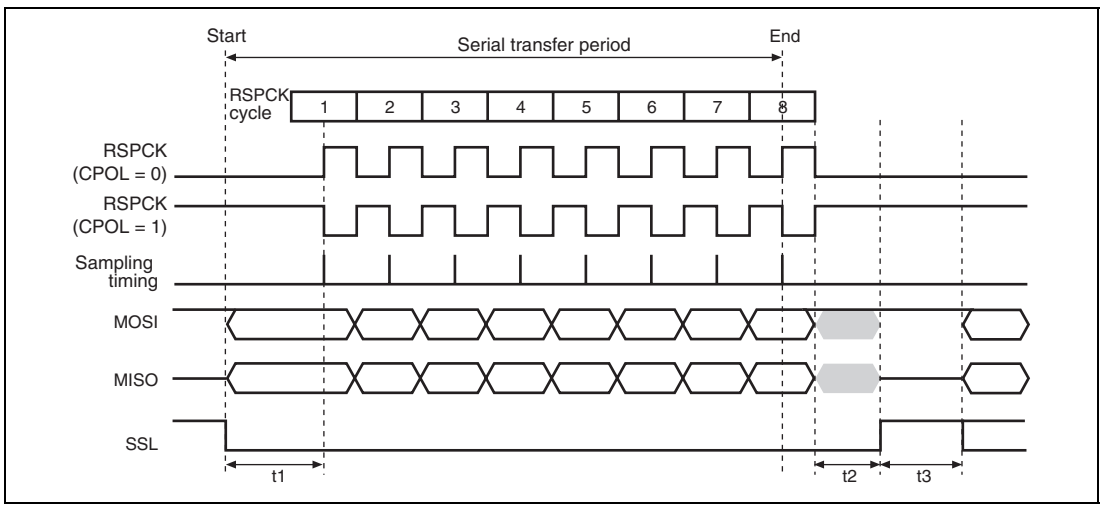


Figure 20.6 Transfer Format (CPHA = 0)

(2) CPHA = 1

Figure 20.7 shows a sample transfer format for the serial transfer of 8-bit data when the CPHA bit in the command register (SPCMD) is 1. In figure 20.7, RSPCK (CPOL = 0) indicates the RSPCK signal waveform when the CPOL bit in SPCMD is 0; RSPCK (CPOL = 1) indicates the RSPCK signal waveform when the CPOL bit is 1. The sampling timing represents the timing at which this module fetches serial transfer data into the shift register. The input/output directions of the signals depend on the modes (master or slave). For details, see section 20.4.2, Pin Control.

When the CPHA bit is 1, the driving of invalid data to the MOSI and MISO signals commences at an SSL signal assertion timing. The driving of valid data to the MOSI and MISO signals commences at the first RSPCK signal change timing that occurs after the SSL signal assertion. After this timing, data is updated at every 1 RSPCK cycle. The transfer data fetch timing is always 1/2 RSPCK cycle after the data update timing. The settings in the CPOL bit do not affect the RSPCK signal operation timing; they only affect the signal polarity.

t1, t2, and t3 are the same as those in the case of CPHA = 0. For a description of t1, t2, and t3 when this module is in master mode, see section 20.4.3 (1), Master/Slave (with This LSI Acting as Master).

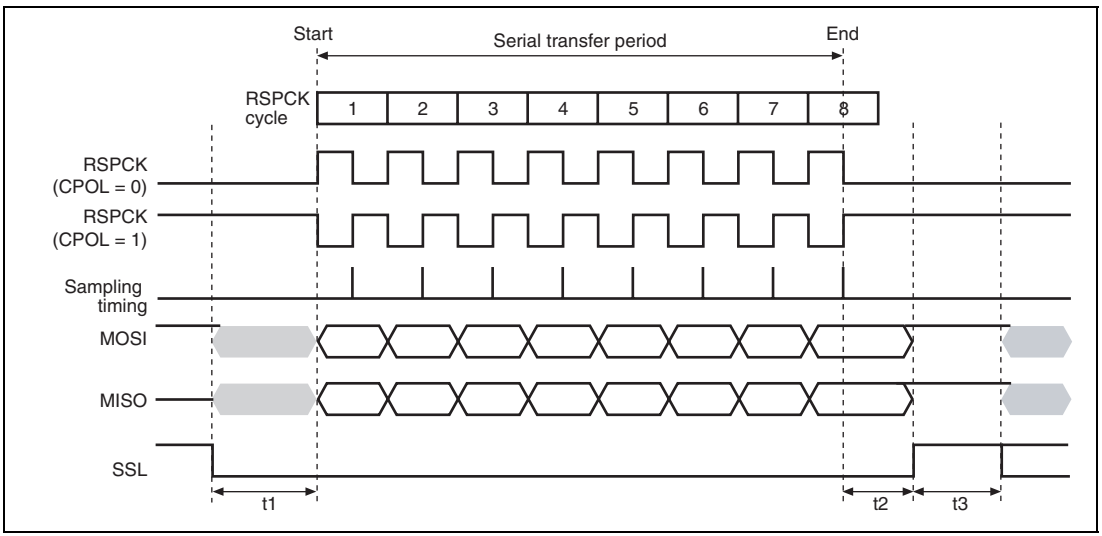


Figure 20.7 Transfer Format (CPHA = 1)

20.4.5 Data Format

The data format depends on the settings in the command register (SPCMD). Irrespective of MSB/LSB first, this module treats the range from the LSB of the data register (SPDR) to the assigned data length as transfer data.

(1) MSB First Transfer (32-Bit Data)

Figure 20.8 shows the operation of the transmit buffer (SPTX) and the shift register when this module performs a 32-bit data length MSB-first data transfer.

The CPU or direct memory access controller writes T31 to T00 to the transmit buffer of SPDR. If the shift register is empty, this module copies the data in the transmit buffer to the shift register, and fully populates the shift register. When serial transfer starts, this module outputs data from the MSB (bit 31) of the shift register, and shifts in the data from the LSB (bit 0) of the shift register. When the RSPCK cycle required for the serial transfer of 32 bits has passed, data R31 to R00 is stored in the shift register. In this state, this module copies the data from the shift register to the receive buffer, and empties the shift register. If the receive buffer does not have a space for the receive data length after the receive data has been copied from the shift register to the receive buffer, another serial transfer will not be started. In order to start another serial transfer, data for the receive data length should be read from the receive buffer to secure the necessary space in the receive buffer.

If another serial transfer is started before the CPU or direct memory access controller writes to the transmit buffer, received data R31 to R00 is shifted out from the shift register.

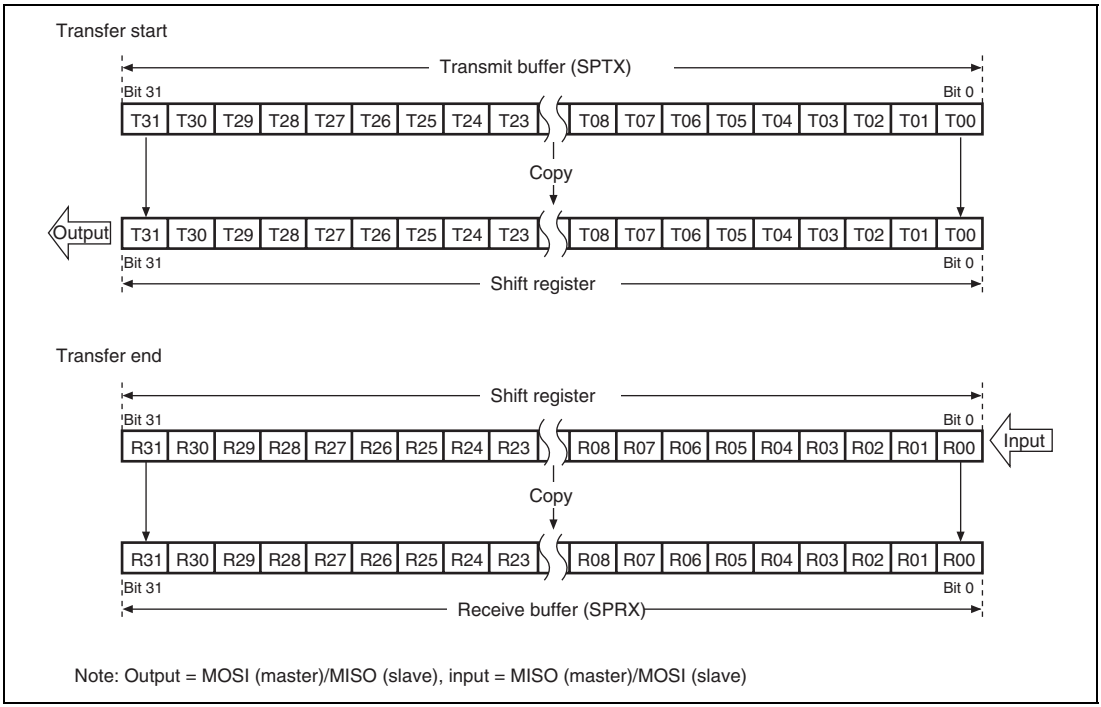


Figure 20.8 MSB First Transfer (32-Bit Data)

(2) MSB First Transfer (16-Bit Data)

Figure 20.9 shows the operation of the transmit buffer (SPTX) and the shift register when this module performs a 16-bit data length MSB-first data transfer.

The CPU or direct memory access controller writes T15 to T00 to the transmit buffer. If the shift register is empty, this module copies the data in the transmit buffer to the shift register, and fully populates the shift register. When serial transfer starts, this module outputs data from bit 15 of the shift register, and shifts in the data from the LSB (bit 0) of the shift register. When the RSPCK cycle required for the serial transfer of 16 bits has passed, received data R15 to R00 is stored in bits 15 to 0 of the shift register. After completion of the serial transfer, data that existed before the transfer is retained in bits 31 to 16 in the shift register. In this state, this module copies the data from the shift register to the receive buffer, and empties the shift register. If the receive buffer does not have a space for the receive data length after receive data has been copied from the shift register to the receive buffer, another serial transfer will not be started. In order to start another serial transfer, data for the receive data length should be read from the receive buffer to secure the necessary space in the receive buffer.

If another serial transfer is started before the CPU or direct memory access controller writes to the transmit buffer, received data R15 to R00 is shifted out from the shift register.

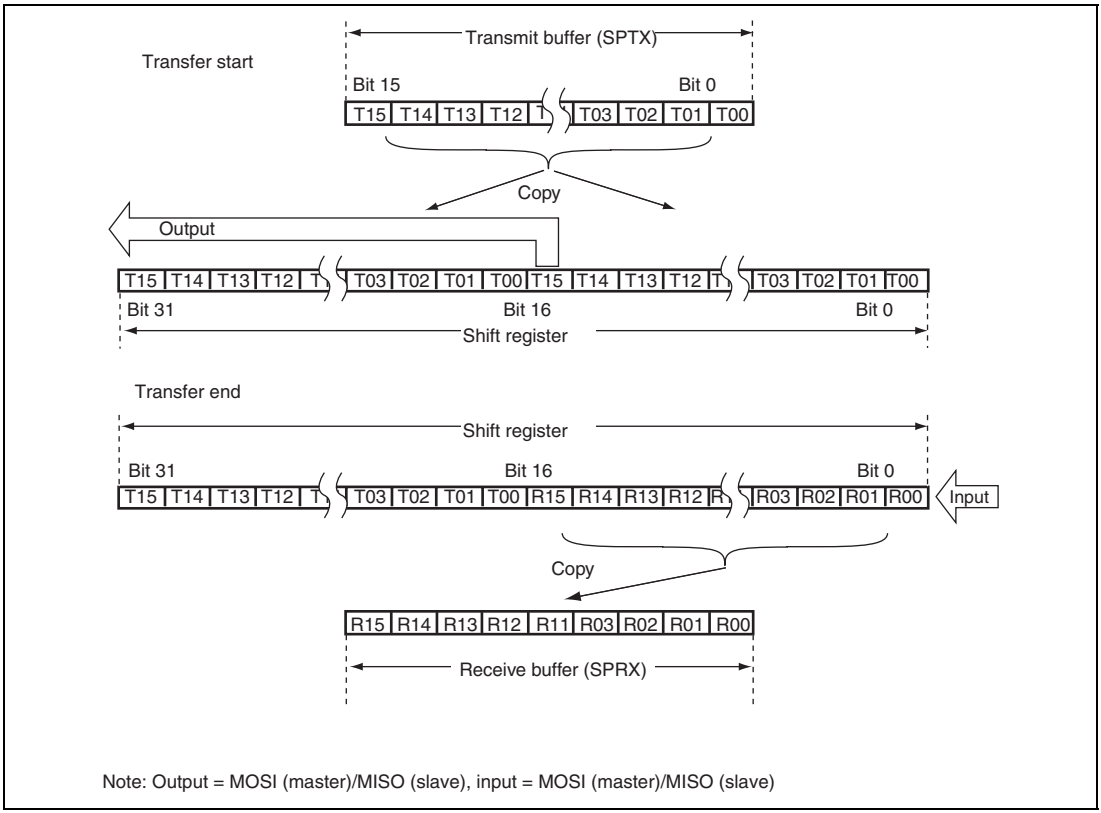


Figure 20.9 MSB First Transfer (16-Bit Data)

(3) MSB First Transfer (8-Bit Data)

Figure 20.10 shows the operation of the transmit buffer (SPDR) and the shift register when this module performs an 8-bit data length MSB-first data transfer.

The CPU or direct memory access controller writes T07 to T00 to the transmit buffer. If the shift register is empty, this module copies the data in the transmit buffer to the shift register, and fully populates the shift register. When serial transfer starts, this module outputs data from bit 7 of the shift register, and shifts in the data from the LSB (bit 0) of the shift register. When the RSPCK cycle required for the serial transfer of 8 bits has passed, received data R07 to R00 is stored in bits 7 to 0 of the shift register. After completion of the serial transfer, data that existed before the transfer is retained in bits 31 to 8 in the shift register. In this state, this module copies the data from the shift register to the receive buffer, and empties the shift register. If the receive buffer does not have a space for the receive data length after receive data has been copied from the shift register to the receive buffer, another serial transfer will not be started. In order to start another serial transfer, data for the receive data length should be read from the receive buffer to secure the necessary area in the receive buffer.

If another serial transfer is started before the CPU or direct memory access controller writes to the transmit buffer, received data R07 to R00 is shifted out from the shift register.

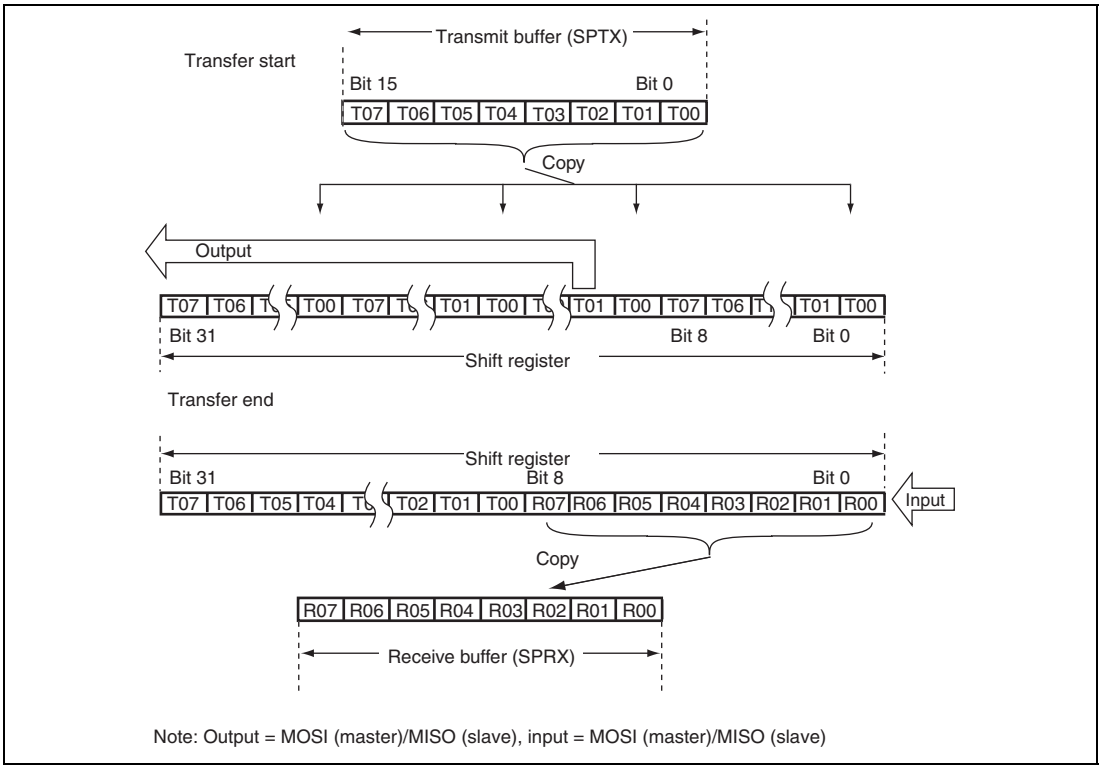


Figure 20.10 MSB First Transfer (8-Bit Data)

(4) LSB First Transfer (32-Bit Data)

Figure 20.11 shows the operation of the transmit buffer (SPTX) and the shift register when this module performs a 32-bit data length LSB-first data transfer.

The CPU or direct memory access controller writes T31 to T00 to the transmit buffer. If the shift register is empty, this module reverses the order of the bits of the data in the transmit buffer, copies it to the shift register, and fully populates the shift register. When serial transfer starts, this module outputs data from the MSB (bit 31) of the shift register, and shifts in the data from the LSB (bit 0) of the shift register. When the RSPCK cycle required for the serial transfer of 32 bits has passed, data R00 to R31 is stored in the shift register. In this state, this module copies the data, in which the order of the bits is reversed, from the shift register to the receive buffer, and empties the shift register.

If the receive buffer does not have a space for the receive data length after receive data has been copied from the shift register to the receive buffer, another serial transfer will not be started. In order to start another serial transfer, data for the receive data length should be read from the receive buffer to secure the necessary space in the receive buffer.

If another serial transfer is started before the CPU or direct memory access controller writes to the transmit buffer of the SPDR, received data R00 to R31 is shifted out from the shift register.

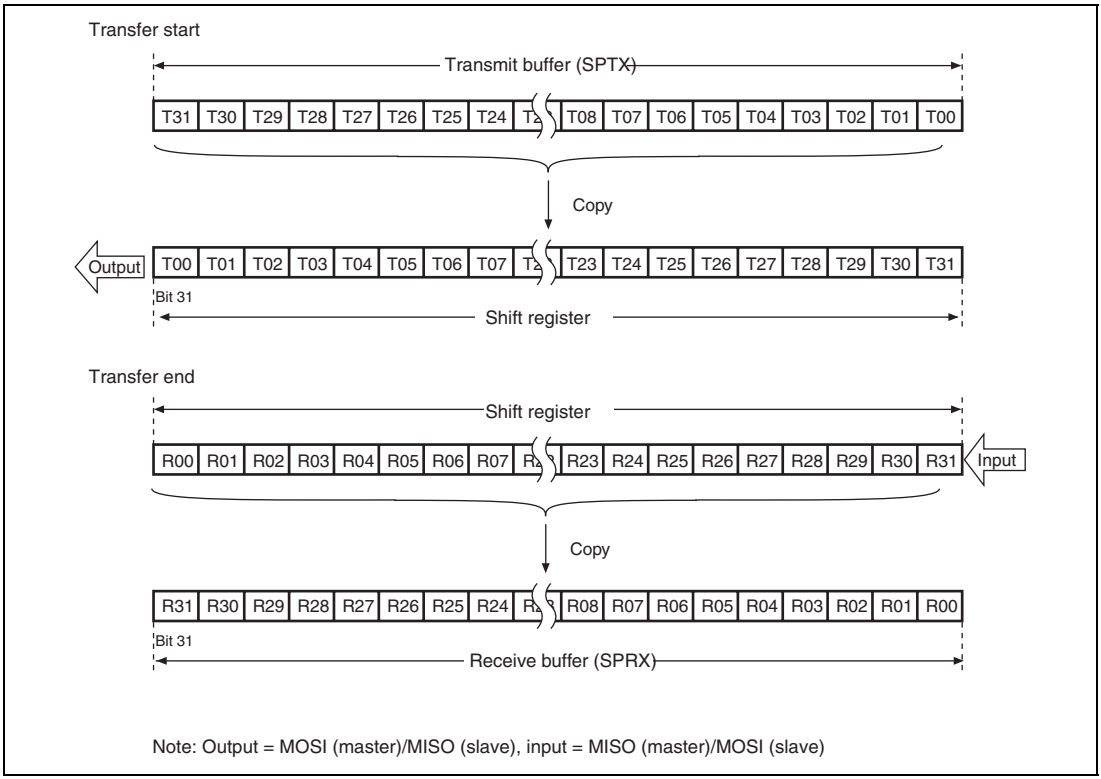


Figure 20.11 LSB First Transfer (32-Bit Data)

(5) LSB First Transfer (16-Bit Data)

Figure 20.12 shows the operation of the transmit buffer (SPTX) and the shift register when this module performs a 16-bit data length LSB-first data transfer.

The CPU or direct memory access controller writes T15 to T00 to the transmit buffer. If the shift register is empty, this module reverses the order of the bits of the data in the transmit buffer, copies it to the shift register, and fully populates the shift register. When serial transfer starts, this module outputs data from the MSB (bit 31) of the shift register, and shifts in the data from bit 16 of the shift register. When the RSPCK cycle required for the serial transfer of 16 bits has passed, received data R00 to R15 is stored in bits 31 to 16 of the shift register. After completion of the serial transfer, data that existed before the transfer is retained in bits 15 to 0 of the shift register. In this state, this module copies the data, in which the order of the bits is reversed, from the shift register to the receive buffer of SPDR, and empties the shift register.

If the receive buffer does not have a space for the receive data length after receive data has been copied from the shift register to the receive buffer, another serial transfer will not be started. In order to start another serial transfer, data for the receive data length should be read from the receive buffer to secure the necessary space in the receive buffer.

If another serial transfer is started before the CPU or direct memory access controller writes to the transmit buffer of SPDR, received data R00 to R15 is shifted out from the shift register.

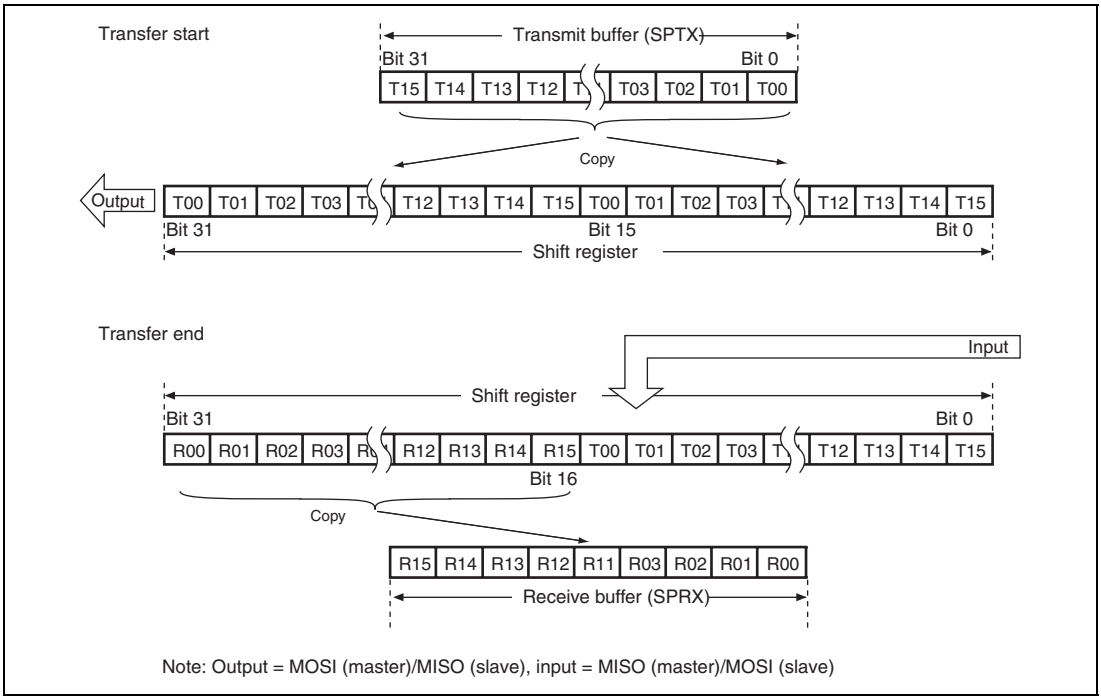


Figure 20.12 LSB First Transfer (16-Bit Data)

(6) LSB First Transfer (8-Bit Data)

Figure 20.13 shows the operation of the transmit buffer (SPTX) and the shift register when this module performs an 8-bit data length LSB-first data transfer.

The CPU or direct memory access controller writes T07 to T00 to the transmit buffer. If the shift register is empty, this module reverses the order of the bits of the data in the transmit buffer, copies it to the shift register, and fully populates the shift register. When serial transfer starts, this module outputs data from the MSB (bit 31) of the shift register, and shifts in the data from bit 24 of the shift register. When the RSPCK cycle required for the serial transfer of 8 bits has passed, received data R00 to R07 is stored in bits 31 to 24 of the shift register. After completion of the serial transfer, data that existed before the transfer is retained in bits 23 to 0 of the shift register. In this state, this module copies the data, in which the order of the bits is reversed, from the shift register to the receive buffer of SPDR, and empties the shift register.

If the receive buffer does not have a space for the receive data length after the receive data has been copied from the shift register to the receive buffer, another serial transfer will not be started. In order to start another serial transfer, data for the receive data length should be read from the receive buffer to secure the necessary space in the receive buffer.

If another serial transfer is started before the CPU or direct memory access controller writes to the transmit buffer of SPDR, received data R00 to R07 is shifted out from the shift register.

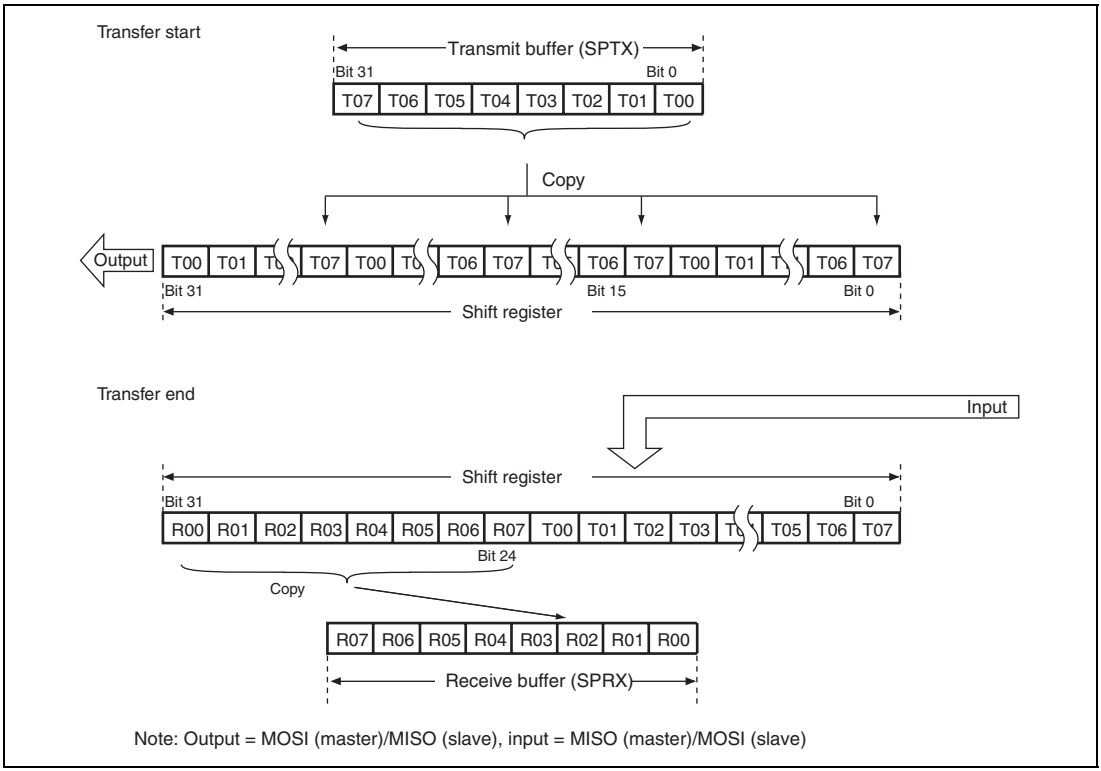


Figure 20.13 LSB First Transfer (8-Bit Data)

20.4.6 Error Detection

In the normal serial transfer, the data written from the data register (SPDR) to the transmit buffer is serially transmitted, and the serially received data can be read from the receive buffer of SPDR. If access is made to SPDR, depending on the status of the transmit buffer/receive buffer or the status at the beginning or end of serial transfer, in some cases non-normal transfers can be executed.

If a non-normal transfer operation occurs, this module detects the event as an overrun error or a mode fault error. Table 20.7 shows the relationship between non-normal transfer operations and the error detection function.

Table 20.7 Relationship between Non-Normal Transfer Operations and Error Detection Function

	Occurrence Condition	Operation	Error Detection
A	SPDR is written when the transmit buffer is full.	Missing write data.	None
B	Serial transfer is started in slave mode when transmit data is still not loaded on the shift register.	Data received in previous serial transfer is serially transmitted.	None
C	SPDR is read when the receive buffer is empty.	The output data is undefined.	None
D	Serial transfer terminates when the receive buffer is full.	Missing serial receive data.	Overrun error (only in slave mode)
E	The SSL input signal is negated during serial transfer in slave mode.	Serial transfer suspended. Missing send/receive data. Operation disabled.	Mode fault error

On operation A shown in table 20.7, this module does not detect an error. Whether SPDR can be written to or not can be checked using the T[3:0] bits in the buffer data count setting register (SPBFDR).

Likewise, this module does not detect an error on operation B. In a serial transfer that was started before the shift register was updated, this module sends the data that was received in the previous serial transfer, and does not treat the operation indicated in B as an error. Note that the received data from the previous serial transfer is retained in the receive buffer of SPDR, thus it can be correctly read.

Similarly, this module does not detect an error on operation C. To prevent extraneous data from being read, the number of receive data units stored in the receive buffer should be read from the R[5:0] bits in the buffer data count setting register (SPBFDR).

An overrun error shown in D is described in section 20.4.6 (1), Overrun Error. A mode fault error shown in E is described in section 20.4.6 (2), Mode Fault Error.

(1) Overrun Error

If serial transfer ends when the receive buffer of the data register (SPDR) is full, this module detects an overrun error, and sets the OVRF bit in SPSR to 1. When the OVRF bit is 1, this module does not copy data from the shift register to the receive buffer so that the data prior to the occurrence of the error is retained in the receive buffer. To reset the OVRF bit in SPSR to 0, either perform a power-on reset, or write a 0 to the OVRF bit after SPSR has been read with the OVRF bit set to 1.

Figure 20.14 shows an example of operation of the SPRF and OVRF bits in SPSR. The SPSR and SPDR accesses shown in figure 20.14 indicates the condition of accesses to SPSR and SPDR, respectively, where I denotes an idle cycle, W a write cycle, and R a read cycle. In the example of figure 20.14, this module performs an 8-bit serial transfer in which the CPHA bit in the command register (SPCMD) is 1, and CPOL is 0. The numbers given under the RSPCK waveform represent the number of RSPCK cycles (i.e., the number of transferred bits).

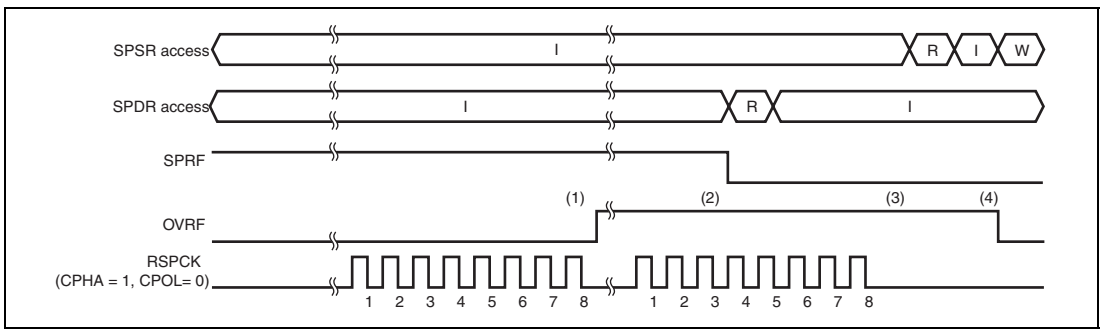


Figure 20.14 SPRF and OVRF Bit Operation Example

The operation of the flags at the timing shown in steps (1) to (4) in the figure is described below.

1. If a serial transfer terminates when the receive buffer does not have a space for the receive data length, this module detects an overrun error, and sets the OVRF bit to 1. This module does not copy the data in the shift register to the receive buffer.
2. The OVFR bit is not cleared even when SPDR is read and thus the number of data bytes in the receive buffer becomes less than the number of the receive buffer data triggering number specified by the RXTRG bits.
3. If the serial transfer terminates in an overrun error state, this module determines that the shift register is empty; in this manner, data transfer is enabled from the transmit buffer to the shift register.
4. If 0 is written to the OVRF bit after SPSR is read with OVRF = 1, this module clears the OVRF bit.

The occurrence of an overrun can be checked either by reading SPSR or by using an error interrupt and reading SPSR. When using an error interrupt, set the SPEIE bit in the control register (SPCR) to 1. When executing a serial transfer without using an error interrupt, measures should be taken to ensure the early detection of overrun errors, such as reading SPSR immediately after SPDR is read.

The OVRF bit is cleared to 0 under the following conditions:

- After SPSR is read in a condition in which the OVRF bit is set to 1, 0 is written to the OVRF bit.
- Power-on reset

Note: When the receive buffer has area enough to store receive data with an overrun error, this module receives receive data.

(2) Mode Fault Error

When the MSTR bit is 0, this module operates in slave mode. This module detects a mode fault error if the SSL input signal is negated during the serial transfer period (from the time the driving of valid data is started to the time the final valid data is fetched) when the MODFEN bit is 1 in slave mode.

Upon detecting a mode fault error, this module stops driving of the output signals and clears the SPE bit in SPCR to 0. When the SPE bit is cleared to 0, the function of this module is disabled and this module stops driving external signals. For details of disabling the function of this module by clearing the SPE bit to 0, see section 20.4.7, Initialization.

The occurrence of a mode fault error can be checked either by reading SPSR or by using an error interrupt and reading SPSR. When using an error interrupt, set the SPEIE bit in the control register (SPCR) to 1. To detect a mode fault error without using an error interrupt, it is necessary to poll SPSR.

When the MODF bit is 1, writing 1 to the SPE bit is ignored. To enable the function of this module after the detection of a mode fault error, the MODF bit must be set to 0. The MODF bit is cleared to 0 under the following conditions:

- After SPSR is read in a condition where the MODF bit has turned 1, 0 is written to the MODF bit.
- Power-on reset

20.4.7 Initialization

If 0 is written to the SPE bit in the control register (SPCR) or this module clears the SPE bit to 0 because of the detection of a mode fault error, this module disables the module function, and initializes a part of the module function. When a power-on reset is generated, this module initializes all of the module function. An explanation follows of initialization by the clearing of the SPE bit and initialization by a power-on reset.

(1) Initialization by Clearing SPE Bit

When the SPE bit in SPCR is cleared, this module performs the following initialization:

- Suspending any serial transfer that is being executed
- Stopping the driving of output signals (Hi-Z) in slave mode
- Initializing the internal state
- Initializing the TEND bit in SPSR

Initialization by the clearing of the SPE bit does not initialize the control bits of this module. For this reason, this module can be started in the same transfer mode as prior to the initialization if the SPE bit is re-set to 1.

20.4.8 SPI Operation

(1) Multi-Master Mode Operation

This section explains the operation in multi-master mode.

(a) Starting Serial Transfer

A serial transfer is started when transmit data is copied from the transmit buffer to the shift register, the shift register becomes full, and the receive buffer has a space for the receive data length. If transmit data has already been written to the shift register, data is not copied from the transmit buffer to the shift register.

For details of the transfer format, see section 20.4.4, Transfer Format.

(b) Terminating Serial Transfer

Irrespective of the CPHA bit in the command register (SPCMD), this module terminates the serial transfer after transmitting an RSPCK edge corresponding to the final sampling timing. After the serial transfer is completed, receive data is copied from the shift register to the receive buffer. If the receive buffer does not have a space for the receive data length after receive data is copied from the shift register to the receive buffer, another serial transfer will not be performed. In order to perform another serial transfer, data for the receive data length should be read from the receive buffer to secure the space for the receive data.

It should be noted that the final sampling timing varies depending on the bit length of transfer data. In master mode, the data length depends on the settings in bits SPB3 to SPB0 in SPCMD. For details on the transfer format, see section 20.4.4, Transfer Format.

(c) Sequence Control

The transfer format that is employed in master mode is determined by the sequence control register (SPSCR), command registers 0 to 3 (SPCMD0 to SPCMD3), the bit rate register (SPBR), the clock delay register (SPCKD), the slave select negation delay register (SSLND), and the next-access delay register (SPND).

SPSCR is a register used to determine the sequence configuration for serial transfers that are executed by this module in master mode. The following items are set in command registers SPCMD0 to SPCMD3: SSL output signal value, MSB/LSB first, data length, some of the bit rate settings, RSPCK polarity/phase, whether SPCKD is to be referenced, whether SSLND is to be referenced, and whether SPND is to be referenced. SPBR holds some of the bit rate settings; SPCKD, a clock delay value; SSLND, an SSL negation delay; and SPND, a next-access delay value.

According to the sequence length that is assigned to SPSCR, this module makes up a sequence comprised of a part or all of SPCMD0 to SPCMD3. This module contains a pointer to the SPCMD that makes up the sequence. The value of this pointer can be checked by reading bits SPCP1 and SPCP0 in the sequence status register (SPSSR). When the SPE bit in the control register (SPCR) is set to 1 and the function of this module is enabled, this module loads the pointer to the commands in SPCMD0, and incorporates the SPCMD0 settings into the transfer format at the beginning of serial transfer. This module increments the pointer each time the next-access delay period for a data transfer ends. Upon completion of the serial transfer that corresponds to the final command comprising the sequence, this module sets the pointer in SPCMD0, and in this manner the sequence is executed repeatedly.

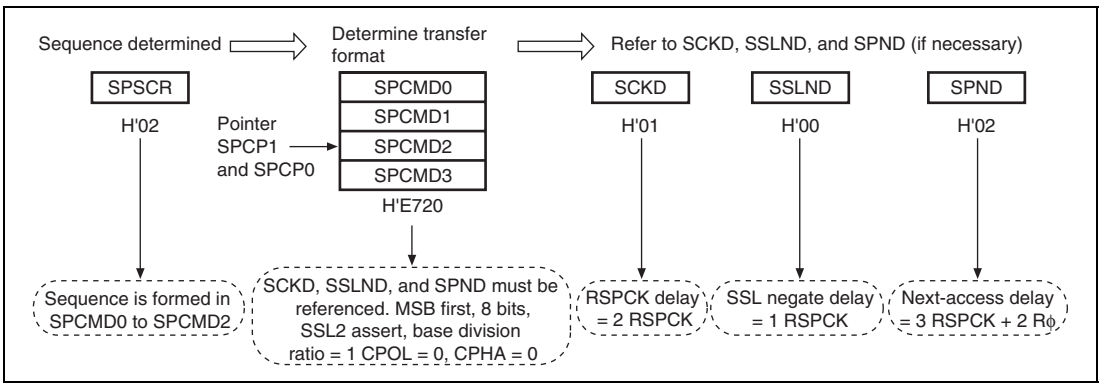


Figure 20.15 Determination Procedure of Serial Transfer Mode in Master Mode

(d) Burst Transfer

If the SSLKP bit in the command register (SPCMD) that this module references during the current serial transfer is 1, this module keeps the SSL signal level during the serial transfer until the beginning of the SSL signal assertion for the next serial transfer. If the SSL signal level for the next serial transfer is the same as the SSL signal level for the current serial transfer, this module can execute continuous serial transfers while keeping the SSL signal assertion status (burst transfer).

Figure 20.16 shows an example of an SSL signal operation for the case where a burst transfer is implemented using SPCMD0 and SPCMD1 settings. The text below explains operations (1) to (7) as depicted in figure 20.16. It should be noted that the polarity of the SSL output signal depends on the settings in the slave select polarity register (SSLP).

1. Based on SPCMD0, this module asserts the SSL signal and inserts RSPCK delays.
2. Serial transfers are executed according to SPCMD0.
3. SSL negation delays are inserted.
4. Because the SSLKP bit in SPCMD0 is 1, this module keeps the SSL signal value on SPCMD0. This period is sustained, at the shortest, for a period equal to the next-access delay of SPCMD0. If the shift register is empty after the passage of a minimum period, this period is sustained until such time as the transmit data is stored in the shift register for another transfer.
5. Based on SPCMD1, this module asserts the SSL signal and inserts RSPCK delays.
6. Serial transfers are executed according to SPCMD1.
7. Because the SSLKP bit in SPCMD1 is 0, this module negates the SSL signal. In addition, a next-access delay is inserted according to SPCMD1.

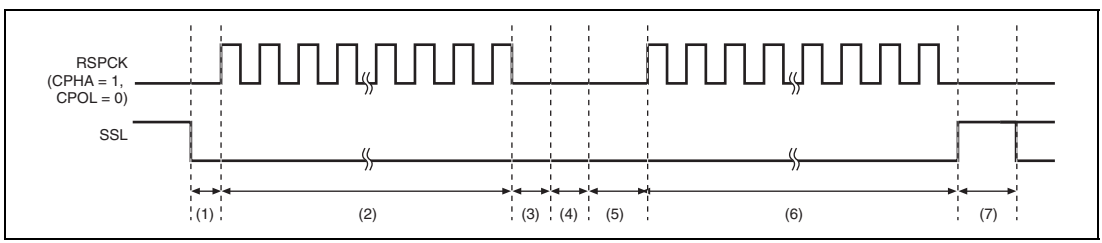


Figure 20.16 Example of Burst Transfer Operation using SSLKP Bit

If the SSL signal settings in the SPCMD in which 1 is assigned to the SSLKP bit are different from the SSL signal output settings in the SPCMD to be used in the next transfer, this module switches the SSL signal status to SSL signal assertion ((5) in figure 20.17) corresponding to the command for the next transfer. Notice that if such an SSL signal switching occurs, the slaves that drive the MISO signal compete, and the possibility arises of the collision of signal levels.

This module in master mode references within the module the SSL signal operation for the case where the SSLKP bit is not used. Even when the CPHA bit in SPCMD is 0, this module can accurately start serial transfers by asserting the SSL signal for the next transfer. For this reason, burst transfers in master mode can be executed irrespective of CPHA bit settings (see section 20.4.8 (2), Slave Mode Operation).

(e) RSPCK Delay (t1)

The RSPCK delay value in master mode depends on SCKDEN bit settings in the command register (SPCMD) and on clock delay register (SPCKD) settings. This module determines the SPCMD to be referenced during serial transfer by pointer control, and determines an RSPCK delay value during serial transfer by using the SCKDEN bit in the selected SPCMD and SPCKD, as shown in table 20.8. For a definition of RSPCK delay, see section 20.4.4, Transfer Format.

Table 20.8 Relationship among SCKDEN and SPCKD Settings and RSPCK Delay Values

SCKDEN	SPCKD	RSPCK Delay Value
0	000 to 111	1 RSPCK
1	000	1 RSPCK
	001	2 RSPCK
	010	3 RSPCK
	011	4 RSPCK
	100	5 RSPCK
	101	6 RSPCK
	110	7 RSPCK
	111	8 RSPCK

(f) SSL Negation Delay (t2)

The SSL negation delay value in master mode depends on SLNDEN bit settings in the command register (SPCMD) and on SSL negation delay register (SSLND) settings. This module determines the SPCMD to be referenced during serial transfer by pointer control, and determines an SSL negation delay value during serial transfer by using the SLNDEN bit in the selected SPCMD and SSLND, as shown in table 20.9. For a definition of SSL negation delay, see section 20.4.4, Transfer Format.

Table 20.9 Relationship among SLNDEN and SSLND Settings and SSL Negation Delay Values

SLNDEN	SSLND	SSL Negation Delay Value
0	000 to 111	1 RSPCK
1	000	1 RSPCK
	001	2 RSPCK
	010	3 RSPCK
	011	4 RSPCK
	100	5 RSPCK
	101	6 RSPCK
	110	7 RSPCK
	111	8 RSPCK

(g) Next-Access Delay (t3)

The next-access delay value in master mode depends on SPNDEN bit settings in the command register (SPCMD) and on next-access delay register (SPND) settings. This module determines the SPCMD to be referenced during serial transfer by pointer control, and determines a next-access delay value during serial transfer by using the SPNDEN bit in the selected SPCMD and SPND, as shown in table 20.10. For a definition of next-access delay, see section 20.4.4, Transfer Format.

Table 20.10 Relationship among SPNDEN and SPND Settings and Next-Access Delay Values

SPNDEN	SPND	Next-Access Delay Value
0	000 to 111	1 RSPCK + 2 Bϕ
1	000	1 RSPCK + 2 Bϕ
	001	2 RSPCK + 2 Bϕ
	010	3 RSPCK + 2 Bϕ
	011	4 RSPCK + 2 Bϕ
	100	5 RSPCK + 2 Bϕ
	101	6 RSPCK + 2 Bϕ
	110	7 RSPCK + 2 Bϕ
	111	8 RSPCK + 2 Bϕ

(h) Initialization Flowchart

Figure 20.17 is a flowchart illustrating an example of initialization in SPI operation when this module is used in master mode. For a description of how to set up the interrupt controller, direct memory access controller, and input/output ports, see the descriptions given in the individual blocks.

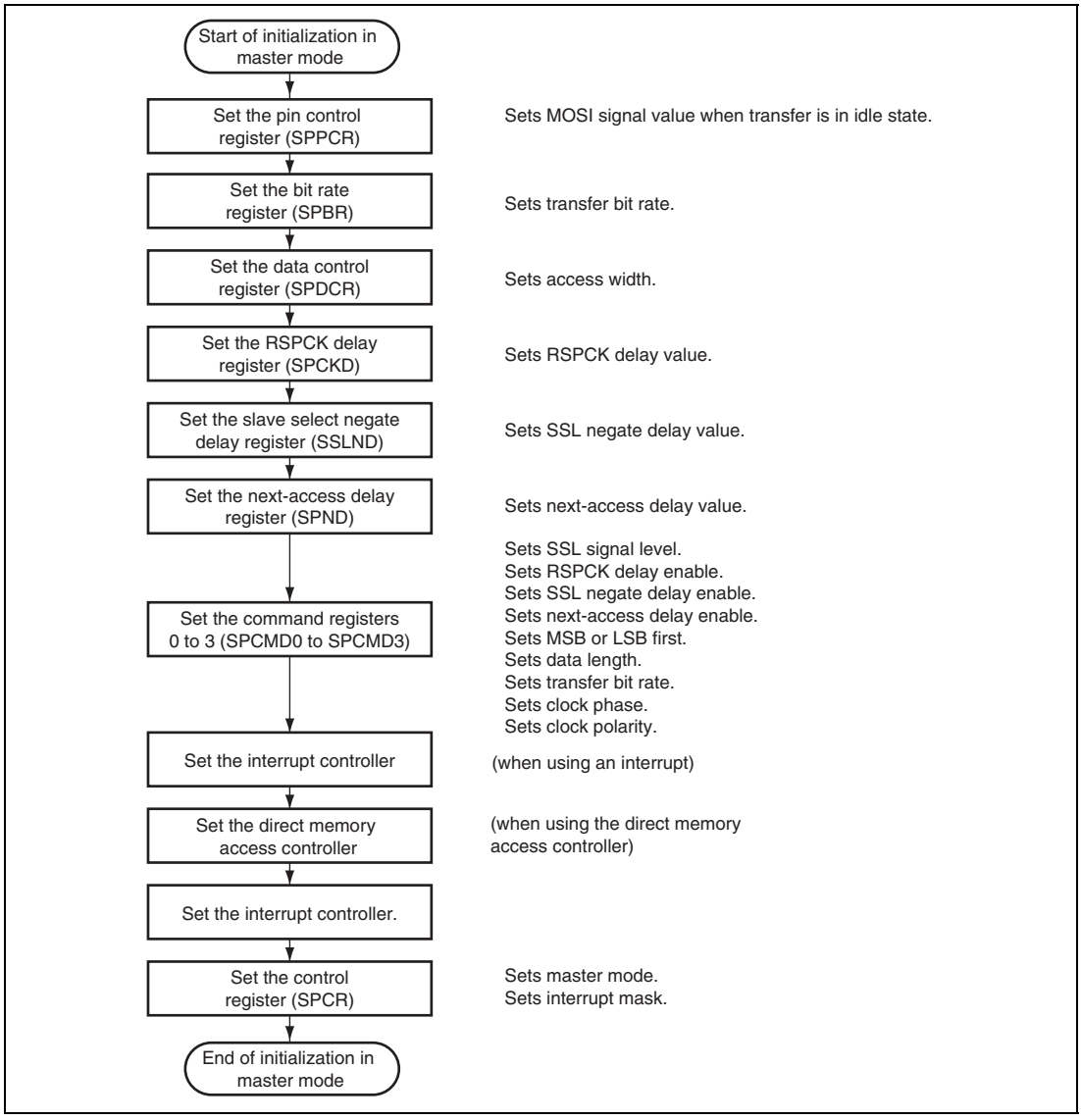


Figure 20.17 Example of Initialization Flowchart in Master Mode

(i) Transfer Operation Flowchart

Figure 20.18 is a flowchart illustrating a transfer in SPI operation when this module is used in master mode.

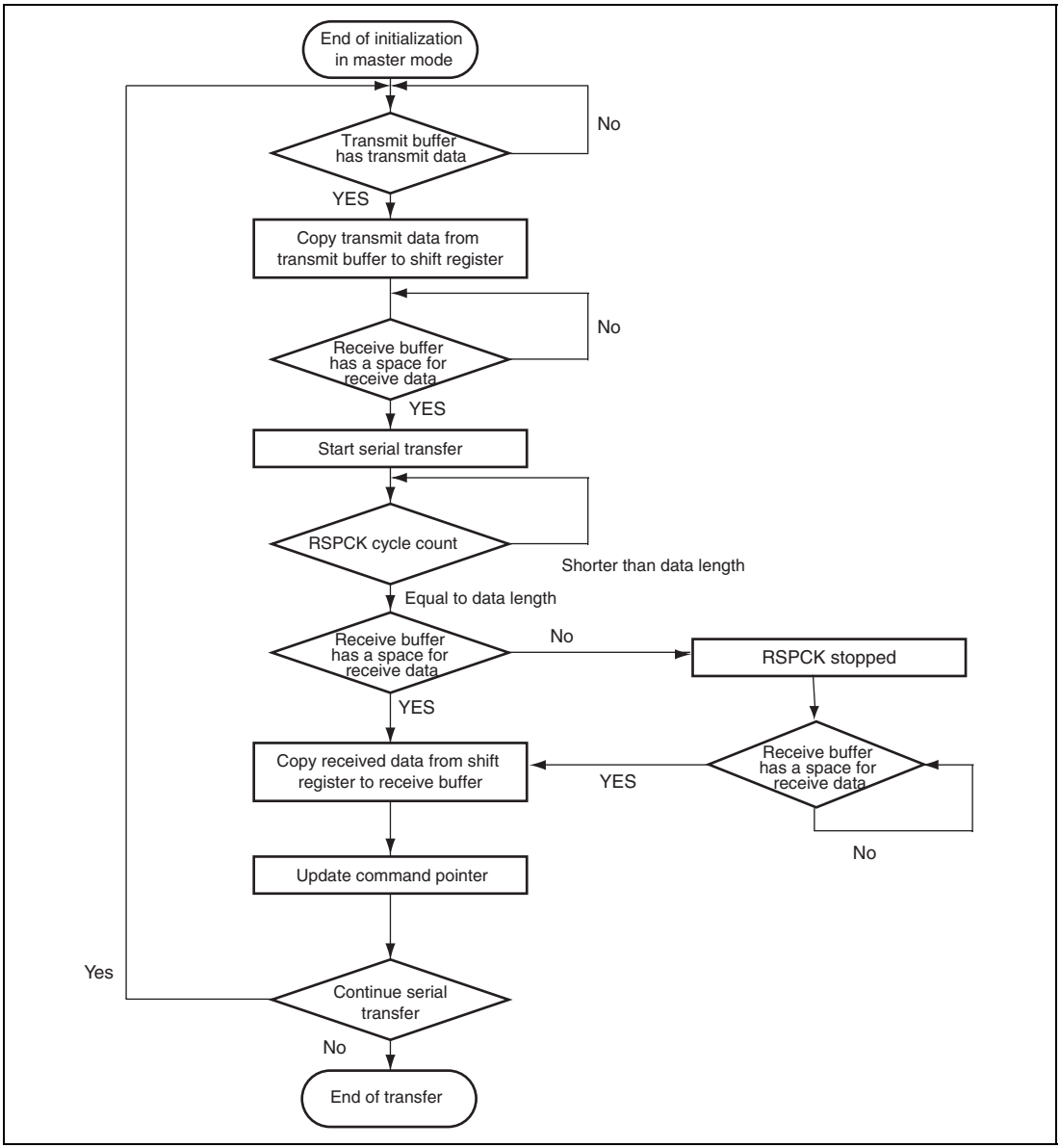


Figure 20.18 Transfer Operation Flowchart in Master Mode

(2) Slave Mode Operation

(a) Starting Serial Transfer

If this module detects an SSL input signal assertion when the CPHA bit in the command register 0 (SPCMD0) is 0, this module is required to start driving valid data to the MISO output signal. For this reason, when the CPHA bit is 0, the asserting of the SSL input signal triggers the start of a serial transfer.

If this module detects the first RSPCK edge in an SSL signal asserted condition when the CPHA bit is 1, this module is required to start driving valid data to the MISO output signal. For this reason, when the CPHA bit is 1, the first RSPCK edge in an SSL signal asserted condition triggers the start of a serial transfer.

When detecting the start of a serial transfer in a condition in which the shift register is empty, this module changes the status of the shift register to "full", so that data cannot be copied from the transmit buffer to the shift register when serial transfer is in progress. If the shift register was full before the serial transfer started, this module leaves the status of the shift register intact, in the full state.

Irrespective of CPHA bit settings, this module starts driving MISO output signals at the SSL signal assertion timing. Whether the data output from this module is valid or invalid differs depending on CPHA bit settings.

For details on the transfer format, see section 20.4.4, Transfer Format. The polarity of the SSL input signal depends on the setting of the SSL0P bit in the slave select polarity register (SSLP).

(b) Terminating Serial Transfer

Irrespective of the CPHA bit in the command register 0 (SPCMD0), this module terminates the serial transfer after detecting an RSPCK edge corresponding to the final sampling timing. When the receive buffer has an enough space for receive data, this module copies received data from the shift register to the receive buffer of the data register (SPDR) upon termination of the serial transfer. Irrespective of the value of the SPRF bit, this module changes the status of the shift register to "empty" upon termination of the serial transfer. If this module detects an SSL input signal negation from the beginning of serial transfer to the end of serial transfer, a mode fault error occurs (see section 20.4.6, Error Detection).

The final sampling timing changes depending on the bit length of the transfer data. In slave mode, the data length depends on the settings in bits SPB3 to SPB0 bits in SPCMD0. The polarity of the SSL input signal depends on the setting in the SSL0P bit in the slave select polarity register (SSLP). For details on the transfer format, see section 20.4.4, Transfer Format.

(c) Notes on Slave Operations

If the CPHA bit in the command register 0(SPCMD0) is 0, this module starts serial transfers when it detects the assertion edge for an SSL input signal. In the type of configuration shown in figure 20.4 as an example, if this module is used in single-slave mode, the SSL signal is always fixed at active state. Therefore, when the CPHA bit is set to 0, this module cannot correctly start a serial transfer. To correctly execute send/receive operation in a configuration in which the SSL input signal is fixed at active state, the CPHA bit should be set to 1. When it is necessary to set the CPHA bit to 0, the SSL input signal should not be fixed.

(d) Burst Transfer

If the CPHA bit in the command register 0 (SPCMD0) is 1, continuous serial transfer (burst transfer) can be executed while retaining the assertion state for the SSL input signal. If the CPHA bit is 1, the period from the first RSPCK edge to the sampling timing for the reception of the final bit in an SSL signal active state corresponds to a serial transfer period. Even when the SSL input signal remains at the active level, this module can accommodate burst transfers because it can detect the start of access.

If the CPHA bit is 0, for the reason given in section 20.4.8 (2) (c), Notes on Slave Operations, second and subsequent serial transfers during the burst transfer cannot be executed correctly.

(e) Initialization Flowchart

Figure 20.19 is a flowchart illustrating an example of initialization in SPI operation when this module is used in slave mode. For a description of how to set up the interrupt controller, direct memory access controller, and input/output ports, see the descriptions given in the individual blocks.

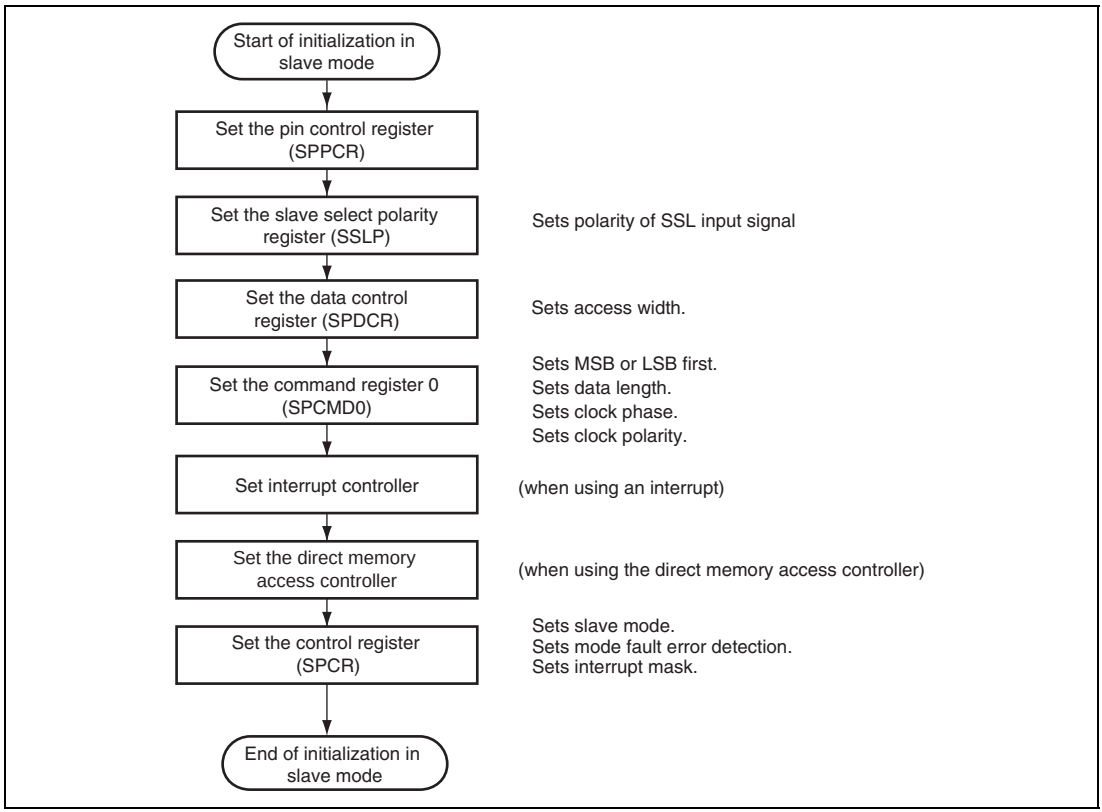


Figure 20.19 Example of Initialization Flowchart in Slave Mode

(f) Transfer Operation Flowchart (CPHA = 0)

Figure 20.20 is a flowchart illustrating a transfer in SPI operation when this module is used in slave mode with the CPHA bit in the command register 0 (SPCMD0) set to 0.

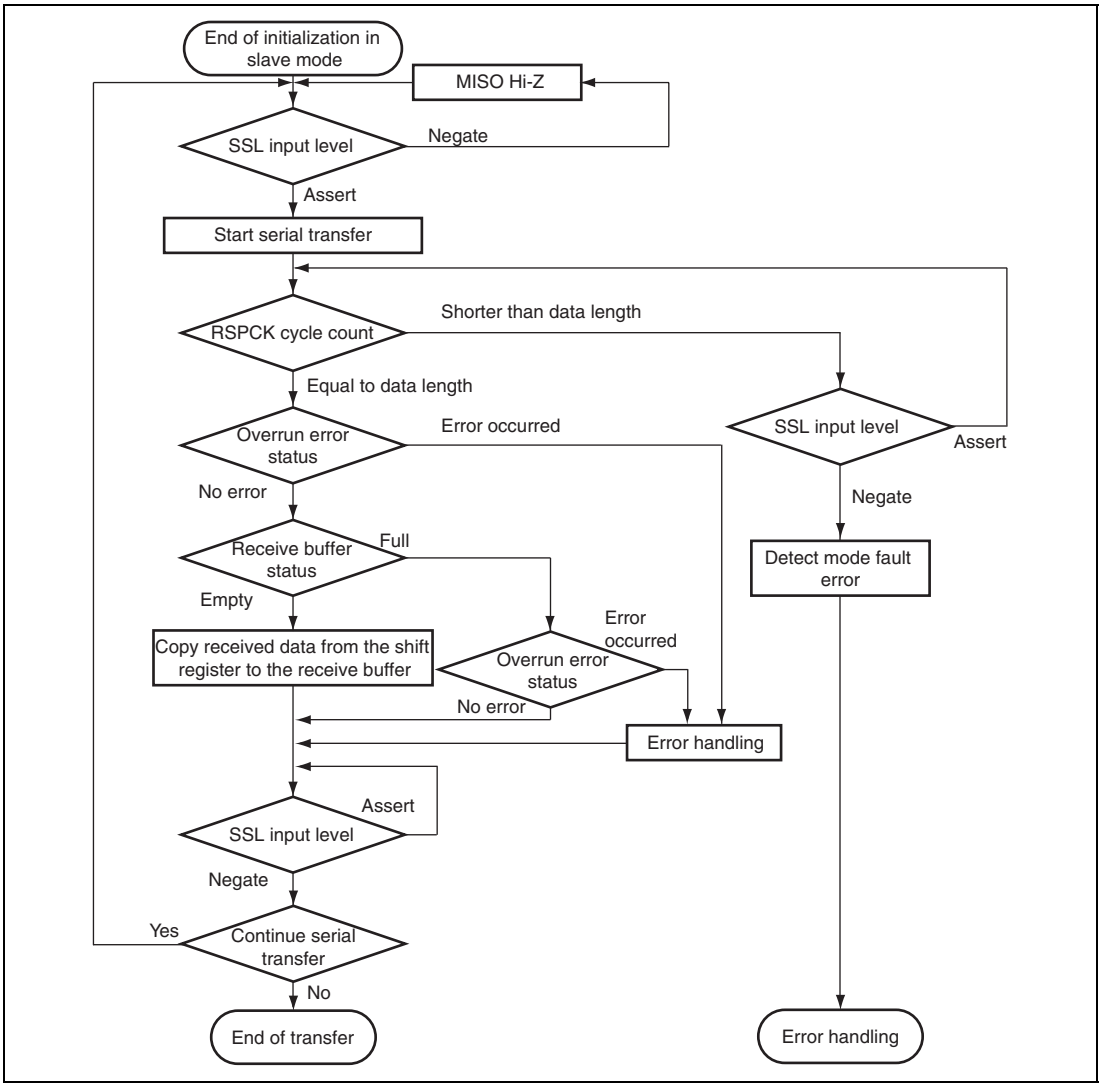


Figure 20.20 Transfer Operation Flowchart in Slave Mode (CPHA = 0)

(g) Transfer Operation Flowchart (CPHA = 1)

Figure 20.21 is a flowchart illustrating a transfer in SPI operation when this module is used in slave mode with the CPHA bit in the command register 0 (SPCMD0) and the MODFEN bit in the control register (SPCR) set to 1, respectively. The subsequent operation is not guaranteed when the serial transfer is started with the MODFEN bit set to 0 and the SSL input level is negated with the number of RSPCK cycles shorter than the data length.

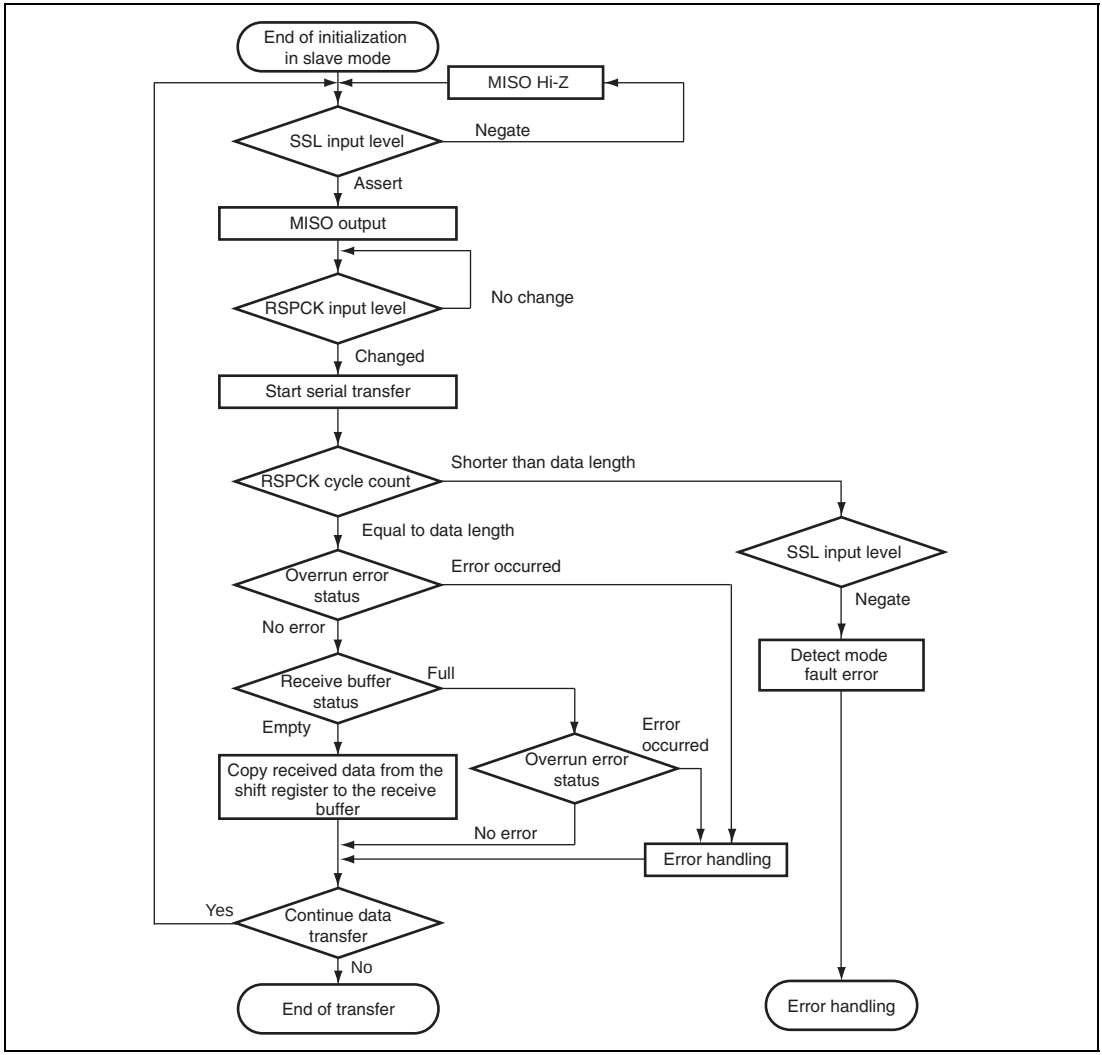


Figure 20.21 Transfer Operation Flowchart in Slave Mode (CPHA = 1)

20.4.9 Error Handling

Figures 20.22 and 20.23 show the error handling. The following error handling is used to return from the error state after an error in master or slave mode.

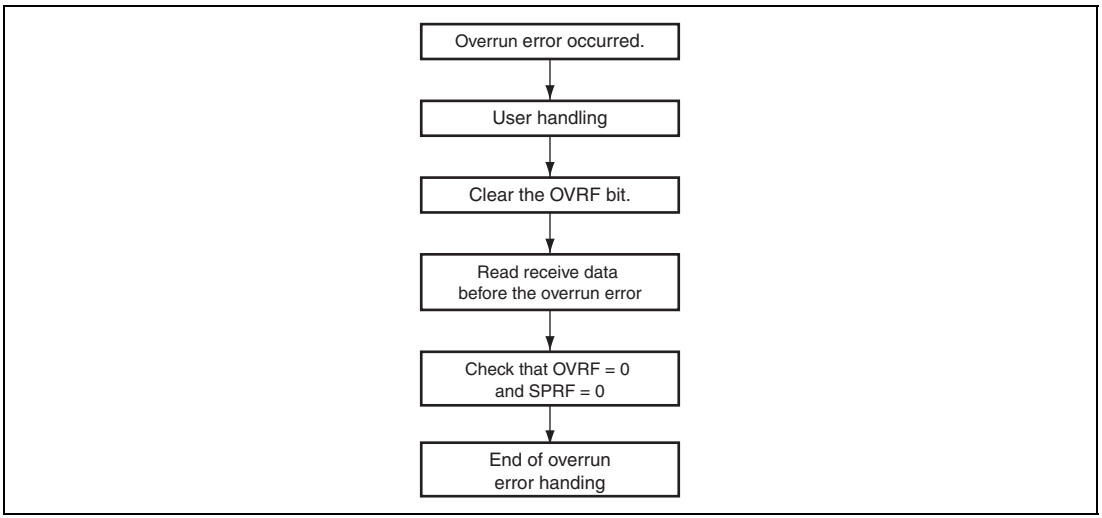


Figure 20.22 Error Handling (Overrun Error)

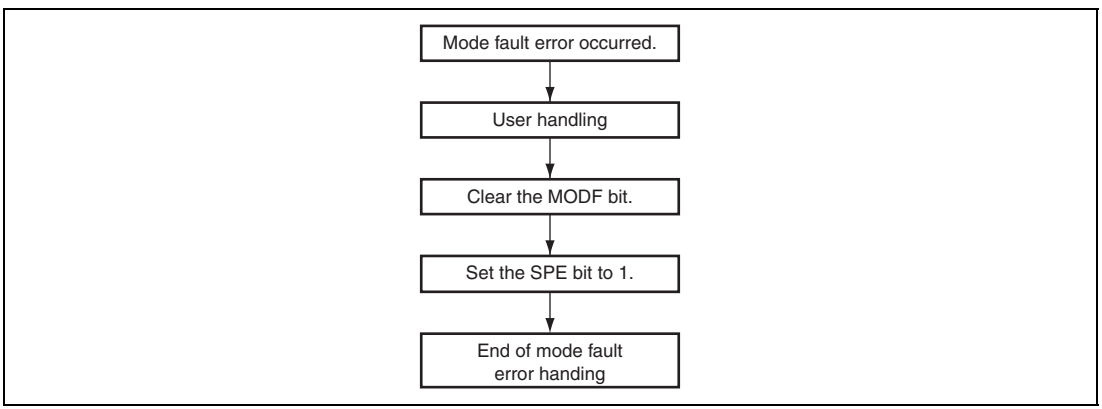


Figure 20.23 Error Handling (Mode Fault Error)

20.4.10 Loopback Mode

When 1 is written to the SPLP bit in the pin control register (SPPCR), this module shuts off the path between the MISO pin and the shift register, and between the MOSI pin and the shift register, and connects the input path and the output path (reversed) of the shift register. This is called loopback mode. When a serial transfer is executed in loopback mode, the transmit data becomes the received data. Figure 20.24 shows the configuration of the shift register input/output paths for the case where this module in master mode is set in loopback mode.

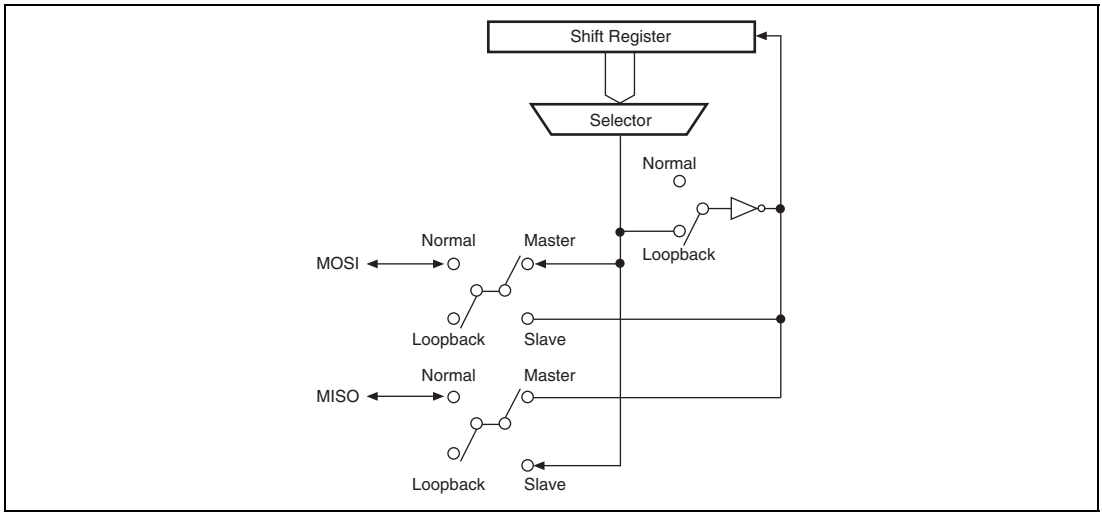


Figure 20.24 Configuration of Shift Register Input/Output Paths in Loopback Mode (Master Mode)

20.4.11 Interrupt Sources

This module has interrupt sources of receive buffer full, transmit buffer empty, mode fault, and overrun. In addition, the direct memory access controller can be activated by the receive buffer full or transmit buffer empty interrupt for data transfer.

Table 20.11 shows the interrupt sources.

When any of the interrupt conditions in table 20.11 is met, an interrupt is generated. The interrupt sources should be cleared with data transfer by the CPU or direct memory access controller (HPB-DMAC).

Table 20.11 Interrupt Sources

Name	Interrupt Source	Abbreviation	Interrupt Condition	Activation of Direct Memory Access Controller (HPB-DMAC)
SPRI	Receive buffer full	RXI	(SPRIE = 1) • (SPRF = 1)	Possible
SPTI	Transmit buffer empty	TXI	(SPTIE = 1) • (SPTEF = 1)	Possible
SPEI	Mode fault	MOI	(SPEIE = 1) • (MODF = 1)	—
	Overrun	OVI	(SPEIE = 1) • (OVRF = 1)	—

Section 21 Host Interface (HIF)

This LSI incorporates a host interface (HIF) for use in high-speed transfer of data between external devices which cannot utilize the system bus.

The HIF allows external devices to read from and write to 4 Kbytes (2 Kbytes × 2 banks) of the on-chip RAM exclusively for HIF use (HIFRAM) within this LSI, in 32-bit units. Interrupts issued to this LSI by an external device, interrupts sent from this LSI to the external device, and DMA transfer requests sent from this LSI to the external device are also supported. By using HIFRAM and these interrupt functions, software-based data transfer between external devices and this LSI becomes possible, and connection to external devices without releasing bus mastership is enabled.

Using HIFRAM, the HIF also supports HIF boot mode allowing this LSI to be booted.

21.1 Features

The HIF has the following features.

- An external device can read from or write to HIFRAM in 32-bit units via the HIF pins (access in 8-bit or 16-bit units not allowed). The on-chip CPU can read from or write to HIFRAM in 8-bit, 16-bit, or 32-bit units, via the internal peripheral bus. The HIFRAM access mode can be specified as bank mode or non-bank mode.
- When an external device accesses HIFRAM via the HIF pins, automatic increment of addresses and the endian can be specified with the HIF internal registers.
- By writing to specific bits in the HIF internal registers from an external device, or by accessing the end address of HIFRAM from the external device, interrupts (internal interrupts) can be issued to the on-chip CPU. Conversely, by writing to specific bits in the HIF internal registers from the on-chip CPU, interrupts (external interrupts) or DMAC transfer requests can be sent from the on-chip CPU to the external device.
- There are seven interrupt source bits each for internal interrupts and external interrupts. Accordingly, software control of 128 different interrupts is possible, enabling high-speed data transfer using interrupts.
- In HIF boot mode, this LSI can be booted from HIFRAM by an external device storing the instruction code in HIFRAM.

Figure 21.1 shows a block diagram of the HIF.

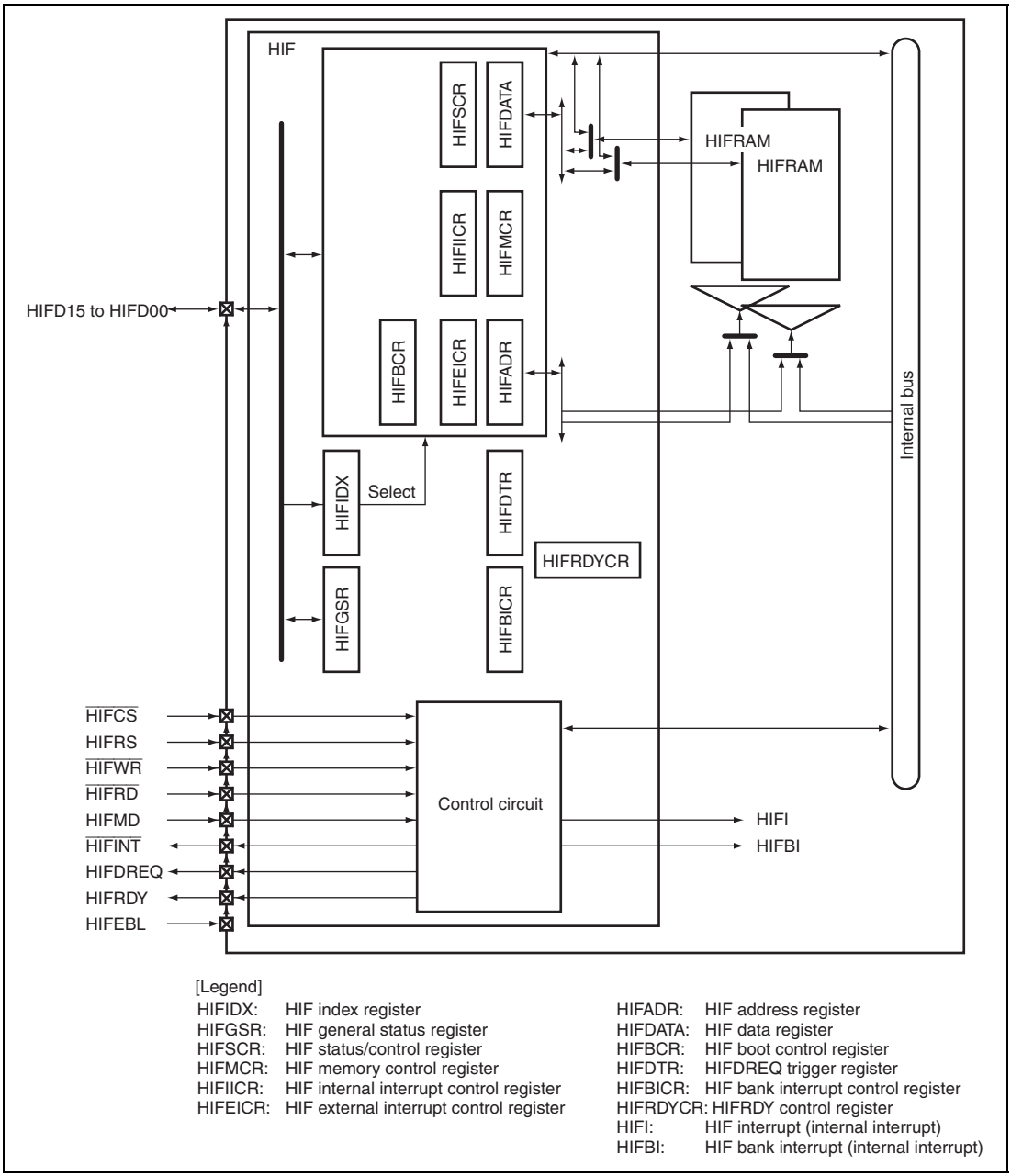


Figure 21.1 Block Diagram of HIF

21.2 Input/Output Pins

Table 21.1 shows the HIF pin configuration.

Table 21.1 Pin Configuration

Name	Abbreviation	I/O	Description
HIF data pins	HIFD15 to HIFD00	I/O	Address, data, or command input/output to the HIF
HIF chip select	$\overline{\text{HIFCS}}$	Input	Chip select input to the HIF
HIF register select	HIFRS	Input	Switching between HIF access types 0: Normal access (other than below) 1: Index register write
HIF write	$\overline{\text{HIFWR}}$	Input	Write strobe signal. Low level is input when an external device writes data to the HIF.
HIF read	$\overline{\text{HIFRD}}$	Input	Read strobe signal. Low level is input when an external device reads data from the HIF.
HIF interrupt	$\overline{\text{HIFINT}}$	Output	Interrupt request to an external device from the HIF
HIF mode	HIFMD	Input	Selects whether or not this LSI is started up in HIF boot mode. If a power-on reset is released when high level is input, this LSI is started up in HIF boot mode.
HIFDMAC transfer request	HIFDREQ	Output	To an external device, DMAC transfer request with HIFRAM as the destination
HIF boot ready	HIFRDY	Output	Indicates that the chip has been released from the HIF reset state and access from an external device to the HIF can be accepted. Use the HIFRDY control register to set the HIFRDY pin to the ready state (HIFRDY = low) in deep standby mode.
HIF pin enable	HIFEBL	Input	All HIF pins other than this pin are asserted by high-level input.

21.3 Parallel Access

21.3.1 Operation

The HIF can be accessed by combining the $\overline{\text{HIFCS}}$, HIFRS, $\overline{\text{HIFWR}}$, and $\overline{\text{HIFRD}}$ pins. Table 21.2 shows the correspondence between combinations of these signals and HIF operations.

Table 21.2 HIF Operations

$\overline{\text{HIFCS}}$	HIFRS	$\overline{\text{HIFWR}}$	$\overline{\text{HIFRD}}$	Operation
1	*	*	*	No operation (NOP)
0	1	0	1	Write to index register (HIFIDX[7:0])
0	0	0	1	Write to register specified by HIFIDX[7:0]
0	0	1	0	Read from register specified by HIFIDX[7:0]
0	*	1	1	No operation (NOP)
0	*	0	0	Setting prohibited

[Legend]
*: Don't care

21.3.2 Connection Method

When connecting the HIF to an external device, a method like that shown in figure 21.2 should be used.

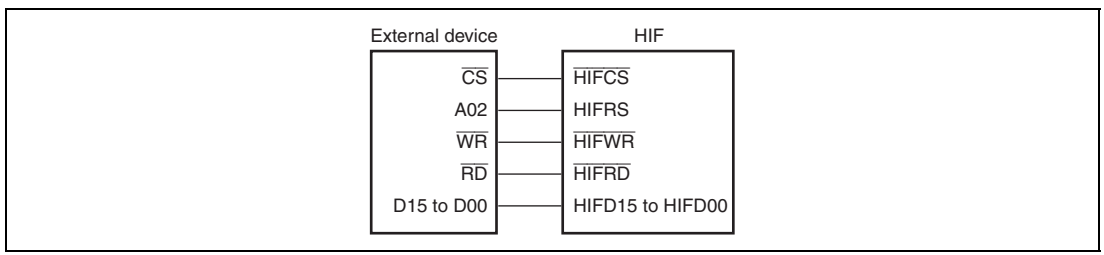


Figure 21.2 HIF Connection Example

21.4 Register Descriptions

Tables 21.3 (1) and 21.3 (2) show the register configuration and register state, respectively.

Table 21.3 (1) Register Configuration

Name	Abbreviation	Area 7 Address	Access Size
HIF index register	HIFIDX	H'FF830000	32
HIF general status register	HIFGSR	H'FF830004	32
HIF status/control register	HIFSCR	H'FF830008	32
HIF memory control register	HIFMCR	H'FF83000C	32
HIF internal interrupt control register	HIFIICR	H'FF830010	32
HIF external interrupt control register	HIFEICR	H'FF830014	32
HIF address register	HIFADR	H'FF830018	32
HIF data register	HIFDATA	H'FF83001C	32
HIFDREQ trigger register	HIFDTR	H'FF830020	32
HIF bank interrupt control register	HIFBICR	H'FF830024	32
HIF boot control register	HIFBCR	H'FF830040	32
HIFRDY control register	HIFRDYCR	H'FF830080	32

Table 21.3 (2) Register State in Each Operating Mode

Abbreviation	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
HIFIDX	Initialized	Initialized	Retained	Retained	Retained	Initialized
HIFGSR	Initialized	Initialized	Retained	Retained	Retained	Initialized
HIFSCR	Initialized	Initialized	Retained	Retained	Retained	Initialized
HIFMCR	Initialized	Initialized	Retained	Retained	Retained	Initialized
HIFIICR	Initialized	Initialized	Retained	Retained	Retained	Initialized
HIFEICR	Initialized	Initialized	Retained	Retained	Retained	Initialized
HIFADR	Initialized	Initialized	Retained	Retained	Retained	Initialized
HIFDATA	Initialized	Initialized	Retained	Retained	Retained	Initialized
HIFDTR	Initialized	Initialized	Retained	Retained	Retained	Initialized
HIFBICR	Initialized	Initialized	Retained	Retained	Retained	Initialized
HIFBCR	Initialized	Initialized	Retained	Retained	Retained	Initialized
HIFRDYCR	Initialized	Initialized	Retained	Retained	Retained	Initialized

21.4.1 HIF Index Register (HIFIDX)

HIFIDX is a 32-bit register used to specify the register read from or written to by an external device when the HIFRS pin is held low. HIFIDX can be only read by the on-chip CPU. HIFIDX can be only written to by an external device while the HIFRS pin is driven high.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	REG5	REG4	REG3	REG2	REG1	REG0	BYTE1	BYTE0
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	REG5	0	R/W	HIF Internal Register Select These bits specify which register among HIFGSR, HIFSCR, HIFMCR, HIFIICR, HIFEICR, HIFADR, HIFDATA, and HIFBCR is accessed by an external device. 000000: HIFGSR 000001: HIFSCR 000010: HIFMCR 000011: HIFIICR 000100: HIFEICR 000101: HIFADR 000110: HIFDATA 001111: HIFBCR Other than above: Setting prohibited
6	REG4	0	R/W	
5	REG3	0	R/W	
4	REG2	0	R/W	
3	REG1	0	R/W	
2	REG0	0	R/W	

Bit	Bit Name	Initial Value	R/W	Description
1	BYTE1	0	R/W	Internal Register Byte Specification
0	BYTE0	0	R/W	These bits specify in advance the target word location before the external device accesses a register among HIFGSR, HIFSCR, HIFMCR, HIFIICR, HIFEICR, HIFADR, HIFDATA, and HIFBCR. See also section 21.8, Alignment Control. • When HIFSCR.BO = 0 00: Bits 31 to 16 in register 01: Setting prohibited 10: Bits 15 to 0 in register 11: Setting prohibited • When HIFSCR.BO = 1 00: Bits 15 to 0 in register 01: Setting prohibited 10: Bits 31 to 16 in register 11: Setting prohibited However, when HIFDATA is selected using bits REG5 to REG0, each time reading or writing of HIFDATA occurs, these bits change according to the following rule. 00 → 10 → 00 → 10... repeated

21.4.2 HIF General Status Register (HIFGSR)

HIFGSR is a 32-bit register, which can be freely used for handshaking between an external device connected to the HIF and the software of this LSI. HIFGSR can be read from and written to by the on-chip CPU. Access to HIFGSR by an external device should be performed with HIFGSR specified by bits REG5 to REG0 in HIFIDX and the HIFRS pin low.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	STATUS[15:0]															
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	STATUS[15:0]	All 0	R/W	General Status This register can be read from and written to by an external device connected to the HIF, and by the on-chip CPU. These bits are initialized only at a power-on reset.

21.4.3 HIF Status/Control Register (HIFSCR)

HIFSCR is a 32-bit register used to control the HIFRAM access mode and endian setting. HIFSCR can be read from and written to by the on-chip CPU. Access to HIFSCR by an external device should be performed with HIFSCR specified by bits REG5 to REG0 in HIFIDX and the HIFRS pin low.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	DMD	DPOL	BMD	BSEL	—	—	MD1	—	—	WBSWP	EDN	BO
Initial Value:	0	0	0	0	0	0	0	0	0	1	0/1	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11	DMD	0	R/W	DREQ Mode
10	DPOL	0	R/W	DREQ Polarity Controls the assert mode for the HIFDREQ pin. For details on the negate timing, see section 21.7, External DMAC Interface. 00: For a DMAC transfer request to an external device, low level is generated at the HIFDREQ pin. The default for the HIFDREQ pin is high-level output. 01: For a DMAC transfer request to an external device, high level is generated at the HIFDREQ pin. The default for the HIFDREQ pin is low-level output. 10: For a DMAC transfer request to an external device, falling edge is generated at the HIFDREQ pin. The default for the HIFDREQ pin is high-level output. 11: For a DMAC transfer request to an external device, rising edge is generated at the HIFDREQ pin. The default for the HIFDREQ pin is low-level output.

Bit	Bit Name	Initial Value	R/W	Description
9	BMD	0	R/W	HIFRAM Bank Mode
8	BSEL	0	R/W	HIFRAM Bank Select Controls the HIFRAM access mode. 00: Both an external device and the on-chip CPU can access bank 0. When access by both of these conflict, even though the access addresses differ, access by the external device is processed before access by the on-chip CPU. Bank 1 cannot be accessed. 01: Both an external device and the on-chip CPU can access bank 1. When access by both of these conflict, even though the access addresses differ, access by the external device is processed before access by the on-chip CPU. Bank 0 cannot be accessed. 10: An external device can access only bank 0 while the on-chip CPU can access only bank 1. 11: An external device can access only bank 1 while the on-chip CPU can access only bank 0.
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
5	MD1	0/1	R	HIF Mode 1 Indicates whether this LSI was started up in HIF boot mode or non-HIF boot mode. This bit stores the value of the HIFMD pin sampled at a power-on reset 0: Started up in non-HIF boot mode (booted from the memory connected to area 0) 1: Started up in HIF boot mode (booted from HIFRAM)
4, 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2	WBSWP	0	R/W	Byte Order for Access of HIFDATA Specifies the byte order when an external device accesses HIFDATA. See also section 21.8, Alignment Control. 0: Aligned according to the BO bit. 1: Swapped in word units from the big endian order and then swapped in byte units within each word. The setting of the BO bit is ignored.
1	EDN	0	R/W	Endian for HIFRAM Access Specifies the byte order when HIFRAM is accessed by the on-chip CPU. 0: Big endian (MSB first) 1: Little endian (LSB first)
0	BO	0	R/W	Byte Order for Access of All HIF Registers Including HIFDATA Specifies the byte order when an external device accesses all HIF registers including HIFDATA. However, for the HIFDATA alignment, this bit is referred to only when WBSWP = 0 and ignored when WBSWP = 1. See also section 21.8, Alignment Control. 0: Big endian (MSB first) 1: Little endian (LSB first)

21.4.4 HIF Memory Control Register (HIFMCR)

HIFMCR is a 32-bit register used to control HIFRAM. HIFMCR can be only read by the on-chip CPU. Access to HIFMCR by an external device should be performed with HIFMCR specified by bits REG5 to REG0 in HIFIDX and the HIFRS pin low.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	LOCK	—	WT	—	RD	—	—	AI/AD
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W*	R	R/W*	R	R/W*	R	R	R/W*

Note: * Changing the HIFRAM banks accessible from an external device by setting the BMD and BSEL bits in HIFSCR does not affect the setting of this bit.

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	LOCK	0	R/W*	Lock This bit is used to lock the access direction (read or write) for consecutive access of HIFRAM by an external device via HIFDATA. When this bit is set to 1, the values of the RD and WT bits set at the same time are held until this bit is next cleared to 0. When the RD bit and this bit are simultaneously set to 1, consecutive read mode is entered. When the WT bit and this bit are simultaneously set to 1, consecutive write mode is entered. Both the RD and WT bits should not be set to 1 simultaneously.
6	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
5	WT	0	R/W*	Write When this bit is set to 1, the HIFDATA value is written to the HIFRAM position corresponding to HIFADR. If this bit and the LOCK bit are set to 1 simultaneously, HIFRAM consecutive write mode is entered, and high-speed data transfer becomes possible. This mode is maintained until this bit is next cleared to 0, or until the LOCK bit is cleared to 0. If the LOCK bit is not simultaneously set to 1 with this bit, writing to HIFRAM is performed only once. Thereafter, the value of this bit is automatically cleared to 0.
4	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
3	RD	0	R/W*	Read When this bit is set to 1, the HIFRAM data corresponding to HIFADR is fetched to HIFDATA. If this bit and the LOCK bit are set to 1 simultaneously, HIFRAM consecutive read mode is entered, and high-speed data transfer becomes possible. This mode is maintained until this bit is next cleared to 0, or until the LOCK bit is cleared to 0. If the LOCK bit is not simultaneously set to 1 with this bit, reading of HIFRAM is performed only once. Thereafter, the value of this bit is automatically cleared to 0.
2, 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	AI/AD	0	R/W*	Address Auto-Increment/Decrement This bit is valid only when the LOCK bit is 1. The value of HIFADR is automatically incremented by 4 or decremented by 4 according to the setting of this bit each time reading or writing of HIFRAM is performed. 0: Auto-increment mode (+4) 1: Auto-decrement mode (–4)

Note: * Changing the HIFRAM banks accessible from an external device by setting the BMD and BSEL bits in HIFSCR does not affect the setting of this bit.

21.4.5 HIF Internal Interrupt Control Register (HIFIICR)

HIFIICR is a 32-bit register used to issue interrupts from an external device connected to the HIF to the on-chip CPU. Access to HIFIICR by an external device should be performed with HIFIICR specified by bits REG5 to REG0 in HIFIDX and the HIFRS pin low.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	IIC6	IIC5	IIC4	IIC3	IIC2	IIC1	IIC0	IIR
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	IIC6	0	R/W	Internal Interrupt Source These bits specify the source for interrupts generated by the IIR bit. These bits can be written to from both an external device and the on-chip CPU. By using these bits, fast execution of interrupt exception handling is possible. These bits are completely under software control, and their values have no effect on the operation of this LSI.
6	IIC5	0	R/W	
5	IIC4	0	R/W	
4	IIC3	0	R/W	
3	IIC2	0	R/W	
2	IIC1	0	R/W	
1	IIC0	0	R/W	
0	IIR	0	R/W	Internal Interrupt Request While this bit is 1, an interrupt request (HIFI) is issued to the on-chip CPU.

21.4.6 HIF External Interrupt Control Register (HIFEICR)

HIFEICR is a 32-bit register used to issue interrupts to an external device connected to the HIF from this LSI. Access to HIFEICR by an external device should be performed with HIFEICR specified by bits REG5 to REG0 in HIFIDX and the HIFRS pin low.

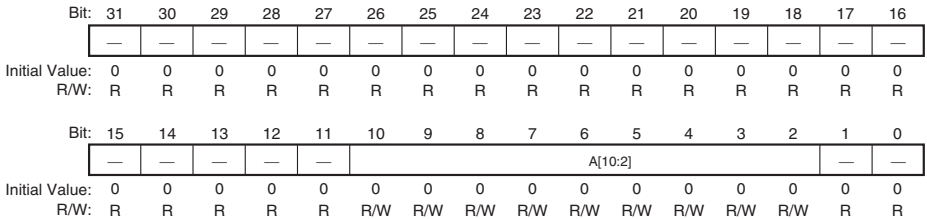
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	EIC6	EIC5	EIC4	EIC3	EIC2	EIC1	EIC0	EIR
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	EIC6	0	R/W	External Interrupt Source These bits specify the source for interrupts generated by the EIR bit. These bits can be written to from both an external device and the on-chip CPU. By using these bits, fast execution of interrupt exception handling is possible. These bits are completely under software control, and their values have no effect on the operation of this LSI.
6	EIC5	0	R/W	
5	EIC4	0	R/W	
4	EIC3	0	R/W	
3	EIC2	0	R/W	
2	EIC1	0	R/W	
1	EIC0	0	R/W	
0	EIR	0	R/W	External Interrupt Request While this bit is 1, the $\overline{\text{HIFINT}}$ pin is asserted to issue an interrupt request to an external device from this LSI.

21.4.7 HIF Address Register (HIFADR)

HIFADR is a 32-bit register which indicates the address in HIFRAM to be accessed by an external device. When using the LOCK bit setting in HIFMCR to specify consecutive access of HIFRAM, auto-increment (+4) or auto-decrement (-4) of the address, according to the AI/AD bit setting in HIFMCR, is performed automatically, and HIFADR is updated. HIFADR can be only read by the on-chip CPU. Access to HIFADR by an external device should be performed with HIFADR specified by bits REG5 to REG0 in HIFIDX and the HIFRS pin low.



Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 2	A[10:2]	All 0	R/W	HIFRAM Address Specification These bits specify the address of HIFRAM to be accessed by an external device, with 32-bit boundary.
1, 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

21.4.8 HIF Data Register (HIFDATA)

HIFDATA is a 32-bit register used to hold data to be written to HIFRAM and data read from HIFRAM for external device accesses. If HIFDATA is not used when accessing HIFRAM, it can be used for data transfer between an external device connected to the HIF and the on-chip CPU. HIFDATA can be read from and written to by the on-chip CPU. Access to HIFDATA by an external device should be performed with HIFDATA specified by bits REG5 to REG0 in HIFIDX and the HIFRS pin low.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	D[31:16]															
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	D[15:0]															
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	D[31:0]	All 0	R/W	32-bit data

21.4.9 HIF Boot Control Register (HIFBCR)

HIFBCR is a 32-bit register for exclusive control of an external device and the on-chip CPU regarding access of HIFRAM. HIFBCR can be only read by the on-chip CPU. Access to HIFBCR by an external device should be performed with HIFBCR specified by bits REG5 to REG0 in HIFIDX and the HIFRS pin low.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	AC
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0/1
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 1	—	All 0	R/W	AC-Bit Writing Assistance These bits should be used to write the bit pattern (H'A5) needed to set the AC bit to 1. These bits are always read as 0.
0	AC	0/1	R/W	HIFRAM Access Exclusive Control Controls accessing of HIFRAM by the on-chip CPU for the HIFRAM bank selected by the BMD and BSEL bits in HIFSCR as the bank allowed to be accessed by this LSI. 0: The on-chip CPU can perform reading/writing of HIFRAM. 1: When an HIFRAM read/write operation by the on-chip CPU occurs, the CPU enters the wait state, and execution of the instruction is halted until this bit is cleared to 0. When booted in non-HIF boot mode, the initial value of this bit is 0. When booted in HIF boot mode, the initial value of this bit is 1. After an external device writes a boot program to HIFRAM via the HIF, clearing this bit to 0 boots the on-chip CPU from HIFRAM. When 1 is written to this bit by an external device, H'A5 should be written to bits 7 to 0 to prevent erroneous writing.

21.4.10 HIFDREQ Trigger Register (HIFDTR)

HIFDTR is a 32-bit register. Writing to HIFDTR by the on-chip CPU asserts the HIFDREQ pin. HIFDTR cannot be accessed by an external device.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DTRG
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	DTRG	0	R/W	HIFDREQ Trigger When 1 is written to this bit, the HIFDREQ pin is asserted according to the setting of the DMD and DPOL bits in HIFSCR. This bit is automatically cleared to 0 in synchronization with negate of the HIFDREQ pin. Though this bit can be set to 1 by the on-chip CPU, it cannot be cleared to 0. To avoid conflict between clearing of this bit by negate of the HIFDREQ pin and setting of this bit by the on-chip CPU, make sure this bit is cleared to 0 before setting this bit to 1 by the on-chip CPU. Writing 0 is invalid.

21.4.11 HIF Bank Interrupt Control Register (HIFBICR)

HIFBICR is a 32-bit register that controls HIF bank interrupts. HIFBICR cannot be accessed by an external device.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	BIE	BIF
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	BIE	0	R/W	Bank Interrupt Enable Enables or disables a bank interrupt request (HIFBI) issued to the on-chip CPU. 0: HIFBI disabled 1: HIFBI enabled

Bit	Bit Name	Initial Value	R/W	Description
0	BIF	0	R/W	Bank Interrupt Request Flag While this bit is 1, a bank interrupt request (HIFBI) is issued to the on-chip CPU according to the setting of the BIE bit. In auto-increment mode (AI/AD bit in HIFMCR is 0), this bit is automatically set to 1 when an external device has completed access to the 32-bit data in the end address of HIFRAM and the $\overline{\text{HIFCS}}$ pin has been negated. In auto-decrement mode (AI/AD bit in HIFMCR is 1), this bit is automatically set to 1 when an external device has completed access to the 32-bit data in the start address of HIFRAM and the $\overline{\text{HIFCS}}$ pin has been negated. Though this bit can be cleared to 0 by the on-chip CPU, it cannot be set to 1. Make sure setting of this bit by HIFRAM access from an external device and clearing of this bit by the on-chip CPU do not conflict using software. Writing 1 is invalid.

21.4.12 HIFRDY Control Register (HIFRDYCR)

HIFRDYCR is a 32-bit register used to mask the HIFRDY signal of this chip. The signal is masked by writing to this register, which is inaccessible from external devices.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MASK
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	MASK	0	R/W	HIFRDY Mask The HIFRDY pin indicates that the chip has been released from the HIF reset state and access from external devices to the HIF can be accepted. By writing 1 to this bit, the accessible state (HIFRDY = high) can be masked (HIFRDY = low). Write 0 to clear this bit.

21.5 Memory Map

Table 21.4 shows the memory map of HIFRAM.

Table 21.4 Memory Map

Classification	Start Address	End Address	Memory Size
Map from external device*	H'0000	H'07FF	2 Kbytes
Map from on-chip CPU*	H'FF82_F000	H'FF82_07FF	2 Kbytes

Note: * Map for a single HIFRAM bank. Which bank is to be accessed by an external device or the on-chip CPU depends on the BMD and BSEL bits in HIFSCR. The mapping addresses are common between the banks.

21.6 Interface

21.6.1 Basic Sequence

Figure 21.3 shows the basic read/write sequence. HIF read is defined by the overlap period of the $\overline{\text{HIFRD}}$ low-level period and $\overline{\text{HIFCS}}$ low-level period, and HIF write is defined by the overlap period of the $\overline{\text{HIFWR}}$ low-level period and $\overline{\text{HIFCS}}$ low-level period. The HIFRS signal indicates whether this is normal access or index register access; low level indicates normal access and high level indicates index register access.

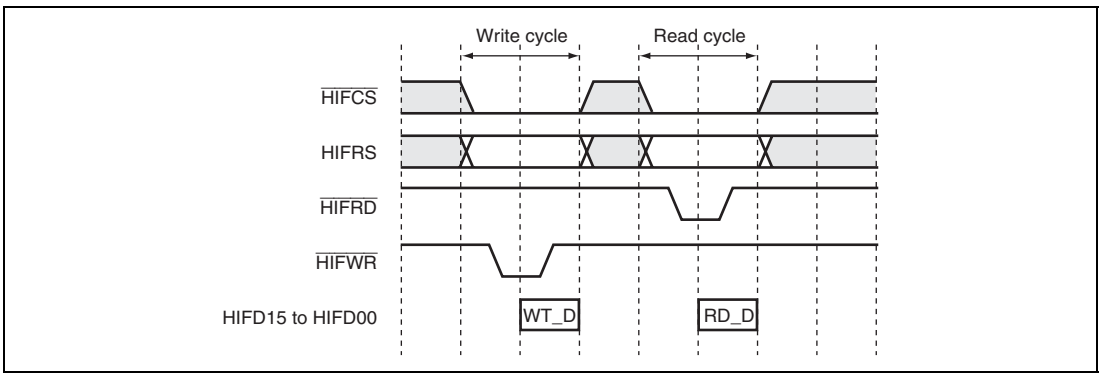


Figure 21.3 Basic Timing for HIF Interface

21.6.2 Reading/Writing of HIF Registers other than HIFIDX and HIFIDX

As shown in figure 21.4, in reading and writing of HIF internal registers other than HIFIDX and HIFIDX, first HIFRS is held high and HIFIDX is written to in order to select the register to be accessed and the byte location. Then HIFRS is held low, and reading or writing of the register selected by HIFIDX is performed.

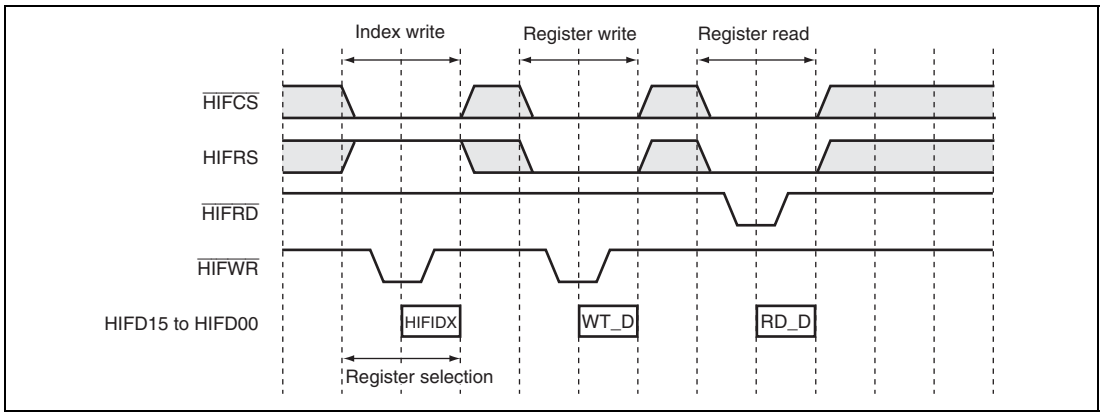


Figure 21.4 HIF Register Settings

21.6.3 Consecutive Data Writing to HIFRAM by External Device

Figure 21.5 shows the timing chart for consecutive data transfer from an external device to HIFRAM. As shown in this timing chart, by setting the start address and the data to be written first, consecutive data transfer can subsequently be performed.

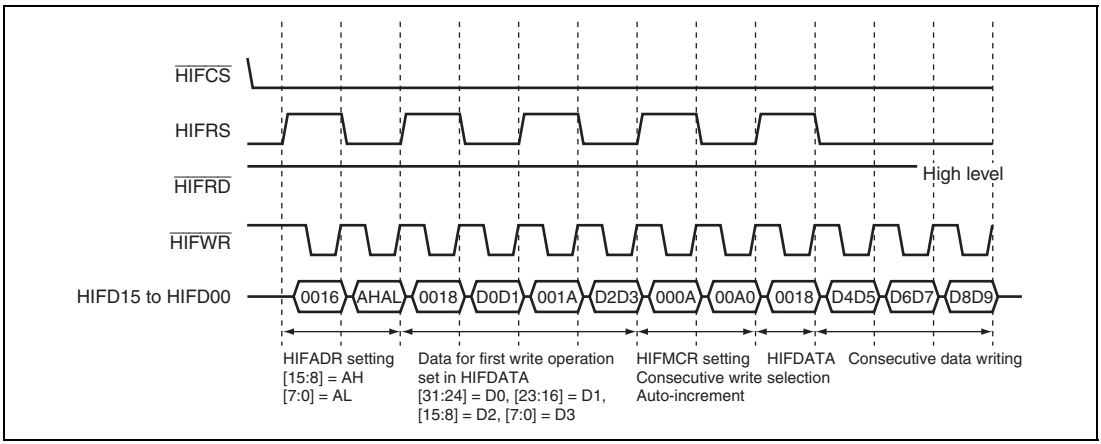


Figure 21.5 Consecutive Data Writing to HIFRAM

21.6.4 Consecutive Data Reading from HIFRAM to External Device

Figure 21.6 shows the timing chart for consecutive data reading from HIFRAM to an external device. As this timing chart indicates, by setting the start address, data can subsequently be read out consecutively.

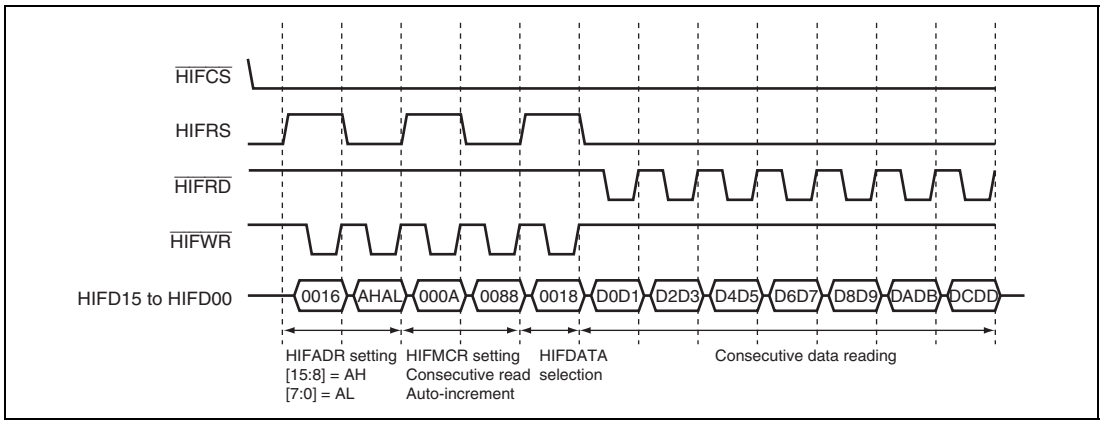


Figure 21.6 Consecutive Data Reading from HIFRAM

21.7 External DMAC Interface

Figures 21.7 to 21.10 show the HIFDREQ output timing. The start of the HIFDREQ assert synchronizes with the DTRG bit in HIFDTR being set to 1. The HIFDREQ negate timing and assert level are determined by the DMD and DPOL bits in HIFSCR, respectively.

When the external DMAC is specified to detect low level of the HIFDREQ signal, set DMD = 0 and DPOL = 0. After writing 1 to the DTRG bit, the HIFDREQ signal remains low until a read from or write to the HIFIDX-specified register is detected. Writing to the index register (HIFIDX) does not negate the signal.

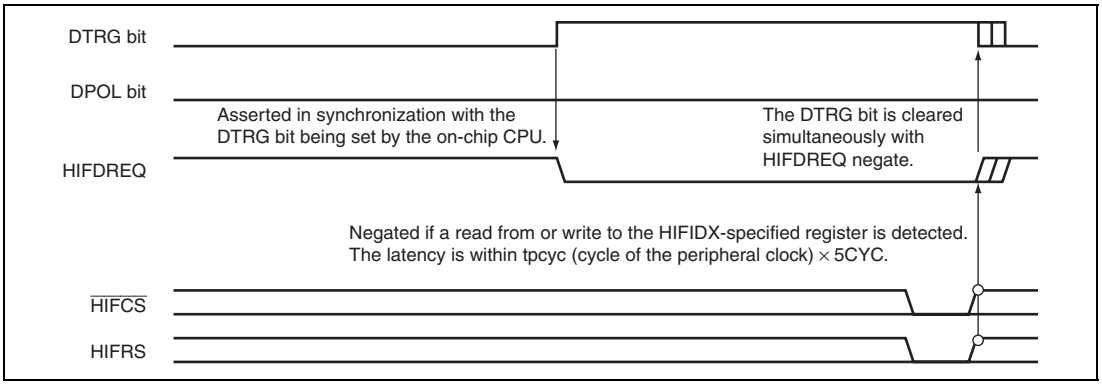


Figure 21.7 HIFDREQ Timing (When DMD = 0 and DPOL = 0)

When the external DMAC is specified to detect high level of the HIFDREQ signal, set DMD = 0 and DPOL = 1. At the time the DPOL bit is set to 1, HIFDREQ becomes low. After this, the HIFDREQ signal remains low from when 1 is written to the DTRG bit until a read from or write to the HIFIDX-specified register is detected. Writing to the index register (HIFIDX) does not negate the signal.

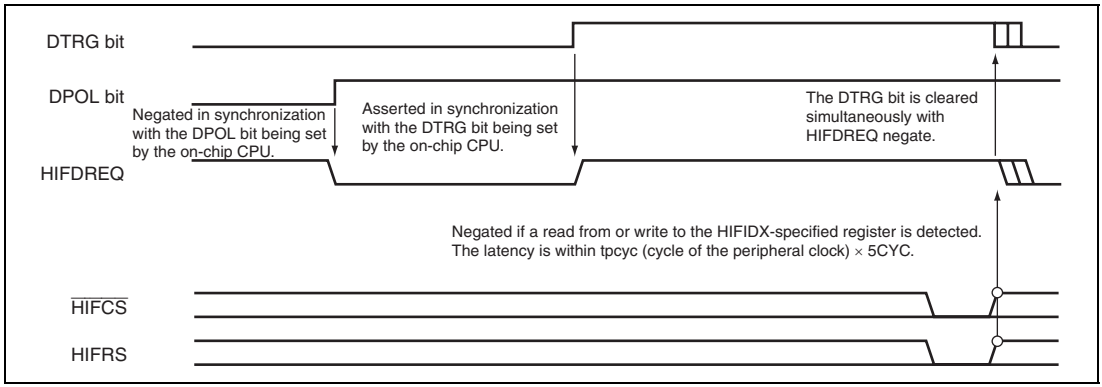


Figure 21.8 HIFDREQ Timing (When DMD = 0 and DPOL = 1)

When the external DMAC is specified to detect the falling edge of the HIFDREQ signal, set DMD = 1 and DPOL = 0. After writing 1 to the DTRG bit, a low pulse of 32 peripheral clock cycles is generated at the HIFDREQ pin.

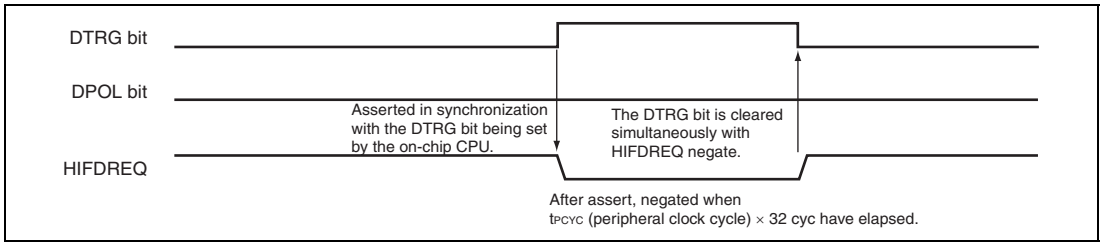


Figure 21.9 HIFDREQ Timing (When DMD = 1 and DPOL = 0)

When the external DMAC is specified to detect the rising edge of the HIFDREQ signal, set DMD = 1 and DPOL = 1. At the time the DPOL bit is set to 1, HIFDREQ becomes low. Then after writing 1 to the DTRG bit, a low pulse of 32 peripheral clock cycles is generated at the HIFDREQ pin.

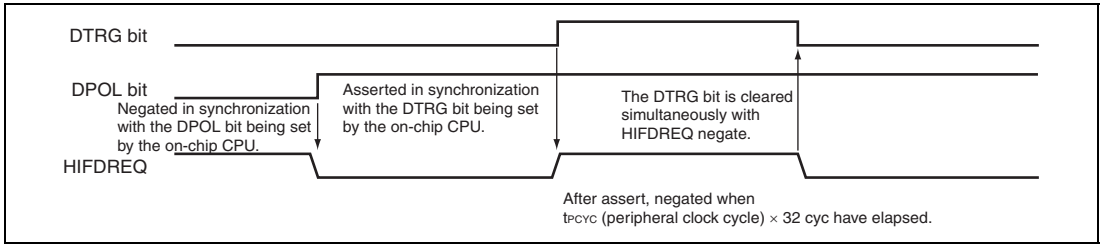


Figure 21.10 HIFDREQ Timing (When DMD = 1 and DPOL = 1)

When the external DMAC supports intermittent operating mode (block transfer mode), efficient data transfer can be implemented by using the HIFRAM consecutive access and bank functions.

Table 21.5 Consecutive Write Procedure to HIFRAM by External DMAC

	External Device		This LSI	
No.	**CPU**	**DMAC**	**HIF**	**CPU**
1	HIF initial setting			HIF initial setting
2	DMAC initial setting			
3	Set HIFADR to HIFRAM end address – 8			
4	Select HIFDATA and write dummy data (4 bytes) to HIFDATA			
5	Set HIFRAM consecutive write with address increment in HIFMCR			
6	Select HIFDATA and write dummy data (4 bytes) to HIFDATA →		→ HIF bank interrupt occurs	→ HIFRAM bank switching by HIF bank interrupt handler (external device accesses bank 1 and on-chip CPU accesses bank 0)
7		Activate DMAC	← Assert HIFDREQ	← Set DTRG bit to 1
8		Consecutive data write to bank 1 in HIFRAM		
9		Write to end address of bank 1 in HIFRAM completes and operation halts	→ HIF bank interrupt occurs	→ HIFRAM bank switching by HIF bank interrupt handler (external device accesses bank 0 and on-chip CPU accesses bank 1)
10		Re-activate DMAC	← Assert HIFDREQ	← Set DTRG bit to 1

	External Device		This LSI			
No.	**CPU**	**DMAC**		**HIF**		**CPU**
11		Consecutive data write to bank 0 in HIFRAM				Read data from bank 1 in HIFRAM
12		Write to end address of bank 0 in HIFRAM completes and operation halts	→	HIF bank interrupt occurs	→	HIFRAM bank switching by HIF bank interrupt handler (external device accesses bank 1 and on-chip CPU accesses bank 0)
13		Re-activate DMAC	←	Assert HIFDREQ	←	Set DTRG bit to 1
Hereafter No. 11 to 13 are repeated. When a register other than HIFDATA is accessed (except that HIFGSR read with HIFRS = low), HIFRAM consecutive write is interrupted, and No. 3 to 6 need to be done again.						

Table 21.6 Consecutive Read Procedure from HIFRAM by External DMAC

	External Device		This LSI	
No.	**CPU**	**DMAC**	**HIF**	**CPU**
1	HIF initial setting			HIF initial setting
2	DMAC initial setting			
3	Set HIFADR to HIFRAM start address			
4	Set HIFRAM consecutive read with address increment in HIFMCR			
5	Select HIFDATA			
6				Write data to bank 1 in HIFRAM

	External Device			This LSI		
No.	**CPU**	**DMAC**		**HIF**		**CPU**
7						After writing data to end address of bank 1 in HIFRAM, perform HIFRAM bank switching (external device accesses bank 1 and on-chip CPU accesses bank 0)
8		Activate DMAC	←	Assert HIFDREQ	←	Set DTRG bit to 1
9		Consecutive data read from bank 1 in HIFRAM				Write data to bank 0 in HIFRAM
10		Read from end address of bank 1 in HIFRAM completes and operation halts	→	HIF bank interrupt occurs	→	HIFRAM bank switching by HIF bank interrupt handler (external device accesses bank 0 and on-chip CPU accesses bank 1)
11		Re-activate DMAC	←	Assert HIFDREQ	←	Set DTRG bit to 1
12		Consecutive data read from bank 0 in HIFRAM				Write data to bank 1 in HIFRAM
13		Read from end address of bank 0 in HIFRAM completes and operation halts	→	HIF bank interrupt occurs	→	HIFRAM bank switching by HIF bank interrupt handler (external device accesses bank 1 and on-chip CPU accesses bank 0)
14		Re-activate DMAC	←	Assert HIFDREQ	←	Set DTRG bit to 1

Hereafter No. 12 to 14 are repeated. When a register other than HIFDATA is accessed (except that HIFGSR read with HIFRS = low), HIFRAM consecutive read is interrupted, and No. 3 to 5 need to be done again.

21.8 Alignment Control

Tables 21.7 and 21.8 show the alignment control when an external device accesses the HIFDATA register, and the HIF registers other than the HIFDATA register, respectively.

Table 21.7 HIFDATA Register Alignment for Access by an External Device

Data in HIFDATA	WBSWP Bit	BO Bit	BYTE[1:0] Bits	Alignment in HIFD[15:0] Pins
H'76543210	0	0	B'00	H'7654
			B'10	H'3210
		1	B'00	H'3210
			B'10	H'7654
	1	0	B'00	H'1032
			B'10	H'5476
		1	B'00	H'5476
			B'10	H'1032

Table 21.8 HIF Registers (other than HIFDATA) Alignment for Access by an External Device

Data in HIFDATA	WBSWP Bit	BO Bit	BYTE[1:0] Bits	Alignment in HIFD[15:0] Pins
H'76543210	Don't care	0	B'00	H'7654
			B'10	H'3210
		1	B'00	H'3210
			B'10	H'7654

21.9 Interface When External Device Power is Cut Off

When the power supply of an external device interfacing with the HIF is cut off, intermediate levels may be applied to the HIF input pins or the HIF output pins may drive an external device not powered, thus causing the device to be damaged. The HIFEBL pin is provided to prevent this from happening. The system power monitor block controls this pin in synchronization with the cutoff of the external device power so that all pins of this module excluding HIFMD can be set to the high-impedance state. Figure 21.11 shows an image of high-impedance control of the HIF pins. Table 21.9 lists the input/output control for the HIF pins.

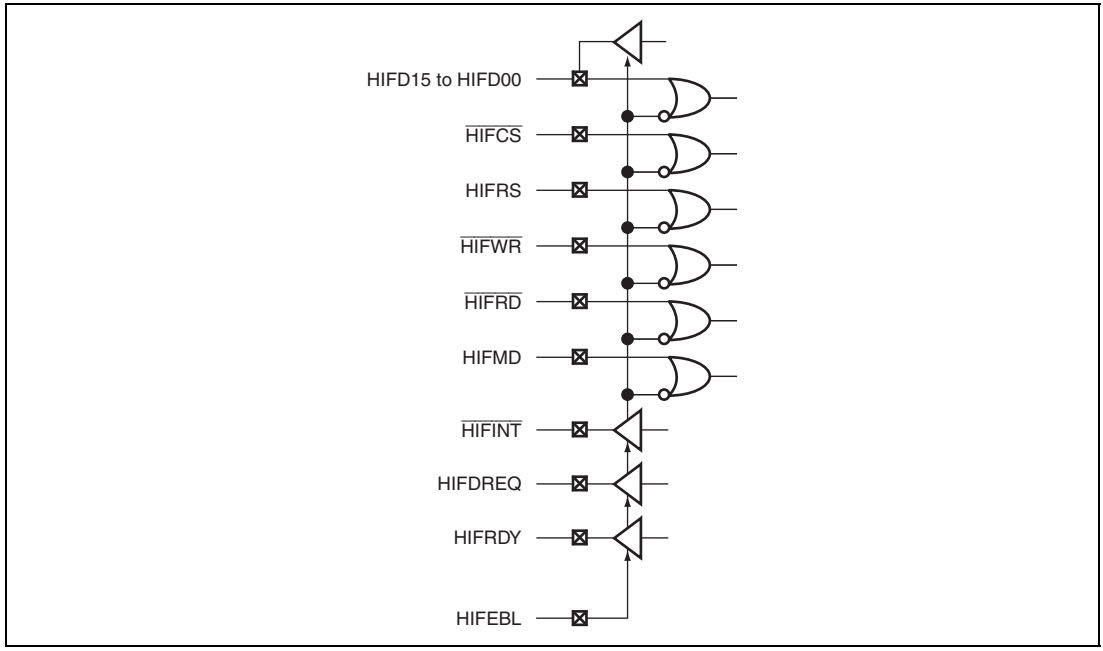


Figure 21.11 Image of High-Impedance Control of HIF Pins by HIFEBL Pin

Table 21.9 Input/Output Control for HIF Pins

LSI Status	**Reset State by $\overline{\text{RES}}$ Pin**		**Reset by $\overline{\text{RES}}$ Pin is Released**		**Reset by $\overline{\text{RES}}$ Pin is Released**	
HIFMD (MD19) input level	**High (Boot setting) Select HIF function.**		**The reset by the $\overline{\text{RES}}$ pin is released while the HIFMD signal is high. (boot mode established) Select HIF function.**		**The reset by the $\overline{\text{RES}}$ pin is released while the HIFMD signal is low. (non-boot mode established) Then, select HIF function by setting the PFC registers.**	
HIFEBL input level	**Low**	**High**	**Low**	**High**	**Low**	**High**
HIFRDY output control	Output buffer: On (Low output)	Output buffer: On (Low output)	Output buffer: Off	Output buffer: On (Sequence output)	Output buffer: Off	Output buffer: On (Sequence output)
$\overline{\text{HIFINT}}$ output control	Output buffer: Off	Output buffer: Off	Output buffer: Off	Output buffer: On (Sequence output)	Output buffer: Off	Output buffer: On (Sequence output)
HIFDREQ output control	Output buffer: Off	Output buffer: Off	Output buffer: Off	Output buffer: On (Sequence output)	Output buffer: Off	Output buffer: On (Sequence output)
HIFD15 to HIFD0 I/O control	I/O buffer: Off	I/O buffer: Off	I/O buffer: Off	I/O buffer controlled according to states of $\overline{\text{HIFCS}}$, $\overline{\text{HIFWR}}$, and $\overline{\text{HIFRD}}$	I/O buffer: Off	I/O buffer controlled according to states of $\overline{\text{HIFCS}}$, $\overline{\text{HIFWR}}$, and $\overline{\text{HIFRD}}$
$\overline{\text{HIFCS}}$ input control	Input buffer: Off	Input buffer: Off	Input buffer: Off	Input buffer: On	Input buffer: Off	Input buffer: On
HIFRS input control	Input buffer: Off	Input buffer: Off	Input buffer: Off	Input buffer: On	Input buffer: Off	Input buffer: On
$\overline{\text{HIFWR}}$ input control	Input buffer: Off	Input buffer: Off	Input buffer: Off	Input buffer: On	Input buffer: Off	Input buffer: On
$\overline{\text{HIFRD}}$ input control	Input buffer: Off	Input buffer: Off	Input buffer: Off	Input buffer: On	Input buffer: Off	Input buffer: On

Note: Not setting a PFC register to select a pin for the HIFEBL pin function means that the input and/or output buffers remain off when pins are selected for these HIF module functions. Such settings are prohibited.

Section 22 Controller Area Network (RCAN-TL1)

Renesas CAN time trigger level 1 (RCAN-TL1, hereafter RCAN) is a module to control the controller area network (CAN) that provides realtime communication for automobiles or industrial equipment systems.

This LSI is equipped with two RCAN interfaces (RCAN0 and RCAN1). No distinction between the two is made in most of this section.

This section describes the program interface with the RCAN.

For the CAN data link controller functions, refer to the references listed below:

References:

1. CAN Specification Version 2.0 part A, Robert Bosch GmbH, 1991
2. CAN Specification Version 2.0 part B, Robert Bosch GmbH, 1991
3. Implementation Guide for the CAN Protocol, CAN Specification 2.0 Addendum, CAN In Automation, Erlangen, Germany, 1997
4. Road vehicles - Controller area network (CAN): Part 1: Data link layer and physical signaling (ISO-CD-11898-1, 2003)
5. CAN License Specification, Robert Bosch GmbH, 1992

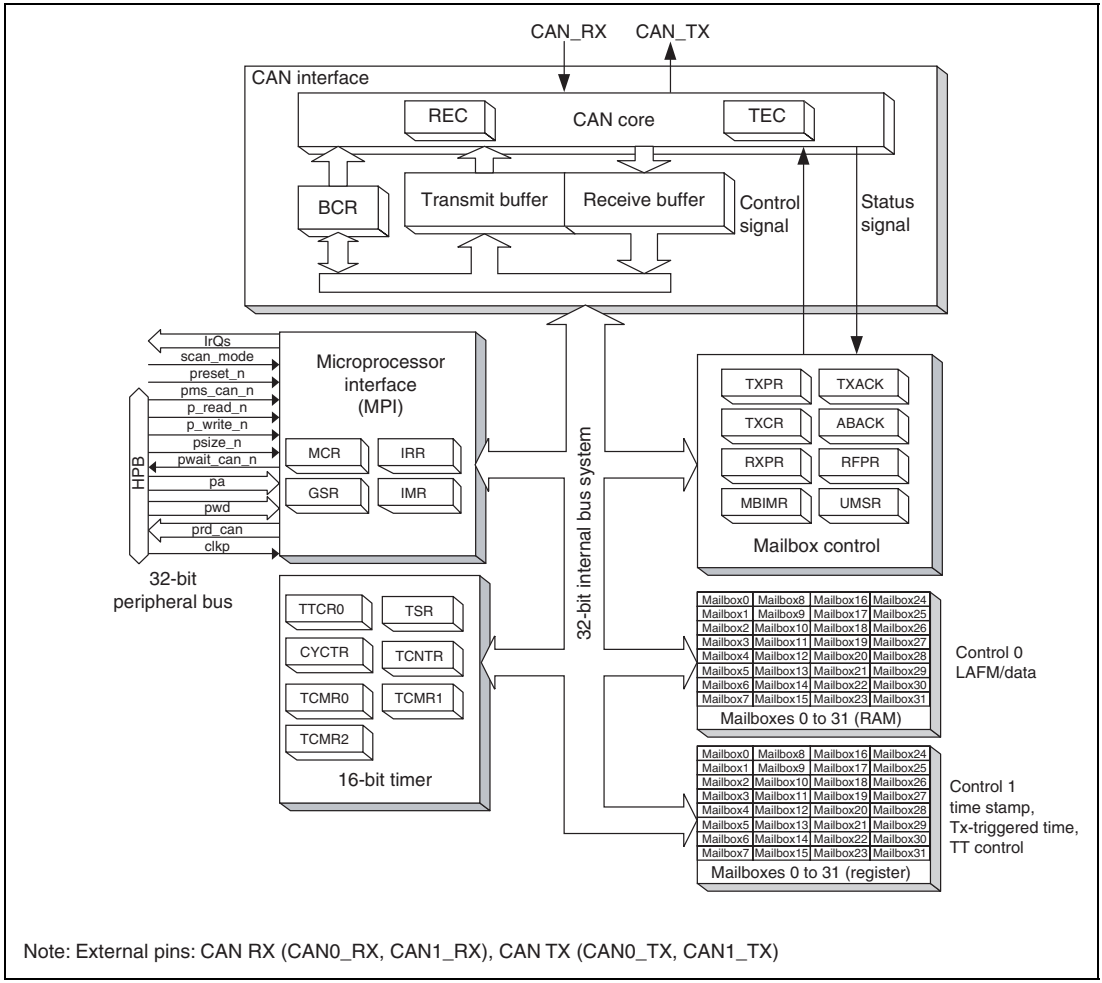
22.1 Features

- The RCAN has the following features.
- Supports CAN specification 2.0B
- Two RCAN interfaces (RCAN0 and RCAN1)
- Bit timing compliant with ISO-11898-1
- 32 mailboxes
- 31 programmable mailboxes for transmission/reception and one receive-only mailbox
- CAN sleep mode for low power consumption and automatic recovery from CAN sleep mode by detecting CAN bus activity
- Programmable receive filter mask (standard and extended identifiers) supported by all mailboxes
- Programmable CAN data rate up to 1 Mbps
- Transmit message queuing with internal priority sorting mechanism against the problem of priority inversion for realtime applications
- Flexible interrupt sources
- On-chip test functions (receive-only mode and error passive mode)
- 16-bit free running timer with flexible clock sources, prescaler, and three timer compare match registers
- Interrupt generation by the timer compare match registers
- Timer counter clear/set capability
- Flexible timestamp at SOF for both transmission and reception supported
- Periodic transmission supported (in addition to event trigger transmission)
- Supports external clock synchronous mode (allowing the input of external clock signals at frequencies from 16 to 50 MHz).

Note: The time taking for RCAN to become operational after the system has been reset depends on the execution time of the OS and the software. Take care on this point if the module is to be used in an application that imposes restrictions on reaction time.

Figure 22.1 shows the block diagram of the RCAN Module.

The RCAN device offers a flexible way to organize and control CAN frames, providing the compliance to CAN2.0B Active and ISO-11898-1. The RCAN consists of five function blocks: microprocessor interface (MPI), mailbox, mailbox control, timer, and CAN interface.



Note: External pins: CAN RX (CAN0_RX, CAN1_RX), CAN TX (CAN0_TX, CAN1_TX)

Figure 22.1 Block Diagram of RCAN

Figure 22.2 is a schematic view of external clock synchronous mode.

The internal clock (clkp) generated by the CPG or the external clock input from the external-clock pin (CAN_CLK) can be selected as the RCAN operating clock by the settings of the RCAN0CTL register (for RCAN0) and RCAN1CTL register (for RCAN1) in the HPB address space.

clkp is selected in the initial state. To select external clock input, make the appropriate settings in RCAN0CTL or RCAN1CTL before access to the RCAN module. When using the external clock, be sure to wait until the clock oscillation is stable before access to the RCAN registers. Modifying the RCAN0CTL and RCAN1CTL settings during RCAN operation is prohibited (refer to section 6, HPB for details of RCAN0CTL and RCAN1CTL).

"Peripheral bus clock" in this section refers to clkp in internal clock synchronous mode or CAN_CLK in external clock synchronous mode.

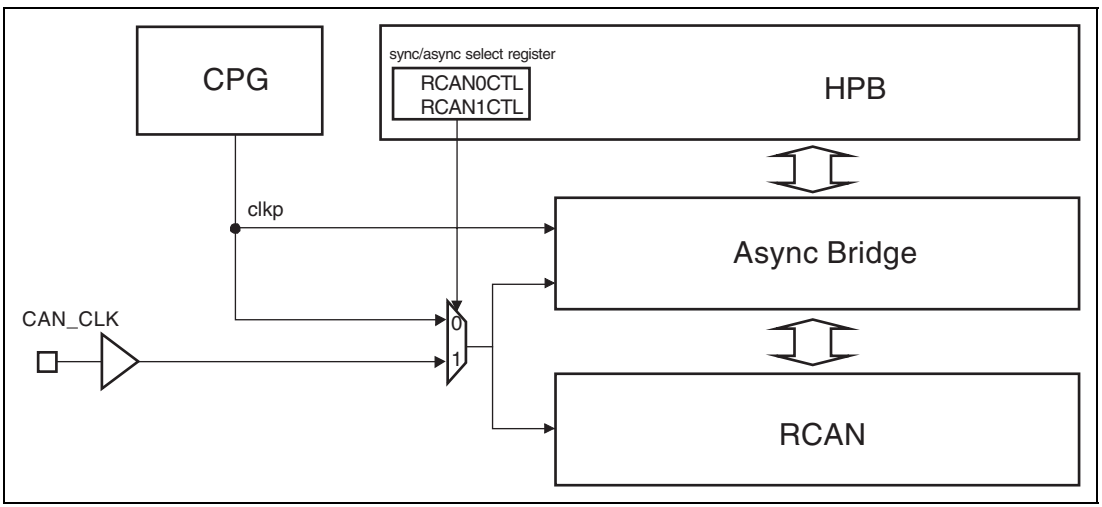


Figure 22.2 Schematic View of External Clock Synchronous Mode

22.1.1 Functions of Each Block

(1) Micro Processor Interface (MPI)

The MPI allows communication between the CPU and the RCAN registers or mailboxes to control the memory interface. It also contains the wakeup control logic that detects the CAN bus activities and notifies the MPI and the other parts of RCAN so that the RCAN can automatically exit the sleep mode.

It contains registers such as MCR, IRR, GSR and IMR.

(2) Mailboxes

The mailboxes are located in RAM and registers as message buffers. There are 32 mailboxes in RAM and registers, and each mailbox has the following information.

a) RAM:

- CAN message control (ID, RTR IDE, etc)
- CAN message data (for CAN data frames)
- Local acceptance filter mask for reception (LAFM)

b) Registers:

- CAN message control (DLC)
- Time stamp for message reception/transmission
- 3-bit mailbox configuration, automatic re-transmission disable bit, auto-transmission for remote request bit, new message control bit

(3) Mailbox Control

The mailbox control supports the following functions.

- During message reception, the mailbox control compares the IDs and generates appropriate RAM addresses/data to store messages from the CAN interface into the mailbox, and sets or clears corresponding registers.
- To transmit event-triggered messages, the mailbox control operates the internal arbitration to select the message of correct priority loads the message from the mailbox to the transmission-buffer of the CAN interface, and sets or clears corresponding registers.
- The mailbox control arbitrates the mailbox accesses between the CPU and the mailbox control.

- The mailbox control incorporates the TXPR, TXCR, TXACK, ABACK, RXPR, RFPR, MBIMR, and UMSR registers.

(4) Timer

The timer is provided to transmit a message within a specific time frame and to record the result.

The timer is a 16-bit free-running up counter which can be controlled by the CPU. The time incorporates one 16-bit compare match register which is compared with the local time and two 16-bit compare match registers which are compared with the cycle time. These compare match registers can generate interrupt signals and clear the counter.

The clock period of the timer can be selected from the clock periods generated based on the system clock. It can also be programmed to be incremented by a bit timing of the CAN bus.

The timer incorporates the TCNTR, TTCR0, TSR, CYCTR, TCMR0, TCMR1 and TCMR2 registers.

(5) CAN Interface

The CAN interface conforms to the requirements for the CAN bus data link controller which is specified in references [2] and [4]. The CAN interface satisfies all the functions of a standard data link controller (DLC) defined by the OSI model. It also provides the registers and the logic which are specific to the CAN bus. For example, it provides the receive error counter, transmit error counter, the bit configuration registers and various useful test modes. It also supports the function to store data received from or transmitted to the CAN data link controller.

22.2 Pin Configuration

Table 22.1 shows the RCAN pin configuration.

Table 22.1 Pin Configuration

Name	Pin Name	I/O	Function
Transmit data pin	CAN0_TX, CAN1_TX	Output	Transmit data pin for the CAN bus
Receive data pin	CAN0_RX, CAN1_RX	Input	Receive data pin for the CAN bus
External clock input pin	CAN_CLK	Input	Input pin for clock signal in external clock synchronous mode

22.3 Register Description

Tables 22.2 to 22.4 show a list of registers, register configurations, and register states in each processing mode, respectively. Do not write to any addresses other than listed below. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than listed below are undefined.

[Legend]

Initial value:	Register value after a reset
—:	Undefined value
R/W:	Readable/writable. The written value can be read.
R:	Read-only. The write value should always be 0.
R/WC1:	Readable/writable. Only writing 1 initializes the bit. Writing 0 is ignored.

Table 22.2 List of Registers

Register Name		Abbreviation	Function
Control register	Master control register	MCR	Setting of RCAN or test mode
	General status register	GSR	RCAN status register
	Bit configuration registers 0 and 1	BCR1, BCR0	Timing setting for baud rate setting
	Interrupt request register	IRR	Interrupt request status
	Interrupt mask register	IMR	Masking of interrupt request
	Transmit error counter/Receive error counter	TEC/REC	For test only
Mailbox control register	Transmit pending registers 1 and 0	TXPR1, TXPR0	Transmission request
	Transmit cancel registers 1 and 0	TXCR1, TXCR0	Transmission abort request
	Transmit acknowledge registers 1 and 0	TXACK1, TXACK0	Flags indicating whether transmission has succeeded or not.
	Abort acknowledge registers 1 and 0	ABACK1, ABACK0	Flags indicating that a transmission has been aborted.

Register Name		Abbreviation	Function
Mailbox control register	Data frame receive pending registers 1 and 0	RXPR1, RXPR0	Flags indicating that the mailbox has received a data frame.
	Remote frame receive pending registers 1 and 0	RFPR1, RFPR0	Flags indicating that the mailbox received a remote frame.
	Mailbox interrupt mask registers 1 and 0	MBIMR1, MBIMR0	Masking of mailbox interrupts
	Unread message status registers 1 and 0	UMSR1, UMSR0	Flags indicating that an unread received message has been overwritten.
Timer register	Time trigger control register 0	TTCR0	Timer setting
	Timer status register	TSR	Timer status flag
	Timer counter register	TCNTR	Indicates the value of the free running timer.
	Cycle time register	CYCTR	CYCTR = TCNTR
	Timer compare match registers 0 to 2	TCMRi (i = 0 to 2)	Values for comparison with the timer counter

The following are the RCAN register base addresses:

RCAN0: H'FFFD 0000

RCAN1: H'FFFD 1000

Table 22.3 Register Configurations

Name	Abbreviation	R/W	Offset Address from Base Address	Access Size
Master control register	MCR	R/W	H'000	16
General status register	GSR	R	H'002	16
Bit configuration register 1	BCR1	R/W	H'004	16
Bit configuration register 0	BCR0	R/W	H'006	16
Interrupt request register	IRR	R/W	H'008	16
Interrupt mask register	IMR	R/W	H'00A	16
Transmit error counter/receive error counter	TEC/REC	R/W	H'00C	16
Transmit pending register 1	TXPR1	R/W	H'020	32
Transmit pending register 0	TXPR0	R/W	H'022	—
Transmit cancel register 1	TXCR1	R/W	H'028	16/32
Transmit cancel register 0	TXCR0	R/W	H'02A	16
Transmit acknowledge register 1	TXACK1	R/WC1	H'030	16/32
Transmit acknowledge register 0	TXACK0	R/WC1	H'032	16
Abort acknowledge register 1	ABACK1	R/WC1	H'038	16/32
Abort acknowledge register 0	ABACK0	R/WC1	H'03A	16
Data frame receive pending register 1	RXPR1	R/WC1	H'040	16/32
Data frame receive pending register 0	RXPR0	R/WC1	H'042	16
Remote frame receive pending register 1	PFPR1	R/WC1	H'048	16/32
Remote frame receive pending register 0	PFPR0	R/WC1	H'04A	16
Mailbox interrupt mask register 1	MBIMR1	R/W	H'050	16/32
Mailbox interrupt mask register 0	MBIMR0	R/W	H'052	16
Unread message status register 1	UMSR1	R/WC1	H'058	16/32
Unread message status register 0	UMSR0	R/WC1	H'05A	16
Time trigger control register 0	TTCR0	R/W	H'080	16
Timer status register	TSR	R	H'088	16
Timer counter register	TCNTR	R/W*	H'08C	16

Name	Abbreviation	R/W	Offset Address from Base Address	Access Size
Cycle time register	CYCTR	R	H'090	16
Timer compare match register 0	TCMR0	R/W	H'098	16
Timer compare match register 1	TCMR1	R/W	H'09C	16
Timer compare match register 2	TCMR2	R/W	H'0A0	16

Note: * Only writing 1 to clear is accepted.

Table 22.4 Register status for processing modes (for both channels 0 and 1)

Abbreviation	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
MCR	Initialized	Initialized	Retained	Retained	Retained	Initialized
GSR	Initialized	Initialized	Retained	Retained	Retained	Initialized
BCR1	Initialized	Initialized	Retained	Retained	Retained	Initialized
BCR0	Initialized	Initialized	Retained	Retained	Retained	Initialized
IRR	Initialized	Initialized	Retained	Retained	Retained	Initialized
IMR	Initialized	Initialized	Retained	Retained	Retained	Initialized
TEC/REC	Initialized	Initialized	Retained	Retained	Retained	Initialized
TXPR1	Initialized	Initialized	Retained	Retained	Retained	Initialized
TXPR0	Initialized	Initialized	Retained	Retained	Retained	Initialized
TXCR1	Initialized	Initialized	Retained	Retained	Retained	Initialized
TXCR0	Initialized	Initialized	Retained	Retained	Retained	Initialized
TXACK1	Initialized	Initialized	Retained	Retained	Retained	Initialized
TXACK0	Initialized	Initialized	Retained	Retained	Retained	Initialized
ABACK1	Initialized	Initialized	Retained	Retained	Retained	Initialized
ABACK0	Initialized	Initialized	Retained	Retained	Retained	Initialized
RXPR1	Initialized	Initialized	Retained	Retained	Retained	Initialized
RXPR0	Initialized	Initialized	Retained	Retained	Retained	Initialized
PFPR1	Initialized	Initialized	Retained	Retained	Retained	Initialized
PFPR0	Initialized	Initialized	Retained	Retained	Retained	Initialized
MBIMR1	Initialized	Initialized	Retained	Retained	Retained	Initialized
MBIMR0	Initialized	Initialized	Retained	Retained	Retained	Initialized
UMSR1	Initialized	Initialized	Retained	Retained	Retained	Initialized
UMSR0	Initialized	Initialized	Retained	Retained	Retained	Initialized
TTCR0	Initialized	Initialized	Retained	Retained	Retained	Initialized
TSR	Initialized	Initialized	Retained	Retained	Retained	Initialized
TCNTR	Initialized	Initialized	Retained	Retained	Retained	Initialized
CYCTR	Initialized	Initialized	Retained	Retained	Retained	Initialized
TCMR0	Initialized	Initialized	Retained	Retained	Retained	Initialized
TCMR1	Initialized	Initialized	Retained	Retained	Retained	Initialized
TCMR2	Initialized	Initialized	Retained	Retained	Retained	Initialized

22.3.1 Memory Map

The RCAN memory map is shown in figures 22.3 (1) and 22.3 (2).

Note that the addresses in the memory map are given as offsets from this address.

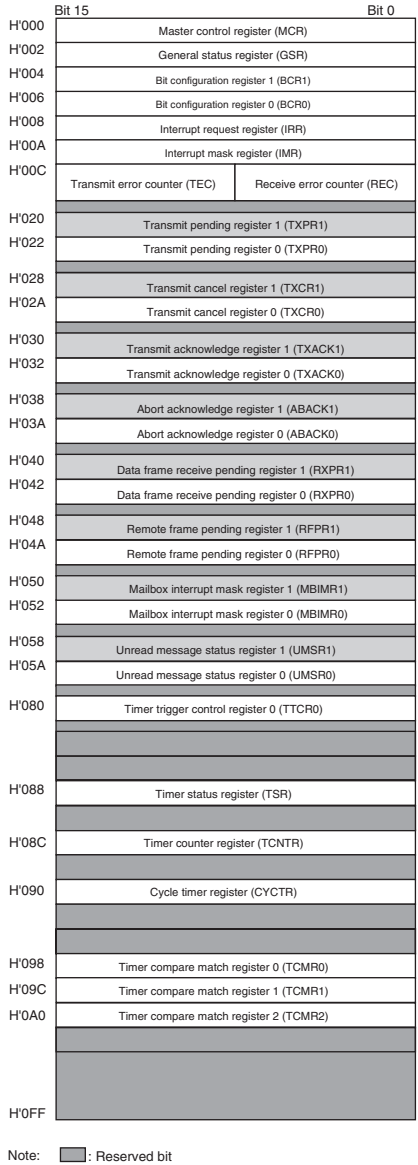


Figure 22.3 RCAN Memory Map (1)

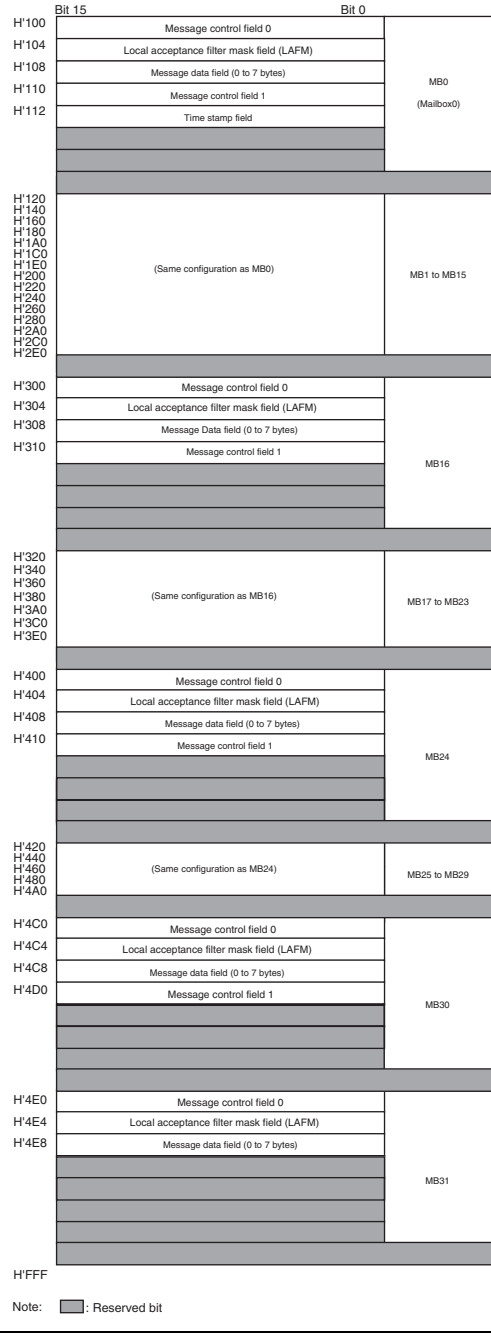


Figure 22.3 RCAN Memory Map (2)

22.4 Mailbox

22.4.1 Mailbox Structure

Mailboxes function as message buffers to transmit or receive CAN frames. Each mailbox is comprised of three identical storage fields: message control, local acceptance filter mask, and message data. In addition, some mailboxes contain the following extra field: time stamp.

Table 22.5 shows the address map of the message control, LAFM, message data, timestamp, and for each mailbox. Note that addresses in table show the offset addresses. To obtain actual addresses, add base address (H'FFDF_0000) to the offset addresses.

Table 22.5 Address Maps for Mailboxes (for One Channel)

	Address				
	RAM			Register	
	Message Control 0	LAFM	Message Data	Message Control 1	Time Stamp
Mailbox (MB)	4 Bytes	4 Bytes	8 Bytes	2 Bytes	2 Bytes
MB0 (receive only)	100 – 103	104– 107	108 – 10F	110 – 111	112 – 113
MB1	120 – 123	124 – 127	128 – 12F	130 – 131	132 – 133
MB2	140 – 143	144 – 147	148 – 14F	150 – 151	152 – 153
MB3	160 – 163	164 - 167	168 – 16F	170 – 171	172 – 173
MB4	180 – 183	184 – 187	188 – 18F	190 – 191	192 – 193
MB5	1A0 – 1A3	1A4 – 1A7	1A8 – 1AF	1B0 – 1B1	1B2 – 1B3
MB6	1C0 – 1C3	1C4 – 1C7	1C8 – 1CF	1D0 – 1D1	1D2 – 1D3
MB7	1E0 – 1E3	1E4 – 1E7	1E8 – 1EF	1F0 – 1F1	1F2 – 1F3
MB8	200 – 203	204 – 207	208 – 20F	210 – 211	212 – 213
MB9	220 – 223	224 – 227	228 – 22F	230 – 231	232 – 233
MB10	240 – 243	244 – 247	248 – 24F	250 – 251	252 – 253
MB11	260 – 263	264 – 267	268 – 26F	270 – 271	272 – 273
MB12	280 – 283	284 – 287	288 – 28F	290 – 291	292 – 293
MB13	2A0 – 2A3	2A4 – 2A7	2A8 – 2AF	2B0 – 2B1	2B2 – 2B3
MB14	2C0 – 2C3	2C4 – 2C7	2C8 – 2CF	2D0 – 2D1	2D2 – 2D3
MB15	2E0 – 2E3	2E4 – 2E7	2E8 – 2EF	2F0 – 2F1	2F2 – 2F3
MB16	300 – 303	304 – 307	308 – 30F	310 – 311	—
MB17	320 – 323	324 – 327	328 – 32F	330 – 331	—
MB18	340 – 343	344 –347	348 – 34F	350 – 351	—
MB19	360 – 363	364 – 367	368 – 36F	370 – 371	—
MB20	380 – 383	384 – 387	388 – 38F	390 – 391	—
MB21	3A0 – 3A3	3A4 – 3A7	3A8 – 3AF	3B0 – 3B1	—
MB22	3C0 – 3C3	3C4 – 3C7	3C8 – 3CF	3D0 – 3D1	—
MB23	3E0 – 3E3	3E4 – 3E7	3E8 – 3EF	3F0 – 3F1	—
MB24	400 – 403	404 – 407	408 – 40F	410 – 411	—
MB25	420 – 423	424 – 427	428 – 42F	430 – 431	—

	Address				
	RAM			Register	
	Message Control 0	LAFM	Message Data	Message Control 1	Time Stamp
Mailbox (MB)	4 Bytes	4 Bytes	8 Bytes	2 Bytes	2 Bytes
MB26	440 – 443	444 – 447	448 – 44F	450 – 451	—
MB27	460 – 463	464 – 467	468 – 46F	470 – 471	—
MB28	480 – 483	484 – 487	488 – 48F	490 – 491	—
MB29	4A0 – 4A3	4A4 – 4A7	4A8 – 4AF	4B0 – 4B1	—
MB30	4C0 – 4C3	4C4 – 4C7	4C8 – 4CF	4D0 – 4D1	4D2 – 4D3 (local time)
MB31	4E0 – 4E3	4E4 – 4E7	4E8 – 4EF	4F0 – 4F1	4F2 – 4F3 (local time)

Mailbox 0 is a receive-only mailbox. All other mailboxes can be used as both receive and transmit mailboxes, depending on the setting of the MBC (Mailbox Configuration) bits in the message control register. Figures 22.4 through 22.6 show the mailbox configuration in details.

Table 22.6 Mailbox Functions

	Event Trigger		Remark
Mailbox (MB)	Transmit (Tx)	Receive (Rx)	Time Stamp Register
MB0	—	Possible by setting MBC bits	Available
MB15-1	Possible by setting MBC bits	Possible by setting MBC bits	Available
MB23-16	Possible by setting MBC bits	Possible by setting MBC bits	—
MB29-24	Possible by setting MBC bits	Possible by setting MBC bits	—
MB30	Possible by setting MBC bits	Possible by setting MBC bits	Available
MB31	Possible by setting MBC bits	Possible by setting MBC bits	Available

MB0 (Receive mailbox with timestamp)

Address	Data bus																Access size	Field name
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
H'100+N*32	IDE	RTR	0	STDID[10:0]											EXTID[17:16]		16/32	Control 0 (RAM)*1*2*3*4
H'102+N*32	EXTID[15:0]																16	
H'104+N*32	IDE LAFM	0	0	STDID_LAFM[10:0]											EXTID LAFM[17:16]		16/32	LAFM*4
H'106+N*32	EXTID_LAFM[15:0]																16	
H'108+N*32	MSG_DATA_0 (first Rx/Tx byte)								MSG_DATA_1								8/16/32	Data
H'10A+N*32	MSG_DATA_2								MSG_DATA_3								8/16	
H'10C+N*32	MSG_DATA_4								MSG_DATA_5								8/16/32	
H'10E+N*32	MSG_DATA_6								MSG_DATA_7								8/16	
H'110+N*32	0	0	NMC	0	0	MBC[2:0]*			0	0	0	0	DLC[3:0]				8/16	Control 1
H'112+N*32	Timestamp[15:0] (CYCTR[15:0] at SOF)																16	Timestamp

[Legend] N: 0 (mailbox number)

Mailboxes 15 to 1 (Mailboxes with timestamp)

Address	Data bus																Address size	Field name
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
H'100+N*32	IDE	RTR	0	STDID[10:0]											EXTID[17:16]		16/32	Control 0 (RAM)*2*4
H'102+N*32	EXTID[15:0]																16	
H'104+N*32	IDE LAFM	0	0	STDID_LAFM[10:0]											EXTID LAFM[17:16]		16/32	LAFM*4
H'106+N*32	EXTID_LAFM[15:0]																16	
H'108+N*32	MSG_DATA_0 (first Rx/Tx byte)								MSG_DATA_1								8/16/32	Data
H'10A+N*32	MSG_DATA_2								MSG_DATA_3								8/16	
H'10C+N*32	MSG_DATA_4								MSG_DATA_5								8/16/32	
H'10E+N*32	MSG_DATA_6								MSG_DATA_7								8/16	
H'110+N*32	0	0	NMC	ATX	DART	MBC[2:0]			0	0	0	0	DLC[3:0]				8/16	Control 1
H'112+N*32	Timestamp[15:0] (CYCTR[15:0] at SOF)																16	Timestamp

[Legend] N: 15 to 1 (mailbox number)

Notes: 1. Mailbox 0 is a receive-only mailbox, so in the setting of bits MBC[2:0] for this mailbox, MBC[1] is fixed to 1. Other limitations apply to the setting of MBC[2:0] for mailbox 0.
2. All the bits shadowed in gray of the control 0 (RAM) field are reserved and read as undefined. The write value should always be 0.
3. The ATX and DART bits have transmission functions, so these bits are not supported for mailbox 0 (these bits are reserved in the case of mailbox 0).
4. The ID-order changing bit (MCR.MCR15) of the master control register can be used to change the order of STDID, RTR, IDE and EXTID of the message control 0 field, and STDID_LAFM, RTR_LAFM, IDE_LAFM, and EXTID_LAFM of the LAFM field. The above chart depicts the case where MCR.MCR15 = B'1 (the initial value.)

Figure 22.4 Configuration of Mailboxes 0 to 15

Mailboxes 23 to 16 (Mailboxes without timestamp)

Address	Data bus																Access size	Field name
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
H'100+N*32	IDE	RTR	0	STDID[10:0]											EXTID[17:16]		16/32	Control 0
H'102+N*32	EXTID[15:0]																16	(RAM)*1,*2
H'104+N*32	IDE LAFM	0	0	STDID_LAFM[10:0]											EXTID LAFM[17:16]		16/32	LAFM*1,*2
H'106+N*32	EXTID_LAFM[15:0]																16	
H'108+N*32	MSG_DATA_0 (first Rx/Tx byte)								MSG_DATA_1								8/16/32	Data
H'10A+N*32	MSG_DATA_2								MSG_DATA_3								8/16	
H'10C+N*32	MSG_DATA_4								MSG_DATA_5								8/16/32	
H'10E+N*32	MSG_DATA_6								MSG_DATA_7								8/16	
H'110+N*32	0	0	NMC	ATX	DART	MBC[2:0]			0	0	0	0	DLC[3:0]				8/16	Control 1

[Legend] N: 23 to 16 (mailbox number)

Mailboxes 29 to 24

Address	Data bus																Access size	Field name
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
H'100+N*32	IDE	RTR	0	STDID[10:0]											EXTID[17:16]		16/32	Control 0
H'102+N*32	EXTID[15:0]																16	(RAM)
H'104+N*32	IDE LAFM	0	0	STDID_LAFM[10:0]											EXTID LAFM[17:16]		16/32	LAFM
H'106+N*32	EXTID_LAFM[15:0]																16	
H'108+N*32	MSG_DATA_0 (first Rx/Tx byte)								MSG_DATA_1								8/16/32	Data
H'10A+N*32	MSG_DATA_2								MSG_DATA_3								8/16	
H'10C+N*32	MSG_DATA_4								MSG_DATA_5								8/16/32	
H'10E+N*32	MSG_DATA_6								MSG_DATA_7								8/16	
H'110+N*32	0	0	NMC	ATX	DART	MBC[2:0]			0	0	0	0	DLC[3:0]				8/16	Control 1
H'112+N*32	Reserved																—	—

[Legend] N: 29 to 24 (mailbox number)

Notes: 1. All the bits shadowed in gray of the control 0 (RAM) field are reserved and read as undefined. The write value should always be 0.
2. The ID-order changing bit (MCR.MCR15) of the master control register can be used to change the order of STDID, RTR, IDE and EXTID of the message control 0 field, and STDID_LAFM, RTR_LAFM, IDE_LAFM, and EXTID_LAFM of the LAFM field. The above chart depicts the case where MCR.MCR15 = B'1 (the initial value.)

Figure 22.5 Configuration of Mailboxes 16 to 29

Mailbox 30

Address	Data bus																Access size	Field name
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
H'100+N*32	IDE	RTR	0	STDID[10:0]											EXTID[17:16]		16/32	Control 0
H'102+N*32	EXTID[15:0]																16	(RAM)*1,*2
H'104+N*32	IDE_LAFM	0	0	STDID_LAFM[10:0]											EXTID_LAFM[17:16]		16/32	LAFM*1,*2
H'106+N*32	EXTID_LAFM[15:0]																16	
H'108+N*32	MSG_DATA_0 (first Rx/Tx byte)								MSG_DATA_1								8/16/32	Data
H'10A+N*32	MSG_DATA_2								MSG_DATA_3								8/16	
H'10C+N*32	MSG_DATA_4								MSG_DATA_5								8/16/32	
H'10E+N*32	MSG_DATA_6								MSG_DATA_7								8/16	
H'110+N*32	0	0	NMC	ATX	DART	MBC[2:0]			0	0	0	0	DLC[3:0]				8/16	Control 1

[Legend] N: 30 (mailbox number)

Mailbox 31

Address	Data bus																Access size	Field name
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
H'100+N*32	IDE	RTR	0	STDID[10:0]											EXTID[17:16]		16/32	Control 0
H'102+N*32	EXTID[15:0]																16	(RAM)
H'104+N*32	IDE_LAFM	0	0	STDID_LAFM[10:0]											EXTID_LAFM[17:16]		16/32	LAFM
H'106+N*32	EXTID_LAFM[15:0]																16	
H'108+N*32	MSG_DATA_0 (first Rx/Tx byte)								MSG_DATA_1								8/16/32	Data
H'10A+N*32	MSG_DATA_2								MSG_DATA_3								8/16	
H'10C+N*32	MSG_DATA_4								MSG_DATA_5								8/16/32	
H'10E+N*32	MSG_DATA_6								MSG_DATA_7								8/16	

[Legend] N: 31 (mailbox number)

Notes: 1. All the bits shadowed in gray of the control 0 (RAM) field are reserved and read as undefined. The write value should always be 0.
2. The ID-order changing bit (MCR.MCR15) of the master control register can be used to change the order of STDID, RTR, IDE and EXTID of the message control 0 field, and STDID_LAFM, RTR_LAFM, IDE_LAFM, and EXTID_LAFM of the LAFM field. The above chart depicts the case where MCR.MCR15 = B'1 (the initial value.)

Figure 22.6 Configuration of Mailboxes 30 and 31

22.4.2 Message Control Field

(1) Message Control 0

- Mailbox N Message Control 0 (MB[N].CONTROL0H + MB[N].CONTROL0L) (N = 0 to 31)

Register Name	Address	Bit	Bit Name	Description
MB[N] CONTROL0H	H'100 + N × 32	15	IDE	Identifier Extension Used to distinguish between standard format and extended format in CAN data and remote frames. 0: Standard format 1: Extended format
		14	RTR	Remote Transmission Request Used to distinguish between a data frame and remote frame. This bit is modified depending on whether the received CAN frame is data frame or remote frame. Important: Note that when the ATX bit is set with the setting MBC[2:0] = B'001, the RTR bit cannot be set. When a remote frame is received, the CPU will be notified by the corresponding RFPR bit or IRRT (remote frame receive interrupt) bit. However, as the RCAN needs to transmit the current message as a data frame, the RTR bit remains unchanged. Important: In order to support automatic answer to a remote frame when MBC[2:0] = B'001 and ATX = 1, the RTR flag should be cleared to 0 to enable data frame transmission. 0: Data frame 1: Remote frame
		13	—	Reserved The initial value is undefined. The write value should always be 0.
		12 to 2	STDID [10:0]	Standard ID These bits set ID (standard ID) of a data frame and remote frame.
		1, 0	EXTID [17:16]	Extended ID These bits set ID (extended ID) of a data frame and remote frame.
MB[N] CONTROL0L	H'102+ N × 32	15 to 0	EXTID [15:0]	

[Legend] N: Mailbox numbers 31 to 0

(2) Message Control 1

- Mailbox 0 Message Control 1 (MB0.CONTROL1)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	NMC	—	—	MBC[2:0]			—	—	—	—	DLC[3:0]			
Initial value:	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R	R	R/W	R	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial value	R/W	Description
15, 14	—	00	R	Reserved These bits are always read as 0. The write value should always be 0.
13	NMC	0	R/W	New Message Control When this bit is cleared to 0, the mailbox where the RXPR or RFPR bit has already been set does not store the new message but maintains the previous one and sets the corresponding UMSR bit. When this bit is set to 1, the mailbox where the RXPR or RFPR bit has already been set is rewritten by the new message and sets the corresponding UMSR bit. Important: Note that if a remote frame is rewritten by a data frame or if a data frame is rewritten by a remote frame, both RXPR and RFPR flags (together with UMSR) are set simultaneously in the same mailbox. In this case, the RTR bit in the mailbox control field will also be rewritten. 0: Overrun mode 1: Overwrite mode
12, 11	—	00	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial value	R/W	Description
10	MBC[2]	1	R/W	Mailbox Configuration
9	MBC[1]	1	R	These bits configure the mailbox function as shown in table 22.7. When MBC[2:0] = B'111, the mailbox is inactive, i.e., it does not receive or transmit a message regardless of TXPR and other settings. When MBC[2:0] is set as reception, TSPR should not be set because the RCAN supports no hardware protection and TXPR remains set. MBC[1] of mailbox 0 is fixed to 1 by hardware because mailbox 0 is a receive-only mailbox. x0x: Setting prohibited 010: Mailbox 0 can be used. 011: Mailbox 0 can be used. 110: Setting prohibited 111: Mailbox is inactive. Note: x = Undefined
8	MBC[0]	1	R/W	
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	DLC [3:0]	0000	R/W	Data Length Code These bits encode the number of data bytes from 0 to 8. 0000: 0 byte 0001: 1 byte 0010: 2 bytes 0011: 3 bytes 0100: 4 bytes 0101: 5 bytes 0110: 6 bytes 0111: 7 bytes 1xxx: 8 bytes Note: x = Don't care

Note: The value of bit 9 (MCB[1]) in MBC [2:0] is always 1.

- Mailboxes 1 to 31 Message Control 1 (MB1.CONTROL1 to MB31.CONTROL1)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	NMC	ATX	DART	MBC[2:0]			0	0	0	0	DLC[3:0]			
Initial value:	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial value	R/W	Description
15, 14	—	00	R	Reserved These bits are always read as 0. The write value should always be 0.
13	NMC	0	R/W	New Message Control When this bit is cleared to 0, the mailbox where the RXPR or RFPR bit has already been set does not store the new message but maintains the previous one and sets the corresponding UMSR bit. When this bit is set to 1, the mailbox where the RXPR or RFPR bit has already been set is rewritten by the new message and sets the corresponding UMSR bit. 0: Overrun mode 1: Overwrite mode Important: Note that if a remote frame is rewritten by a data frame or if a data frame is rewritten by a remote frame, both RXPR and RFPR flags (together with UMSR) are set simultaneously in the same mailbox. In this case, the RTR bit in the mailbox control field will also be rewritten.

Bit	Bit Name	Initial value	R/W	Description
12	ATX	0	R/W	Automatic Transmission of Data Frame When this bit is set to 1 and a mailbox receives a remote frame, the DLC is stored and TXPR is set automatically. Then, a data frame is transmitted from the same mailbox using the DLC updated by the same mailbox. The mailboxes, which are specified as automatic transmission, are scheduled according to the ID priority or Mailbox priority depending to the message transmission priority control bit (MCR2). In order to use this function, MBC[2:0] should be set to B'001. When transmission is performed by this function, the data length code (DLC) that has been received will be used. Application needs to guarantee that the DLC of the remote frame corresponds to that of the data frame to be requested. 0: Automatic Transmission of Data Frame disabled 1: Automatic Transmission of Data Frame enabled Important: 1. When ATX is used and MBC[2:0] = B'001, the filter for the IDE bit cannot be used because the ID of remote frame should be exactly the same as that of data frame in the same way as a reply message. 2. When this function is used, note that the RTR bit will not be set even if a remote frame is received. When a remote frame is received, the CPU will be notified by the corresponding RFPR bit setting. However, as the RCAN needs to transmit the current message as a data frame, the RTR bit remains unchanged. 3. Automatic transmission of a remote frame may not be started. Note that the overrun condition (the UMSR flag set when the Mailbox has its NMC=0) the message received is discarded. 4. In case a remote frame is causing overrun into a Mailbox configured with ATX = 1, the automatic transmission request for the previous message might be accepted. Note: This bit is provided only for mailboxes 1 to 31. For mailbox 0, this bit is reserved.

Bit	Bit Name	Initial value	R/W	Description
11	DART	0	R/W	Disable Automatic Re-Transmission When this bit is set to 1, the automatic re-transmission of a message is disabled if a CAN bus error has occurred or if an arbitration lost on the CAN bus has occurred. In effect, when this function is used, the corresponding TXCR bit is automatically set at the start of transmission. When this bit is cleared to 0, the RCAN tries to transmit the message as many times as required until the transmission is successfully completed or it is cancelled by the TXCR. 0: Enables the retransmission 1: Disables the retransmission Note: This bit is provided only for mailboxes 1 to 31. For mailbox 0, this bit is reserved.
10 to 8	MBC [2:0]	111	R/W	Mailbox Configuration These bits configure the mailbox function as shown in table 22.7. When MBC[2:0] = B'111, the mailbox is inactive, i.e., it does not receive or transmit a message regardless of TXPR and other settings. Setting the MBC[2:0] bits to B'110, B'101' and B'100 are prohibited. When the MBC[2:0] is set to any other value, the LAFM field can be used. When MBC[2:0] is set as reception, TSPR should not be set because the RCAN supports no hardware protection and TXPR remains set. MBC[1] of mailbox 0 is fixed to 1 by hardware because mailbox 0 is a receive-only mailbox. Refer to table 22.7, Mailbox Function Setting for details.
7 to 4	—	All 0	—	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial value	R/W	Description
3 to 0	DLC [3:0]	0000	R/W	Data Length Code These bits encode the number of data bytes from 0 to 8. 0000: 0 byte 0001: 1 byte 0010: 2 bytes 0011: 3 bytes 0100: 4 bytes 0101: 5 bytes 0110: 6 bytes 0111: 7 bytes 1xxx: 8 bytes Note: x = Undefined

Table 22.7 Mailbox Function Setting

MBC[2]	MBC[1]	MBC[0]	Data Frame Trans-mission	Remote Frame Trans-mission	Data Frame Recep-tion	Remote Frame Recep-tion	Remarks
0	0	0	Enabled	Enabled	Disabled	Disabled	• Not allowed for mailbox 0
0	0	1	Enabled	Enabled	Disabled	Enabled	• Can be used with ATX* • Not allowed for mailbox 0 • LAFM can be used
0	1	0	Disabled	Disabled	Enabled	Enabled	• Allowed for mailbox 0 • LAFM can be used
0	1	1	Disabled	Disabled	Enabled	Disabled	• Allowed for mailbox 0 • LAFM can be used
1	0	0	Setting prohibited				
1	0	1	Setting prohibited				
1	1	0	Setting prohibited				
1	1	1	Mailbox inactive (Initial value)				

Note In order to support automatic retransmission, RTR should be cleared to 0 when MBC = B'001 and ATX = 1. When ATX = 1, the filter for IDE must not be used.

22.4.3 Local Acceptance Filter Mask (LAFM)

When MBC[2:0] is set to B'001, B'010, or B'011, this field is used as the LAFM field. The FAFM field allows a mailbox to accept multiple receive IDs. The LAFM is comprised of two 16-bit readable/writable areas as shown in figure 22.7.

Register name	Address	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Access size	Field name
MB[N].LAFMH	H'104+N*32	IDE_LAFM	0	0	STDID_LAFM[10:0]											EXTID_LAFM[17:16]		16/32	LAFM
MB[N].LAFML	H'106+N*32	EXTID_LAFM[15:0]																16	

[Legend] N: 31 to 0 (mailbox number)

Figure 22.7 Local Acceptance Filter Mask (LAFM)

If a bit in the LAFM field is set, the corresponding bit of a received CAN ID is ignored while the RCAN searches for a mailbox whose CAN ID matches. If a bit in LAFM is cleared, the corresponding bit of a received CAN ID must match STDID/IDE/EXTID set in the mailbox to be stored. The structure of the LAFM is the same as that of the message control in a mailbox. If this function is not used, the LAFM field must be filled with 0s.

Notes: 1. The RCAN starts to search a matching ID from mailbox 31 to mailbox 0. As soon as the RCAN detects a matching ID, it stops searching. The message will be stored regardless of the NMC and RXPR/RFPR flag settings. This means that a received message can only be stored in one mailbox even if the LAFM field function is used.

2. When a message is received and a matching mailbox is found, the whole message is stored in the mailbox. Accordingly, if the LAFM is used, STDID, RTR, IDE and EXTID of the received message may differ from those of the previous settings because they are updated with the STDID, RTR, IDE and EXTID of the received message.

Register Name	Address	Bit	Bit Name	Description
MB[N] LAFMH	H'104+N*32	15	IDE_ LAFM	Filter mask bit for the IDE bit. 0: Enables the corresponding IDE bit. 1: Disables the corresponding IDE bit.
		14, 13	—	Reserved The initial value is undefined. The write value should always be 0.
		12 to 2	STDID_ LAFM [10:0]	Filter mask bit for the STDID[10:0] bits 0: Enables the corresponding STDID bit. 1: Disables the corresponding STDID bit.
		1, 0	EXTID_ LAFM [17:16]	Filter mask bit for the EXTID[10:0] bits 0: Enables the corresponding EXTID bit. 1: Disables the corresponding EXTID bit.
MB[N] LAFML	H'106+N*32	15 to 0	EXTID_ LAFM [15:0]	

[Legend] N: Mailbox numbers 15 to 0

22.4.4 Message Data Fields

The message data field stores the CAN message data that is transmitted or received. MSG_DATA_0 corresponds to the first data byte that is transmitted or received. The bit order on the CAN bus is from bit 7 to bit 0.

22.4.5 Timestamp

The timestamp stores the timestamp recorded in a transmit or receive message. The timestamp will be a useful function to monitor if a message is received or transmitted within expected schedule.

(1) Timestamp

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TS15	TS14	TS13	TS12	TS11	TS10	TS9	TS8	TS7	TS6	TS5	TS4	TS3	TS2	TS1	TS0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

(2) Message Reception

For messages received in mailboxes 15 to 0, the CYCTR[15:0] (cycle time register) value is captured at SOF and stored as the timestamp.

For messages received in mailboxes 30 and 31, the timer counter register (TCNTR) value is captured at SOF and stored as the timestamp.

(3) Message Transmission

For messages transmitted from mailboxes 15 to 0, the cycle time register (CYCTR) value is captured at SOF and stored as the timestamp.

For messages transmitted from mailboxes 30 and 31, the timer counter register (TCNTR) value is captured at SOF and stored as the timestamp.

Important: Note that the timestamp is stored in a temporary register. When transmission or reception is completed successfully, the timestamp value is copied into the relevant mailbox field. If the CPU clears RXPR[N]/RFPR[N] at the same time when UMSR[N] is set by an overrun, only the timestamp may be updated. At this time, the correct timestamp value before clearing RXPR[N]/RFPR[N] can be read.
[Legend] N: Mailbox numbers 31, 30, and 15 to 0

22.5 RCAN Control Registers

The following sections describe the RCAN control registers. These registers can only be accessed in word size (16 bits).

Table 22.8 shows the control registers of the RCAN.

Table 22.8 Configuration of RCAN Control Registers

Register Name	Abbreviation	Address	Access Size (Bits)
Master control register	MCR	H'000	16
General status register	GSR	H'002	16
Bit configuration register 1	BCR1	H'004	16
Bit configuration register 0	BCR0	H'006	16
Interrupt request register	IRR	H'008	16
Interrupt mask register	IMR	H'00A	16
Transmit error counter/ receive error counter	TEC/REC	H'00C	16

22.5.1 Master Control Register (MCR)

The master control register (MCR) is a 16-bit readable/writable register that controls the RCAN.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MCR15	MCR14	—	—	—	TST[2:0]			MCR7	MCR6	MCR5	—	—	MCR2	MCR1	MCR0
Initial value:	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	MCR15	1	R/W	ID Order Change of Mailbox (RAM) Changes the ID order of STDID, RTR, IDE and EXTID of the message control 0 field, and STDID_LAFM, RTR_LAFM, IDE_LAFM, and EXTID_LAFM of the LAFM field. This bit can be modified only in reset mode. For details on ID order change, refer to figure 22.8.

Bit	Bit Name	Initial Value	R/W	Description
14	MCR14	0	R/W	Auto Bus Off Halt If both this bit and MCR6 are set, MCR1 is automatically set as soon as the RCAN enters bus off state This bit can be modified only in reset mode. 0: The RCAN remains in bus off state with the normal recovery sequence (128 x 11 recessive bits) 1: The RCAN enters directly from bus off state to halt mode if MCR6 is set to 1.
13 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 8	TST[2:0]	000	R/W	Test Mode These bits enable or disable the test modes. Note that the RCAN should be placed in halt mode or reset mode before the test mode is activated. This is to avoid that the transition to test mode affects a transmission or reception in progress. For details, refer to section 22.8.2, Test Mode Settings. Note that the test modes can be used only for diagnosis and tests and cannot be used when the RCAN is used in normal operation. 000: Normal mode 001: Listen-only mode (Receive-only mode) 010: Self test mode 1 (External) 011: Self test mode 2 (Internal) 100: Write error counter 101: Error passive mode 110: Setting prohibited 111: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
7	MCR7	0	R/W	Auto-Wake Mode Enables or disables the auto-wake mode. If this bit is set to 1, the RCAN automatically cancels the sleep mode (MCR5) by detecting CAN bus activity (dominant bit). If MCR7 is cleared to 0, the RCAN does not automatically cancel the sleep mode. The RCAN cannot store the message that wakes it up. 0: Disables auto-wake by the CAN bus activity. 1: Enables auto-wake by the CAN bus activity. Note: The MCR7 bit cannot be modified in CAN sleep mode.
6	MCR6	0	R/W	Halt during Bus Off Enables or disables the RCAN to enter Halt mode immediately when MCR1 is set in bus off state. This bit can be modified only in Reset or Halt mode. Note that the CAN engine immediately returns to error active mode when the RCAN enters halt mode in bus off state. 0: Does not enter halt mode in Bus Off state but waits for the end of recovery sequence. 1: Enters halt mode immediately in bus off state if MCR[1] is set.

Bit	Bit Name	Initial Value	R/W	Description
5	MCR5	0	R/W	CAN Sleep Mode Enables or disables a transition to CAN sleep mode. If this bit is set while the RCAN is in halt mode, the transition to CAN sleep mode is enabled. This bit setting becomes valid after the RCAN has entered halt mode. Two error counters (REC and TEC) remain unchanged in sleep mode. CAN sleep mode can be cancelled in two methods: • Writing 0 to MCR5 • Detecting a dominant bit on the CAN bus if MCR7 is valid If auto wake up mode is disabled, the RCAN will ignore all CAN bus activities until CAN sleep mode is cancelled. Before CAN sleep mode is cancelled, the RCAN is synchronized with the CAN bus by checking for 11 recessive bits before entering the CAN Bus activity. This means that, when the second method described above is used, the RCAN cannot receive the first message. In the same way, the CAN transceivers cannot receive the first message when stand by mode is cancelled. Accordingly, the software should be designed considering the above descriptions. In CAN sleep mode, only the MCR, GSR, IRR and IMR registers can be accessed. For details, refer to section 22.8.1 (3), CAN Sleep Mode. 0: CAN sleep mode is cancelled. 1: A transition to CAN sleep mode is enabled. Note: The RCAN should be placed in halt mode before making a transition to CAN sleep mode. This allows the CPU to clear all pending interrupts before entering CAN sleep mode. Once all interrupts are cleared, the RCAN must should make a transition from halt mode to CAN sleep mode simultaneously (by writing MCR[5]=1 and MCR[1]=0 at the same time).
4, 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2	MCR2	0	R/W	Message Transmission Priority Selects the transmission order for pending transmit data. If this bit is set to 1, pending transmit data is sent in order of the bit position in the transmission pending register (TXPR). The transmission starts from mailbox 31 (highest priority) to mailbox 1 (when those mailboxes are configured for transmission). If MCR2 is cleared to 0, all messages for transmission are transmitted depending on their ID priority (by operating the internal arbitration). The message of the highest priority has the arbitration field (STDID + IDE bit + EXTID (if IDE=1) + RTR bit) with the lowest digital value and it is transmitted first. The internal arbitration includes the RTR and IDE bits (the internal arbitration operates in the same way as the arbitration on the CAN bus between two CAN nodes starting transmission simultaneously). This bit can be modified only in reset or halt mode. 0: Transmits depending on the message ID priority 1: Transmits depending on the mailbox numbers (from mailbox 31 to mailbox 1)

Bit	Bit Name	Initial Value	R/W	Description
1	MCR1	0	R/W	Halt Request Setting the MCR1 bit to 1 causes the CAN controller to complete its current operation and then enter halt mode (then it is cut off from the CAN bus). The RCAN remains in halt mode until the MCR1 is cleared. In halt mode, the CAN Interface does not perform the CAN bus activity and does not store or transmit messages. All the user registers (including mailbox contents and TEC/REC) remain unchanged, with the exception of IRR0 and GSR4 which are used to notify the CPU of the halt status. If the CAN bus is in idle or intermission state, the RCAN will enter halt mode within one bit time regardless of the MCR6 setting. If MCR6 is set, a halt request during Bus Off will also be processed within one bit time. In other cases, the RCAN waits for the completion of the bus off recovery sequence and then enters halt mode. Entering the halt mode can be notified by IRR0 and GSR4. If both MCR14 and MCR6 are set, MCR1 is automatically set as soon as the RCAN enters bus off state. In halt mode, the RCAN configuration can be modified with the exception of the bit timing setting, because the RCAN does not perform the bus activity. To rejoin the CAN bus activity, the MCR1 bit should be cleared by writing 0. After this bit has been cleared, the RCAN waits until it detects 11 recessive bits, and then joins the CAN bus. 0: Clears the halt request 1: Requests a transition to halt mode Notes: 1. After issuing a halt request, the CPU cannot access TXPR or TXCR or clear MCR1 until the transition to halt mode is completed (notified by IRR0 and GSR4). After MCR1 has been set, MCR1 can be cleared only after entering halt mode or by reset operation (software or hardware). 2. A transition to or from halt mode is enabled only when an appropriate baud rate is selected by the BCR1 and BCR0 registers.

Bit	Bit Name	Initial Value	R/W	Description
0	MCR0	0	R/W	Reset Request Controls resetting of the RCAN module. When this bit changes from 0 to 1, the RCAN controller enters its reset routine, re-initializes the internal logic, then sets GSR3 and IRR0 to notify the reset mode. During a re-initialization, all user registers are initialized. The RCAN can be re-configured while the MCR0 bit is set. This bit should be cleared by writing 0 to join the CAN bus operation. After this bit has been cleared, the RCAN module waits until it detects 11 recessive bits, and then joins the CAN bus operation. The baud rate should be set appropriately in order to sample the value on the CAN bus. After power-on reset, this bit and GSR3 are always set to 1. This means that a reset has been requested and the RCAN needs to be configured. The reset request is equivalent to a power -on reset, but it is controlled by software. 0: Clears a reset request [Clearing condition] 0 is written to this bit after the RCAN has been reset. 1: Requests a transition to reset mode of CAN interface

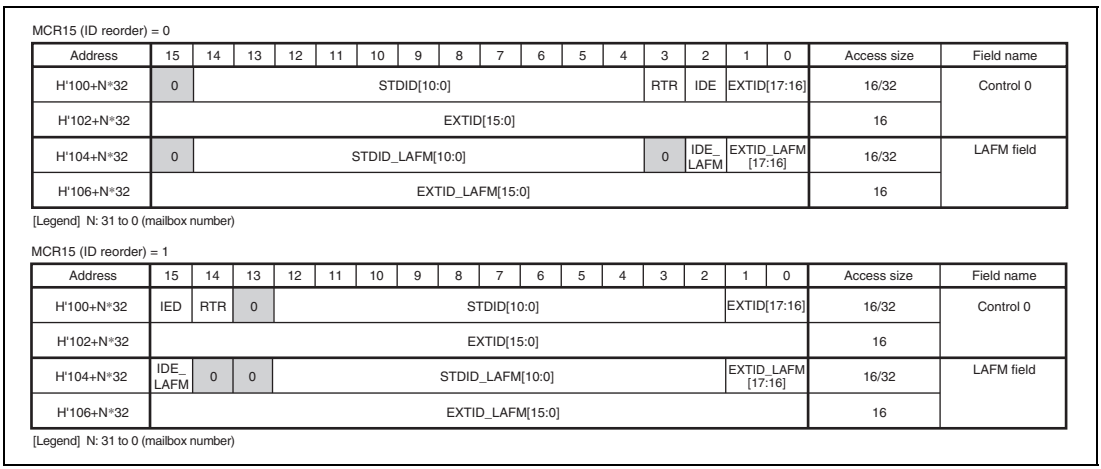


Figure 22.8 ID Order Change

22.5.2 General Status Register (GSR)

The General Status Register (GSR) is a 16-bit read-only register that indicates the RCAN status.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	GSR5	GSR4	GSR3	GSR2	GSR1	GSR0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	GSR5	0	R	Error Passive Status Indicates whether the CAN interface is in error passive or not. This bit will be set to 1 as soon as the RCAN enters the error passive state and is cleared to 0 when the RCAN re-enters the error active state (this means the GSR5 remains set to 1 in error passive state and bus off state). Consequently, to acquire the correct state, both GSR5 and GSR0 should be checked. 0: Indicates that the RCAN is not in error passive or in bus off state. [Clearing condition] The RCAN is in error active state. 1: Indicates that the RCAN is in error passive (GSR0 = 0) or bus of (GSR0 = 1) state. [Setting condition] When TEC ≥128 or REC ≥128 or error passive state is selected in test mode.

Bit	Bit Name	Initial Value	R/W	Description
4	GSR4	0	R	Halt/Sleep Status Indicates whether the CAN controller is in halt or sleep state or not. Note that the clearing time of this flag is not the same as the setting time of IRR12. In addition, note that this flag reflects the status of the CAN engine but it does not of reflect entire RCAN status. The RCAN can be accessed after it has returned from sleep mode and MCR5 is cleared. The CAN sleep mode of the CAN controller will be cancelled after two additional transmission clocks on the CAN bus have passed. 0: Indicates that the RCAN is not in the halt state or sleep state 1: Indicates that the RCAN –TL1 is in halt mode (MCR1 = 1) or CAN sleep mode (MCR5 = 1) [Setting condition] If the CAN bus is either in intermission or idle state while MCR1 is set, if the RCAN is in halt mode while MCR5 is set, or if the RCAN enters bus off state when both MCR14 and MCR6 are set.
3	GSR3	1	R	Reset Status Indicates whether the RCAN is in reset state or not. 0: Indicates that the RCAN is not in reset state 1: Indicates that the RCAN is in reset state [Setting condition] After RCAN software or hardware reset
2	GSR2	1	R	Message Transmission in progress Flag Indicates to the CPU whether the RCAN is in bus off or transmitting a message or an error/overload flag due to an error detected during message transmission. The timing to set TXACK is different from the timing to clear GSR2. TXACK is set at the 7th bit of End Of Frame (EOF). GSR2 is set at the 3rd bit of intermission if there are no more messages ready to be transmitted. It is also set by an arbitration lost, bus idle, reception, reset or, a transition to halt state. 0: Indicates that the RCAN is in bus off or a transmission is in progress 1: [Setting condition] When the RCAN is not in bus off or no message is being transmitted.

Bit	Bit Name	Initial Value	R/W	Description
1	GSR1	0	R	Transmit/Receive Warning Flag Indicates an error warning. 0: [Clearing condition] When (TEC < 96 and REC < 96) or bus off 1: [Setting condition] When 96 ≤ TEC < 256 or 96 ≤ REC < 256 Note: REC is incremented during bus off to count the recurrences of 11 recessive bits because the count is necessary for the bus off recovery sequence. Note, however, that the GSR1 is not set in bus off state.
0	GSR0	0	R	Bus Off Flag Indicates that RCAN is in the bus off state. 0: [Clearing condition] After recovery from bus off state or after hardware or software reset 1: [Setting condition] When TEC ≥ 256 (bus off state)

22.5.3 Bit Configuration Registers 0 and 1 (BCR0 and BCR1)

The bit configuration registers 0 and 1 (BCR0 and BCR1) are 16-bit readable/writable registers that are used to set CAN bit timing parameters and the baud rate prescaler for the CAN interface.

The time quanta is defined as follows:

$$\text{Time quanta} = 2 \times (\text{BPR} + 1) / f_{clk}$$

Where f_{clk} indicates the peripheral bus frequency.

(1) BCR1

For details on the TSEG1 and TSEG2 settings, see table 22.9.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG1[3:0]				—	TSG2[2:0]			—	—	SJW[1:0]		—	—	—	BSP
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	TSG1[3:0]	0000	R/W	Time Segment 1 These bits are used to set the segment TSEG1 (=PRSEG + PHSEG1) to compensate for edges on the CAN bus with a positive phase error. A value from 4 to 16 time quanta can be set. 0000: Setting prohibited 0001: Setting prohibited 0010: Setting prohibited 0011: PRSEG + PHSEG1 = 4 time quanta 0100: PRSEG + PHSEG1 = 5 time quanta : 1111: PRSEG + PHSEG1 = 16 time quanta
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
10 to 8	TSG2[2:0]	000	R/W	Time Segment 2 These bits are used to set the segment TSEG2 (=PHSEG2) to compensate for edges on the CAN bus with a negative phase error. A value from 2 to 8 time quanta can be set as shown below. 000: Setting prohibited 001: PHSEG2 = 2 time quanta (conditionally prohibited. For details, refer to table 22.9.) 010: PHSEG2 = 3 time quanta 011: PHSEG2 = 4 time quanta 100: PHSEG2 = 5 time quanta 101: PHSEG2 = 6 time quanta 110: PHSEG2 = 7 time quanta 111: PHSEG2 = 8 time quanta
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5, 4	SJW[1:0]	00	R/W	Re-Synchronization Jump Width These bits set the synchronization jump width. 00: Synchronization jump width = 1 time quantum 01: Synchronization jump width = 2 time quanta 10: Synchronization jump width = 3 time quanta 11: Synchronization jump width = 4 time quanta
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	BSP	0	R/W	Bit Sample Point Sets the point at which data is sampled. 0: Performs bit sampling at one point (end of time segment 1) 1: Performs bit sampling at three points (rising edge of the last three clock cycles of PHSEG1)

(2) BCR0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	BRP[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	BRP[7:0]	0000 0000	R/W	Baud Rate Prescaler These bits are used to specify the peripheral bus clock periods contained in a time quantum. 00000000: 2 × peripheral bus clock 00000001: 4 × peripheral bus clock 00000010: 6 × peripheral bus clock : : 2 × (register value + 1) × peripheral bus clock 11111111: 512 × peripheral bus clock

(3) Bit Configuration Register

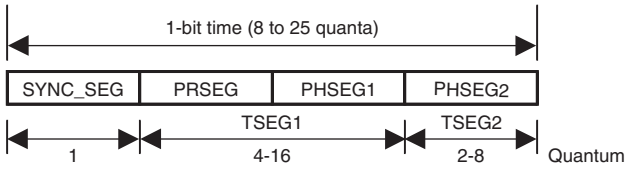


Figure 22.9 One-Bit Time Configuration

SYNC_SEG: Segment for establishing synchronization of nodes on the CAN bus. (Normal bit edge transitions occur in this segment.)

PRSEG: Segment for compensating the physical delay between networks.

PHSEG1: Buffer segment for correcting the phase drift (positive). (This segment is extended when synchronization (resynchronization) is established.)

PHSEG2: Buffer segment for correcting the phase drift (negative). (This segment is shortened when synchronization (resynchronization) is established)

TSEG1: TSG1 + 1

TSEG2: TSG2 + 1

The RCAN bit rate is calculated by the following formula:

Bit rate = fclk / (2*(BRP + 1) * (TSEG1 + TSEG2 + 1))

where BRP is given by the register value, and TSEG1 and TSEG2 are calculated by using the table described above. Note that they are not the actual setting value. The '+ 1' in the above formula is for the SYNC_SEG whose duration is 1 time quanta.

fclk = Peripheral clock

Restrictions on BCR Setting

TSEG1min > TSEG2 ³ SJWmax (SJW = 1 to 4)

8 ≤ TSEG1 + TSEG2 + 1 ≤ 25 time quanta (TSEG1 + TSEG2 + 1 =7 is not allowed)

TSEG2 ≥ 2

If the setting values of TSEG1 and TSEG2 in the bit configuration register are within the range shown in table 22.9 below, the above restrictions are satisfied. Table 22.9 shows the available settings of SJW. "No" indicates that there is no available combination of TSEG1 and TSEG2.

Table 22.9 Relationship between TSG and TSEG

		001	010	011	100	101	110	111	TSG2
		2	3	4	5	6	7	8	TSEG2
TSG1	TSEG1								
0011	4	No	1-3	No	No	No	No	No	
0100	5	1-2	1-3	1-4	No	No	No	No	
0101	6	1-2	1-3	1-4	1-4	No	No	No	
0110	7	1-2	1-3	1-4	1-4	1-4	No	No	
0111	8	1-2	1-3	1-4	1-4	1-4	1-4	No	
1000	9	1-2	1-3	1-4	1-4	1-4	1-4	1-4	
1001	10	1-2	1-3	1-4	1-4	1-4	1-4	1-4	
1010	11	1-2	1-3	1-4	1-4	1-4	1-4	1-4	
1011	12	1-2	1-3	1-4	1-4	1-4	1-4	1-4	
1100	13	1-2	1-3	1-4	1-4	1-4	1-4	1-4	
1101	14	1-2	1-3	1-4	1-4	1-4	1-4	1-4	
1110	15	1-2	1-3	1-4	1-4	1-4	1-4	1-4	
1111	16	1-2	1-3	1-4	1-4	1-4	1-4	1-4	

Example: To obtain a bit rate of 500 kbps with a frequency of fclk = 50 MHz, it is possible to set: BRP = 1, TSEG1 = 16, TSEG2 = 8.
In this case, BCR1 and BCR0 are specified as H'F700 and H'0001, respectively.

22.5.4 Interrupt Request Register (IRR)

IRR is a 16-bit readable/writable-clearable register that incorporates the status flags for the various interrupt sources.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IRR15	IRR14	IRR13	IRR12	IRR11	—	IRR9	IRR8	IRR7	IRR6	IRR5	IRR4	IRR3	IRR2	IRR1	IRR0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	IRR15	0	R/W	Timer Compare Match Interrupt 1 Indicates that a compare match has occurred in the timer compare match register 1 (TCMR1). When the value set in the TCMR1 matches the cycle time (TCMR1 = CYCTR), this bit is set to 1. 0: Indicates that a timer compare match has not occurred in TCMR1 [Clearing condition] Writing 1 to this bit. 1: Indicates that a timer compare match has occurred in TCMR1 [Setting condition] TCMR1 matches the cycle time (TCMR1 = CYCTR)
14	IRR14	0	R/W	Timer Compare Match Interrupt 0 Indicates that a compare match has occurred in the timer compare match register 0 (TCMR0). When the value set in the TCMR0 matches the local time (TCMR0 = TCNTR), this bit is set to 1. 0: Indicates that a timer compare match has not occurred in TCMR0. [Clearing condition] Writing 1 to this bit. 1: Indicates that a timer compare match has occurred in TCMR0. [Setting condition] TCMR0 matches the timer value (TCMR0 = TCNTR).

Bit	Bit Name	Initial Value	R/W	Description
13	IRR13	0	R/W	Timer Overrun Interrupt/Message Error Interrupt Indicates the following interrupt occurrence according to the RCAN mode. • Indicates that a timer (TCNTR) overrun has occurred when the RCAN operates in event-trigger mode (including test modes). • Indicates that a message error has occurred when the RCAN operates in Note that this bit will not be set even if a message overload has occurred in test mode. 0: Indicates that no timer (TCNTR) overrun has occurred in event-trigger mode (including test modes), or that no message error has occurred in test mode. [Clearing condition] Writing 1 to this bit. 1: [Setting condition] Indicates that a timer (TCNTR) overrun has occurred and TCNTR has changed from H'FFFF to H'0000 in event-trigger mode (including test modes), or that a message error has occurred in test mode.

Bit	Bit Name	Initial Value	R/W	Description
12	IRR12	0	R/W	Bus Activity in CAN Sleep Mode Indicates that a CAN bus activity is present. While the RCAN is in sleep mode and a dominant bit is detected on the CAN bus, this bit is set. This interrupt is cleared by writing 1 to this bit. Writing 0 to this bit is ignored. If the auto-wakeup function is not used and if this interrupt is not requested, this interrupt should be disabled by the related interrupt mask bit. If the auto-wakeup function is not used and if this interrupt is requested, this bit should be cleared only after recovering from sleep mode. This is to avoid that a new falling edge of the reception line sets this interrupt again. Note that the setting time of this interrupt is different from the clearing time of GSR4. 0: Indicates that the RCAN is in bus idle state [Clearing condition] Writing 1 to this bit. 1: Indicates that the CAN bus activity detected in CAN sleep mode [Setting condition] Dominant bit level detection on the CAN_RX line in CAN sleep mode
11	IRR11	0	R/W	Timer Compare Match Interrupt 2 Indicates that a compare match has occurred in the timer compare match register 2 (TCMR2). When the value set in the TCMR2 matches the cycle time (TCMR2 = CYCTR), this bit is set to 1. 0: Indicates that no timer compare match has occurred in TCMR2. [Clearing condition] Writing 1 to this bit. 1: Indicates that a timer compare match has occurred in TCMR2. [Setting condition] TCMR2 matches the cycle time (TCMR2 = CYCTR)
10	—	0	R	Reserved. This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
9	IRR9	0	R	Message Overrun/Overwrite Interrupt Flag Indicating that a message has been received but the existing message in the matching mailbox has not been read because the corresponding RXPR or RFPR has already been set to 1 and not yet cleared by the CPU. The received message is either abandoned (overrun) or overwritten dependant on the new message control (NMC) bit. This bit is cleared when a corresponding bit in all bit in UMSR (Unread Message Status Register) are cleared (by writing '1') or by setting the mailbox interrupt mast register (MBIMR) in UMSR. It is also cleared by writing 1 to the corresponding bit in MBIMR. Writing to this bit is invalid. 0: Indicates that no pending notification of message overrun/overwrite [Clearing condition] Clearing all bits in UMSR/setting MBIMR for all UMSR set 1: Indicates that a receive message has been discarded due to an overrun or a message has been overwritten [Setting condition] Message is received while the corresponding RXPR and/or RFPR = 1 and MBIMR = 0

Bit	Bit Name	Initial Value	R/W	Description
8	IRR8	0	R	Mailbox Empty Interrupt Flag This bit is set when one of the messages set for transmission has been successfully sent (corresponding TXACK flag is set) or has been successfully aborted (corresponding ABACK flag is set). In event triggered mode the related TXPR is also cleared and this mailbox is now ready to accept a new message data for the next transmission. In effect, this bit is set by a signal obtained by ORing the TXACK and ABACK bits not masked by the corresponding MBIMR flag. Therefore, this bit is automatically cleared when all the TXACK and ABACK bits are cleared. It is also cleared by writing 1 to the correspondent bit in MBIMR. Writing to this bit is invalid. 0: Indicates that no message for transmission or transmission cancellation is processed. [Clearing Condition] All the TXACK and ABACK bits are cleared or MBIMR corresponding to all TXACK and ABACK set is set. 1: Indicates that a message has been transmitted or aborted, and that a new message can be stored. [Setting condition] When a TXACK or ABACK bit is set (if related MBIMR = 0).
7	IRR7	0	R/W	Overload Frame Indicates that the RCAN has detected a condition that should initiate the transmission of an overload frame. This bit is cleared when 1 is written to IRR7. Writing 0 to this bit is ignored. 0: [Clearing condition] Writing 1 to this bit. 1: [Setting conditions] Overload condition is detected.

Bit	Bit Name	Initial Value	R/W	Description
6	IRR6	0	R/W	Bus Off Interrupt Flag This bit is set when the RCAN enters the bus off state or when the RCAN makes a transition from bus off state to error-active state. The cause therefore is the existing condition TEC ≥ 256 at the node, the end of the Bus-off recovery sequence (128X11 consecutive recessive bits), or a transition from bus off to halt (automatic or manual). This bit remains set even if the RCAN cancels the bus off state, and needs to be cleared by software The software is expected to read the GSR0 to check whether the RCAN is in the bus off or error active state. It is cleared by writing 1 to this bit even if the node is still in bus off state. Writing 0 to this bit is ignored. 0: [Clearing condition] Writing 1 to this bit. 1: Indicates that the RCAN enters bus off state caused by a transmit error or it makes a transition from bus off state to error active state [Setting condition] When TEC becomes ≥ 256 or End of Bus-off after 128 × 11 consecutive recessive bits or transition from bus off to halt.
5	IRR5	0	R/W	Error Passive Interrupt Flag Indicates that error passive state has been entered by transmit or receive error counter or by test mode. This bit is reset by writing 1 to this bit. Writing 0 to this bit is ignored. If this bit is cleared, the node may still be in error passive. Note that the software needs to check GSR0 and GSR5 to determine whether the RCAN is in error passive state or bus off state. 0: [Clearing condition] Writing 1 to this bit. 1: Indicates that the RCAN is in error passive state caused by a transmit/receive error [Setting condition] When TEC ≥ 128 or REC ≥ 128 or error passive test mode is used.

Bit	Bit Name	Initial Value	R/W	Description
4	IRR4	0	R/W	Receive Error Counter Warning Interrupt Flag This bit is set if the receive error counter (REC) reaches a value greater than 95 when the RCAN is not in bus off state. The interrupt is cleared by writing 1 to this bit. Writing 0 to this bit is ignored. 0: [Clearing condition] Writing 1 to this bit. 1: indicates that the RCAN is in error warning state caused by a receive error. [Setting condition] When REC ≥ 96 and RCAN is not in bus off state.
3	IRR3	0	R/W	Transmit Error Counter Warning Interrupt Flag This bit is set if the transmit error counter (TEC) reaches a value greater than 95. The interrupt is cleared by writing 1 to this bit. Writing 0 to this bit is ignored. 0: [Clearing condition] Writing 1 to this bit. 1: indicates that the RCAN is in error warning state caused by a transmit error. [Setting condition] When TEC ≥ 96
2	IRR2	0	R	Remote Frame Receive Interrupt Flag Indicates that a mailbox has received in a remote frame. This bit is set if at least one receive mailbox, with related MBIMR not set, stores a remote frame transmission request. This bit is automatically cleared when all bits in the remote frame receive pending register (RFPR) are cleared. It is also cleared by writing 1 to all the correspondent bits in MBIMR. Writing to this bit is ignored. 0: [Clearing condition] Clearing of all bits in RFPR 1: Indicates that one or more remote requests are pending. [Setting conditions] When remote frame is received and the corresponding MBIMR = 0

Bit	Bit Name	Initial Value	R/W	Description
1	IRR1	0	R	Data Frame Received Interrupt Flag Indicates that there are pending receive data frames. If this bit is set, at least one receive mailbox contains a pending message. This bit is cleared when all bits in the data frame receive pending register (RXPR) are cleared, i.e. there is no pending message in any receive mailbox. The RXPR flags of receive mailboxes for which the relevant bits in MBIMR are not 0 are logically ORed and the result is set in this bit. This bit is cleared by writing 1 to all the corresponding bits in MBIMR. Writing to this bit is ignored. 0: [Clearing condition] Clearing all bits in RXPR 1: Indicates that data frame has been received and stored in a mailbox. [Setting conditions] When data is received and the corresponding MBIMR = 0.

Bit	Bit Name	Initial Value	R/W	Description
0	IRR0	1	R/W	Reset/Halt/Sleep Interrupt Flag This flag is set by the following three different reasons. It can indicate that: • Reset mode has been entered after a software (MCR0) or hardware reset • Halt mode has been entered after a halt request (MCR1) • Sleep mode has been entered after a sleep request (MCR5) has been made while in halt mode. The GSR may be read after this bit is set to check the RCAN state. Note: When a sleep mode request needs to be made, the halt mode must be used beforehand. Refer to the MCR5 description and figure 22.12, State Transitions. IRR0 is set when GSR3 or GSR4 changes from 0 to 1 or when the RCAN makes a transition from halt mode to sleep mode. So, IRR0 is not set if the RCAN enters halt mode again immediately after exiting from halt mode, without GSR4 being cleared. Similarly, IRR0 is not set by direct transition from sleep mode to halt request. At the transition from halt/sleep mode to transition/reception, clearing GSR4 requires the time period of (one-bit time - TSEG2) to (one-bit time * 2 - TSEG2). In reset mode, IRR0 is set, however, the interrupt to the CPU is not asserted since IMR0 is automatically set by initialization. 0: [Clearing condition] Writing 1 to this bit. 1: Indicates that the RCAN makes a transition to software reset mode, halt mode, or sleep mode. [Setting condition] When reset/halt/sleep transition is completed after a reset (MCR0 or HW) or halt mode (MCR1) or sleep mode (MCR5) is requested

22.5.5 Interrupt Mask Register (IMR)

IMR is a 16-bit register that protects all corresponding interrupts in IRR from generating an output signal on the IRQ. An interrupt request is masked if the corresponding bit is set to 1. This register can be read or written at any time. The IMR directly controls the generation of IRQ, but does not prevent the setting of the corresponding bit in the IRR.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IMR15	IMR14	IMR13	IMR12	IMR11	—	IMR9	IMR8	IMR7	IMR6	IMR5	IMR4	IMR3	IMR2	IMR1	IMR0
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	IMR[15:0]	All 1	R/W	Maskable Interrupt Sources Corresponding to IRR [15:0] When a bit is set, the interrupt signal is not generated, although setting the corresponding IRR bit is still performed. 0: Corresponding IRR is not masked (IRQ is generated for interrupt conditions) 1: Corresponding interrupt of IRR is masked
10	—	1	R	Reserved. This bit is always read as 1. The write value should always be 1.
9 to 0	IMR[9:0]	All 1	R/W	Maskable Interrupt Sources Corresponding to IRR [9:0] When a bit is set, the interrupt signal is not generated, although setting the corresponding IRR bit is still performed. 0: Corresponding IRR is not masked (IRQ is generated for interrupt conditions) 1: Corresponding interrupt of IRR is masked

22.5.6 Transmit Error Counter (TEC)/Receive Error Counter (REC)

TEC/REC is a 16-bit readable/(writable) register that functions as a counter indicating the number of transmit/receive message errors on the CAN Interface. The count value is stipulated in the CAN protocol specification Refs. [2], [3], [4] and [5]. When not in (write error counter) test mode this register is read only, and can only be modified by the CAN Interface. This register can be cleared by a reset request (MCR0) or entering to bus off.

In write error counter test mode (i.e. TST[2:0] = B'100), it is possible to write to this register. The same value can only be written to TEC/REC, and the value written into TEC is set to TEC and REC. When writing to this register, RCAN needs to be put into halt mode. This feature is only intended for test purposes.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TEC7	TEC6	TEC5	TEC4	TEC3	TEC2	TEC1	TEC0	REC7	REC6	REC5	REC4	REC3	REC2	REC1	REC0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	TEC[7:0]	All 0	R/W*	Transmit Error Counter This register value is incremented when an error related to the CAN specifications occurred during transmission.
7 to 0	REC[7:0]	All 0	R/W*	Receive Error Counter This register value is incremented when an error related to the CAN specifications occurred during reception.

Note: * It is only possible to write the value in test mode when TST[2:0] in MCR is B'100. REC is incremented during Bus Off to count the recurrences of 11 recessive bits as requested by the Bus Off recovery sequence.

22.6 RCAN Mailbox Registers

The following describes RCAN mailbox registers that control the mailboxes and indicate mailbox states. Table 22.10 shows the RCAN mailbox registers.

Table 22.10 RCAN Mailbox Registers

Register Name	Abbreviation	Address	Access Size (Bits)
Transmit pending register 1	TXPR1	H'020	32
Transmit pending register 0	TXPR0	H'022	—
		H'024	
		H'026	
Transmit cancel register 1	TXCR1	H'028	16/32
Transmit cancel register 0	TXCR0	H'02A	16
		H'02C	
		H'02E	
Transmit acknowledge register 1	TXACK1	H'030	16/32
Transmit acknowledge register 0	TXACK0	H'032	16
		H'034	
		H'036	
Abort acknowledge register 1	ABACK1	H'038	16/32
Abort acknowledge register 0	ABACK0	H'03A	16
		H'03C	
		H'03E	
Data frame receive pending register 1	RXPR1	H'040	16/32
Data frame receive pending register 0	RXPR0	H'042	16
		H'044	
		H'046	
Remote frame receive pending register 1	RFPR1	H'048	16/32
Remote frame receive pending register 0	RFPR0	H'04A	16
		H'04C	
		H'04E	

Register Name	Abbreviation	Address	Access Size (Bits)
Mailbox interrupt mask register 1	MBIMR1	H'050	16/32
Mailbox interrupt mask register 0	MBIMR0	H'052	16
		H'054	
		H'056	
Unread message status register 1	UMSR1	H'058	16/32
Unread message status register 0	UMSR0	H'05A	16
		H'05C	
		H'05E	

22.6.1 Transmit Pending Registers 1 and 0 (TXPR1 and TXPR0)

The concatenation of TXPR1 and TXPR0 is a 32-bit register that contains any transmit pending flags for the CAN module.

The TXPR1 controls mailboxes 31 to 16, and the TXPR0 controls mailboxes 15 to 1. The CPU may set the TXPR bits to affect any message being considered for transmission by writing 1 to the corresponding bit location. Writing 0 has no effect, and TXPR cannot be cleared by writing 0 and must be cleared by setting the corresponding TXCR bits. TXPR may be read by the CPU to determine which, if any, transmissions are pending or in progress. In effect there is a transmit pending bit for all mailboxes except for the mailbox 0. Writing 1 to a bit location when the mailbox is not configured to transmit is not allowed.

In event triggered mode, the RCAN will clear a transmit pending flag after successful transmission of its corresponding message or when a transmission abort is requested successfully from the TXCR. The TXPR flag is not cleared if the message is not transmitted due to the CAN node losing the arbitration process or due to errors on the CAN bus, and RCAN automatically tries to transmit it again unless its DART bit (disable automatic re-transmission) is set in the message-control of the corresponding mailbox. In such case (DART set), the transmission is cleared and notified through the mailbox empty interrupt flag (IRR8) and the correspondent bit within the abort acknowledgement register (ABACK).

If the status of the TXPR changes, the RCAN shall ensure that in the identifier priority scheme (MCR2 = 0), the highest priority message is always presented for transmission in an intelligent way even under circumstances such as bus arbitration losses or errors on the CAN bus. Refer to section 22.8, Operation, for details.

When the RCAN changes the state of any TXPR bit to 0, an empty slot interrupt (IRR8) may be generated. This indicates that either a successful or an aborted mailbox transmission has just been

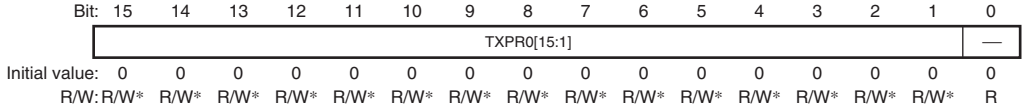
made. If a message transmission is successful it is signaled in the TXACK register, and if a message transmission abortion is successful it is signaled in the ABACK register. By checking these registers, the contents of the message of the corresponding mailbox may be modified to prepare for the next transmission.

(1) TXPR1

Note: Only 1 can be written to a bit corresponding to the mailbox that has been set for transmission.

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	TXPR1 [15:0]	H'0000	R/W	Requests the corresponding mailbox to transmit a CAN frame. Bits 15 to 0 correspond to mailboxes 31 to 16, respectively. When multiple bits are set, the transmissions are performed in message ID order or mailbox number order depending on the MCR2 bit setting. 0: Indicates that the corresponding mailbox is in transmit message idle state. [Clearing condition] Completion of message transmission (for Event Triggered Messages) or message transmission abortion (automatically cleared) 1: Indicates that transmission is requested to the corresponding mailbox.

(2) TXPR0



Note: Only 1 can be written to a bit corresponding to the mailbox that has been set for transmission. TXPR1 and TXPR0 should be accessed in longword.

Bit	Bit Name	Initial Value	R/W	Description
15 to 1	TXPR0 [15:1]	H'0000	R/W	These bits indicate that the corresponding mailbox is requested to transmit a CAN frame. Bits 15 to 1 correspond to mailboxes 15 to 1, respectively. When multiple bits are set, is the transmission is performed in message ID order or mailbox number order depending on the MCR2 bit setting. 0: Indicates that the corresponding mailbox is in transmit message idle state. [Clearing condition] Completion of message transmission (for event triggered messages) or message transmission abortion (automatically cleared) 1: Indicates that transmission is requested to the corresponding mailbox.
0	—	0	R	Reserved This bit is a receive-only mailbox and is always read as 0. The write value should always be 0.

22.6.2 Transmit Cancel Registers 1 and 0 (TXCR1 and TXCR0)

The TXCR1 and TXCR0 are 16-bit readable/conditionally-writable registers. The TXCR1 controls mailboxes 31 to 16, and the TXCR0 controls mailboxes 15 to 1. This register is used by the CPU to request the pending transmission requests in the TXPR to be cancelled. To clear the corresponding bit in the TXPR the CPU must write 1 to the bit position in the TXCR. Writing 0 has no effect.

When an abort has succeeded the CAN controller clears the corresponding TXPR + TXCR bits, and sets the corresponding ABACK bit. However, once a mailbox has started a transmission, it cannot be cancelled by this bit. In such a case, if the transmission is completed successfully, the CAN controller clears the corresponding TXPR + TXCR bit, and sets the corresponding TXACK bit, however, if the transmission fails due to a bus arbitration loss or an error on the bus, the CAN controller clears the corresponding TXPR + TXCR bit, and sets the corresponding ABACK bit. If an attempt is made by the CPU to clear a mailbox transmission that is not transmit-pending it has no effect. In this case the CPU will be not able at all to set the TXCR flag.

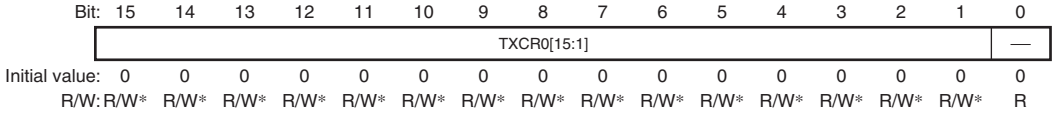
(1) TXCR1

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TXCR1[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

Note: Only 1 can be written to a bit corresponding to the transmit mailbox that has received a transmission request.

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	TXCR1 [15:0]	H'0000	R/W	Requests the corresponding mailbox that is placed in the transmission queue to cancel its transmission. Bits 15 to 0 correspond to mailboxes 31 to 16 (and TXPR1[15:0]), respectively. 0: Indicates that the corresponding mailbox is in transmit message cancellation idle state. [Clearing condition] Completion of transmit message cancellation (automatically cleared) 1: Indicates that the transmission cancellation is requested to the corresponding mailbox.

(2) TXCR0



Note: Only 1 can be written to a bit corresponding to the transmit mailbox that has received a transmission request.

Bit	Bit Name	Initial Value	R/W	Description
15 to 1	TXCR0 [15:1]	H'0000	R/W	Requests the corresponding mailbox that is in the transmission queue to cancel its transmission. Bits 15 to 1 correspond to mailboxes 15 to 1 (and TXPR0[15:1]), respectively. 0: Indicates that the corresponding mailbox is in transmit message cancellation idle state. [Clearing condition] Completion of transmit message cancellation (automatically cleared) 1: Indicates that the transmission cancellation is requested to the corresponding mailbox.
0	—	0	R	Reserved This bit is a receive-only mailbox and is always read as 0. The write value should always be 0.

22.6.3 Transmit Acknowledge Registers 1 and 0 (TXACK1 and TXACK0)

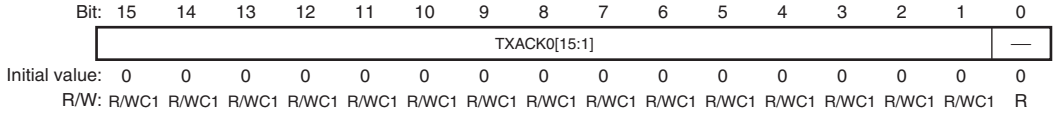
The TXACK1 and TXACK0 are 16-bit readable/conditionally-writable registers. These registers are used to signal to the CPU that a mailbox transmission has been completed successfully. When a transmission has succeeded, the RCAN sets the corresponding bit in the TXACK register. The CPU may clear a TXACK bit by writing 1 to the corresponding bit. Writing 0 to the corresponding bit is ignored.

(1) TXACK1

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TXACK1[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/WC1	R/WC1	R/WC1	R/WC1	R/WC1	R/WC1	R/WC1	R/WC1	R/WC1	R/WC1	R/WC1	R/WC1	R/WC1	R/WC1	R/WC1	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	TXACK1 [15:0]	H'0000	R/WC1	Notifies that the requested transmission of the corresponding mailbox has been completed successfully. Bits 15 to 0 correspond to Mailboxes 31 to 16, respectively. 0: [Clearing condition] Writing '1 to the corresponding bit. 1: Notifies that the corresponding mailbox has successfully transmitted a message (data or remote frame) [Setting condition] Completion of message transmission for corresponding mailbox

(2) TXACK0

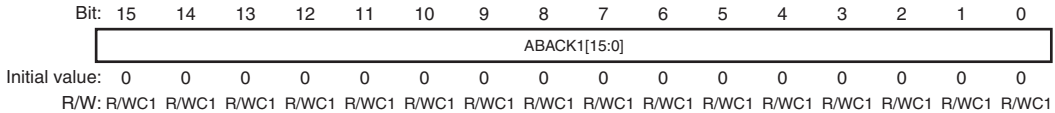


Bit	Bit Name	Initial Value	R/W	Description
15 to 1	TXACK0 [15:1]	H'0000	R/WC1	Notifies that the requested transmission of the corresponding mailbox has been completed successfully. Bits 15 to 1 correspond to mailboxes 15 to 1, respectively. 0: [Clearing condition] Writing 1 to the corresponding bit. 1: Notifies that the corresponding mailbox has successfully transmitted a message (data or remote frame) [Setting condition] Completion of message transmission for corresponding mailbox
0	—	0	R	Reserved This bit is a receive-only mailbox and is always read as 0. The write value should always be 0.

22.6.4 Abort Acknowledge Registers 1 and 0 (ABACK1 and ABACK0)

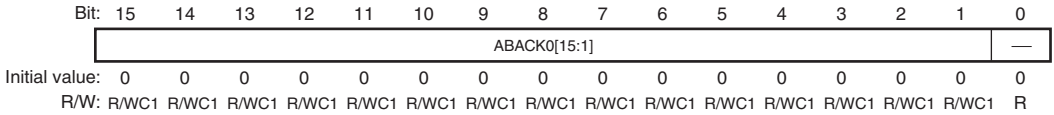
The ABACK1 and ABACK0 are 16-bit readable/conditionally-writable registers. These registers are used to signal to the CPU that a mailbox transmission has been aborted according to the request. When a mailbox transmission is aborted, the RCAN sets the corresponding bit in the ABACK register. The CPU may clear the abort acknowledge bit by writing 1 to the corresponding bit. Writing 0 to the corresponding bit is ignored. An ABACK bit is set by the RCAN to acknowledge that the TXPR bit has been cleared by the corresponding TXCR bit.

(1) ABACK1



Bit	Bit Name	Initial Value	R/W	Description
15 to 0	ABACK1 [15:0]	H'0000	R/WC1	Notifies that the requested transmission of the corresponding mailbox has been cancelled successfully. Bits 15 to 0 correspond to mailboxes 31 to 16, respectively. 0: [Clearing condition] Writing 1 to the corresponding bit. 1: Notifies that the message (data or remote frame) transmission of the corresponding mailbox has been cancelled. [Setting condition] Completion of transmission cancellation for the corresponding mailbox

(2) ABACK0

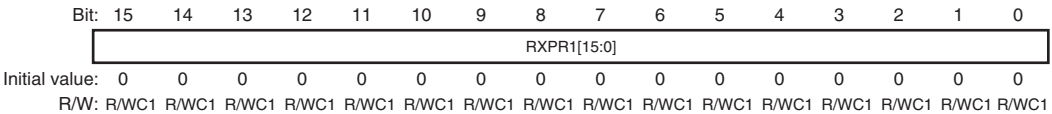


Bit	Bit Name	Initial Value	R/W	Description
15 to 1	ABACK0 [15:1]	H'0000	R/WC1	Notifies that the requested transmission of the corresponding mailbox has been cancelled successfully. Bits 15 to 1 correspond to mailboxes 15 to 1, respectively. 0: [Clearing condition] Writing 1 to the corresponding bit. 1: Notifies that the message (data or remote frame) transmission of the corresponding mailbox has been cancelled. [Setting condition] Completion of transmission cancellation for the corresponding mailbox
0	—	0	R	Reserved This bit is a receive-only mailbox and is always read as 0. The write value should always be 0.

22.6.5 Data Frame Receive Pending Registers 1 and 0 (RXPR1, RXPR0)

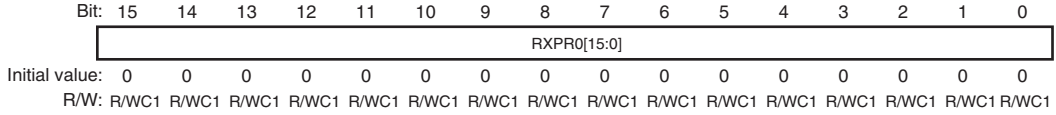
The RXPR1 and RXPR0 are 16-bit readable/conditionally-writable registers. The RXPR is a register that contains the received data frames pending flags for the receive mailboxes. When a CAN data frame is stored successfully in a receive mailbox, the corresponding bit in the RXPR is set. The bit may be cleared by writing 1 to it. Writing 0 to the corresponding bit is ignored. However, the bit may only be set if the mailbox is configured by its MBC (mailbox configuration) to receive data frames. When a bit in RXPR is set, the data frame received interrupt flag (IRR1) is set if the corresponding mailbox interrupt mask register (MBIMR) is not set, and the interrupt signal is generated if IMR1 is not set. Note that these bits are set only by receiving data frames and not by receiving remote frames.

(1) RXPR1



Bit	Bit Name	Initial Value	R/W	Description
15 to 0	RXPR1 [15:0]	H'0000	R/WC1	Bits 15 to 0 correspond to mailboxes 31 to 16 which are configured as receive mailboxes. 0: [Clearing condition] Writing 1 to the corresponding bit. 1: Indicates that the corresponding mailbox has received a CAN data frame [Setting condition] Data frame has been received in the corresponding mailbox.

(2) RXPR0



Bit	Bit Name	Initial Value	R/W	Description
15 to 0	RXPR0 [15:0]	H'0000	R/WC1	Bits 15 to 0 correspond to mailboxes 15 to 0 which are configured as receive mailboxes. 0: [Clearing condition] Writing 1 to the corresponding bit. 1: Indicates that the corresponding mailbox has received a CAN data frame [Setting condition] Data frame has been received in the corresponding mailbox.

22.6.6 Remote Frame Receive Pending Registers 1 and 0 (RFPR1 and RFPR0)

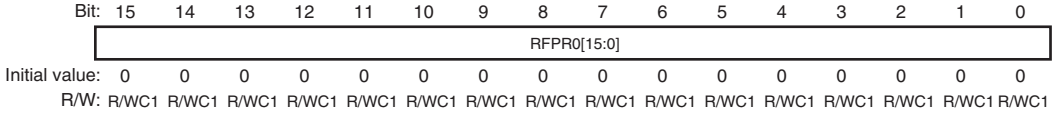
The RFPR1 and RFPR0 are 16-bit readable/conditionally-writable registers. The RFPR is a register that contains the received remote frame pending flags for the receive mailboxes. When a CAN remote frame is successfully stored in a receive mailbox, the corresponding bit is set in the RFPR. The bit may be cleared by writing 1 to the corresponding bit. Writing 0 to the corresponding bit is ignored. Bits in this register correspond to all mailboxes, however, the bit may only be set if the corresponding mailbox is configured to receive remote frames by the MBC (mailbox configuration). When a RFPR bit is set, it also sets the remote frame receive interrupt flag (IRR2) if its mailbox interrupt mask register (MBIMR) mailbox interrupt mask register is not set, and the interrupt signal is generated if IMR2 is not set. Note that these bits are set only by receiving remote frames and not by receiving data frames.

(1) RFPR1

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFPR1[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/WC1	R/WC1	R/WC1	R/WC1	R/WC1	R/WC1	R/WC1	R/WC1	R/WC1	R/WC1	R/WC1	R/WC1	R/WC1	R/WC1	R/WC1	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	RFPR1 [15:0]	H'0000	R/WC1	Remote request pending flags for mailboxes 31 to 16, respectively. 0: [Clearing condition] Writing to the corresponding bit. 1: Indicates that the corresponding mailbox received a remote frame [Setting condition] The remote frame has been received in the corresponding mailbox.

(2) RFPR0



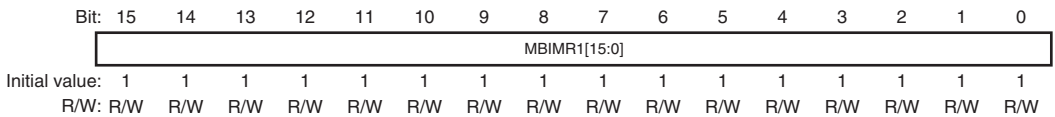
Bit	Bit Name	Initial Value	R/W	Description
15 to 0	RFPR0 [15:0]	H'0000	R/WC1	Remote request pending flags for mailboxes 15 to 0, respectively. 0: [Clearing condition] Writing 1 to the corresponding bit. 1: Indicates that the corresponding mailbox received a remote frame. [Setting condition] The remote frame has been received in the corresponding mailbox.

22.6.7 Mailbox Interrupt Mask Registers 0 and 1 (MBIMR1 and MBIMR0)

The MBIMR1 and MBIMR0 are 16-bit readable/writable registers. The MBIMR only prevents the setting of IRR related to the mailbox activities, that are IRR[1] – data frame received interrupt, IRR[2] – remote frame receive interrupt, IRR[8] – mailbox empty interrupt, and IRR[9] – message overrun/overwrite interrupt. If a mailbox is configured as a receive mailbox, a mask at the corresponding bit prevents the generation of a receive interrupt (IRR[1] and IRR[2] and IRR[9]) but does not prevent the setting of the corresponding bit in the RXPR, RFPR, or UMSR. Similarly, when a mailbox has been configured a transmit mailbox, a mask prevents the generation of an interrupt signal and setting of an mailbox empty interrupt due to successful transmission or transmission abortion (IRR[8]). However, it does not prevent the RCAN from clearing the corresponding TXPR/TXCR bit or setting the TXACK bit for successful transmission, and it does not prevent the RCAN from clearing the corresponding TXPR/TXCR bit or setting the ABACK bit for transmission abortion.

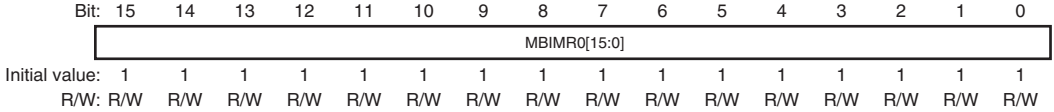
The mailbox activity can be masked by writing 1 to the corresponding bit in this register. At reset all mailbox interrupts are masked.

(1) MBIMR1



Bit	Bit Name	Initial Value	R/W	Description
15 to 0	MBIMR1 [15:0]	H'FFFF	R/W	These bits enable or disable interrupt requests from mailboxes 31 to mailbox 16, respectively. 0: Enables interrupt requests from IRR1, IRR2, IRR8, and IRR9. 1: Disables interrupt requests from IRR1, IRR2, IRR8, and IRR9.

(2) MBIMR0



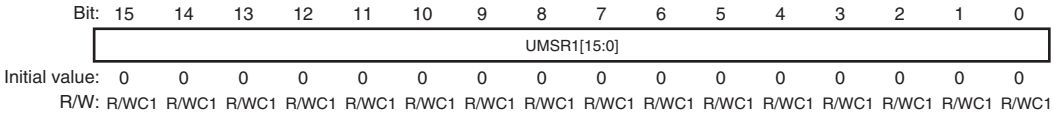
Bit	Bit Name	Initial Value	R/W	Description
15 to 0	MBIMR0 [15:0]	H'FFFF	R/W	These bits enable or disable interrupt requests from individual mailboxes 15 to 0, respectively. 0: Enables interrupt requests from IRR1, IRR2, IRR8, and IRR9. 1: Disables interrupt requests from IRR1, IRR2, IRR8, and IRR9. [Setting condition] The remote frame has been received in the corresponding mailbox.

22.6.8 Unread Message Status Registers 1 and 0 (UMSR1 and UMSR0)

These registers are 16-bit readable/conditionally-writable register that records the mailboxes whose contents have not been accessed by the CPU prior to a new message being received. If the CPU has not cleared the corresponding bit in the RXPR or RFPR when a new message for that mailbox is received, the corresponding UMSR bit is set to 1. This bit may be cleared by writing 1 to the corresponding bit in the UMSR. Writing 0 to the corresponding bit is ignored.

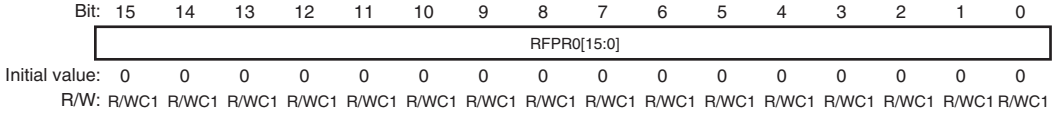
If a mailbox is configured as a transmit mailbox, the corresponding UMSR will not be set.

(1) UMSR1



Bit	Bit Name	Initial Value	R/W	Description
15 to 0	UMSR1 [15:0]	H'0000	R/WC1	These bits indicate that an unread received message has been overwritten or that an overrun has occurred in a corresponding mailbox among mailboxes 31 to 16. 0: [Clearing condition] Writing 1 to the corresponding bit. 1: indicates that an unread received message is overwritten by a new message or that an overrun has occurred. [Setting condition] When a new message is received before RXPR or RFPR is cleared

(2) UMSR0



Bit	Bit Name	Initial Value	R/W	Description
15 to 0	UMSR0 [15:0]	H'0000	R/WC1	These bits indicate that an unread received message has been overwritten or that an overrun has occurred in a corresponding mailbox among mailboxes 15 to 0. 0: [Clearing Condition] Writing to the corresponding bit. 1: indicates that an unread received message is overwritten by a new message or that an overrun has occurred. [Setting Condition] When a new message is received before RXPR or RFPR is cleared

22.7 Timer Registers

The timer is 16 bits and supports several source clocks. A prescale counter can be used to reduce the speed of the clock. It also supports three compare match registers (TCMR2, TCMR1, and TCMR0). The address map is as follows.

Important: The timer registers can only be accessed in word size (16 bits).

Table 22.11 RCAN Timer Registers

Register Name	Abbreviation	Address	Access Size (Bits)
Time trigger control register 0	TTCR0	H'080	16
Timer status register	TSR	H'088	16
Timer counter register	TCNTR	H'08C	16
Cycle time register	CYCTR	H'090	16
Timer compare match register 0	TCMR0	H'098	16
Timer compare match register 1	TCMR1	H'09C	16
Timer compare match register 2	TCMR2	H'0A0	16

22.7.1 Time Trigger Control Register 0 (TTCR0)

TTCR0 is a 16-bit readable/writable register that controls timer operation. Before setting periodic transmission or monitor registers, this register should be set to operate the timer.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TCR15	—	TCR13	TCR12	TCR11	TCR10	—	—	—	TCR6	TPSC5	TPSC4	TPSC3	TPSC2	TPSC1	TPSC0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	TCR15	0	R/W	Enable Timer When this bit is set, the timer TCNTR is operating. When this bit is cleared, TCNTR and CCR are cleared. 0: Clears TCNTR and CCR to stop the timer. 1: The timer is operating.

Bit	Bit Name	Initial Value	R/W	Description
14	—	0	R	Reserved This bit is always read as 0. The write value should always be 0. If 1 is written to this bit, the operation is not guaranteed.
13	TCR13	0	R/W	Cancellation by TCMR2 Cancels the messages in the transmission queue by setting TXCR bits corresponding to the TXPR bit, when both this bit and bit 12 are set and a compare match occurs when the RCAN is not in halt state 0: Disables the transmission cancellation by a TCMR2 compare match. 1: Enables the transmission cancellation by a TCMR2 compare match.
12	TCR12	0	R/W	TCMR2 Compare Match Enable When this bit is set, IRR11 is set by TCMR2 compare match. 0: Does not set IRR11 by a TCMR2 compare match. 1: Sets IRR11 by a TCMR2 compare match.
11	TCR11	0	R/W	TCMR1 compare match enable When this bit is set, IRR15 is set by a TCMR1 compare match. 0: Does not set IRR15 by a TCMR1 compare match. 1: Sets IRR15 by a TCMR1 compare match.
10	TCR10	0	R/W	TCMR0 compare match enable When this bit is set, IRR14 is set by a TCMR0 compare match. 0: Does not set IRR14 by a TCMR0 compare match. 1: Sets IRR14 by a TCMR0 compare match.
9 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6	TCR6	0	R/W	Timer Clear-Set Control by TCMR0 Specifies if the timer is to be cleared and set to H'0000 when the TCMR0 matches the TCNTR. Note that the TCMR0 can also generate an interrupt signal to the CPU via IRR14. 0: Timer is not cleared by the TCMR0. 1: Timer is cleared by the TCMR0.

Bit	Bit Name	Initial Value	R/W	Description
5 to 0	TPSC[5:0]	All 0	R/W	Timer Prescaler These bits set the timer source clock (4*[RCAN system clock]) to be divided before it is used for the timer. This prescaler function is available only in event-trigger mode. The following shows the relationship between source clock period and the timer period 000000: 1 × source clock 000001: 2 × source clock 000010: 3 × source clock 000011: 4 × source clock 000100: 5 × source clock : 111111: 64 × source clock

22.7.2 Timer Status Register (TSR)

TSR is a 16-bit read-only register that allows the CPU to monitor the timer compare match status and the timer-overrun status.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	TSR3	TSR2	TSR1	TSR0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	TSR3	0	R*	Timer Compare Match Flag 2 Indicates that a compare match has occurred in the timer compare match register 2 (TCMR2). When the value set in TCMR2 matches the cycle time register (TCMR2 = CYCTR), this bit is set if TTCR0 bit 12 = 1. Note that this bit is read-only and is cleared when IRR11 (timer compare match interrupt 2) is cleared. 0: indicates that a timer compare match has not occurred in TCMR2. [Clearing condition] Writing 1 to IRR11 (timer compare match interrupt 1) 1: Indicates that a timer compare match has occurred in TCMR2. [Setting condition] TCMR2 matches the cycle time (TCMR2=CYCTR) when TTCR0 bit 12 = 1.

Bit	Bit Name	Initial Value	R/W	Description
2	TSR2	0	R*	Timer Compare Match Flag 1 Indicates that a compare match has occurred in the timer compare match register 1 (TCMR1). When the value set in TCMR1 matches to the cycle time register (TCMR1 = CYCTR), this bit is set if TTCR0 bit 11 =1. Note that this bit is read-only and is cleared when IRR15 (timer compare match interrupt 1) is cleared. 0: Indicates that a timer compare match has not occurred in TCMR1. [Clearing condition] Writing 1 to IRR15 (timer compare match interrupt 1) 1: Indicates that a timer compare match has occurred in TCMR1. [Setting condition] TCMR1 matches the cycle time (TCMR1 = CYCTR) when TTCR0 bit 11 = 1.
1	TSR1	0	R*	Timer Compare Match Flag 0 Indicates that a compare match has occurred in the compare match register 0 (TCMR0). When the value set in TCMR0 matches to the timer value (TCMR0 = TCNTR), this bit is set if TTCR0 bit 10 = 1. Note that this bit is read-only and is cleared when IRR14 (timer compare match interrupt 0) is cleared. 0: Indicates that a compare Match has not occurred in TCMR0. [Clearing condition] Writing 1 to IRR14 (timer compare match interrupt 0) 1: Indicates that a compare match has occurred in TCMR0. [Setting condition] TCMR0 matches to the timer value (TCMR0 = TCNTR).

Bit	Bit Name	Initial Value	R/W	Description
0	TSR0	0	R*	Timer Overrun/Message Error This flag is assigned to three different functions. It indicates that the timer has overrun when it operates in event-trigger mode, and that an error detected on the CAN bus has occurred in test mode, respectively. Test mode has higher priority with respect to the other settings. 0: Indicates that the timer (TCNTR) has not overrun in event-trigger mode, or that a message error has not occurred in test mode. [Clearing condition] Writing 1 to IRR13 1: [Setting condition] The timer (TCNTR) has overrun and changed from H'FFFF to H'0000 in event-trigger mode, or a message error has occurred in test mode.

Note: These bits can only be read. The CPU can monitor the cycle counter, timer and compare-match registers by reading these bits. These bits cannot be modified.

22.7.3 Timer Counter Register (TCNTR)

TCNTR is a 16-bit readable/writable register that allows the CPU to monitor and modify the free running timer counter. The timer starts counting by setting TTCR0[15] = 1.

The value of the timer counter prescaler is set by the TPSC[5:0] bits in the TTCR0 register

When the timer matches TCMR0 (Timer Compare Match Register 0) + TTCR0 [6] is set to 1, the TCNTR is cleared to H'0000 and starts running again.

Notes: 1. When TTCR0 bit 15 = 0, TCNTR is always H'0000.
2. There could be a delay of a few clock cycles from the timer enable to TCNTR increment start. This is caused by the prescaler internal logic.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TCNTR[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

Note: This bit can only be written when the timer is enabled (TTCR0 [15] = 1).

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	TCNTR [15:0]	H'0000	R/W	These bits indicate the value of the free running timer.

22.7.4 Cycle Time Register (CYCTR)

This register is the exact copy of TCNTR.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CYCTR[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	CYCTR[15:0]	H'0000	R	These bits indicate the cycle time.

22.7.5 Timer Compare Match Registers 0 to 2 (TCMR0 to TCMR2)

TCMR0 to TCMR2 are 16-bit readable/writable registers that are capable of generating interrupt signals, clearing-setting the timer value (only supported by TCMR0) or clear the transmission messages in the queue (only supported by TCMR2). TCMR0 is compared with TCNTR, however, TCMR1 and TCMR2 are compared with CYCTR.

The value used for the compare can be configured independently for each register. In order to set flags, bits 12 to 10 in TTCR0 should be set.

(1) Interrupt:

The interrupts are flagged by the bit 11, bit 15 and bit 14 in the IRR accordingly when a compare match occurs, and setting these bits can be enabled by bits 12 to 10 in TTCR0. The generation of interrupt signal itself can be prevented by bit 11, bit 15 and bit 14 in the IMR. When a compare match occurs and the IRR11 (or IRR15 or IRR14) is set, bit 3, bit 2, or bit 1 in the TSR (timer status register) is also set. Clearing the IRR bit also clears the corresponding bit of TSR.

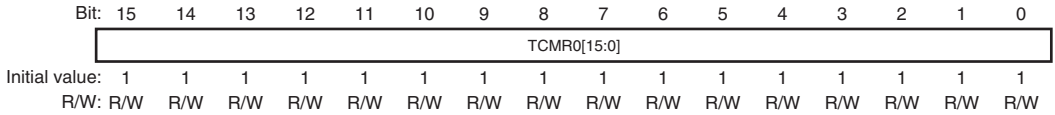
(2) Timer Clear-Set:

The timer value can only be cleared when a compare match occurs if it is enabled by the bit 6 in the TTCR0. TCMR1 and TCMR2 do not support this function.

(3) Cancellation of Messages in Transmission Queue:

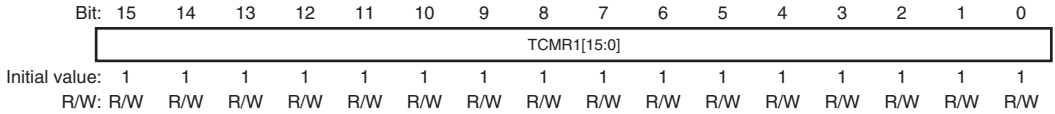
The messages in the transmission queue can only be cleared by the TCMR2 through setting TXCR when a compare match occurs while the RCAN is not in the halt state. TCMR1 and TCMR0 do not support this function.

(a) TCMR0



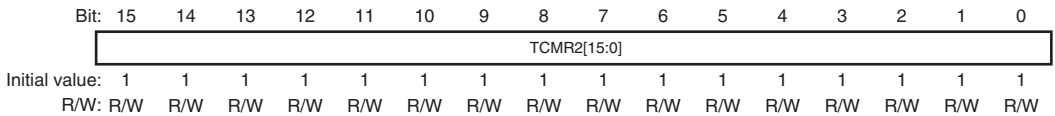
Bit	Bit Name	Initial Value	R/W	Description
15 to 0	TCMR0 [15:0]	H'FFFF	R/W	Timer Compare Match Register These bits indicate the TCNTR value when a compare match occurs.

(b) TCMR1



Bit	Bit Name	Initial Value	R/W	Description
15 to 0	TCMR1 [15:0]	H'FFFF	R/W	Timer Compare Match Register These bits Indicate the CYCTR value when a compare match occurs.

(c) TCMR2



Bit	Bit Name	Initial Value	R/W	Description
15 to 0	TCMR2 [15:0]	H'FFFF	R/W	Timer Compare Match Register These bits Indicate the CYCTR value when a compare match occurs.

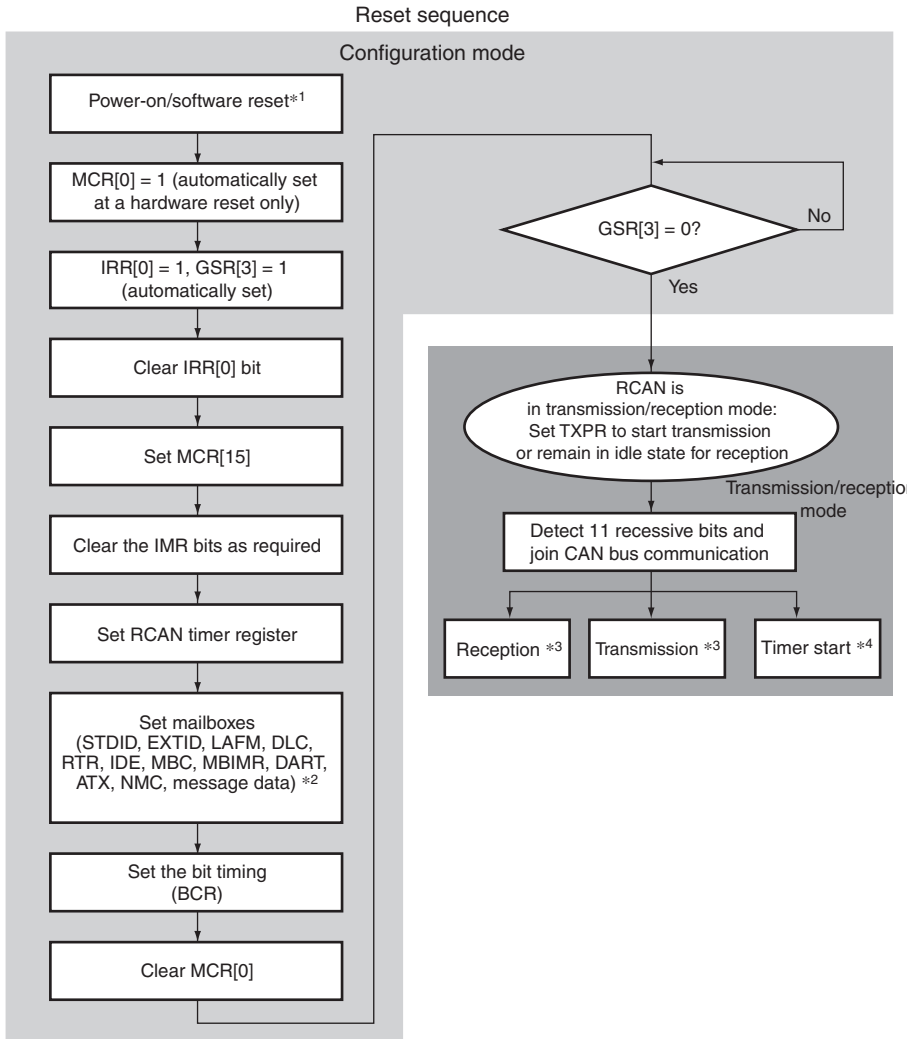
22.8 Operation

22.8.1 RCAN Setting

This section describes the RCAN settings in configuration mode or halt mode after hardware reset (power-on reset) or software reset (MCR0). In both conditions, the RCAN cannot join the CAN bus activity and configuration changes does not affect the traffic on the CAN bus.

(1) Reset Sequence

Figure 22.10 shows the procedure to set the RCAN after software reset or hardware reset. The RCAN setting is required to join the CAN bus activity because all the registers are initialized after reset. For details, see the notes in figure 22.10.



Notes: 1. A software reset can be executed at any time by setting MCR[0] to 1.
2. Mailboxes are comprised of RAM. Therefore, all of the mailboxes that have been enabled by MBC should be initialized even if some of them are not used.
3. If no bit in TXPR is set, the RCAN will receive the next message. When a bit in TXPR is set, the RCAN starts message transmission and arbitration is performed between the RCAN and CAN bus. If the RCAN loses the arbitration, it enters reception state.
4. The timer can start operation at any time after the timer control register is set.

Figure 22.10 Reset Sequence

(2) Halt Mode

When the RCAN is in halt mode, it cannot join the CAN bus activity. Consequently, the user can modify all the requested registers without influencing existing traffic on the CAN bus. It is important for this that the user waits for the RCAN to be in halt mode before to modify the requested registers - note that the transition to halt mode is not always immediate (transition will occurs when the CAN Bus is idle or in intermission). After the RCAN has made a transition to halt mode, GSR4 is set.

Once the configuration is completed the Halt request needs to be released. The RCAN will join CAN bus activity after the detection of 11 recessive bits on the CAN bus.

(3) CAN Sleep Mode

When the RCAN is in CAN sleep mode, the clock for the main blocks of the RCAN is stopped in order to reduce power consumption. Only the following user registers are clocked and can be accessed: MCR, GSR, IRR and IMR. Because interrupts related to transmission (TXACK and ABACK) and reception (RXPR and RFPR) cannot be cleared in sleep mode, they must to be cleared in advance.

Figure 22.11 shows the flow to follow to move RCAN into CAN sleep mode.

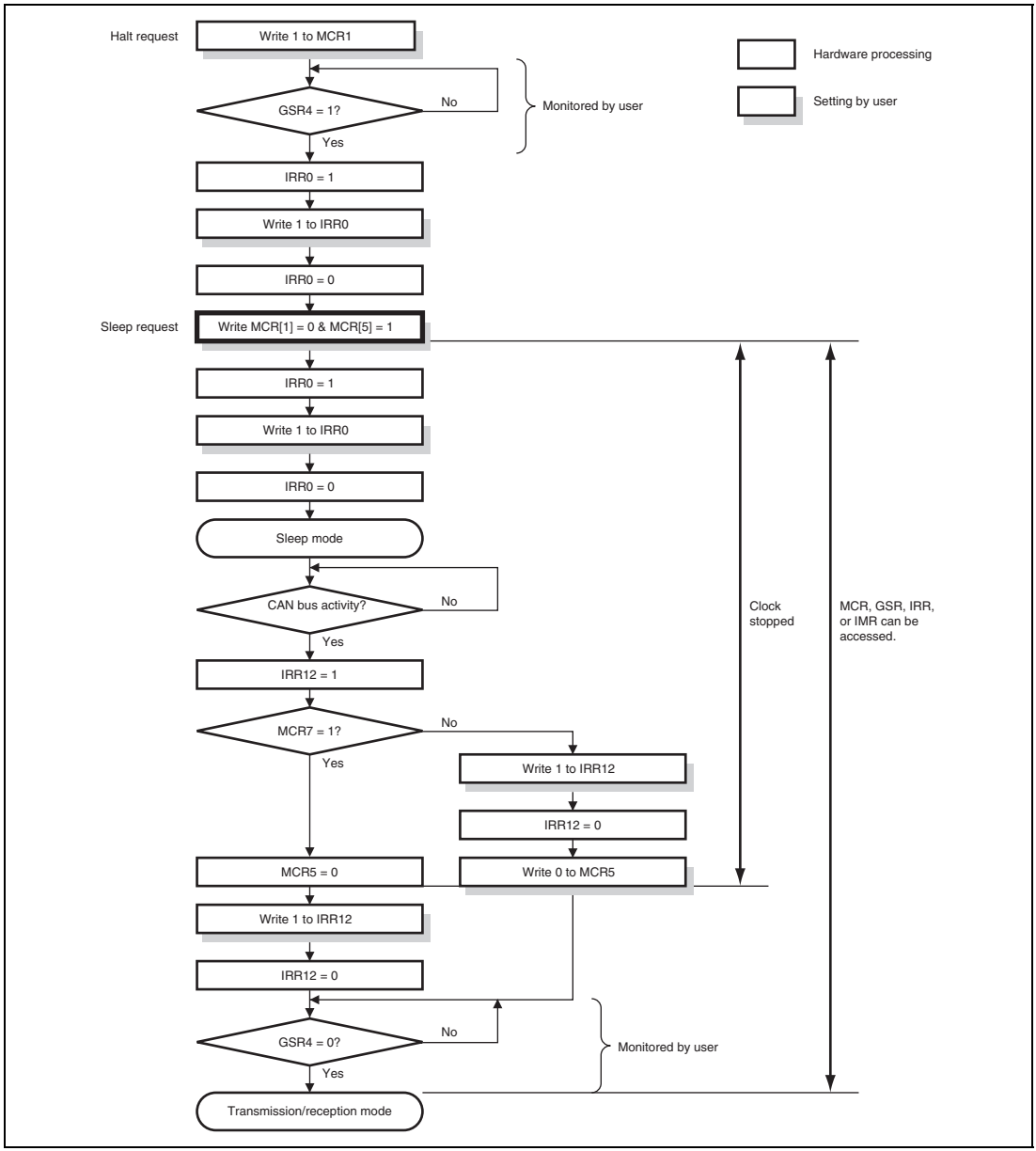
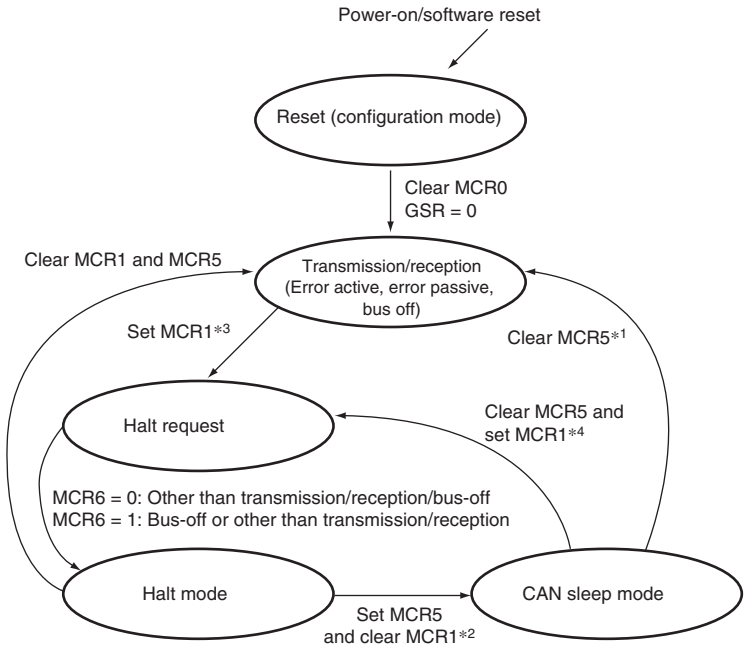


Figure 22.11 CAN Sleep Mode

Figure 22.12 shows allowable state transitions. The MCR5 bit (CAN sleep mode) should be set in halt mode only. After MCR1 is set, GSR4 should be set and then the RCAN is placed in halt mode before the MCR1 is cleared.



Notes: 1. MCR5 is automatically cleared when a dominant bit is detected in the CAN bus with MCR7 = 1. Writing 0 can also clear MCR5.
2. MCR1 is cleared by the software. Clearing MCR1 and setting MCR5 should be performed by the same instruction.
3. MCR1 should not be cleared by the software before GSR4 is set. If the RCAN enters bus-off mode when MCR14 and MCR6 are set, MCR1 is automatically set by the hardware.
4. If MCR5 is cleared and MCR1 is set at the same time, the RCAN enters halt request state, then a transition is made to halt mode without transmission/reception.

Figure 22.12 State Transitions

Table 22.12 shows conditions to access registers.

Table 22.12 Register Access Conditions

	Registers									
Status Mode	**MCR, GSR**	**IRR, IMR**	**BCR**	**MBIMR Timer**	**Flag Register**	**Mail-boxes (Control 0, LAFM)**		**Mail-boxes (Data)**	**Mail-boxes (Control 1)**	
Reset	Possible	Possible	Possible	Possible	Possible	Possible		Possible	Possible	
Transmit/ Receive	Possible	Possible	Impossible	Possible	Possible	Impossible	Possible*	Possible*	Impossible	Possible*
Halt request	Possible	Possible	Impossible	Possible	Possible	Impossible	Possible*	Possible*	Impossible	Possible*
Halt mode	Possible	Possible	Impossible	Possible	Possible	Possible		Possible	Possible	
CAN sleep mode	Possible	Possible	Impossible	Impossible	Impossible	Impossible		Impossible	Impossible	

[Legend]

Yes: The register can be accessed.

No: The register cannot be accessed.

Note: * The register can be accessed when TXPR0 is not set.

22.8.2 Test Mode Settings

The RCAN has various test modes. The TST[2:0] bits in the MCR register are used to select the RCAN test mode. The default (initialized) settings allow RCAN to operate in normal mode. Table 22.13 shows the test mode settings.

The test mode can be selected only in configuration mode. The user must then exit the configuration mode (ensuring BCR0/BCR1 is set) in order to run the selected test mode.

Table 22.13 Test Mode Settings

TST2	TST1	TST0	Mode
0	0	0	Normal mode
0	0	1	Receive-only mode
0	1	0	Self test mode 1 (external)
0	1	1	Self test mode 2 (internal)
1	0	0	Write error counter
1	0	1	Error passive mode
1	1	0	Setting prohibited
1	1	1	Setting prohibited

- Normal Mode

  The RCAN operates in normal mode.

- Receive-Only Mode

  ISO-11898 requires this mode for baud rate detection. The error counters are cleared and disabled so that the TEC/REC does not increase the values, and the Tx Output is disabled so that the RCAN does not generate error frames or acknowledgment bits. IRR13 is set when a message error occurs.

- Self Test Mode 1 (External)

  The RCAN generates its own acknowledge bit, and can store its own messages into a reception mailbox (if required). The Rx/Tx pins must be connected to the CAN bus.

- Self Test Mode 2 (Internal)

  The RCAN generates its own acknowledge bit, and can store its own messages into a reception mailbox (if required). The Rx/Tx pins do not need to be connected to the CAN bus or any external devices, as the internal Tx is looped back to the internal Rx. Tx pin outputs only recessive bits and Rx pin is disabled.

- Write Error Counter

  The TEC/REC can be written in this mode. The RCAN can be forced to become error passive mode by writing a value greater than 127 into the error counters. The value written into the TEC is used to write into the REC, so only the same value can be set to these registers. Similarly, the RCAN can be forced to become an error warning by writing a value greater than 95 into them.

  The RCAN needs to be in halt mode when writing into TEC/REC (MCR1 must be set to 1 when writing to the Error Counter). Furthermore, this test mode needs to be cancelled before halt mode is cancelled.

- Error Passive Mode

  The RCAN can be forced to enter error passive mode.

  The REC will not be modified by implementing error passive mode. However, once running in error passive mode, the REC will increment normally when an error has been received. In this mode, the RCAN will enter bus off state if the TEC reaches 256 (Dec). However, when this mode is used, the RCAN cannot enter error active mode. Consequently, at the end of the bus off recovery sequence, the RCAN will make a transition to error passive mode not to error active mode.

When a message error occurs, IRR13 is set in all test modes.

22.8.3 Message Transmission Sequence

(1) Message Transmission Request

Figure 22.13 shows a sample sequence used to transmit a CAN frame onto the bus.

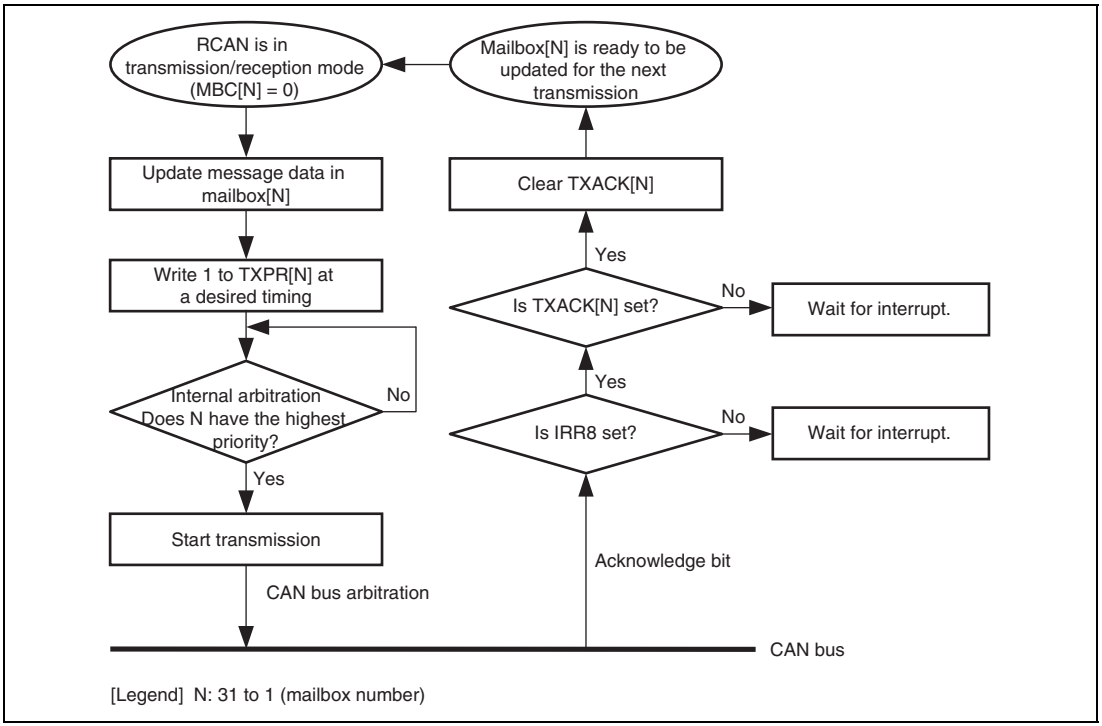
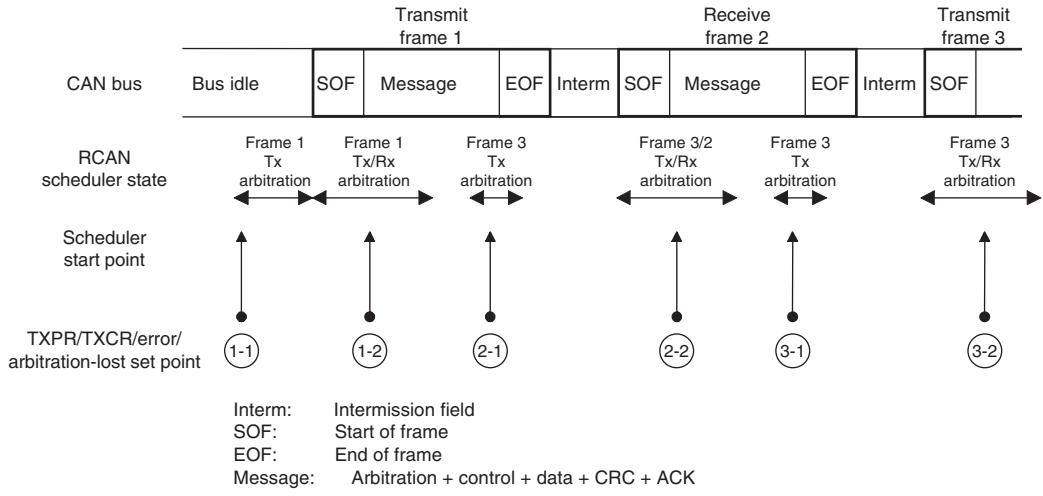


Figure 22.13 Message Transmission Request

As described in the previous register section, note that IRR8 is set when one of the TXACK or ABACK bits is set. This means that one of the mailboxes has completed its transmission or transmission abortion and is now ready to be updated for the next transmission, whereas, the GSR2 means that there is currently no transmission request made (No TXPR flags set).

(2) Internal Arbitration for Transmission

Figure 22.14 illustrates how the RCAN schedules the transmission-requested messages in the correct order based on the CAN identifier. 'Internal arbitration' picks up the message of highest priority among transmit-requested messages.



1-1: When a TXPR bit(s) is set while the CAN bus is in idle state, the internal arbitration starts operating immediately and the transmission is started.
1-2: Operations for both transmission and reception starts at SOF. Since there is no reception frame, the RCAN becomes transmitter.
2-1: At CRC delimiter, internal arbitration to search next message transmitted starts.
2-2: Operations for both transmission and reception starts at SOF. Because of a reception frame with higher priority, the RCAN becomes a receiver. Therefore, reception is carried out instead of transmitting frame 3.
3-1: At CRC delimiter, the internal arbitration starts to search the next message to be transmitted.
3-2: Operations for both transmission and reception starts at SOF. Since a transmission frame has higher priority than a reception frame, the RCAN becomes a transmitter.

Figure 22.14 Internal Arbitration for Transmission

If an error is detected on the CAN bus, the internal arbitration for the next transmission is also performed at the beginning of each error delimiter. It is also performed at the beginning of error delimiters following overload frame.

Because the arbitration for transmission is performed at CRC delimiter, if a remote frame request is received into a mailbox with ATX = 1, a message to be transmitted for the request will join the transmission arbitration only at the following bus idle, CRC delimiter, or error delimiter.

The corresponding message abortion processing will be performed with a delay of maximum 1 CAN frame following the assertion of TXCR, depending on the CAN bus status.

(3) Timer Operation

Figure 22.15 shows the timing diagram of the timer.

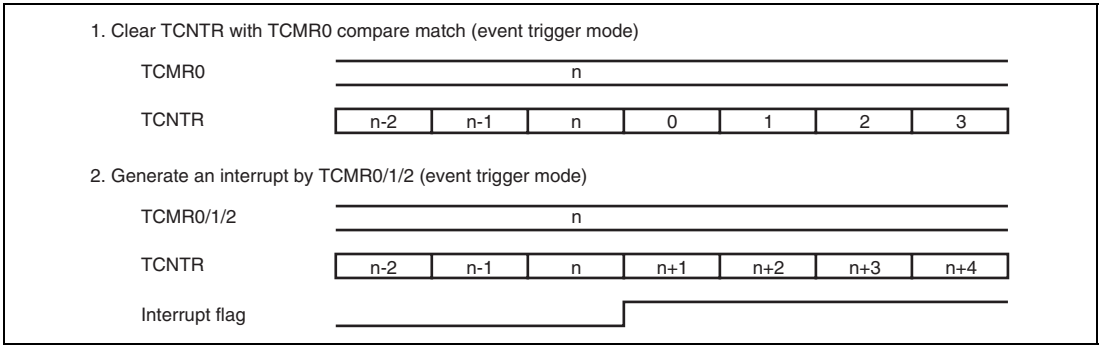
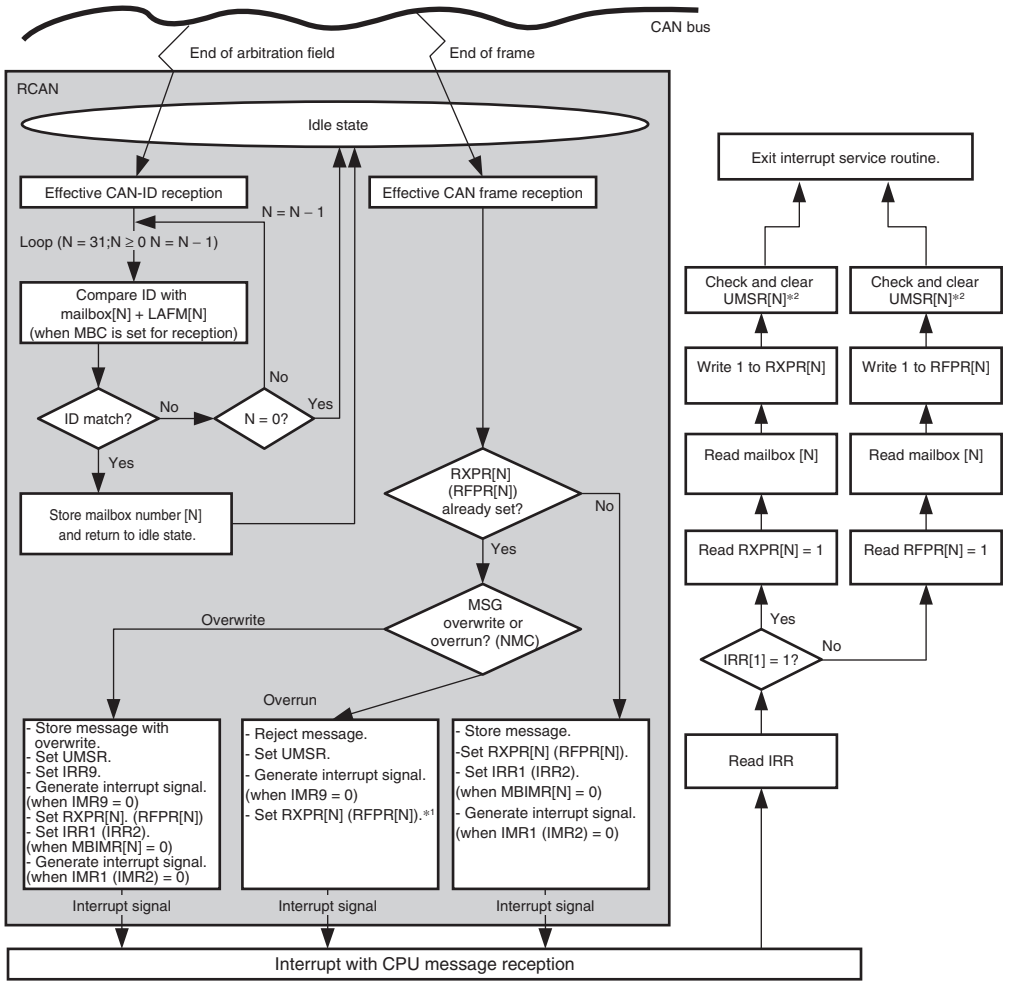


Figure 22.15 Timing Diagram of Timer

22.8.4 Message Reception Sequence

Figure 22.16 shows the message reception sequence.



[Legend] N: 31 to 0 (mailbox number)

Notes: 1. RXPR[N]/RFPR[N] is set again only when the CPU clears RXPR[N]/RFPR[N] at the same time that UMSR is set with overrun, even though the message has not been updated.
Though the timestamp is updated, the value before clearing RXPR[N]/RFPR[N] can be read by reading the mailbox.

2. When the mailbox N is set to be overwritten (NMC = 1), the message should be discarded while UMSR[1] = 1. At this time, UMSR[N] is cleared and all the interrupt service routines are exited. When the mailbox N is set to be overrun (NMC = 0), RXPR[N]/RFPR[N]/UMSR[N] should be cleared and the message should be considered unused while UMSR[N] = 1.

Figure 22.16 Message Reception Sequence

When the RCAN recognizes the end of the Arbitration field while receiving a message, it starts comparing the received identifier to the identifiers set in the mailboxes, starting from mailbox 31 to mailbox 0. It first checks the MBC if it is configured as a receive mailbox, and reads the CAN-ID of mailbox 31 (if configured as receive) to finally compare them to the received ID. If it does not match, the same check takes place at mailbox 30 (if configured as receive mailbox). Once the RCAN finds a matching identifier, it stores the number of mailbox [N] into an internal buffer, stops the search, and returns to idle state, waiting for the EndOfFrame (EOF) to come. When the 6th bit of EOF is notified by the CAN interface logic, the received message is written or abandoned, depending on the NMC bit.

Modification of the RCAN message IDs or LAFM settings are not allowed during communication. Entering halt mode and configuration mode are one of ways to modify configuration. If it is written into the corresponding mailbox, including the CAN-ID, i.e., there is a possibility that the CAN-ID is overwritten by a different CAN-ID of the received message due to the LAFM used. This also implies that, if the identifier of a received message matches to ID + LAFM of 2 or more mailboxes, the higher numbered mailbox will always store the relevant messages and the lower numbered mailbox will never receive messages. Therefore, the settings of the identifiers and LAFMs need to be carefully selected.

With regards to the reception of data and remote frames described in figure 22.16, the clearing of the UMSR flag after the reading of IRR is to detect situations where a message is overwritten by a new incoming message stored in the same mailbox (if its NMC = 1) while the interrupt service routine is running. If a overwrite condition is detected in the final check of UMSR, the message should be discarded and read again.

If UMSR is set and the mailbox is configured for overrun (NMC = 0) the message is still valid, however it is obsolete as it is not reflecting the latest message monitored on the CAN bus. Access the full mailbox content before clearing the related RXPR/RFPR flag.

Note that in the case a received remote frame is overwritten by a data frame, both the remote frame receive interrupt (IRR2) and data frame received interrupt (IRR1) and also the Receive Flags (RXPR and RFPR) are set. In an analogous way, the overwriting of a data frame by a remote frame, leads to setting both IRR2 and IRR1.

When a message is received and stored into a mailbox all the fields of the data not received are stored as zero. The same applies when a standard frame is received. The extended identifier part (EXTID [17:0]) is written as zero.

22.8.5 Reconfiguration of Mailbox

When re-configuration of mailboxes is required, the following procedures should be taken.

(1) Changing Transmit Box Settings

Two cases are possible.

- Change of ID, RTR, IDE, LAFM, Data, DLC, NMC, ATX, DART

  The settings can be changed only when MBC = B'000. Confirm that the corresponding TXPR is not set. The settings (except MBC bit) can be changed at any time.

- Change from transmit box to receive box

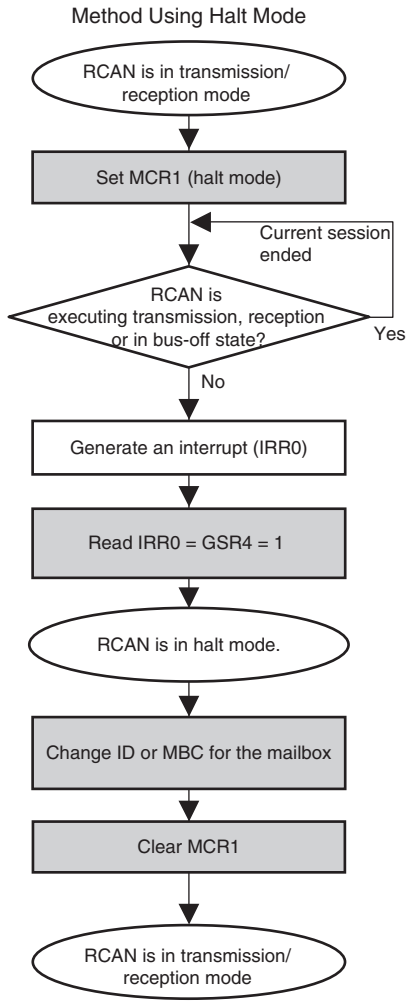
  Confirm that the corresponding TXPR is not set. The change can be made only in the halt or reset state. Note that it may take long for the RCAN to transit to halt state if the change is made when the RCAN is receiving or transmitting a message (because the transition to the halt state is delayed until the reception/transmission is completed). The RCAN cannot receive or transmit messages during the halt state.

  If the RCAN is in the bus off state the transition to halt state depends on the settings of bits 6 and 14 of MCR.

(2) Changing Receiver Box Settings (ID, RTR, IDE, LAFM, Data, DLC, NMC, ATX, DART, MBC) and Changing from Receiver Box to Transmitter Box

The settings can be changed only in halt mode. When a message is currently on the CAN bus and the RCAN is a receiver mode, the message is not lost. The RCAN transits to halt mode after the current reception is completed. Note that the transition may take long if settings are changed when the RCAN is receiving or transmitting a message (because the transition to the halt state is delayed until the reception/transmission is completed). The RCAN cannot receive/transmit messages during the halt state.

If the RCAN is in bus off state, a transition to halt mode is performed depending on the settings of bits 6 and 14 in MCR.



Note: The shadowed box indicates processing which should be done by the software (CPU).

Figure 22.17 Change of Receive Box ID or Change from Receive Box to Transmit Box

22.9 Interrupt Sources of RCAN

Table 22.14 shows the interrupt sources of RCAN. These sources can be masked by the mailbox interrupt mask registers (MBIMR1, MBIMR0) and the interrupt mask register (IMR). For the interrupt vectors corresponding to the respective interrupt requests, refer to section 7, INTC/INTC2.

Table 22.14 Interrupt Sources of RCAN

Module Name	Type	Interrupt Sources		Interrupt Flag (IRR Register)
For both RCAN0 and RCAN1	Error processing	Error passive (TEC ≥ 128 or REC ≥ 128)		IRR5
		Bus off (TEC ≥ 256)/recovery from bus off		IRR6
		Transmit error warning (TEC ≥ 96)		IRR3
		Receive error warning (REC ≥ 96)		IRR4
	Overrun processing	Transition to reset/halt/CAN sleep		IRR0
		Over load frame transmission		IRR7
		Overwrite of unread message (overrun)		IRR9
		TCMR2 compare match		IRR11
		Detection of CAN bus operation during CAN sleep		IRR12
		Timer overrun/message error		IRR13
		TCMR0 compare match		IRR14
		TCMR1 compare match		IRR15
	Mailbox processing	Mailbox 0 reception (MB0*3)	Data frame reception	IRR1*2
			Remote frame reception	IRR2*1
		Mailbox 1 to 31 reception (MB1 to MB31*4)	Data frame reception	IRR1*2
			Remote frame reception	IRR2*1
		Transmission of message/cancellation of transmission (slot empty)		IRR8

Notes: 1. IRR2 corresponds to the remote frame reception flags (RFPR1[15:0]) and RFPR0[15:0]) of mailboxes 31 to 0.
2. IRR1 corresponds to the data frame reception flags (RXPR1[15:0]) and RXPR0[15:0]) of mailboxes 31 to 0.
3. MB0 indicates remote frame reception or data frame reception by mailbox 0. Refer to section 7.4.14, INTC2/Interrupt Detailed Source Registers (INT2B0 to INT2B52).

4. MB1 to MB31 indicate remote frame reception or data frame reception by mailboxes 31 to 1. Refer to section 7.4.14, INTC2/Interrupt Detailed Source Registers (INT2B0 to INT2B52).

Section 23 USB

The USB module consists of the USB-LINK circuits (EHCI, OHCI, and FUNCTION), PHY circuits, interface circuits, and common registers (REGS). This section describes the common registers (REGS) and external USB-PHY circuits. For the USB-LINK (EHCI, OHCI, and FUNCTION) registers, refer to sections 23A, USB2.0 Host Controller, 23B, USB1.1 Host Controller, and 23C, USB 2.0 Function Module (USBF).

23.1 Features

The USB module provides the following control functions.

- Initializing the circuits in the USB module
- Port: Switching host and function
- Polarity: Switching OVC and VBUS

Figure 23.1 shows a block diagram.

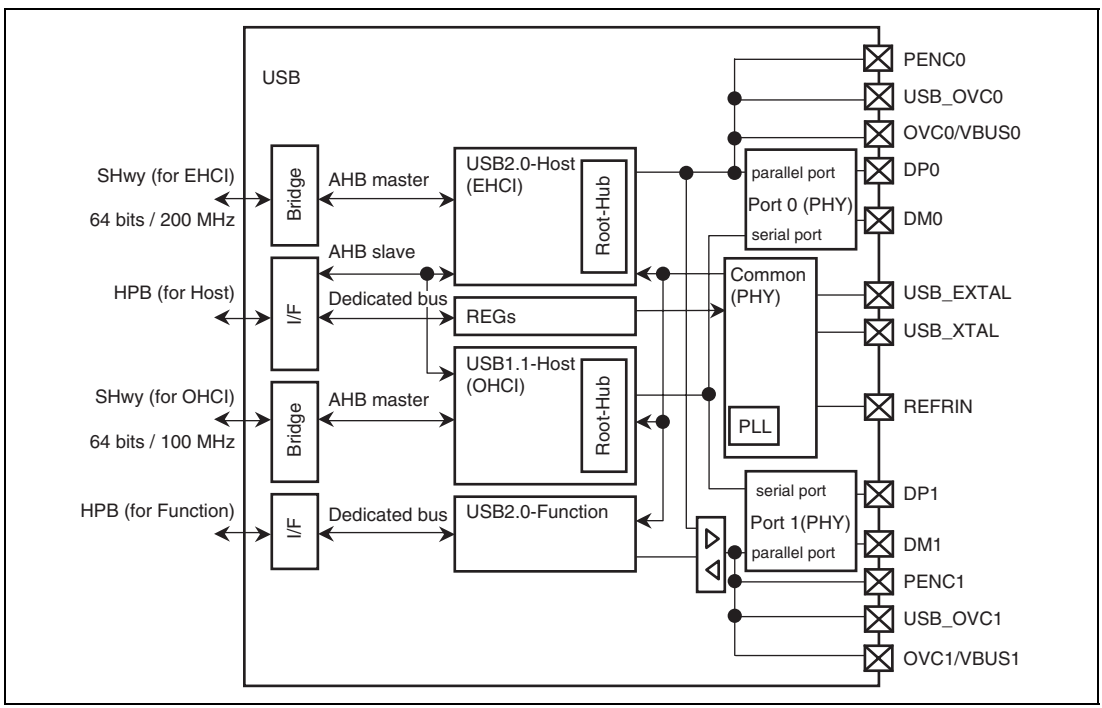


Figure 23.1 USB Block Diagram

23.2 External Pins

Table 23.1 Pin Configuration

Pin Name	Function	I/O	Description
USB_EXTAL	PLL oscillation clock	Input	Connect this pin to the crystal oscillator for USB.
USB_XTAL	PLL oscillation clock	Output	An external clock signal can also be input through the USB_EXTAL pin.
DP0	USB port 0 D+	I/O	USB port0 D+
DM0	USB port 0 D-	I/O	USB port0 D-
PENC0	Port Enable 0	Output	Host: Power-supply control 1: Power-supply IC is turned on. 0: Power-supply IC is turned off. The initial value is 0. (power supply is turned off)
OVC0/VBUS0	Port 0 overcurrent (5-V I/F)	Input	Host: Power-supply IC overcurrent detection Active-low at initial state, and active mode can be changed by a register.
USB_OVC0	Port 0 overcurrent (3.3-V I/F)	Input	Host: Power-supply IC overcurrent detection Active-low at initial state, and active mode can be changed by a register.
DP1	USB port 1 D+	I/O	USB port1 D+
DM1	USB port 1 D-	I/O	USB port1 D-
PENC1	Port Enable 1	Output	Host: Power-supply control 1: Power-supply IC is turned on. 0: Power-supply IC is turned off. Function: Port output function The initial value is 0.
OVC1/VBUS1	Port 1 VBUS/ overcurrent (5-V I/F)	Input	Host: Power-supply IC overcurrent detection Function: Cable connection/disconnection detection Active-low at initial state (host mode), and active mode can be changed by a register.

Pin Name	Function	I/O	Description
USB_OVC1	Port 1 overcurrent (3.3-V I/F)	Input	Host: Power-supply IC overcurrent detection, Active-low at initial state, and active mode can be changed by a register.
REFRIN	A pin for connecting the external resistor	I/O	A pin for connecting the external resistor

Note: GPIO is selected as an initial state of the PENC1 pin. To cause the pin to function as PENC1, change the pin function to PENC1 and then connect the USB module to the USB function device.

23.3 Register Descriptions

The registers can be accessed only by the CPU. When they are accessed by other modules, operation is not guaranteed. Bit widths of these resisters are all 32 bits, so should be accessed in longword (32 bits) units. Access in other units is not supported.

Table 23.2 (1) List of Registers

Register Name	Abbreviation	R/W	Initial Value	Address	Access size
Port Control 0	USBPCTRL0	R/W	H'00000000	H'FFE7 0800	32
Port Control 1	USBPCTRL1	R/W	H'00000000	H'FFE7 0804	32
Port Status	USBST	R	H'00000000	H'FFE7 0808	32
EHCI Control 0	USBEH0	R/W	H'00000000	H'FFE7 080C	32
OHCI Control 0	USBOH0	R/W	H'00000000	H'FFE7 081C	32
USB Control 0	USBCTL0	R/W	H'00000224	H'FFE7 0858	32

Note: Addresses other than the above must not be write-accessed to. If written to, operation is not guaranteed. If read, an undefined value is read.

Table 23.2 (2) Register State in Each Operating Mode

Abbreviation	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
USBPCTRL0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized*
USBPCTRL1	H'00000000	H'00000000	Retained	Retained	Retained	Initialized*
USBST	H'00000000	H'00000000	Retained	Retained	Retained	Initialized*
USBEH0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized*
USBCTL0	H'00000224	H'00000224	Retained	Retained	Retained	Initialized*
USBOH0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized*

Note: * Initialized: The value is the one written in "Power-on Reset".

[Legend for Register Description]

Initial value: Register value after a reset

—: Undefined value

R/W: Readable/writable. The write value can be read.

R/WC1: Readable/writable. Only writing 1 initializes the bit. Writing 0 is ignored.

R: Read-only. For Read Modify Write, follow the description of each bit.

—/W: Write-only. The read value is undefined.

23.3.1 Port Control 0 (USBPCTRL0)

USBPCTRL0 is a register that sets the port function.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	OVC1/ VBUS1	OVC0/ VBUS0	—	—	—	PENC	OVC0	—	OVC1	PORT1
Initial value:	—	—	—	—	—	—	0	0	—	—	—	0	0	—	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R	R/W	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	—	R	Reserved. Setting other than 0 is prohibited.
9	OVC1/ VBUS1	0	R/W	Switches the OVC input pin for port 1. 0: Input is from the OVC1/VBUS1 pin 1: Input is from the USB_OVC1 pin 5V should be input to the OVC1/VBUS1pin if the setting of this bit is 1. Function mode: Set this bit to 0.
8	OVC0/ VBUS0	0	R/W	Switches the OVC input pin for port 0. 0: Use the input from the OVC0/VBUS0 pin 1: Use the input from the USB_OVC0 pin. 5 V should be input to the OVC0/VBUS0 pin if this bit is set to 1.
7 to 5	—	—	R	Reserved Setting other than 0 is prohibited.
4	PENC	0	R/W	The bit for PENC1 at function mode. 0: Output low 1: Output high
3	OVC0	0	R/W	Host mode: OVC0 polarity switching 0: Active-low 1: Active-high
2	—	—	R	Reserved The setting other than 0 is prohibited.

Bit	Bit Name	Initial Value	R/W	Description
1	OVC1	0	R/W	Host mode: OVC1 polarity switching 0: Active-low 1: Active-high Function mode: Set this bit to 1. (zero/one corresponds to disconnected/connected)
0	PORT1	0	R/W	Selects host or function for port 1. 0: Host 1: Function

23.3.2 Port Control 1 (USBPCTRL1)

USBPCTRL1 is a register that is used for setting the port function or applying a software-reset. The target registers that are initialized when the RST bit is used are all USB registers except USBPCTRL0, USBPCTRL1, USBEH0, USBOH0, USBCTL0, and USBST.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RST	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	PHY RST	PLL ENB	PHY ENB
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	RST	0	R/W	Resets the USB module when this bit is set to 1. To restore the USB module from the reset state, clear this bit to 0. 0: Negate 1: Assert
30 to 3	—	—	R	Reserved. Setting other than 0 is prohibited.
2	PHYRST	0	R/W	Resets the USB-PHY. To use the USB module, set this bit to 1 after setting the PHYENB to 1 and the PLLENB to 1 so the PLL frequency is stabilized. 0: Assert 1: Negate
1	PLLENB	0	R/W	Enables the PLL in the USB-PHY. To use the USB module, set this bit to 1 after setting the PHYENB to 1. 0: Disable 1: Enable
0	PHYENB	0	R/W	Enables the USB-PHY. To use the USB module, set this bit to 1. 0: Disable 1: Enable

Procedure of using USB module:

1. Set the USBPCTRL1.PHYENB to 1.
2. Set the USBPCTRL1.PLLENB to 1.
3. Confirm the USBST.PLL and USBST.ACT are set to 1.
4. Set the USBPCTRL1.PHYRST to 1.

23.3.3 USB Status (USBST)

USBST is a register that indicates the USB module state.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ACT	PLL	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	ACT	—*[1]	R	Indicates the USB module state. Confirm both ACT and PLL are set to 1 before setting registers of the USB module. 0: The USB module is initializing. 1: The USB module is active.
30	PLL	—	R	Indicates the USB PLL state.*[2] Confirm both ACT and PLL are set to 1 before setting registers of the USB module. 0: PLL oscillation is not stable. 1: PLL oscillation is stable.
29 to 0	—	—	R	Reserved Setting other than 0 is prohibited.

Notes: 1. Indicates 0 for approximately 1 ms after the LSI starts up.
2. The USB clock input is necessary for stabilization of PLL oscillation.

23.3.4 EHCI Control 0 (USBEH0)

USBEH0 is a register that controls data alignment in the bridge section on the EHCI side.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	USBEH0															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	USBEH0															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	USBEH0	H'0000 0000	R/W	Bus alignment control register For details, refer to section 23.4, Initial Settings.

23.3.5 USB Control 0 (USBCTL0)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CLK SEL	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	1	0	0	0	1	0	0	1	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved. Setting other than 0 is prohibited.
9	—	1	R	Reserved. Setting other than 1 is prohibited.
8	—	0	R	Reserved. Setting other than 0 is prohibited.
7	CLKSEL	0	R/W	USB clock mode select 0: Crystal oscillator mode 1: External clock mode
6	—	0	R	Reserved. Setting other than 0 is prohibited.
5	—	1	R	Reserved. Setting other than 1 is prohibited.
4, 3	—	All 0	R	Reserved. Setting other than 0 is prohibited.
2	—	1	R	Reserved. Setting other than 1 is prohibited.
1, 0	—	All 0	R	Reserved. Setting other than 0 is prohibited.

23.3.6 OHCI Control 0 (USBOH0)

USBOH0 is a register that controls data alignment in the bridge section on the OHCI side.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	USBOH0															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	USBOH0															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	USBOH0	H'00000000	R/W	Bus alignment control register For details, refer to section 23.4, Initial Settings.

23.4 Initial Settings

The following settings are necessary for using the USB modules.

The following settings should be ensured before accessing the EHCI/OHCI/FUNCTION registers.

- Setting 1
  Address: H'FFE7 0804 (USBPCTRL1)
  Write value: H'0000 0001 (Set the PHYENB bit to 1 to release USB-PHY from standby mode.)
- Setting 2
  Address: H'FFE7 0804 (USBPCTRL1)
  Write value: H'0000 0003 (Set the PLLENB bit to 1 to operate the USB-PHY internal PLL circuit.)
- Confirmation
  Address: H'FFE7 0808 (USBST)
  ACL = 1 (USB is active state.)
  PLL = 1 (USB PLL oscillation is stable.)
- Setting 3
  Address: H'FFE7 0804 (USBPCTRL1)
  Write value: H'00000007 (Set the PHYRST bit to 1 to release the USB-PHY internal logic from reset state.)
- Setting 1
  Address: H'FFE7 0094
  Initial value: H'0040 0040
  Write value: H'00FF 0040
- Setting 2
  Address: H'FFE7 009C
  Initial value: H'0000 0000
  Write value: H'0000 0001

The following settings are necessary for bus alignment processing.

Bus alignment on the EHCI side

- Settings
  Address: H'FFE7080C (USBEH0)
  Initial value: H'0000 0000
  Write value: See the table below.

Endian	Swap Setting	Register Value	Remarks
Big endian	No swap	H'0000_0003	Usually set this value.
	Byte swap	H'0000_0002	
	Word swap	H'0000_0001	
	Word-byte swap	H'0000_0000	
Little endian	No swap	H'0000_0000	Usually set this value.
	Byte swap	H'0000_0001	
	Word swap	H'0000_0002	
	Word-byte swap	H'0000_0003	

Bus alignment on the OHCI side

- Setting
  Address: H'FFE7 081C (USBOH0)
  Initial value: H'0000 0000
  Write value: See the table below.

Endian	Swap Setting	Register Value	Remarks
Big endian	No swap	H'8800_0003	Usually set this value.
	Byte swap	H'8800_0002	
	Word swap	H'8800_0001	
	Word byte swap	H'8800_0000	
Little endian	No swap	H'0000_0000	Usually set this value.
	Byte swap	H'0000_0001	
	Word swap	H'0000_0002	
	Word byte swap	H'0000_0003	

23.5 Examples of Handling Unused Pins

This section describes how to treat unused pins.

23.5.1 Example of DP and DM Connections

Leave these ports open when they are not used.

23.5.2 Example of OVC and PENC Connections

Set the OVC pins to the Not Active state on the host side and to the disconnected state on the function side when they are not used.

Example: When the OVC signal polarity is set to low-active on the host side, the signals should be pulled up if they are not used.

Leave PENC open when it is not used.

Note that the OVC and PENC pins are multiplexed with the GPIO signals. When selecting the GPIO functions for the pins, treat them as required in the GPIO processing.

23.6 Examples of External Circuit Connections

23.6.1 Introduction

The pin names of the USB 2.0 host/function module are used in the following descriptions. Table 23.3 provides an overview of the USB 2.0 host/function module pins.

Table 23.3 Overview of USB 2.0 Host/Function Module Pins

Abbr.	I/O	Name	Description
DP0/1	Input/ Output	USB D+ data	USB bus D+ data
DM0/1	Input/ Output	USB D- data	USB bus D- data
OVCn/ VBUSn (n = 0,1)	Input	VBUS input	Connect this pin to the USB bus VBUS
REFRIN	Input	Reference input	Connect this pin to the AG pin through a resistor of 5.6 kΩ ±1%.
USB_ EXTAL	Input	USB crystal oscillator/external clock	Connect this pin to the crystal oscillator for USB. An external clock can also be input to the USB_EXTAL pin.
USB_ XTAL	Output		
AV33	Power	Transceiver Analog pin power source	3.3-V analog power supply for pins
AV12	Power	Transceiver Analog core power	1.2-V analog power supply for the LSI core
AG	Power	Transceiver Analog core/pin ground	Ground for the 1.2-V supply for the analog core and for the 3.3-V analog supply for the pins
VCCQ*	Power	I/O circuit power source	3.3-V power supply for I/O pins
VSSQ*	Power	I/O circuit ground	3.3-V ground for I/O pins
VDD*	Power	Power	Core 1.2-V power supply for internal use
VSS*	Power	Ground	Core 1.2-V ground for internal use

Note: * The VCCQ, VSSQ, VDD, and VSS power pins are used for the power supply to the core and the I/O of the entire LSI excluding the USB module.

23.6.2 USB Transmission Lines

USB transmission lines indicate a wiring pattern that connects the USB connector and the USB transceiver.

USB 2.0 includes the following three communication modes: Hi-Speed, Full-Speed, and Low-Speed. Of these, Hi-Speed provides the communication speed of 480 Mbps, so it is necessary to design the USB transmission lines as a high-frequency circuit. Impedance control is required for the USB transmission lines.

The following are notes on designing a wiring pattern of the USB transmission lines:

- The characteristic impedance required for USB Hi-Speed transmission lines is a differential impedance of 90 Ω ±15%.
- The pattern width/interval for impedance control varies depending on the thickness, material, and layer structure of the substrate. For details, consult the board manufacturer.
- The wiring pattern length of the USB transmission lines from the LSI to the USB connector must be determined so that the delay time does not exceed its maximum specified in the USB standard. Table 23.4 shows the recommended values for designing a wiring pattern of the USB transmission lines on a printed circuit board made of common material.

Table 23.4 Recommended Values for Designing USB Transmission Line

	Maximum Delay Time (USB standard)	Wiring Length	Difference between Lengths of D+ and D-
Host Controller	3 ns	150 mm or less	2.5 mm or less
Function Controller	1 ns	50 mm or less	2.5 mm or less

- The layer below USB transmission lines must be a solid ground plane. 2-mm or more interval is required between the edge of the solid ground plane and the USB transmission lines.
- Do not place other signal lines near the USB transmission lines. Particularly, those for considerably fluctuating signals, such as clock and data bus lines, must be placed far from the USB transmission lines. Also, the USB transmission lines must not be intersected by other signal lines.
- It is recommended that in the same layer as the USB transmission lines (surface layer), grounded guard rings be applied about 2 mm outward from them.
- Arrange the USB transmission lines in the same layer without using through-hole vias. Also, arrange the USB transmission lines so that they do not branch.
- The interval between the USB transmission lines must be constant.

- The USB transmission lines must be kept away from the oscillator, the power supply circuit, and other I/O connectors.
- Arrange the USB transmission lines as straight as possible. If necessary, bend them at an angle of 135° or curve them gently. Do not bend them at an acute angle of about 90°.
- It is recommended that grounded guard rings be applied to signals for clock, reset, read, write, and chip select.

Figures 23.2 and 23.3 show USB transmission line pattern design examples for host and function controllers, respectively.

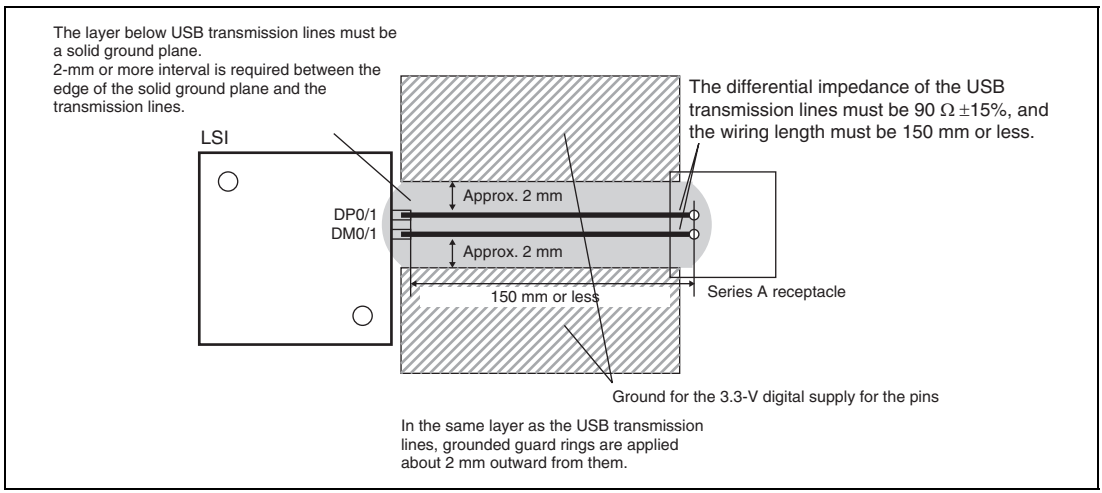


Figure 23.2 USB Transmission Line Pattern Design Example for Host Controller

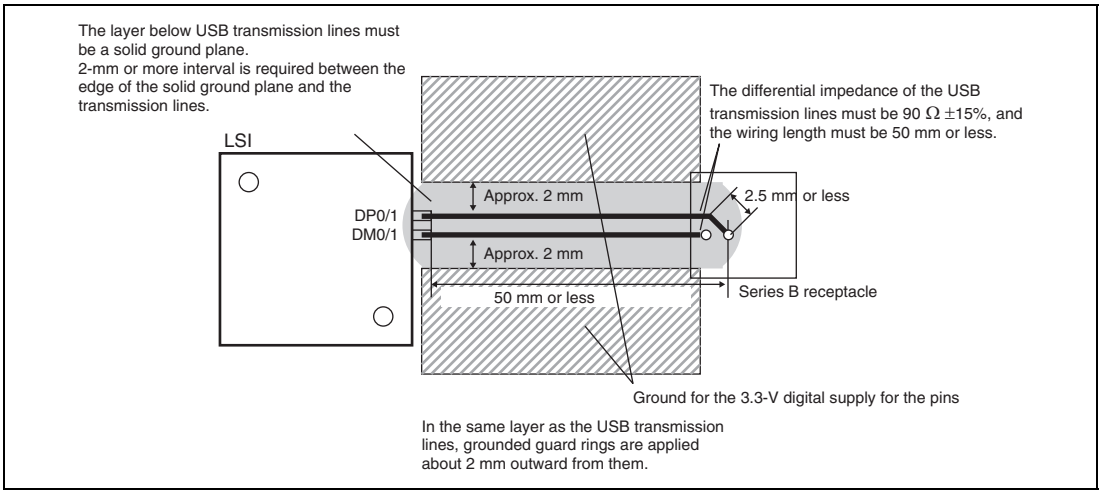


Figure 23.3 USB Transmission Line Pattern Design Example for Function Controller

23.6.3 Power Supply and Ground Pattern

Given below are notes on designing power supply and ground pattern.

- Power supply and ground must be separated into digital and analog. Tables 23.5 and 23.6 show the classification of power supply and ground, respectively.

Table 23.5 USB Power Supply Classification

	Power Supply Classification			
Pin Name	**Analog Power Supply (1.2 V)**	**Digital Power Supply (1.2 V)**	**Analog Power Supply (3.3 V)**	**Digital Power Supply (3.3 V)**
AV12	√			
AV33			√	
VDD		√		
VCCQ				√

Note: √: Indicates the power supply to be used.

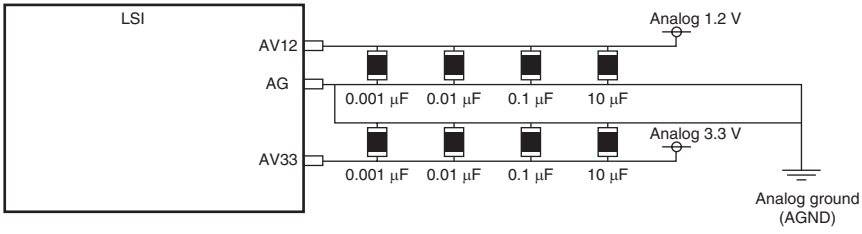
Table 23.6 USB Ground Classification

	Ground Classification	
Pin Name/USB Connector	**Analog Ground (AGND)**	**Digital Ground (DGND)**
AG	√	
VSS		√
VSSQ		√
USB connector ground (Including frame ground)		√

Note: √: Indicates the ground to be used.

- On the board, low-impedance connection must be made to the digital power supply/ground near USB signals.
- The pattern must be designed so that the layer of power supply and ground is as wide as possible.
- Tantalum solid electrolytic capacitors or ceramic capacitors having excellent high-frequency characteristics are recommended for power supply.
- Aluminum electrolytic capacitors affect the jitter value during the measurement of EYE pattern. Therefore, use the capacitors after careful design and tests.

- It is recommended that decoupling capacitors having the capacitances of 0.001, 0.01, 0.1, and 10 µF be placed near the USB power pins. Figure 23.4 gives an example of arranging decoupling capacitors.



* Place the capacitors near the USB power pins in ascending order of capacity.

Figure 23.4 Decoupling Capacitor Arrangement Example

23.6.4 Oscillation Circuit

Notes on designing an oscillation circuit are given below.

- The oscillation circuit must be placed near the USB clock input pin USB_EXTAL. It is recommended to apply a grounded guard ring to USB_EXTAL.
- The frequencies of all necessary oscillation devices must be 48 MHz ± 100 ppm.
- When using a crystal resonator, consult the oscillator manufacturer for the circuit parameters.

Figures 23.5 and 23.6 show examples of connecting a crystal resonator and an oscillator, respectively.

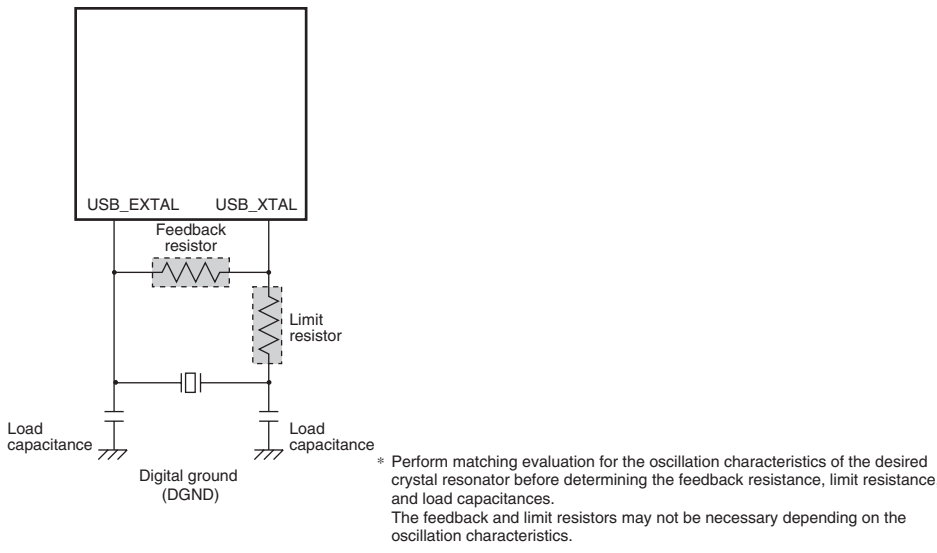


Figure 23.5 Example of Connecting Crystal Resonator

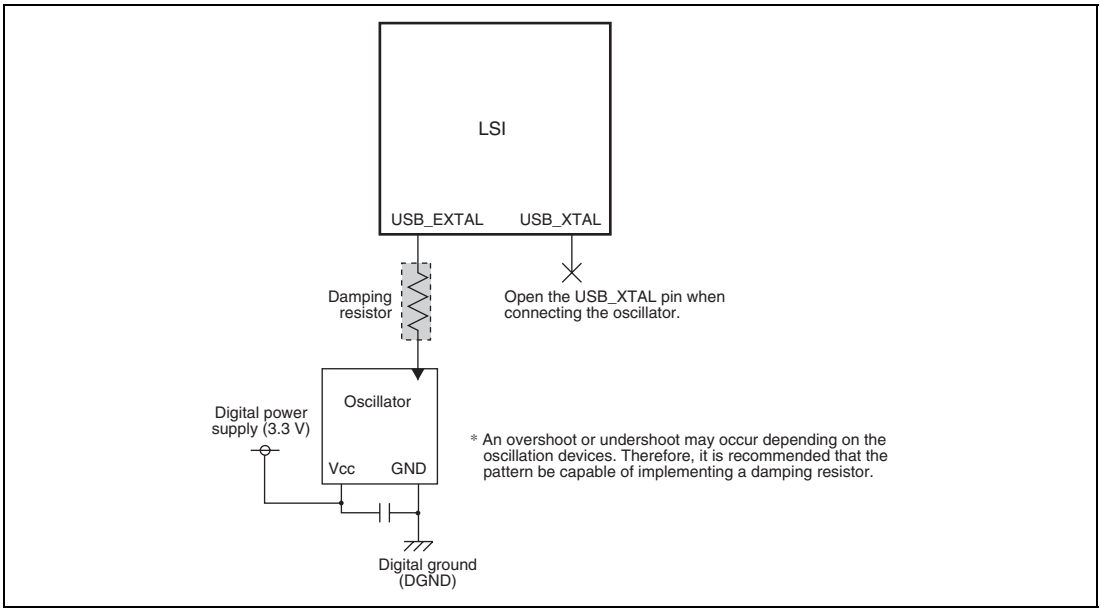


Figure 23.6 Example of Connecting Oscillator

23.6.5 VBUS Power Supply Circuit

Notes on designing a VBUS power supply circuit are given below.

- To use the USB module as a host controller, the circuit must be designed so that the load capacitance of the VBUS line is 120 µF or more.
- To use the USB module as a function controller, the circuit must be designed so that the load capacitance of the VBUS line ranges from 1.0 to 10 µF.
- It is possible that a VBUS line overshoot could occur due to impedance mismatching when a USB cable is connected. Therefore, a filter circuit must be implemented. Use a capacitor of 1.0 to 10 µF and a resistor of 100 Ω to 1 KΩ to make a filter circuit. Determine their resistance and capacitance by checking whether an overshoot occurs on the board. Note that a resistor of 1 KΩ or higher must not be used.
- In order to use the USB module as a host controller, it is necessary to supply VBUS power to USB function devices. In order to control the VBUS power supply, it is recommended to use a power switch IC with an overcurrent limiter for USB power supply bus (hereafter referred to as USB power switch IC).
  The current limit for the VBUS power line must be determined based on the current necessary for the target system and the USB function devices to be communicated with. Also, design the VBUS power supply control circuit by referring to circuit examples described in the data sheet of the USB power switch IC to be used.
- In order to use the USB module as a host controller while the LSI power supply is off, it is necessary to turn off the input to the OVCn/VBUSn (n = 0, 1) pin used for the host controller. When the USB module is used as a function controller, the VBUS voltage from the USB host can be applied to the OVCn/VBUSn (n = 0, 1) pin used for the function controller, even if the LSI power supply is off.

Figures 23.7, 23.8, and 23.9 show examples of the VBUS power supply circuit for host controller use. Also, figure 23.10 shows an example of that for function controller use.

Since the USB_OVC0/1 pin is 3.3-V I/O in the host controller examples (figures 23.7 and 23.8), 0 and 3.3 V must be input at low and high levels, respectively.

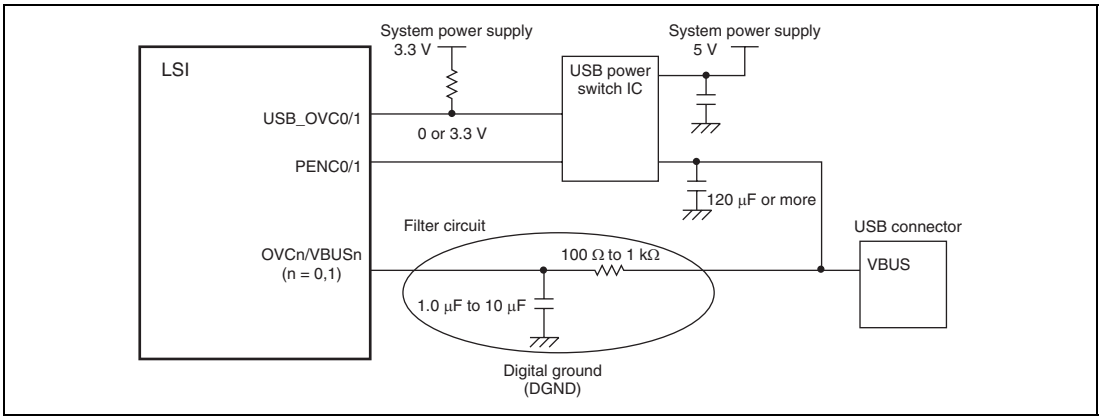


Figure 23.7 Host Controller VBUS Circuit Example (OVCn/VBUSn (n = 0, 1) and USB_OVC0/1 pins are used individually [1])

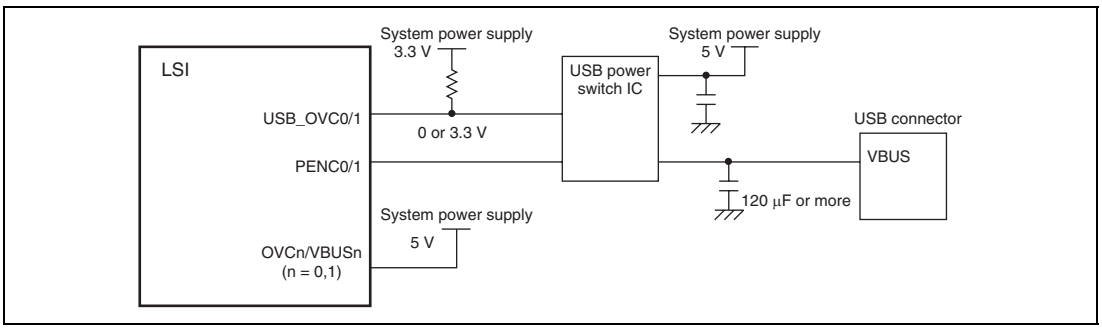


Figure 23.8 Host Controller VBUS Circuit Example (OVCn/VBUSn (n = 0, 1) and USB_OVC0/1 pins are used individually [2])

Since the USB_OVC0/1 pin is 5-V I/O in the host controller example (figure 23.9), 0 and 5 V must be input at low and high levels, respectively. Also, OVC detection signal must be input through 5-V standard logic or such devices because the VBUS pin incorporates a pull-down resistor.

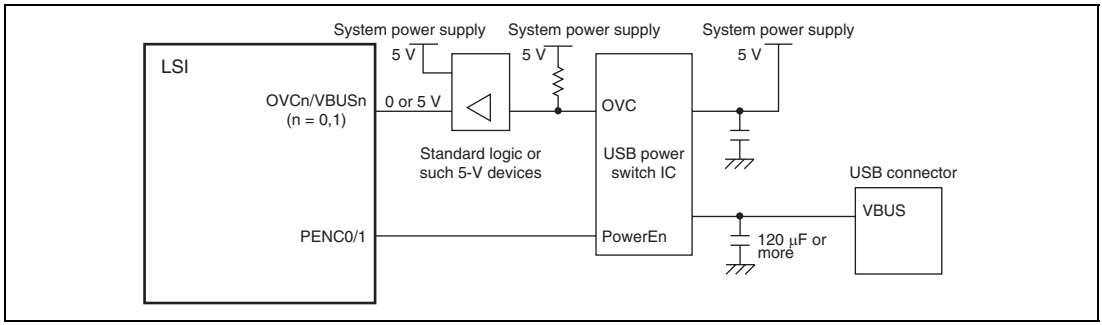


Figure 23.9 Host Controller VBUS Circuit Example (OVCn/VBUSn (n = 0, 1) pin is used as OVC function)

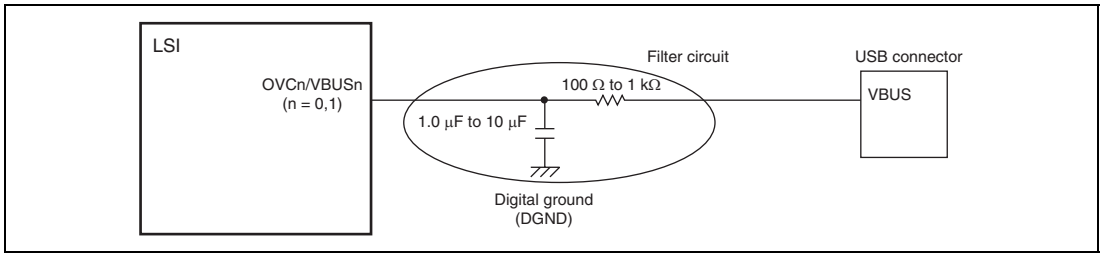


Figure 23.10 Function Controller VBUS Circuit Example

23.6.6 REFRIN Pin

Notes on designing a REFRIN pin peripheral circuit are given below.

- A resistor of 5.6 kΩ ± 1% (hereafter referred to as reference resistor) must be connected between the REFRIN pin and AG.
- The reference resistor must be placed as near as possible to the LSI.
- The REFRIN pin, the reference resistor, and AG must be connected with a wide pattern and the shortest route.
- The reference resistor and AG must be connected with a dedicated pattern leading into analog ground. The pattern must be designed so that no impedance is shared with other signals.
- In order to prevent crosstalk, considerably fluctuating signals (e.g., DP0/1, DM0/1, clock, and address data control signals) must not be transmitted across or along areas near the reference resistor and its pattern. It is recommended to apply a grounded guard ring to the reference resistor and its pattern.

Figure 23.11 shows a connection diagram and a pattern design example for the REFRIN pin.

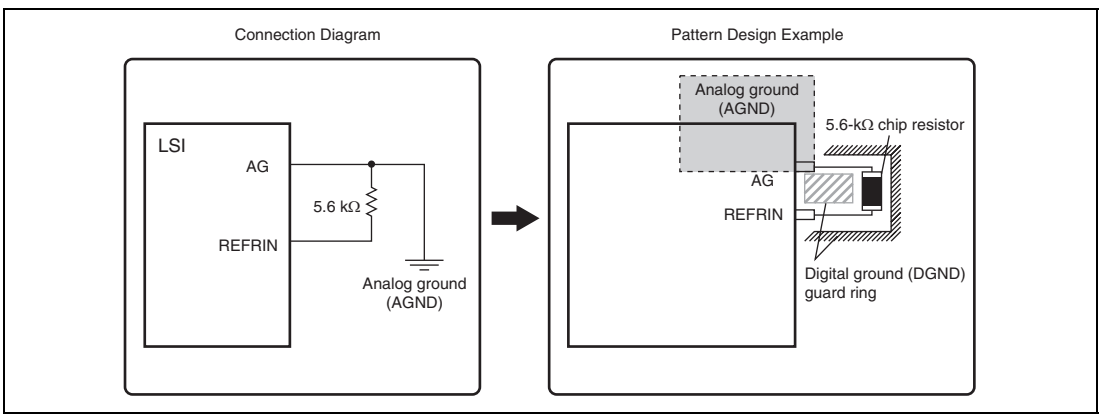


Figure 23.11 Connection Diagram and Pattern Design Example for REFRIN Pin

23.6.7 EMI/ESD Countermeasures

Notes on taking countermeasures for EMI and ESD are given below.

- In order to implement EMI/ESD protection devices such as coils and diodes into the USB transmission lines, they must be placed near the lines, and their wiring lengths must be as short as possible.
- The desired EMI/ESD protection devices must support USB 2.0. Implementing such devices may cause impedance mismatching on the USB transmission lines, resulting in disturbance to waveform. Therefore, fully evaluate them to make a selection.

Figure 23.12 shows a connection diagram for EMI/ESD protection devices.

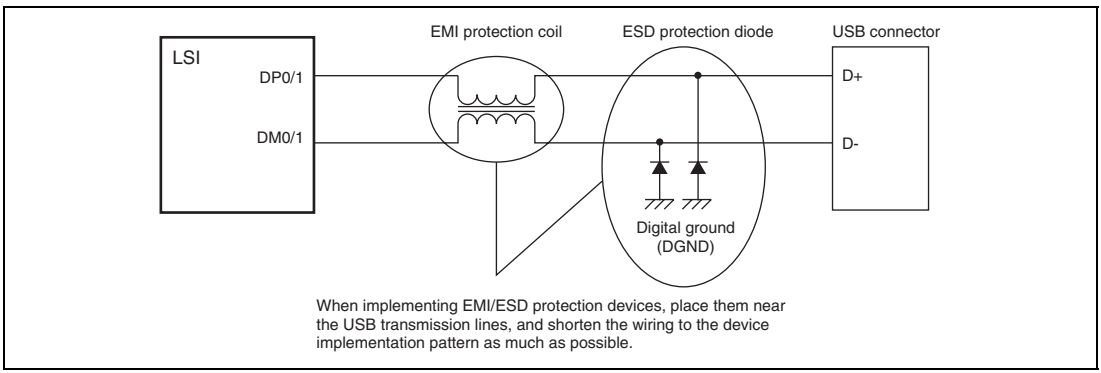


Figure 23.12 Example of Connecting EMI/ESD Protection Devices

23.7 Usage Note

23.7.1 Point for Caution in Function Controller Operation

Establishing a connection when the USB module is operating as a function controller may not be possible if the cable is longer than 1.5 m. Accordingly, use USB cables that are no longer than 1.5 m.

Section 23A USB2.0 Host Controller

This document describes the EHCI standard. Refer to this document when developing USB host systems. For details on the EHCI standard, see the "Enhanced Host Controller Interface Specification for Universal Serial Bus Revision 1.0."

23A.1 Register Descriptions

The registers can be accessed only by the CPU. When they are accessed by other modules, operation is not guaranteed. Bit widths of these resisters are all 32 bits, so should be accessed in longword (32 bits) units. Access in other units is not supported.

Table 23A.1 List of Register

- Host Controller Capability Registers

The address is obtained by H'FFE70000 + offset.

Register name	R/W	Offset	Access Size
HCIVERSION/CAPLENGTH	R	0	32
HCSPARAMS	R	4	32
HCCPARAMS	R	8	32
HCSP-PORTROUTE	R	C	32

- Host Controller Operational Registers

The address is obtained by H'FFE70010 + offset.

Register name	R/W	Offset	Access Size
USBCMD	R/W	0	32
USBSTS	R/W	4	32
USBINTR	R/W	8	32
FRINDEX	R/W	C	32
CTRLDSSEGMENT	R/W	10	32
PERIODICLISTBASE	R/W	14	32
ASYNCLISTADDR	R/W	18	32
CONFIGFLAG	R/W	40	32
PORTSC (1 to N_PORT)	R/W	44	32

Note: Addresses other than the above must not be write-accessed to. If written to, operation is not guaranteed. If read, an undefined value is read.

Table 23A.2 Register State in Each Operating Mode

Register Address	Register Name	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
H'FFE7 0000	HCIVERSION/CAPLENGTH	H'0100 xx10	H'0100 xx10	Retained	Retained	Retained	Initialized
H'FFE7 0004	HCSPARAMS	H'xx00 1212	H'xx00 1212	Retained	Retained	Retained	Initialized
H'FFE7 0008	HCCPARAMS	H'xxxx A016	H'xxxx A016	Retained	Retained	Retained	Initialized
H'FFE7 000C	HCSP-PORTROUTE	H'0000 0000	H'0000 0000	Retained	Retained	Retained	Initialized
H'FFE7 0010	USBCMD	H'xx08 xB00	H'xx08 xB00	Retained	Retained	Retained	Initialized
H'FFE7 0014	USBSTS	H'xxxx 1x00	H'xxxx 1x00	Retained	Retained	Retained	Initialized
H'FFE7 0018	USBINTR	H'xxxx xx00	H'xxxx xx00	Retained	Retained	Retained	Initialized
H'FFE7 001C	FRINDEX	H'xxxx 0000	H'xxxx 0000	Retained	Retained	Retained	Initialized
H'FFE7 0020	CTRLDSSEGMENT	H'0000 0000	H'0000 0000	Retained	Retained	Retained	Initialized
H'FFE7 0024	PERIODICLISTBASE	H'0000 0xxx	H'0000 0xxx	Retained	Retained	Retained	Initialized
H'FFE7 0028	ASYNCLISTADDR	H'0000 00xx	H'0000 00xx	Retained	Retained	Retained	Initialized
H'FFE7 0050	CONFIGFLAG	H'xxxx xxx0	H'xxxx xxx0	Retained	Retained	Retained	Initialized
H'FFE7 0054	PORTSC (1 to N_PORT)	H'xx00 2000	H'xx00 2000	Retained	Retained	Retained	Initialized

23A.1.1 HCIVERSION Register

HCIVERSION is a register that indicates the version of the EHCI standard supported by the host controller in the BCD format.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	HCIVERSION															
Initial value:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—								CAPLENGTH							
Initial value:	—	—	—	—	—	—	—	—	0	0	0	1	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	HCIVERSION	H'0100	R	Host Controller Interface Version Number These bits indicate the EHCI standard version supported by the host controller (BCD format).
15 to 8	—	Undefined	R	Reserved These bits are read as undefined. The write value should always be 0.
7 to 0	CAPLENGTH	H'10	R	Capability Register Length These bits indicate the size of the capability register area. Since the operational register area is located immediately after the capability register area, the location of the operational register area can be known by using this register value as an offset.

23A.1.2 HCSPARAMS Register

HCSPARAMS is a register that indicates host controller structure parameters, such as the number of ports.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	DPN				—	—	—	P_INDI
Initial value:	—	—	—	—	—	—	—	—	0	0	0	0	—	—	—	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	N_CC				N_PCC				PRR	—	—	PCC	N_PORTS			
Initial value:	0	0	0	1	0	0	1	0	0	—	—	1	0	0	1	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	Undefined	R	Reserved These bits are read as undefined. The write value should always be 0.
23 to 20	DPN	H'0	R	Debug Port Number (Optional) These bits indicate the DebugPort number in N_PORTS ports. 0: There is no DebugPort. Other than 0: There is at least one DebugPort. The field value indicates the port number. A number greater than N_PORTS cannot be set in this field.
19 to 17	—	Undefined	R	Reserved These bits are read as undefined. The write value should always be 0.
16	P_INDI	0	R	Port Indicator (P_INDICATOR) Indicates whether or not the port supports the port indicator control function. 0: The port indicator control function is not supported. 1: The port indicator control function is supported. The port indicator can be controlled by the Port Indicator Control field in the register PORTSC.

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	N_CC	0001	R	Number of Companion Controller (N_CC) These bits indicate the number of the mounted USB1.1 companion host controllers. 0: There is no companion host controller. Only the EHCI host controller is mounted. Other than 0: Companion host controllers are mounted. The field value indicates the number of companion host controllers.
11 to 8	N_PCC	0010	R	Number of Ports per Companion Controller (N_PCC) These bits indicate the number of ports used for one companion host controller.
7	PRR	0	R	Port Routing Rule Specifies the routing rules that define to which companion host controller a given port is to be assigned. 0: Assigns N_PCC ports each in ascending order of cHC numbers. If N_PORT = 8, N_CC = 2, and N_PCC = 4, assigns ports No.1 to No.4 to the first companion host controller, and ports No.5 to No.8 to the second companion host controller. 1: Defines assignment rules in HCSP-PORTROUTE, and assigns the ports accordingly.
6, 5	—	Undefined	R	Reserved These bits are read as undefined. The write value should always be 0.
4	PPC	1	R	Port Power Control (PPC) Indicates whether the port power supply can be switched on and off. 1: Port power supply can be switched on and off. 0: Port power supply cannot be switched on and off. When this bit is 1, the Port Power bit in the register PORTSC can be set.

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	N_PORTS	0010	R	N_PORTS These bits indicate the number of downstream ports of the host controller. The number of port registers (PORTSC) in the operational register area is determined by this value. The value can be set in the range from H'1 to H'F.

23A.1.3 HCCPARAMS Register

HCCPARAMS is a register that indicates parameters relating to the host controller capabilities.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EECP								IST				—	ASPC	PFLF	64AC
Initial value:	1	0	1	0	0	0	0	0	0	0	0	1	—	1	1	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	Undefined	R	Reserved These bits are read as undefined. The write value should always be 0.
15 to 8	EECP	H'A0	R	EHCI Extended Capabilities Pointer (EECP) To add capability information, an additional capability area can be provided in the PCI configuration area. This field indicates an offset for the added capability area.

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	IST	0001	R	Isochronous Scheduling Threshold These bits indicate cache mode for isochronous schedule data. When 7th bit (bit 7) is 0: Microframe cache mode. Field[6:4] specifies mframe to be cached. When 7th bit (bit 7) is 1: Frame cache mode. The entire frame is cached. When IST = 1000, the entire frame is cached. Therefore, the data structure corresponding to the current frame cannot be modified by software. Data structures of the next and the following microframes can be modified. When IST = 0010, two microframes are cached. Therefore, data structures of the 3rd and the following microframes can be modified. When IST = 0000, data structures are not cached; data structures are acquired at each microframe. Therefore, data structures of the next and the following microframes can be modified.
3	—	Undefined	R	Reserved This bit is read as undefined. The write value should always be 0.
2	ASPC	1	R	Asynchronous Schedule Park Capability Specifies whether or not the park function is supported for queue heads for HS in asynchronous schedules. 0: Park function is not supported. 1: Park function is supported. Park function can be enabled using Asynchronous Schedule Park Mode Enable and Asynchronous Schedule Park Mode Count in the register USBCMD.

Bit	Bit Name	Initial Value	R/W	Description
1	PFLF	1	R	Programmable Frame List Flag Specifies whether or not the size of the frame list can be changed. 0: The number of frame list elements is fixed at 1024. Frame List Size in USBCMD is read-only and is set to 0. 1: The number of frame list elements can be changed between 512 and 256 using Frame List Size in USBCMD.
0	64AC	0	R	64-Bit Addressing Capability Specifies addressing mode for accessing data structures. 0: 32-bit addressing 1: 64-bit addressing

23A.1.4 HCSP-PORTROUTE Register

HCSP-PORTROUTE is a register that specifies to which companion host controller a DownStream port is to be assigned.

This register is valid only when the Port Routing Rules field in HCSPARAMS is set to 1.

Bit:	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48
	—	—	—	—	HCSP-PORTROUTE											
Initial value:	—	—	—	—	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
	HCSP-PORTROUTE															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	HCSP-PORTROUTE															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HCSP-PORTROUTE															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
63 to 60	—	Undefined	R	Reserved These bits are read as undefined. The write value should always be 0.
59 to 0	HCSP-PORTROUTE	All 0	R	HCSP-PORTROUTE Specifies the correspondence between N_PORTS ports and companion host controllers. The number for a companion host controller is specified in 4 bits. There is a maximum of 15 N_PORTS, and the size of this register is 15 × 4 = 60 bits. For example, if ports are numbered 0h 1h 0h 1h ..., the corresponding companion host controllers are numbered 1, 2, 1, 2 ...

23A.1.5 USBCMD Register

USBCMD controls the ON/OFF of the host controller, host controller resets, and ON/OFF of schedule processing.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	ITC							
Initial value:	—	—	—	—	—	—	—	—	0	0	0	0	1	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	ASPME	—	ASPMC		LHCR	IAAD	ASE	PSE	FLS		HCR	RS
Initial value:	—	—	—	—	1	—	1	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	Undefined	R	Reserved These bits are read as undefined. The write value should always be 0.
23 to 16	ITC	H'08	R/W	Interrupt Threshold Control These bits specify the frequency (maximum interval) of hardware interrupts generated by the host controller. H'01: 1 microframe H'02: 2 microframes H'04: 4 microframes H'08: 8 microframes (initial value, 1 ms) H'10: 16 microframes (2 ms) H'20: 32 microframes (4 ms) H'40: 64 microframes (8 ms) This field must not be set when the HC Halted bit in USBSTS is 0.
15 to 12	—	Undefined	R	Reserved These bits are read as undefined. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
11	ASPME	1	R/W or R	Asynchronous Schedule Park Mode Enable (Optional) Enables park mode for asynchronous schedules. 0: Park mode is disabled. 1: Park mode is enabled. This field is fixed at 0 and can only be read unless the Asynchronous Schedule Park Capability bit in the register HCCPARAMS is set to 1. Conversely, if Asynchronous Schedule Park Capability is set to 1, the initial value of this bit is 1, and the bit can be read and written.
10	—	Undefined	R	Reserved This bit is read as undefined. The write value should always be 0.
9, 8	ASPMC	11	R/W or R	Asynchronous Schedule Park Mode Count (Optional) These bits specify the number of transactions that can be issued in a single operation from one QH in the asynchronous schedule. For example, if a transaction count of 3 is specified, bus transactions are executed 3 times from the fetched QH, and then the next QH is fetched. If Asynchronous Schedule Park Mode is set to 1, the value 0 must not be assigned to this field. This field is fixed at 0 and can only be read unless the Asynchronous Schedule Park Capability bit in the register HCCPARAMS is set to 1. Conversely, if Asynchronous Schedule Park Capability is set to 1, the initial value of this bit is 3, and the bit can be read and written.
7	LHCR	0	R/W	Light Host Controller Reset (Optional) Resets the EHCI host controller without affecting the status of the port or port ownership relations. In other words, this bit initializes all the registers except the PORTSC or CF register. By assigning the value 1 to this bit, the Light Host Controller Reset command can be executed. If this bit is read and it is 0, it can be concluded that the resetting is complete. If the bit is 1, the resetting operation is still in progress.

Bit	Bit Name	Initial Value	R/W	Description
6	IAAD	0	R/W	Interrupt on Async Advance Doorbell Enables System Software to request the host controller to generate an interrupt when the asynchronous schedule processing is executed. When this bit is set, the host controller clears any cached asynchronous schedule data, and sets the Interrupt on Async Advance bit in the register USBSTS. If the Interrupt on Async Advance Enable bit in USBINTR is set, the host controller generates an interrupt on the next interrupt threshold.
5	ASE	0	R/W	Asynchronous Schedule Enable Specifies that the host controller executes or skips asynchronous schedules. 0: Asynchronous schedules are not executed. 1: Asynchronous schedules are executed using the register ASYNCLISTADDR.
4	PSE	0	R/W	Periodic Schedule Enable Specifies that the host controller executes or skips periodic schedules. 0: Periodic schedules are not executed. 1: Periodic schedules are executed using the register PERIODICLISTBASE.
3, 2	FLS	00	R/W	Frame List Size These bits specify the frame list size. Depending on the frame list size specified with this field, which bit in the register Frame List Index is used for the Frame List Current Index. These bits can be written only when Programmable Frame List Flag in HCCPARAMS is set. 00: 1024 elements (4096 bytes) 01: 512 elements (2048 bytes) 10: 256 elements (1024 bytes)

Bit	Bit Name	Initial Value	R/W	Description
1	HCR	0	R/W	Host Controller Reset (HCRESET) Resets the host controller. If the host controller is reset by this bit, the RootHub register operates in the same manner as when the hardware reset is applied to the chip. When this bit is set, the host controller resets the pipeline, timer, counter, and state machine in the host controller, and sets the initial value. In addition, the host controller immediately terminates any transfer operation that is being conducted at the time. This resetting is not driven to any of the downstream ports. Although the resetting does not initialize the registers in the PCI configuration register area, all registers in the operational register area, including the Port register and the State Machine of the Port, are reset to their initial value. The ownership of the port is returned to the companion host controller. For this reason, after resetting the software must re-initialize the host controller in order to reset the host controller to the operating condition. The host controller clears this bit (i.e., if the bit is set to 1, resets it to 0) upon completion of the resetting operation. During the resetting process, software cannot clear this bit and interrupt the resetting process. Further, software cannot set the HCRESET when the Halted bit in the register USBSTS is 0. In other words, USBRESET must not be performed when the host controller is in the execution status.

Bit	Bit Name	Initial Value	R/W	Description
0	RS	0	R/W	Run/Stop Controls the ON/OFF action of the entire host controller. 1: While the bit is 1, the host controller continues to execute the schedule. 0: When the bit is set to 0, the host controller terminates any communication currently in progress, and stops the operation. After the software has cleared this bit, the host controller must halt its operation within 16 microframes. The Halted bit in USBSTS can be used to verify that the host controller has terminated any transfer being executed by the host controller and transitioned to the Stop state. When the host controller is in the Halt state (USBSTS/Halted = 1), the software must not set the Run/Stop bit to 1.

23A.1.6 USBSTS Register

USBSTS indicates status information such as interrupt status and schedule status.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ASS	PSS	R	HCH	—	—	—	—	—	—	IAA	HSE	FLR	PCD	UEI	UI
Initial value:	0	0	0	1	—	—	—	—	—	—	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	Undefined	R	Reserved These bits are read as undefined. The write value should always be 0.
15	ASS	0	R	Asynchronous Schedule Status Indicates the current (actual) status of asynchronous schedules. 1: Asynchronous schedules are enabled. 0: Asynchronous schedules are disabled. It is not necessary for the host controller to enable/disable asynchronous schedules immediately after the software sets/clears the Asynchronous Schedule Enable bit in USBCMD to enable/disable asynchronous schedules (it is not necessary to reflect the software request immediately). When this bit matches the Asynchronous Schedule Enable bit in USBCMD, asynchronous schedules are recognized to be enabled or disabled.

Bit	Bit Name	Initial Value	R/W	Description
14	PSS	0	R	Periodic Schedule Status Indicates the current (actual) status of periodic schedules. 1: Periodic schedules are enabled. 0: Periodic schedules are disabled. It is not necessary for the host controller to enable/disable periodic schedules immediately after the software sets/clears the Periodic Schedule Enable bit in USBCMD to enable/disable periodic schedules (it is not necessary to reflect the software request immediately). When this bit matches the Periodic Schedule Enable bit in USBCMD, periodic schedules are recognized to be enabled or disabled.
13	R	0	R	Reclamation This bit is set when asynchronous schedule empty state is detected.
12	HCH	1	R	HC Halted Indicates the host controller status. 0: The Run/Stop bit in USBCMD is set to 1. 1: The Run/Stop bit is cleared to 0 and the host controller is stopped. The Run/Stop bit is cleared by software or hardware (for example, by Internal Error).
11 to 6	—	Undefined	R	Reserved These bits are read as undefined. The write value should always be 0.
5	IAA	0	R/W*	Interrupt Async Advance By setting the Interrupt On Async Advance Doorbell bit in USBCMD, the software can force the host controller to generate an interrupt when executing the asynchronous schedule. Thus, this bit indicates that the host controller has executed the asynchronous schedule. This bit, being a status bit, indicates the occurrence of an interrupt due to this interrupt source. This interrupt is used to delete queue heads from asynchronous schedules.

Bit	Bit Name	Initial Value	R/W	Description
4	HSE	0	R/W*	Host System Error The host controller sets this bit if a severe error occurs during access by the host system involving the use of the host controller module. In the PCI system, this bit is set in conditions including "PCI parity error", "PCI master abort", and "PCI target abort". If this error occurs, the host controller sets the Run/Stop bit to 0 to prevent any scheduled TD from being executed.
3	FLR	0	R/W*	Frame List Rollover The host controller sets this bit if the Frame List Index (FRINDEX) register rolls over from its maximum value to 0. At what value a RollOver occurs depends on the Frame List Size of USBCMD. If the Frame List Size is set to 1024, a RollOver occurs each time FRINDEX[13] toggles. Similarly, if the Frame List Size is 512, a RollOver occurs each time FRINDEX[12] toggles.
2	PCD	0	R/W*	Port Change Detect This bit is set by the host controller in the following cases: • When the Port Owner bit is changed from 0 to 1 in a port. • When a transition from J to K is detected in a suspended port and the Force Resume Transition bit is changed from 0 to 1. • When the Port Owner bit is written to 1 by software and the port ownership is released.
1	UEI	0	R/W*	USB Error interrupt This bit is set by the host controller when a USB transfer ends with an error (when the error counter underflows). When a transfer ends with an error at the TD with ioc = 1, both the UEI and USBINT bits are set.

Bit	Bit Name	Initial Value	R/W	Description
0	UI	0	R/W*	USB Interrupt This bit is set by the host controller if the USB transfer ends when ioc = 1 and the TD retires. This bit is also set when a short packet is received (when the size of received data is smaller than expected.)

Note: * Only 1 can be written to clear these bits. Writing 0 is ignored.

23A.1.7 USBINTR Register

USBINTR specifies the on/off of hardware interrupts.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	IAAE	HSEE	FLRE	PCDE	UEIE	UIE
Initial value:	—	—	—	—	—	—	—	—	—	—	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	Undefined	R	Reserved These bits are read as undefined. The write value should always be 0.
5	IAAE	0	R/W	Interrupt on Async Advance Enable The host controller generates hardware interrupts when the Interrupt on Async Advance bit in USBSTS is set while this bit is 1.
4	HSEE	0	R/W	Host System Error Enable The host controller generates hardware interrupts when the Host System Error Status bit in USBSTS is set while this bit is 1.
3	FLRE	0	R/W	Frame List Rollover Enable The host controller generates hardware interrupts when the Frame List Rollover bit in USBSTS is set while this bit is 1.

Bit	Bit Name	Initial Value	R/W	Description
2	PCDE	0	R/W	Port Change Detect Enable The host controller generates hardware interrupts when the Port Change Detect bit in USBSTS is set while this bit is 1.
1	UEIE	0	R/W	USB Error Interrupt Enable The host controller generates hardware interrupts when the USBERRINT bit in USBSTS is set while this bit is 1.
0	UIE	0	R/W	USB Interrupt Enable The host controller generates hardware interrupts when the USBINT bit in USBSTS is set while this bit is 1.

23A.1.8 FRINDEX Register

FRINDEX indicates the current frame number which is used when the host controller references the Periodic Frame List. The register value is updated every 125 µs.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	FI													
Initial value:	—	—	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	Undefined	R	Reserved These bits are read as undefined. The write value should always be 0.
13 to 0	FI	H'0000	R/W	Frame Index This value is incremented at every end of microframe. Bits[N:3] are used for the Current Index of the framelist. In other words, accesses to framelist are performed every 8 microframes (1 frame).

Frame List Size (USBCMD)	Number of Elements	N	Actual Size of Framelist
00b	1024	12	2^{12} = 4096
01b	512	11	2^{11} = 2048
10b	256	10	2^{10} = 1024

23A.1.9 CTRLDSSEGMENT Register

CTRLDSSEGMENT indicates the upper 32 bits [63:32] when the host controller accesses the 64-bit data structures.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CTRLDSSEGMENT															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CTRLDSSEGMENT															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CTRLDSSEGMENT	H'0000 0000	R/W	Control Data Structure Data Segment These bits specify the upper 32 bits (bits 63 to 32) when the data structures of EHCI are to be accessed in 64 bits. This field is disabled if the 64-bit Addressing Capability field in HCCPARAMS is 0. If it is 1, access can be made to the data structures of EHCI. The data structures must reside within the same 4-Gbyte boundary.

23A.1.10 PERIODICLISTBASE Register

PERIODICLISTBASE specifies the base address for a periodic framelist.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BA															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BA				—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	BA	H'0 0000	R/W	Base Address Specifies the start address of the periodic framelist placed on memory. This register value is loaded by software before the periodic schedule is executed by the host controller. The host controller can execute the periodic framelists in sequence using this register and FRINDEX.
11 to 0	—	Undefined	R	Reserved These bits are read as undefined. The write value should always be 0.

23A.1.11 ASYNCLISTADDR Register

ASYNCLISTADDR specifies the pointer to queue heads in asynchronous schedules.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LPL															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LPL											—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	LPL	H'000 0000	R/W	Link Pointer Low Specifies the queue head for the next execution in asynchronous schedules. Since a queue head is allocated on the 32-bit boundary, only the upper 27 bits are specified.
4 to 0	—	Undefined	R	Reserved These bits are read as undefined. The write value should always be 0.

23A.1.12 CONFIGFLAG Register

CONFIGFLAG specifies the ownership for all ports.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CF
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	Undefined	R	Reserved These bits are read as undefined. The write value should always be 0.
0	CF	0	R/W	Config Flag Specifies the routing rule for all ports. 0: Each port is routed to the corresponding cHC. 1: All ports are routed to eHC.

23A.1.13 PORTSC (1 to N_PORT) Register

PORTSC controls ports and monitors port status.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	WOE	WDE	WCE	PTC			
Initial value:	—	—	—	—	—	—	—	—	—	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIC		PO	PP	LS		—	PR	S	FPR	OC	OA	PEDC	PED	CSC	CCS
Initial value:	0	0	1	0	0	0	—	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W*	R	R/W*	R/W*	R/W*	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	Undefined	R	Reserved These bits are read as undefined. The write value should always be 0.
22	WOE	0	R/W	Wake on Over-Current Enable (WKOC_E) Enables detecting over-current for ports as a wakeup event to make transition to the resume from the suspend state.
21	WDE	0	R/W	Wake on Disconnect Enable (WKDSCNNT_E) Enables detecting device disconnection as a wakeup event to make transition to the resume from the suspend state.
20	WCE	0	R/W	Wake on Connect Enable (WKCNNT_E) Enables detecting device connection as a wakeup event to make transition to the resume from the suspend state.
19 to 16	PTC	0000	R/W	Port Test Control These bits control port test mode. 0000: Test mode is invalid. 0001: Test J_state 0010: Test K_state 0011: Test SE0_NAK 0100: Test packet 0101: Test Force_Enable Other than above: Setting inhibited.

Bit	Bit Name	Initial Value	R/W	Description
15, 14	PIC	00	R/W	Port Indicator Control These bits control the port indicator. 00: The port indicator function is turned off. 01: Amber 10: Greed This field is invalid when HCSPARAMS/P_INDICATOR is 0.
13	PO	1	R/W	Port Owner Controls the port ownership. 1: The companion host controller has the ownership of this port. 0: The EHCI host controller has the ownership of this port. When the connected device is not a high-speed device, this bit is set to 1 and the ownership is released by software. When the Configured bit in the register Configuration Flag is changed from 0 to 1, this bit is unconditionally cleared to 0. Conversely, when the Configured bit is changed from 1 to 0, this field is unconditionally set to 1.

Bit	Bit Name	Initial Value	R/W	Description
12	PP	0	R/W	Port Power Controls the port power. The meaning of this bit differs depending on the PPC bit in HCSPARAMS.

PPC	PP
0	1 (read-only) The port power is always turned on and this bit is always 1.
1	0 or 1 (readable and writable) The port power can be switched on (when PP = 0) and off (when PP = 1).

When this bit is 0 or the port power is turned off, the port does not function and connection/disconnection is not detected.

When over-current is detected and PPC is 1, the PP bit for the port is cleared to 0 by the host controller and the port power is turned off.

Bit	Bit Name	Initial Value	R/W	Description
11, 10	LS	00	R	Line Status These bits indicate the current D+/D- logic level. D+ is indicated at the 11th bit (bit 11) and D- is indicated at the 10th bit (bit 10). This field is used for detecting a low-speed device connection before port-reset or port-enable processing is executed. This field is only valid when Port Enable = 0 and Current Connect Status = 1. Since this field is valid during time from detecting connection to enabling the port. Therefore, it is used for determining whether the connected device is low speed or not.
9	—	Undefined	R	Reserved This bit is read as undefined. The write value should always be 0.

LS	State	Description
00	SE0	Not a low-speed device. Performs EHCI reset
01	J-state	Not a low-speed device. Performs EHCI reset
10	K-state	Low-speed device. Releases port ownership
11	SE1	Not a low-speed device. Performs EHCI reset

Bit	Bit Name	Initial Value	R/W	Description
8	PR	0	R/W	Port Reset Controls the port resetting processing. 1: The port being reset 0: The port not being reset The setting of this bit by software commences the bus resetting operation that is defined in the USB 2.0 specifications. To complete the bus resetting operation, the software must write a 0 to this bit. The software must keep this bit at 1 until such time as the reset time defined in the USB 2.0 specifications has elapsed. Notes: 1. To change this bit from 0 to 1, the Port Enable bit must be set to 0. 2. Even if the software sets the bit to 0, there is some delay before the bit becomes 0. Even if this bit is read, the 0 is not read until the reset operation is completed. If the port is in the high-speed mode and the resetting has completed, the host controller automatically enables the port (Port Enable = 1). The host controller must complete the resetting operation and stabilize the port within 2 ms from the time the software set the bit to 0. For example, if the connection of a high-speed device is detected during the resetting of the port, the port must be enabled within 2 ms from the time the software set the bit to 0. 3. If the software uses this bit, the HC Halted bit in USBSTS must be set to 0. As long as USBSTS/HC Halted = 1, the host controller must continue to assert the Port Reset bit to 1.

Bit	Bit Name	Initial Value	R/W	Description
7	S	0	R/W	Suspend Controls the Suspend operation of the port. 1: The port is in the Suspend state. 0: The port is not in the Suspend state. Depending on the setting of the Port Enable and Port Suspend bits, the port assumes the following condition: **Bit[Port Enable, Port Suspend]** / **Port State** 0X / Disable 10 / Enable 11 / Suspend In the Suspend state, any propagation of data from this port to the Down Stream is blocked, except that Port Reset is conveyed. If a transfer is being conducted, the data is blocked after the current transfer operation is completed. In the Suspend state, the port can detect a Resume. Notes: 1. The status of this bit does not change after a transition to the Suspend state and when the current transfer operation, being executed, is pending on Suspend transition. 2. Any clearing of this bit by the software is ignored by the host controller. The host controller unconditionally clears this bit if the following conditions are met: — The software setting the Force Port Resume bit from 0 to 1 — The software setting the Port Reset bit from 0 to 1

Bit	Bit Name	Initial Value	R/W	Description
6	FPR	0	R/W	Force Port Resume Controls the Resume operation of the port. 1: It is detected that the port has resumed. Or, it is detected that the port is driving the Resume state. In other words, the port is in the Resume state. 0: It is not detected that the port is in the Resume state (K state) or driving the Resume state. This bit depends on the status of the Suspend bit. For example, if the port is not in the Suspend state (if both the Suspend bit and Port Enable bit are 1, it is Suspend, but not both bits are 1), even if this bit is set, the resulting status is undefined. When setting this bit, the software drives Resume. In other words, if the software itself wants to drive Resume, it sets this bit. The host controller sets this bit when a transition from J to K is detected when the ports is in the Suspend state. In other words, upon detecting a Resume, the host controller itself sets this bit. Since the host controller sets this bit after detecting the transition from J to K, the host controller must also set the Port Change Detect bit in USBSTS. However, if the software has set this bit, the host controller must not set the Port Change Detect bit in USBSTS. Notes (Setting by software): If EHCI is the owner of the port, the Resume operation is performed according to the USB 2.0 specifications. As long as this bit is 1, the Resume signal (Full-Speed K) continues to be driven. The software must resume for an appropriate length of time, and after performing a Resume for a sufficient length of time, it must set this bit to 0. When this bit is set from 1 to 0, the port returns to the high-speed mode (forces the bus to the idle state of the high-speed mode). Even if a 0 is written to this bit, the bit remains 1 until such time as a switching can be made to idle state of high-speed. The host controller must complete the switching within 2 ms after the software wrote a 0 to this bit.

Bit	Bit Name	Initial Value	R/W	Description
5	OC	0	R/WC	Over-Current Change 1: Indicates that the port over-current state is changed from non-active to active. 0: Indicates that the state is not changed. This bit can be cleared by writing 1 by software.
4	OA	0	R	Over-Current Active 1: Indicates that the port is in the over-current state. 0: Indicates that the port is not in the over-current state. This bit is automatically cleared to 0 when the over-current state is cancelled.
3	PEDC	0	R/WC	Port Enable/Disable Change 1: Indicates that the port state has changed between enable and disable. 0: Indicates that the port state has not changed. This bit is set to 1 only when a condition to disable the port exists at EOF2 and the port is actually disabled (see section 11, Port Error, in the USB 2.0 Specification).

Bit	Bit Name	Initial Value	R/W	Description
2	PED	0	R/W	Port Enabled/Disabled 1: The port is enabled. 0: The port is disabled. The port is enabled only when it is reset; the software cannot enable the port by resetting this bit. Only the host controller can enable the port. The host controller can set this bit only when it is verified by means of a reset sequence that the connected device is a high-speed device. The port is disabled only when a fault condition arises (due to the occurrence of a disconnection or other fault events) or by software. Notes: 1. The status of this bit does not change until the port status actually changes. Due to other host controller processing or bus processing, there may be situations where the port is slow to become enabled/disabled. 2. If this bit is 0b and the port is disabled, data does not propagate to the DownStream; any resetting, however, is conveyed.
1	CSC	0	R/WC	Connect Status Change 1: The Current Connect Status bit has changed. 0: Status is not changed. The host controller sets this bit to 1 when the state of device connection to the port is changed even though software has not clear this bit to 0. This bit can be cleared by writing 1 by software.
0	CCS	0	R	Current Connect Status 1: A device detected in the port. 0: No device detected in the port. This bit indicates the port state. This bit is not directly affected by event generation which sets the Connect Status Change bit.

Note: * Only 1 can be written to clear these bits. Writing 0 is ignored.

Section 23B USB1.1 Host Controller

The USB host interface embedded in this LSI has a root hub and a two-port USB transceiver, and operates in Full speed mode. Open HCI interfaces and registers are also embedded in this LSI.

For the development of software, refer to the Open HCI specifications as well.

23B.1 Features

- Support the Open HCI interface.
- Support the USB host interface.
- Root Hub function
- Operate in Full speed mode (12 Mbps) and Low speed mode (1.5 Mbps)
- Support over-current detection and Power source enable management.

23B.2 Register Descriptions

Table 23B.1 is a list of the registers in this module.

The USB host's registers are allocated to the address space in the I/O bus. This description is based on the Open HCI Rev.1.0. For details, refer to the Open HCI Rev1.0. Bit widths of these resisters are all 32 bits, so should be accessed in longword (32 bits) units. Access in other units is not supported.

Table 23B.1 (1) List of Open HCI Registers

Register Address	Register Name	R/W	Access Size
H'FFE70400	HcRevision register	R	32
H'FFE70404	HcControl register	R/W	32
H'FFE70408	HcCommandStatus register	R/W	32
H'FFE7040C	HcInterruptStatus register	R/W	32
H'FFE70410	HcInterruptEnable register	R/W	32
H'FFE70414	HcInterruptDisable register	R/W	32
H'FFE70418	HcHCCA register	R/W	32
H'FFE7041C	HcPeriodCurrentED register	R/W	32
H'FFE70420	HcControlHeadED register	R/W	32
H'FFE70424	HcControlCurrentED register	R/W	32
H'FFE70428	HcBulkHeadED register	R/W	32

Register Address	Register Name	R/W	Access Size
H'FFE7042C	HcBulkCurrentED register	R/W	32
H'FFE70430	HcDoneHead register	R/W	32
H'FFE70434	HcFmInterval register	R/W	32
H'FFE70438	HcFmRemaining register	R	32
H'FFE7043C	HcFmNumber register	R	32
H'FFE70440	HcPeriodicStart register	R/W	32
H'FFE70444	HcLSThreshold register	R/W	32
H'FFE70448	HcRhDescriptorA register	R/W	32
H'FFE7044C	HcRhDescriptorB register	R/W	32
H'FFE70450	HcRhStatus register	R/W	32
H'FFE70454	HcRhPortStatus1 register	R/W	32
H'FFE70458	HcRhPortStatus2 register	R/W	32

Note: Addresses other than the above must not be write-accessed to. If written to, operation is not guaranteed. If read, an undefined value is read.

Table 23B.1 (2) Register State in Each Operating Mode

Register Name	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
HcRevision register	H'xxxxxx10	H'xxxxxx10	Retained	Retained	Retained	Initialized
HcControl register	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
HcCommandStatus register	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
HcInterruptStatus register	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
HcInterruptEnable register	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
HcInterruptDisable register	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
HcHCCA register	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
HcPeriodCurrentED register	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
HcControlHeadED register	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
HcControlCurrentED register	H'00000000	H'00000000	Retained	Retained	Retained	Initialized

Register Name	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
HcBulkHeadED register	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
HcBulkCurrentED register	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
HcDoneHead register	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
HcFmInterval register	H'00002EDF	H'00002EDF	Retained	Retained	Retained	Initialized
HcFmRemaining register	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
HcFmNumber register	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
HcPeriodicStart register	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
HcLSThreshold register	H'00000628	H'00000628	Retained	Retained	Retained	Initialized
HcRhDescriptorA register	H'FF000902	H'FF000902	Retained	Retained	Retained	Initialized
HcRhDescriptorB register	H'00060000	H'00060000	Retained	Retained	Retained	Initialized
HcRhStatus register	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
HcRhPortStatus1 register	H'00000100	H'00000100	Retained	Retained	Retained	Initialized
HcRhPortStatus2 register	H'00000100	H'00000100	Retained	Retained	Retained	Initialized

Note: Initialized: The value is the one written in "Power-on Reset".

[Legend for Register Description]

Initial value: Register value after a reset.

—: The read value is undefined. The write value should always be 0.

R/W: Bit or field is readable and writable.

R: Bit or field is readable. The write value should always be 0.

Note: The register can be set when a 48 MHz clock is input.

Other than control registers for setting, all registers conform to the Open HCIs specification.

These control registers are only for this LSI.

23B.2.1 HcRevision Register

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value :	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	REVISION							
Initial value:	—	—	—	—	—	—	—	—	0	0	0	1	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	Undefined	R	Reserved These bits are always read as undefined. The write value should always be 0.
7 to 0	REVISION	H'10	R	Revision These bits indicate the Open HCI Specification revision number implemented by the Hardware. (X.Y = XYh) USB Host Controller supports the Open HCI1.0 specification.

23B.2.2 HcControl Register

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	RWCE	RWC	IR	HCFS		BLE	CLE	IE	PLE	CBSR	
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
10	RWCE	0	R/W	Remote Wakeup Connected Enable If a remote wakeup signal is supported, this bit enables that operation. Since remote wakeup signal is not supported, this bit is ignored.
9	RWC	0	R/W	Remote Wakeup Connected This bit indicates whether the Host Controller (HC) supports a remote wakeup signal.
8	IR	0	R/W	Interrupt Routing This bit specifies interrupt routing: 0: Interrupts routed to normal interrupt processing unit (INTC2). 1: Interrupts routed to SMI.
7, 6	HCFS	00	R/W	Host Controller Functional State These bits set the Host Controller state. The state encodings are: 00: UsbReset 01: UsbResume 10: UsbOperational 11: UsbSuspend The Host Controller may force a state change from UsbSuspend to UsbResume after detecting resume signaling from a downstream port.
5	BLE	0	R/W	Bulk List Enable Setting this bit enables processing of the Bulk list.
4	CLE	0	R/W	Control List Enable Setting this bit enables processing of the Control list.
3	IE	0	R/W	Isochronous Enable Clearing this bit disables the Isochronous List when the Periodic List is enabled (so Interrupt EDs may be serviced). While processing the Periodic List, the Host Controller will check this bit when it finds an isochronous ED.
2	PLE	0	R/W	Periodic List Enable Setting this bit enables processing of the Periodic (interrupt and isochronous) list. The Host Controller checks this bit prior to attempting any periodic transfers in a frame.

Bit	Bit Name	Initial Value	R/W	Description
1, 0	CBSR	00	R/W	Control Bulk Service Ratio These bits specify the number of Control Endpoints serviced for every Bulk Endpoint. Encoding is N-1 where N is the number of Control Endpoints (i.e. 00 = 1 Control Endpoint; 11 = 4 Control Endpoints).

23B.2.3 HcCommandStatus Register

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SOC	
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	OCR	BLF	CLF	HCR
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17, 16	SOC	00	R	Schedule Overrun Count These bits are incremented every time the SchedulingOverrun bit in HcInterruptStatus is set. After counting has reached 11, the next incrimination makes the value return to 00.
15 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	OCR	0	R/W	Ownership Change Request Setting this bit by software sets the Ownership Change bit in HcInterruptStatus register. This bit is cleared by software.
2	BLF	0	R/W	Bulk List Filled Setting this bit indicates there is an active ED on the Bulk List. The bit can be set by either software or the Host Controller. The bit is cleared by the Host Controller each time it begins processing the head of the Bulk List.

Bit	Bit Name	Initial Value	R/W	Description
1	CLF	0	R/W	Control List Filled Setting this bit indicates there is an active ED on the Control List. The bit can be set by either software or the Host Controller. The bit is cleared by the Host Controller each time it begins processing the head of the Control List.
0	HCR	0	R/W	Host Controller Reset This bit is set to initiate a software reset. This bit is cleared by the Host Controller upon completion of the reset operation.

23B.2.4 HcInterruptStatus Register

All bits are set by hardware and cleared by software.

These bits in this register can be cleared by writing 1 to bit positions to be cleared.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	OC	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	RHSC	FNO	UE	RD	SF	WDH	SO
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved These bits are always read as 0. The write value should always be 0.
30	OC	0	R/W	Ownership Change This bit is set when the OwnershipChangeRequest bit in HcCommandStatus register is set.
29 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
6	RHSC	0	R/W	Root Hub Status Change This bit is set when the content of HcRhStatus register or the content of any HcRhPortStatus register has changed.
5	FNO	0	R/W	Frame Number Overflow This bit is set when bit 15 of FrameNumber changes value from 0 to 1 or from 1 to 0.
4	UE	0	R/W	Unrecoverable Error This bit is set when HC detects a system error that is not USB related.
3	RD	0	R/W	Resume Detected This bit is set when the Host Controller detects resume signaling on a downstream port.
2	SF	0	R/W	Start of Frame This bit is set when the Frame manager signals a Start of Frame's event.
1	WDH	0	R/W	Writeback Done Head This bit is set after the Host Controller has written the value of HcDoneHead register to HccaDoneHead.
0	SO	0	R/W	Scheduling Overrun This bit is set when the List Processor determines a Schedule Overrun has occurred.

23B.2.5 HcInterruptEnable Register

Writing 1 to a bit in this register sets the corresponding bit, while writing a 0 to a bit leaves the bit unchanged.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MIE	OCE	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	RHSCE	FNOE	UEE	RDE	SFE	WDHE	SOE
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	MIE	0	R/W	Master Interrupt Enable This bit is a global interrupt enable. Writing 1 allows interrupts to be enabled via the specific enable bits listed below.
30	OCE	0	R/W	Ownership Change Enable 0: Ignored 1: Interrupt due to Ownership Change is enabled.
29 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6	RHSCE	0	R/W	Root Hub Status Change Enable 0: Ignored 1: Interrupt due to Root Hub Status Change is enabled.
5	FNOE	0	R/W	Frame Number Overflow Enable 0: Ignored 1: Interrupt due to Frame Number Overflow is enabled.
4	UEE	0	R/W	Unrecoverable Error Enable This function is not supported. Writing is ignored.
3	RDE	0	R/W	Resume Detected Enable 0: Ignored 1: Interrupt due to Resume Detected is enabled.

Bit	Bit Name	Initial Value	R/W	Description
2	SFE	0	R/W	Start of Frame Enable 0: Ignored 1: Interrupt due to Start of Frame is enabled.
1	WDHE	0	R/W	Writeback Done Head Enable 0: Ignored 1: Interrupt due to Writeback Done Head is enabled.
0	SOE	0	R/W	Scheduling Overrun Enable 0: Ignored 1: Interrupt due to Scheduling Overrun is enabled.

23B.2.6 HcInterruptDisable Register

Writing 1 to a bit in this register clears the corresponding bit in the HcInterrupt Enable register, while writing 0 to a bit leaves the bit unchanged.

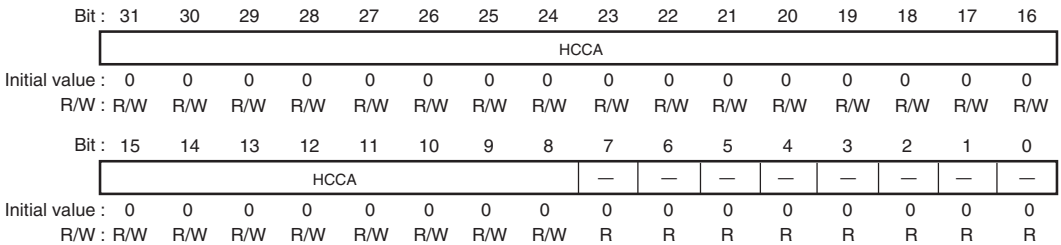
Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MID	OCD	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	RHSCD	FNOD	UED	RDD	SFD	WDHD	SOD
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	MID	0	R/W	Master Interrupt Disable This bit is a global interrupt disable. Writing 1 disables all interrupts.
30	OCD	0	R/W	Ownership Change Disable 0: Ignored 1: Interrupt due to Ownership Change is disabled.
29 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6	RHSCD	0	R/W	Root Hub Status Change Disable 0: Ignored 1: Interrupt due to Root Hub Status Change is disabled.
5	FNOD	0	R/W	Frame Number Overflow Disable 0: Ignored 1: Interrupt due to Frame Number Overflow is disabled.
4	UED	0	R/W	Unrecoverable Error Disable This function is not supported. Writing is ignored.
3	RDD	0	R/W	Resume Detected Disable 0: Ignored 1: Interrupt due to Resume Detected is disabled.

Bit	Bit Name	Initial Value	R/W	Description
2	SFD	0	R/W	Start of Frame Disable 0: Ignored 1: Interrupt due to Start of Frame is disabled.
1	WDHD	0	R/W	Writeback Done Head Disable 0: Ignored 1: Interrupt due to Writeback Done Head is disabled.
0	SOD	0	R/W	Scheduling Overrun Disable 0: Ignored 1: Interrupt generation due to Scheduling Overrun is disabled.

23B.2.7 HcHCCA Register



Bit	Bit Name	Initial Value	R/W	Description
31 to 8	HCCA	All 0	R/W	HCCA Pointer to HCCA base address. (within Unified Memory space)
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

23B.2.8 HcPeriodCurrntED Register

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PCED															
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PCED												—	—	—	—
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	PCED	All 0	R	Period Current ED Pointer to the current Periodic List ED. (within Unified Memory space)
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

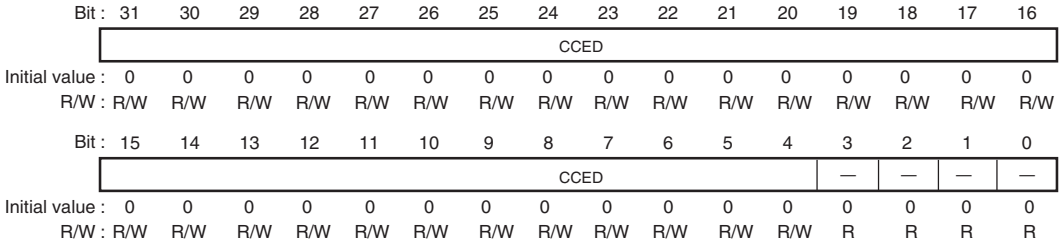
23B.2.9 HcControlHeadED Register

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CHED															
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CHED												—	—	—	—
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

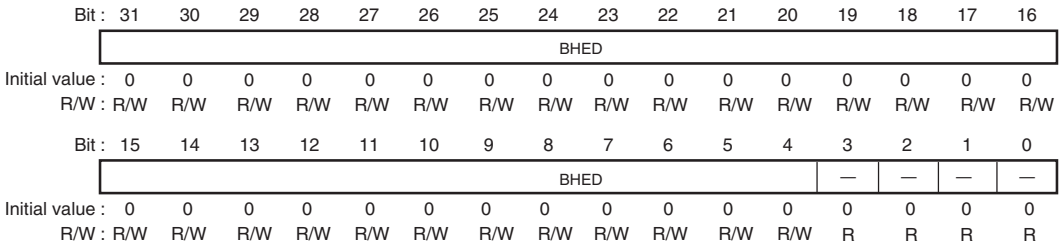
Bit	Bit Name	Initial Value	R/W	Description
31 to 4	CHED	All 0	R/W	Control Head ED Pointer to the Control List Head ED (within Unified Memory space).
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

23B.2.10 HcControlCurrentED Register



Bit	Bit Name	Initial Value	R/W	Description
31 to 4	CCED	All 0	R/W	Control Current ED Pointer to the current Control List ED (within Unified Memory space).
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

23B.2.11 HcBulkHeadED Register



Bit	Bit Name	Initial Value	R/W	Description
31 to 4	BHED	All 0	R/W	Bulk Head ED Pointer to the Bulk List Head ED (within Unified Memory space).
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

23B.2.12 HcBulkCurrentED Register

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BCED															
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BCED												—	—	—	—
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	BCED	All 0	R/W	Bulk Current ED Pointer to the current Bulk List ED (within Unified Memory space).
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

23B.2.13 HcDoneHead Register

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DH															
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DH												—	—	—	—
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	DH	All 0	R	DoneHead Pointer to the current Done List Head ED (within Unified Memory space).
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

23B.2.14 HcFmInterval Register

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FIT	FSLDP														
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	FI	FI	FI	FI	FI	FI	FI	FI	FI	FI	FI	FI	FI	FI
Initial value :	0	0	1	0	1	1	1	0	1	1	0	1	1	1	1	1
R/W :	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	FIT	0	R/W	Frame Interval Toggle This bit is toggled by Host Control Driver (HCD) whenever it loads a new value into FrameInterval bit.
30 to 16	FSLDP	All 0	R/W	FS Largest Data Packet These bits specify a value which is loaded into the Largest Data Packet Counter at the beginning of each frame.
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13 to 0	FI	H'2EDF	R/W	Frame Interval These bits specify the length of a frame as (bit times - 1). For 12,000 bit times in a frame, a value of 11,999 is specified.

23B.2.15 HcFrameRemaining Register

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FRT	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	FR	FR	FR	FR	FR	FR	FR	FR	FR	FR	FR	FR	FR	FR
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bits	Bit Name	Initial Value	R/W	Description
31	FRT	0	R	Frame Remaining Toggle This bit is loaded with FrameIntervalToggle when FrameRemaining is loaded.
30 to 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13 to 0	FR	All 0	R	Frame Remaining These bits are the 14-bit down counter used to time a frame. When the Host Controller is in the USB OPERATIONAL state, the counter decrements each 12 MHz clock period. When the count reaches 0, the end of a frame has been reached. The counter reloads with FrameInterval at that time. In addition, the counter reloads when the Host Controller transitions into USB OPERATIONAL.

23B.2.16 HcFmNumber Register

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FN	FN	FN	FN	FN	FN	FN	FN	FN	FN	FN	FN	FN	FN	FN	FN
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	FN	All 0	R	Frame Number These bits are the 16-bit up counter. The count is incremented coincident with the loading of FrameRemaining bit. The count will roll over from H'FFFF to H'0000.

23B.2.17 HcPeriodicStart Register

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PS	PS	PS	PS	PS	PS	PS	PS	PS	PS	PS	PS	PS	PS
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13 to 0	PS	All 0	R/W	Periodic Start These bits set a value used by the List Processor to determine where in a frame the Periodic List processing must begin.

23B.2.18 HcLSThreshold Register

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	LST	LST	LST	LST	LST	LST	LST	LST	LST	LST	LST	LST
Initial value :	0	0	0	0	0	1	1	0	0	0	1	0	1	0	0	0
R/W :	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	LST	H'628	R/W	LS Threshold These bits are a value used by the Frame manager to determine whether or not a low speed transaction can be started in the current frame.

23B.2.19 HcRhDescriptorA Register

This register is only reset by a power-on reset. It is written during system initialization to configure the Root Hub. This register should not be written during normal operation.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PPGT								—	—	—	—	—	—	—	—
Initial value :	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

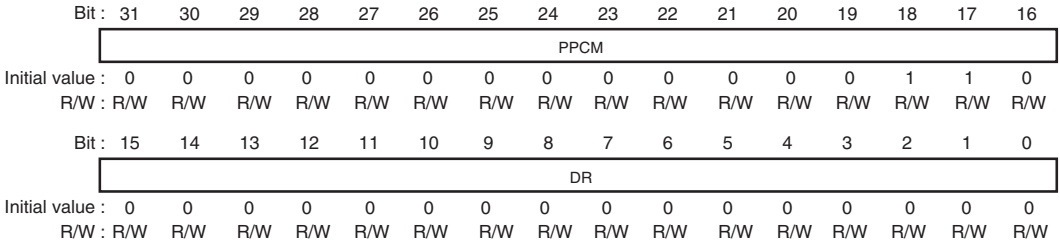
Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	NOCP	OCPM	DT	NPS	PSM	NDP							
Initial value :	0	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0
R/W :	R	R	R	R/W	R/W	R	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	PPGT	H'FF	R/W	Power-on to Power Good Time USB Host Controller power switching is effective within 2 ms. The bit value is represented as the number of 2 ms intervals. Only bits 25 and 24 can be written to. The remaining bits are read only as 0. It is not expected that these bits be written to anything other than 1h, but limited adjustment is allowed. These bits should be written to support the system implementation. These bits should always be written to a non-zero value.
23 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	NOCP	0	R/W	No Over Current Protection USB Host Controller implements global over-current reporting 0: Over-current status is reported 1: Over-current status is not reported This bit should be written to support the external system port over-current implementation.

Bit	Bit Name	Initial Value	R/W	Description
11	OCPM	1	R/W	Over-current Protection Mode USB Host Controller implements global over-current reporting 0: Global Over-Current is reported. 1: Individual Over-Current is reported. This bit is only valid when NoOverCurrentProtection bit is cleared.
10	DT	0	R	Device Type USB Host Controller is not a composite device.
9	NPS	0	R/W	No Power Switching USB Host Controller implements global power switching. 0: Ports are power switched. 1: Ports are always powered on. This bit should be written to support the external system port power switching implementation.
8	PSM	1	R/W	Power Switching Mode USB Host Controller implements a global power switching mode. 0: Global Switching 1: Individual Switching This bit is only valid when NoPowerSwitching is cleared.
7 to 0	NDP	H'02	R	Number Downstream Ports USB Host Controller supports one downstream port.

23B.2.20 HcRhDescriptorB Register

This register is only reset by a power-on reset. It is written during system initialization to configure the Root Hub. This register should not be written during normal operation.



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	PPCM	H'0006	R/W	Port Power Control Mask USB Host Controller implements global-power switching. These bits are only valid if NoPowerSwitching is cleared and PowerSwitchingMode bit is set (individual port switching). When set, the port only responds to individual port power switching commands (set/ClearPortPower). When cleared, the port only responds to global power switching commands (set/ClearGlobalPower). 0: Device not removable 1: Global-power switching is masked Port Bit relationship Bit 16: Reserved Bit 17: Port 1 Bit 18: Port 2 : Bit 31: Port 15 Unimplemented ports are reserved. These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	DR	H'0000	R/W	Device Removable USB Host Controller ports default to removable devices. 0: Device removable 1: Device not removable Port Bit relationship Bit 0: Reserved Bit 1: Port 1 Bit 2: Port 2 : Bit 15: Port 15 Unimplemented ports are reserved. These bits are always read as 0. The write value should always be 0.

23B.2.21 HcRhStatus Register

This register is reset by the UsbReset state.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRWE	—	—	—	—	—	—	—	—	—	—	—	—	—	OCIC	SGP
Initial value :	—	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DRWE	—	—	—	—	—	—	—	—	—	—	—	—	—	OCI	CGP
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	CRWE	—	W	(write) Clear Remote Wakeup Enable Writing 1 to this bit clears DeviceRemoteWakeupEnable bit. Writing 0 has no effect.
30 to 18	—	All 0	—	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
17	OCIC	0	R/W	Over Current Indicator Change This bit is set when OverCurrentIndicator changes. Writing 1 clears this bit. Writing 0 has no effect.
16	SGP	0	R/W	(read) Local Power Status Change Not supported by this LSI. The read value should always be 0. (write) SetGlobalPower Write 1 issues a SetGlobalPower command to the ports. Writing 0 has no effect.
15	DRWE	0	R/W	(read) Device Remote Wakeup Enable This bit enables ports' ConnectStatusChange as a remote wakeup event. 0: Disabled 1: Enabled (write) SetRemoteWakeupEnable Writing 1 sets the DeviceRemoteWakeupEnable bit. Writing 0 has no effect.
14 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	OCI	0	R	Over Current Indicator This bit reflects the state of the OVRCUR pin. This bit is only valid if NoOverCurrentProtection and OverCurrentProtectionMode bits are cleared. 0: No over-current condition 1: Over-current condition
0	CGP	0	R/W	(read) Local Power Status Not Supported by this LSI. The read value should always be 0. (write) ClearGlobalPower Writing 1 issues a ClearGlobalPower command to the ports. Writing 0 has no effect.

23B.2.22 HcRhPortStatus1, 2 Register

This register is reset by the UsbReset state.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	PRSC	POCIC	PSSC	PESC	CSC
Initial value :	—	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	LSDA	PPS	—	—	—	PRS	POCI	PSS	PES	CCS
Initial value :	0	0	0	0	0	0	X*	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	PRSC	0	R/W	Port Reset Status Change This bit indicates that the port reset signal has completed. 0: Port reset is not complete. 1: Port reset is complete.
19	POCIC	0	R/W	Port Over Current Indicator Change This bit is set when OverCurrentIndicator changes. Writing 1 clears this bit. Writing 0 has no effect.
18	PSSC	0	R/W	Port Suspend Status Change This bit indicates the completion of the selective resume sequence for the port. 0: Port is not resumed. 1: Port resume is complete.
17	PESC	0	R/W	Port Enable Status Change This bit indicates that the port has been disabled due to a hardware event (PortEnableStatus bit is cleared). 0: Port has not been disabled. 1: PortEnableStatus bit has been cleared.

Bit	Bit Name	Initial Value	R/W	Description
16	CSC	0	R/W	Connect Status Change This bit indicates a connection or disconnection event has been detected. Writing 1 clears this bit. Writing 0 has no effect. 0: No connect/disconnect event. 1: Hardware detection of connect/disconnect event. Note: If DeviceRemoveable is set, this bit resets to 1.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	LSDA	×*	R/W	(read) Low Speed Device Attached This bit defines the speed (and bus idle) of the attached device. It is only valid when CurrentConnectStatus is set. 0: Full Speed device 1: Low Speed device (write) ClearPortPower Writing 1 clears the PortPowerStatus bit. Writing 0 has no effect.
8	PPS	0	R/W	(read) Port Power Status This bit reflects the power state of the port regardless of power switching mode. 0: Port power is off. 1: Port power is on. Note: If NoPowerSwitching bit is set, the read value should always be 0. (write) SetPortPower Writing 1 sets the PortPowerStatus bit. Writing 0 has no effect.
7 to 5	—	All 0	—	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
4	PRS	0	R/W	(read) Port Reset Status 0: Port reset signal is not active. 1: Port reset signal is active. (write) SetPortReset Writing 1 sets the PortResetStatus bit. Writing 0 has no effect.
3	POCI	0	R/W	(read) Port Over Current Indicator USB Host Controller supports global over-current reporting. This bit reflects the state of the OVRCUR pin dedicated to this port. This bit is only valid if NoOverCurrentProtection bit is cleared and OverCurrentProtectionMode is set. 0: No over-current condition 1: Over-current condition (write) ClearSuspendStatus Writing 1 initiates the selective resume sequence for the port. Writing 0 has no effect.
2	PSS	0	R/W	(read) Port Suspend Status 0: Port is not suspended 1: Port is selectively suspended (write) SetPortSuspend Writing 1 sets the PortSuspendStatus bit. Writing 0 has no effect.
1	PES	0	R/W	(read) Port Enable Status 0: Port disabled. 1: Port enabled. (write) SetPortEnable Writing 1 sets the PortEnableStatus bit. Writing 0 has no effect.
0	CCS	0	R/W	(read) Current Connect Status 0: No device connected. 1: Device connected. Note: If DeviceRemoveable bit is set (not removable) this bit is always read as 1. (write) ClearPortEnable Writing 1 clears the PortEnableStatus bit. Writing 0 has no effect.

Note: *: Will have an effect on the status of the transceiver.

Section 23C USB 2.0 Function Module (USBF)

This module is a USB controller that provides peripheral function operations and supports high-speed and full-speed transfers defined by USB specification 2.0.

This module supports all of the transfer types defined by the USB specification. It has up to 4 Kbytes of buffer memory for data transfer and provides a maximum of ten pipes. Any endpoint numbers can be assigned to PIPE1 to PIPE9, based on the function devices or user system for communication. A local bus interface (dedicated to a DMA interface) is provided separately from the CPU bus interface for systems involving high-speed large-size data transfers.

23C.1 Features

(1) Peripheral Controller Supporting USB High-Speed Operation

- Supporting the interface conforming to UTMI+ specification 1.0 (UTMI+ level 3; the On-the-Go function not supported)

(2) All Types of USB Transfers Supported

Support of all types of USB transfers including isochronous transfer

- Control transfer
- Bulk transfer
- Interrupt transfer (high bandwidth transfer not supported)
- Isochronous transfer (high bandwidth transfer not supported)

(3) Bus Interfaces

- 32-bit-width HPB interface
- Two channels of DMA interface available
  (DMAC interface can be selected separately from the CPU bus interface)
- High-speed data transfer for access to the internal FIFOs at 60 Mbytes/second (in UTMI+ 8-bit mode)

(4) Pipe Configuration

- On-chip 4-Kbyte (max.) buffer memory for USB communications
- Up to ten pipes can be selected (including the default control pipe)
- Programmable pipe configuration
- Endpoint numbers can be assigned flexibly to PIPE1 to PIPE9.
- Transfer conditions that can be set for each pipe:

PIPE0:	Control transfer, 64-byte fixed single buffer
PIPE1 and PIPE2:	Bulk transfers/isochronous transfer, continuous transfer mode, programmable buffer size (up to 1 Kbyte: double buffer can be specified)
PIPE3 to PIPE5:	Bulk transfer, continuous transfer mode, programmable buffer size (up to 1 Kbyte: double buffer can be specified)
PIPE6 to PIPE9:	Interrupt transfer, 64-byte fixed single buffer

(5) Features of Peripheral Function Operation

- Support of high-speed transfer (480 Mbps) and full-speed transfer (12 Mbps)
- Automatic recognition of high-speed operation or full-speed operation based on automatic response to the reset handshake
- Control transfer stage control function
- Device state control function
- Auto response function for SET_ADDRESS request
- NAK response interrupt function (NRDY)
- SOF interpolation function

(6) Other Features

- Byte endian swap function for supporting both big and little endian data formats
- Transfer ending function using transaction count
- DMA transfer ending function using external triggers (TEND or WREND signal)
- SOF pulse output function
- BRDY interrupt event notification timing change function (BFRE)
- Function that automatically clears the buffer memory after the data for the pipe specified at the DnFIFO port has been read (DCLRM)
- NAK setting function for response PID generated by end of transfer (SHTNAK)

(7) Applications

Navigation systems, DVD recorders, set-top boxes, audio equipment, printers, external storage equipment, and other systems incorporating USB functions.

23C.1.1 Block Diagram

Figure 23C.1 shows a block diagram of this module.

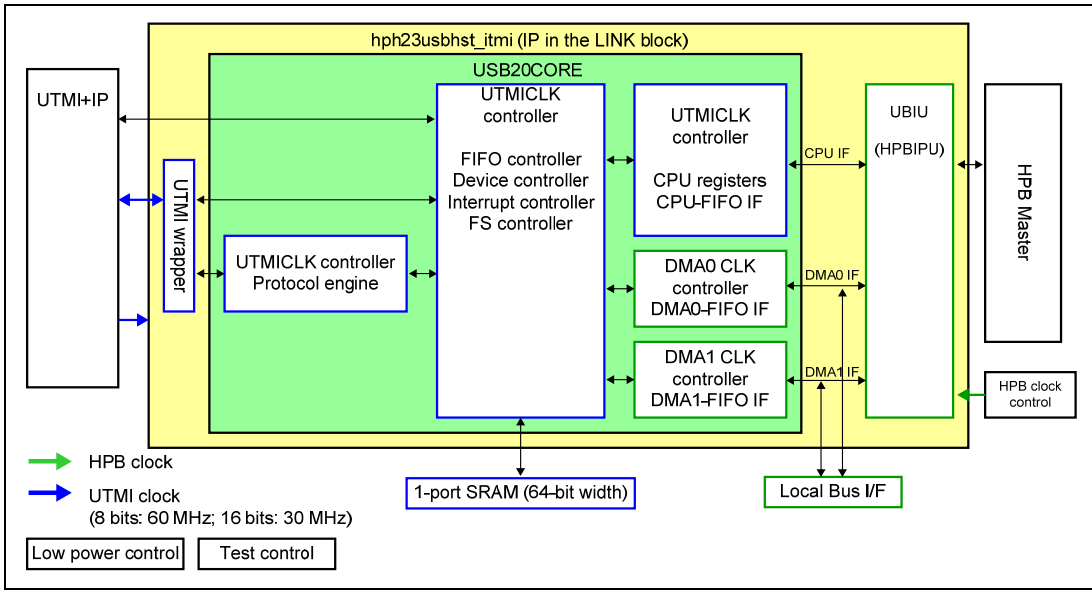


Figure 23C.1 Block Diagram

Data transfer between this module and the host controller connected to the USB bus uses buffer memory assigned to each pipe. For communication between the function and host controllers, this controller converts the data stored in buffer memory into USB data packets and outputs them to the USB bus serially, and also inputs data packets from the USB bus and stores them in buffer memory.

23C.1.2 Functional Overview

(1) Controller Function

This module automatically detects the USB transfer speed.

(2) Bus Interface

This module supports the interface of a 32-bit-width HPB.

(a) FIFO Buffer Memory Access

The following two types of access are available for the FIFO buffer memory for USB data transfers. To read from or write to the FIFO buffer memory, access (read or write) the FIFO ports through the CPU or DMAC.

1. CPU access
   Specify a FIFO port address and write or read data to or from the FIFO buffer memory.
2. DMA access
   Specify a FIFO port address through the DMAC in the CPU or the dedicated DMAC and write or read data to or from the FIFO buffer memory.

USB data transfer is done in little endian. The byte endian swap function is available for FIFO port access; for 16-bit or 32-bit access, the endian can be switched through register settings.

(b) FIFO Buffer Memory Access from Local Bus

A local bus interface (dedicated to the DMA interface) is available separately from the CPU bus interface, enabling FIFO buffer access for high-speed large-size data transfer.

(3) USB Events

This module sends an interrupt to the user system to notify an event in USB operation. This module asserts the UCL_Dx_DREQ signal to notify that the pipe selected for the DMA interface has become ready for access.

Sending interrupts can be enabled or disabled separately for each interrupt type and source through software settings.

(4) USB Data Transfers

This module performs all types of USB data transfer: control transfer, bulk transfer, interrupt transfer, and isochronous transfer. The following numbers of pipes are available for each transfer type.

1. One pipe dedicated to control transfer
2. Four pipes dedicated to interrupt transfer
3. Three pipes dedicated to bulk transfer
4. Two pipes selectively used for bulk transfer or isochronous transfer

Each pipe should be set up as necessary for the desired USB transfer in accordance with the user system; for example, transfer type, endpoint number, or maximum packet size.

This module provides up to 4 Kbytes of buffer memory. For pipes dedicated for bulk transfer or selective pipes for bulk transfer or isochronous transfer, buffer memory assignment, buffer operating mode setting, or other necessary settings should be made in accordance with the user system. By setting the buffer operating mode, such as double buffer structure or continuous data packet transfer mode, fast data transfer is achieved with fewer interrupts.

Use three FIFO port registers to access the buffer memory from the user system controller CPU or DMA controller.

(5) Functions Available for Access from DMAC

This module provides two channels of the DMA interface having the following functions.

1. Transfer end notification function using a transfer end signal (only when the local bus is selected)
2. Automatic FIFO buffer clear function when a zero-length packet is received
3. Transfer ending function using the transaction counter

(6) SOF Pulse Output Function

This module provides a function for outputting an SOF pulse to indicate the SOF packet transfer timing. An SOF pulse is output when an SOF packet is received. Even if an SOF packet is damaged, pulses are output periodically using the SOF interpolation timer.

23C.2 List of Registers

Table 23C.1 (1) is a list of the registers in this module.

Table 23C.1 (1) List of Registers

Name	Abbr.	R/W	Address	Access size
System configuration control register	SYSCFG0	R/W	H'FFE6 0000	16
CPU bus wait register	BUSWAIT	R/W	H'FFE6 0002	16
System configuration status register	SYSSTS0	R	H'FFE6 0004	16
Device state control register	DVSTCTR0	R/W	H'FFE6 0008	16
Test mode register	TESTMODE	R/W	H'FFE6 000C	16
CFIFO port register	CFIFO	R/W	H'FFE6 0014	8, 16, 32
D0FIFO port register	D0FIFO	R/W	H'FFE6 0018	8, 16, 32
D1FIFO port register	D1FIFO	R/W	H'FFE6 001C	8, 16, 32
CFIFO port select register	CFIFOSEL	R/W	H'FFE6 0020	16
CFIFO port control register	CFIFOCTR	R/W	H'FFE6 0022	16
D0FIFO port select register	D0FIFOSEL	R/W	H'FFE6 0028	16
D0FIFO port control register	D0FIFOCTR	R/W	H'FFE6 002A	16
D1FIFO port select register	D1FIFOSEL	R/W	H'FFE6 002C	16
D1FIFO port control register	D1FIFOCTR	R/W	H'FFE6 002E	16
Interrupt enable register 0	INTENB0	R/W	H'FFE6 0030	16
BRDY Interrupt enable register	BRDYENB	R/W	H'FFE6 0036	16
NRDY Interrupt enable register	NRDYENB	R/W	H'FFE6 0038	16
BEMP Interrupt enable register	BEMPENB	R/W	H'FFE6 003A	16
SOF pin configuration register	SOFCFG	R/W	H'FFE6 003C	16
Interrupt status register 0	INTSTS0	R/W	H'FFE6 0040	16
BRDY Interrupt status register	BRDYSTS	R/W	H'FFE6 0046	16
NRDY Interrupt status register	NRDYSTS	R/W	H'FFE6 0048	16
BEMP Interrupt status register	BEMPSTS	R/W	H'FFE6 004A	16
Frame number register	FRMNUM	R/W	H'FFE6 004C	16
µFrame number register	UFRMNUM	R	H'FFE6 004E	16
USB address register	USBADDR	R	H'FFE6 0050	16
USB request type register	USBREQ	R	H'FFE6 0054	16
USB request value register	USBVAL	R	H'FFE6 0056	16

Name	Abbr.	R/W	Address	Access size
USB request index register	USBINDX	R	H'FFE6 0058	16
USB request length register	USBLENG	R	H'FFE6 005A	16
DCP maximum packet size register	DCPMAXP	R/W	H'FFE6 005E	16
DCP control register	DCPCTR	R/W	H'FFE6 0060	16
Pipe window select register	PIPESEL	R/W	H'FFE6 0064	16
Pipe configuration register	PIPECFG	R/W	H'FFE6 0068	16
Pipe buffer setting register	PIPEBUF	R/W	H'FFE6 006A	16
Pipe maximum packet size register	PIPEMAXP	R/W	H'FFE6 006C	16
Pipe cycle control register	PIPEPERI	R/W	H'FFE6 006E	16
PIPE1 control register	PIPE1CTR	R/W	H'FFE6 0070	16
PIPE2 control register	PIPE2CTR	R/W	H'FFE6 0072	16
PIPE3 control register	PIPE3CTR	R/W	H'FFE6 0074	16
PIPE4 control register	PIPE4CTR	R/W	H'FFE6 0076	16
PIPE5 control register	PIPE5CTR	R/W	H'FFE6 0078	16
PIPE6 control register	PIPE6CTR	R/W	H'FFE6 007A	16
PIPE7 control register	PIPE7CTR	R/W	H'FFE6 007C	16
PIPE8 control register	PIPE8CTR	R/W	H'FFE6 007E	16
PIPE9 control register	PIPE9CTR	R/W	H'FFE6 0080	16
PIPE1 transaction counter enable register	PIPE1TRE	R/W	H'FFE6 0090	16
PIPE1 transaction counter register	PIPE1TRN	R/W	H'FFE6 0092	16
PIPE2 transaction counter enable register	PIPE2TRE	R/W	H'FFE6 0094	16
PIPE2 transaction counter register	PIPE2TRN	R/W	H'FFE6 0096	16
PIPE3 transaction counter enable register	PIPE3TRE	R/W	H'FFE6 0098	16
PIPE3 transaction counter register	PIPE3TRN	R/W	H'FFE6 009A	16
PIPE4 transaction counter enable register	PIPE4TRE	R/W	H'FFE6 009C	16
PIPE4 transaction counter register	PIPE4TRN	R/W	H'FFE6 009E	16
PIPE5 transaction counter enable register	PIPE5TRE	R/W	H'FFE6 00A0	16
PIPE5 transaction counter register	PIPE5TRN	R/W	H'FFE6 00A2	16
UTMI suspend mode register	SUSPMODE	R/W	H'FFE6 0102	16

Note: Do not write to any addresses other than listed above. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than listed above are prohibited.

Table 23C.1 (2) Register State in Each Operating Mode

Abbr.	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
SYSCFG0	Undefined	Undefined	Retained	Retained	Retained	Initialized
BUSWAIT	Undefined	Undefined	Retained	Retained	Retained	Initialized
SYSSTS0	Undefined	Undefined	Retained	Retained	Retained	Initialized
DVSTCTR0	Undefined	Undefined	Retained	Retained	Retained	Initialized
TESTMODE	Undefined	Undefined	Retained	Retained	Retained	Initialized
CFIFO	H'0000 0000	H'0000 0000	Retained	Retained	Retained	Initialized
D0FIFO	H'0000 0000	H'0000 0000	Retained	Retained	Retained	Initialized
D1FIFO	H'0000 0000	H'0000 0000	Retained	Retained	Retained	Initialized
CFIFOSEL	Undefined	Undefined	Retained	Retained	Retained	Initialized
CFIFOCTR	Undefined	Undefined	Retained	Retained	Retained	Initialized
D0FIFOSEL	Undefined	Undefined	Retained	Retained	Retained	Initialized
D0FIFOCTR	Undefined	Undefined	Retained	Retained	Retained	Initialized
D1FIFOSEL	Undefined	Undefined	Retained	Retained	Retained	Initialized
D1FIFOCTR	Undefined	Undefined	Retained	Retained	Retained	Initialized
INTENB0	Undefined	Undefined	Retained	Retained	Retained	Initialized
BRDYENB	Undefined	Undefined	Retained	Retained	Retained	Initialized
NRDYENB	Undefined	Undefined	Retained	Retained	Retained	Initialized
BEMPENB	Undefined	Undefined	Retained	Retained	Retained	Initialized
SOFCFG	Undefined	Undefined	Retained	Retained	Retained	Initialized
INTSTS0	Undefined	Undefined	Retained	Retained	Retained	Initialized
BRDYSTS	Undefined	Undefined	Retained	Retained	Retained	Initialized
NRDYSTS	Undefined	Undefined	Retained	Retained	Retained	Initialized
BEMPSTS	Undefined	Undefined	Retained	Retained	Retained	Initialized
FRMNUM	Undefined	Undefined	Retained	Retained	Retained	Initialized
UFRMNUM	Undefined	Undefined	Retained	Retained	Retained	Initialized
USBADDR	Undefined	Undefined	Retained	Retained	Retained	Initialized
USBREQ	H'0000	H'0000	Retained	Retained	Retained	Initialized
USBVAL	H'0000	H'0000	Retained	Retained	Retained	Initialized
USBINDX	H'0000	H'0000	Retained	Retained	Retained	Initialized
USBLENG	H'0000	H'0000	Retained	Retained	Retained	Initialized

Abbr.	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
DCPMAXP	Undefined	Undefined	Retained	Retained	Retained	Initialized
DCPCTR	Undefined	Undefined	Retained	Retained	Retained	Initialized
PIPESEL	Undefined	Undefined	Retained	Retained	Retained	Initialized
PIPECFG	Undefined	Undefined	Retained	Retained	Retained	Initialized
PIPEBUF	Undefined	Undefined	Retained	Retained	Retained	Initialized
PIPEMAXP	Undefined	Undefined	Retained	Retained	Retained	Initialized
PIPEPERI	Undefined	Undefined	Retained	Retained	Retained	Initialized
PIPE1CTR	Undefined	Undefined	Retained	Retained	Retained	Initialized
PIPE2CTR	Undefined	Undefined	Retained	Retained	Retained	Initialized
PIPE3CTR	Undefined	Undefined	Retained	Retained	Retained	Initialized
PIPE4CTR	Undefined	Undefined	Retained	Retained	Retained	Initialized
PIPE5CTR	Undefined	Undefined	Retained	Retained	Retained	Initialized
PIPE6CTR	Undefined	Undefined	Retained	Retained	Retained	Initialized
PIPE7CTR	Undefined	Undefined	Retained	Retained	Retained	Initialized
PIPE8CTR	Undefined	Undefined	Retained	Retained	Retained	Initialized
PIPE9CTR	Undefined	Undefined	Retained	Retained	Retained	Initialized
PIPE1TRE	Undefined	Undefined	Retained	Retained	Retained	Initialized
PIPE1TRN	H'0000	H'0000	Retained	Retained	Retained	Initialized
PIPE2TRE	Undefined	Undefined	Retained	Retained	Retained	Initialized
PIPE2TRN	H'0000	H'0000	Retained	Retained	Retained	Initialized
PIPE3TRE	Undefined	Undefined	Retained	Retained	Retained	Initialized
PIPE3TRN	H'0000	H'0000	Retained	Retained	Retained	Initialized
PIPE4TRE	Undefined	Undefined	Retained	Retained	Retained	Initialized
PIPE4TRN	H'0000	H'0000	Retained	Retained	Retained	Initialized
PIPE5TRE	Undefined	Undefined	Retained	Retained	Retained	Initialized
PIPE5TRN	H'0000	H'0000	Retained	Retained	Retained	Initialized
SUSPMODE	Undefined	Undefined	Retained	Retained	Retained	Initialized

Note: Initialized: The value is the one written in "Power-on Reset".

23C.2.1 System Configuration Control Register (SYSCFG0)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	HSE	—	—	DPRPU	—	—	—	USBE
Initial value:	—	—	—	—	—	—	—	—	0	—	—	0	—	—	—	0
R/W:	R	R	R	R	R	R	R	R	R/W	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	—	Undefined	R	Reserved The write value should always be 0.
7	HSE	0	R/W	High-Speed Operation Enable 0: High-speed operation is disabled (full-speed) 1: High-speed operation is enabled (detected by this module) When HSE = 1, this module executes the reset handshake protocol, and automatically performs high-speed or full-speed operation according to the protocol execution result. This bit should be modified while DPRPU is 0.
6, 5	—	Undefined	R	Reserved. The value should always be 0.
4	DPRPU	0	R/W	D+ Line Resistor Control Enables or disables pulling up D+ line. 0: Pulling up the line is disabled. 1: Pulling up the line is enabled. Setting this bit to 1 allows this module to pull up the D+ line to 3.3 V, thus notifying the USB host of connection. Modifying this bit from 1 to 0 allows this module to cancel pulling up the D+ line, thus notifying the USB host of disconnection.
3 to 1	—	Undefined	R	Reserved The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
0	USBE	0	R/W	USB Block Operation Enable Enables or disables operation of this module. 0: USB block operation is disabled. 1: USB block operation is enabled. Modifying this bit from 1 to 0 initializes some register bits as listed in table 23C.2. Write to this bit only while SuspendM = 1 and the UTMI clock is oscillating.

Note: This register can be written to even while the UTMI clock is stopped. In this case, note that the written value takes effect only after the UTMI clock restarts oscillation.

Table 23C.2 Register Bits Initialized by Writing USBE = 0

Register Name	Bit Name
SYSSTS0	LNST
DVSTCTR0	RHST
INTSTS0	DVSQ
USBADDR	USBADDR
USBREQ	bRequest, bmRequestType
USBVAL	wValue
USBINDX	wIndex
USBLENG	wLength

23C.2.2 CPU Bus Wait Setting Register (BUSWAIT)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	BWAIT			
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	1	1	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 4	—	Undefined	R	Reserved The write value should always be 0.
3 to 0	BWAIT	1111	R/W	CPU Bus Wait Specifies the number of wait cycles to be inserted during an access to this module. 0000: Setting prohibited 0001: One wait cycle (access cycles: 3) 0010 to 1110: Setting prohibited 1111: Fifteen wait cycles (access cycles: 17) (initial value) Note: Be sure to set this bit field to B'0001 during the initialization routine for this module.

23C.2.3 System Configuration Status Register (SYSSTS0)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LNST[1:0]	
Initial value:	0	—	0	0	0	1	0	0	0	0	0	0	0	0	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved The write value should always be 0.
14	—	Undefined	R	Reserved The write value should always be 0.
13 to 11	—	0	R	Reserved The write value should always be 0.
10	—	1	R	Reserved The write value should always be 1.
9 to 2	—	0	R	Reserved The write value should always be 0.
1, 0	LNST	00*	R	USB Data Line Status Monitor Indicates the status of the USB data bus lines (D+ and D-). Read the LNST bit after setting USBE = 1 and the attach processing (setting DPRPU = 1).

Note: * These bits are initialized to undefined values by a USB bus reset.

Table 23C.3 USB Data Bus Line Status

LNST[1]	LNST[0]	During Full-Speed Operation	During High-Speed Operation	During Chirp Operation
0	0	SE0	Squelch	Squelch
0	1	J state	Unsquelch	Chirp J
1	0	K state	Invalid	Chirp K
1	1	SE1	Invalid	Invalid

[Legend]

Chirp:	The reset handshake protocol is being executed in high-speed operation enabled state (the HSE bit in SYSCFG is set to 1).
Squelch:	SE0 or idle state
Unsquelch:	High-speed J state or high-speed K state
Chirp J:	Chirp J state
Chirp K:	Chirp K state

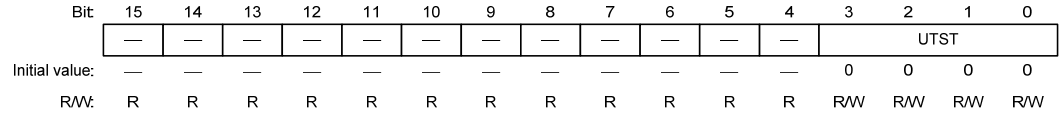
23C.2.4 Device State Control Register (DVSTCTR0)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	WKUP	—	—	—	—	—	RHST		
Initial value:	—	—	—	—	—	—	—	0	—	—	—	—	—	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 9	—	Undefined	R	Reserved The write value should always be 0.
8	WKUP	0	R/W	Wakeup Output Enables or disables outputting the remote wakeup signal (resume signal output) to the USB bus. 0: Remote wakeup signal is not output. 1: Remote wakeup signal is output. The module controls the output time of a remote wakeup signal to the USB bus. When this bit is set to 1, this module clears this bit to 0 after outputting the 10-ms K state. According to the USB specification, the USB bus idle state must be kept for 5 ms or longer before a remote wakeup signal is output. If this module writes 1 to this bit right after detection of suspended state, the K state will be output after 2 ms. Do not write 1 to this bit, unless the device state is in the suspended state (the DVSQ bit in the INTSTS0 register is set to B'1xx) and the USB host enables the remote wakeup signal. To set this bit to 1, do not stop the internal clock even in the suspended state (write 1 to this bit while the SUSPM bit in the SUSPMODE register is 1).

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	Undefined	R	Reserved The write value should always be 0.
2 to 0	RHST	000	R	Reset Handshake Indicates the status of the reset handshake. 000: Powered or disconnected state 010: Connected at full speed 011: Connected at high speed 100: Reset handshake in progress If HSE has been set to 1, these bits indicate B'100 as soon as this module detects the USB bus reset. Then, these bits indicate B'011 as soon as this module outputs Chirp-K and detects Chirp-JK from the USB host three times. If the connection speed is not fixed to high speed within 2.5 ms after Chirp-K output, these bits indicate B'010. If HSE has been set to 0, these bits indicate B'010 as soon as this module detects the USB bus reset. A DVST interrupt is generated as soon as this module detects the USB bus reset and then the value of the RHST bits is fixed to B'010 or B'011.

23C.2.5 Test Mode Register (TESTMODE)



Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	UTST			
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 4	—	Undefined	R	Reserved The write value should always be 0.
3 to 0	UTST	0000	R/W	Test mode This module outputs the USB test signal in high-speed mode according to the value set to these bits. 0000: Normal operation 0001: Test_J 0010: Test_K 0011: Test_SE0_NAK 0100: Test_Packet 0101 to 0111: Reserved These bits should be set according to the SetFeature request from the USB host during high-speed operation. This module will not shift to the suspended state while the setting of these bits is any value from B'0001 to B'0100.

23C.2.6 FIFO Port Registers (CFIFO, D0FIFO, D1FIFO)

The transfer buffer memory in this module is configured as FIFO (FIFO buffers). To access a FIFO buffer, use the corresponding FIFO port register. There are three FIFO ports: CFIFO, D0FIFO, and D1FIFO ports. Each FIFO port is configured of a port register (CFIFO, D0FIFO, or D1FIFO) that handles the reading of data from and the writing of data to the buffer memory, a select register (CFIFOSEL, D0FIFOSEL, or D1FIFOSEL) that is used to select the pipe assigned to the FIFO port, and a control register (CFIFOCTR, D0FIFOCTR, or D1FIFOCTR).

Each FIFO port has the following characteristics.

1. To access the FIFO buffer for the DCP, use the CFIFO port.
2. To access the FIFO buffer using DMA transfer, use the DnFIFO ports.
3. The DnFIFO ports can also be accessed by the CPU.
4. When using the functions specific to the FIFO ports, the pipe number set to the CURPIPE bits (selected pipe) cannot be changed (signal input/output of DMA-related pins, etc.).
5. The registers configuring a FIFO port do not affect other FIFO ports.
6. The same pipe should not be assigned to two or more FIFO ports.
7. There are two types of buffer state: the access right on the CPU side or on the SIE side. When the buffer memory access right is on the SIE side, the buffer cannot be accessed from the CPU.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FIFOPORT (Low)															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FIFOPORT (High)															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	FIFOPORT	H'00000000	R/W	FIFO Port Accessing these bits allow reading the received data from the FIFO buffer or writing the transmit data to the FIFO buffer. When these bits are accessed by software, this module accesses the FIFO buffer assigned to the pipe number set in the CURPIPE bits in the select register (CFIFOSEL, D0FIFOSEL, or D1FIFOSEL). These bits can be accessed only while the FRDY bit in the control register (CFIFOCTR, D0FIFOCTR, or D1FIFOCTR) is 1 (or while this module asserts the UCL_Dx_DREQ output signal). The valid bits in this register depend on the settings of the MBW bits and BIGEND bit as shown in tables 23C.4 to 23C.6.

Table 23C.4 Endian Operation in 32-Bit Access (CFIFOSEL.MBW =B'10)

BIGEND Bit	Bits 31 to 24	Bits 23 to 16	Bits 15 to 8	Bits 7 to 0
0	N + 3 address	N + 2 address	N + 1 address	N + 0 address
1	N + 0 address	N + 1 address	N + 2 address	N + 3 address

Table 23C.5 Endian Operation in 16-Bit Access (CFIFOSEL.MBW =B'01)

BIGEND Bit	Bits 31 to 24	Bits 23 to 16	Bits 15 to 8	Bits 7 to 0
0	Writing: invalid, reading: prohibited*		Odd address	Even address
1	Even address	Odd address	Writing: invalid, reading: prohibited*	

Note: * Reading a word or byte from an invalid register is prohibited.

Table 23C.6 Endian Operation in 8-Bit Access (CFIFOSEL.MBW =B'00)

BIGEND Bit	Bits 31 to 24	Bits 23 to 16	Bits 15 to 8	Bits 7 to 0
0	Writing: invalid, reading: prohibited*			Writing: valid, reading: valid
1	Writing: valid, reading: valid	Writing: invalid, reading: invalid*		

Note: * Reading a word or byte from an invalid register is prohibited.

23C.2.7 CFIFO Port Select Register (CFIFOSEL)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RCNT	REW	—	—	MBW		—	BIGEND	—	—	ISEL	—	CURPIPE			
Initial value:	0	0	—	—	0	0	—	0	—	—	0	—	0	0	0	0
R/W:	R/W	R/W	R	R	R/W	R/W	R	R/W	R	R	R/W	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	RCNT	0	R/W	Read Count Mode Specifies the read mode for the value in the DTLN bits in CFIFOCTR. 0: The DTLN bit is cleared when all of the receive data has been read from the CFIFO. 1: The DTLN bit is decremented when the receive data is read from the CFIFO. When this bit is cleared to 0, this module clears the DTLN bits in CFIFOCTR to 0 when all of the receive data has been read from the FIFO buffer that is assigned to the pipe specified in the CURPIPE bits (called the specified pipe). When this bit is set to 1, this module decrements the DTLN bits in CFIFOCTR every time receive data is read from the FIFO buffer that is assigned to the specified pipe.

Bit	Bit Name	Initial Value	R/W	Description
14	REW	0	R/W	Buffer Pointer Rewind Specifies whether or not to rewind the buffer pointer. 0: The buffer pointer is not rewound. 1: The buffer pointer is rewound. When the selected pipe is in the receiving direction, setting this bit to 1 while the FIFO buffer is being read allows re-reading the FIFO buffer from the first data (in double buffer mode, re-reading the currently-read one FIFO buffer plane from the first data is allowed). Do not set REW to 1 simultaneously with modifying the CURPIPE bits. Before setting REW to 1, be sure to check that FRDY is 1. To re-write to the FIFO buffer again from the first data for the pipe in the transmitting direction, use the BCLR bit.
13, 12	—	Undefined	R	Reserved The write value should always be 0.
11, 10	MBW	00	R/W	CFIFO Port Access Bit Width Specifies the bit width for accessing the CFIFO port. 00: 8-bit width 01: 16-bit width 10: 32-bit width 11: Setting prohibited When the selected pipe is in the receiving direction, once reading data is started after setting these bits, these bits should not be modified until all the data has been read. When the selected pipe is in the receiving direction, set the CURPIPE and MBW bits simultaneously. When the selected pipe is in the transmitting direction, the bit width cannot be changed from the 8-bit width to the 16-/32-bit width or from the 16-bit width to the 32-bit width while data is being written to the buffer memory. The odd number of bytes can also be written through byte-access control even when 16- or 32-bit width is selected.

Bit	Bit Name	Initial Value	R/W	Description
9	—	Undefined	R	Reserved The write value should always be 0.
8	BIGEND	0	R/W	CFIFO Port Endian Control Specifies the byte endian for the CFIFO port. Refer to section 23C.2.6, FIFO Port Registers (CFIFO, D0FIFO, D1FIFO). 0: Little endian 1: Big endian
7, 6	—	Undefined	R	Reserved The write value should always be 0.
5	ISEL	0	R/W	CFIFO Port Access Direction When DCP is Selected Specifies the direction of FIFO port access when the DCP is selected through the CURPIPE bits. 0: Reading from the buffer memory is selected 1: Writing to the buffer memory is selected To modify this bit when the specified pipe is DCP, write to this bit and then read it to check that the read value matches the written value before moving to the next processing. When this bit is modified during access to the FIFO buffer, the access results up to that point can be retained, and after the bit is restored to the previous value, access can be continued. Set this bit and the CURPIPE bits simultaneously.
4	—	Undefined	R	Reserved The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	CURPIPE	0000	R/W	FIFO Port Access Pipe Specification Specifies the pipe number using which data is read or written through the CFIFO port. 0000: DCP 0001: PIPE1 0010: PIPE2 : : 1000: PIPE8 1001: PIPE9 To modify these bits, write to this bit and then read it to check that the read value matches the written value before moving to the next processing. Do not specify the same pipe in the CURPIPE bits in CFIFOSEL, D0FIFOSEL, and D1FIFOSEL. Even if an attempt is made to modify the setting of these bits during access to the FIFO buffer, the current access setting is retained until the access is completed. Then, the modification becomes effective thus enabling continuous access.

23C.2.8 DnFIFO Port Select Register (D0FIFOSEL, D1FIFOSEL)

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RCNT	REW	DCLRM	DREQE	MBW		—	BIG END	—	—	—	—	CURPIPE			
Initial value :	0	0	0	0	0	0	—	0	—	—	—	—	0	0	0	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	RCNT	0	R/W	Read Count Mode Specifies the read mode for the value in the DTLN bits in Dx_FIFOCTR. 0: The DTLN bit is cleared when all of the receive data has been read from the DnFIFO. 1: The DTLN bit is decremented when the receive data is read from the DnFIFO. When this bit is cleared to 0, this module clears the DTLN bits in DnFIFOCTR to 0 when all of the receive data has been read from the one FIFO buffer plane that is assigned to the pipe specified in the CURPIPE bits (called the specified pipe). When this bit is set to 1, this module decrements the DTLN bits in DnFIFOCTR every time receive data is read from the FIFO buffer that is assigned to the specified pipe. When accessing DnFIFO with the BFRE bit set to 1, set this bit to 0.

Bit	Bit Name	Initial Value	R/W	Description
14	REW	0	R/W	Buffer Pointer Rewind Specifies whether or not to rewind the buffer pointer. 0: The buffer pointer is not rewound. 1: The buffer pointer is rewound. When the selected pipe is in the receiving direction, setting this bit to 1 while the FIFO buffer is being read allows re-reading the FIFO buffer from the first data (in double buffer mode, re-reading the currently-read FIFO buffer plane from the first data is allowed). Do not set REW to 1 simultaneously with modifying the CURPIPE bits. Before setting REW to 1, be sure to check that FRDY is 1. When accessing DnFIFO with the BFRE bit set to 1, do not set this bit to 1 in the state in which the short packet data has been read out. To re-write to the FIFO buffer again from the first data for the pipe in the transmitting direction, use the BCLR bit.
13	DCLRM	0	R/W	Auto Buffer Memory Clear Mode Accessed after Specified Pipe Data is Read Enables or disables the buffer memory to be cleared automatically after data has been read out using the selected pipe. 0: Auto buffer clear mode is disabled. 1: Auto buffer clear mode is enabled. With this bit set to 1, this module sets BCLR to 1 for the FIFO buffer of the selected pipe on receiving a zero-length packet while the FIFO buffer assigned to the selected pipe is empty, or on receiving a short packet and reading the data while BFRE is 1. When using this module with the BRDYM bit set to 1, set this bit to 0.

Bit	Bit Name	Initial Value	R/W	Description
12	DREQE	0	R/W	DMA Transfer Request Enable Enable or disable the DMA transfer request. 0: DMA transfer request disabled 1: DMA transfer request enabled To enable the DMA transfer request, set the CURPIPE bits and then set this bit to 1. Before modifying the CURPIPE bits, be sure to clear this bit to 0.
11, 10	MBW	00	R/W	FIFO Port Access Bit Width Specifies the bit width for accessing the DnFIFO port. Refer to section 23C.2.6, FIFO Port Registers (CFIFO, D0FIFO, D1FIFO) for details. 00: 8-bit width 01: 16-bit width 10: 32-bit width 11: Setting prohibited
9	—	Undefined	R	Reserved This bit is always read as 0. The write value should always be 0.
8	BIGEND	0	R/W	FIFO Port Endian Control Specifies the byte endian for the FIFO port. The value should be 0. Refer to section 23C.2.6, FIFO Port Registers (CFIFO, D0FIFO, D1FIFO) for details. 0: Little endian 1: Big endian
7 to 4	—	Undefined	R	Reserved The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	CURPIPE	0000	R/W	FIFO Port Access Pipe Specification Specify a desired pipe number for which data is read or written through the DnFIFO port. 0000: No pipe specified 0001: PIPE1 0010: PIPE2 : : 1000: PIPE8 1001: PIPE9 After writing to these bits, read these bits to check that the written value agrees with the read value before proceeding to the next process. Do not set the same pipe to the CURPIPE bits in CFIFOSEL, D0FIFOSEL, and D1FIFOSEL. Even if an attempt is made to modify the setting of these bits during access to the FIFO buffer, the current access setting is retained until the access is completed. Then, the modification becomes effective thus enabling continuous access.

23C.2.9 CFIFO, DnFIFO Port Control Registers (CFIFOCTR, D0FIFOCTR, D1FIFOCTR)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BVAL	BCLR	FRDY	—	DTLN											
Initial value:	0	0	0	—	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	BVAL	0	R/W	Buffer Memory Valid Flag Set this bit to 1 when writing has completed in the CPU-side FIFO buffer for the pipe specified in CURPIPE (called the specified pipe). 0: Invalid 1: Writing ended When the pipe specified through the CURPIPE bits (called the specified pipe) is set to the transmit direction, set this bit to 1 under any of the following conditions. After this bit is set to 1, this module switches the CPU-side FIFO buffer to the SIE side and makes it ready for transmission. • To transmit a short packet, set this bit to 1 after data has been written. • To transmit a zero-length packet, set this bit to 1 before data is written to the FIFO buffer. • Set this bit to 1 after the number of data bytes has been written for the pipe in discontinuous transfer mode, where the number is a natural integer multiple of the maximum packet size and less than the buffer size. When the data of the maximum packet size has been written for the pipe in continuous transfer mode, this module sets this bit to 1 and switches the FIFO buffer from the CPU side to the SIE side, enabling transmission. Be sure to write 1 to this bit only while FRDY = 1. Do not write 1 to this bit when the specified pipe is in the receive direction.

Bit	Bit Name	Initial Value	R/W	Description
14	BCLR	0	R/W	CPU Buffer Clear This bit should be set to 1 to clear the FIFO buffer on the CPU side for the corresponding pipe. 0: Invalid 1: Clears the buffer memory on the CPU side. When double buffer mode is set for the FIFO buffer assigned to the specified pipe, this module clears only one plane of the FIFO buffer even when both planes are read-enabled. When the specified pipe is the DCP, setting BCLR to 1 allows this module to clear the FIFO buffer regardless of whether the FIFO buffer is on the CPU side or SIE side. To clear the buffer on the SIE side, set the PID bits for the DCP to NAK before setting BCLR to 1. When the selected pipe is in the transmitting direction, if 1 is written to BVAL and BCLR bits simultaneously, this module clears the data that has been written before it, enabling transmission of a zero-length packet. When the specified pipe is not DCP, be sure to write 1 to this bit only while FRDY = 1.
13	FRDY	0	R	FIFO Port Ready Indicates whether the FIFO port can be accessed by the CPU (DMAC). 0: FIFO port access is disabled. 1: FIFO port access is enabled. In the following cases, this module sets FRDY to 1 but data cannot be read via the FIFO port because there is no data to be read. In these cases, set BCLR to 1 to clear the FIFO buffer, and enable transmission and reception of the next data. • A zero-length packet is received when the FIFO buffer assigned to the specified pipe is empty. • A short packet is received and the data is completely read while BFRE is 1.

Bit	Bit Name	Initial Value	R/W	Description
12	—	Undefined	R	Reserved The write value should always be 0.
11 to 0	DTLN	H'000	R	Receive Data Length Indicate the length of received data. During FIFO buffer reading, the value of these bits depends on the RCNT bit setting as follows. • When RCNT = 0: Indicate the receive data length until the CPU (DMAC) has completed reading all receive data from one FIFO buffer. While BFRE = 1, this module retains the receive data length until BCLR = 1 even after reading has been completed. • When RCNT = 1: This module decrements the value in the DTLN bits every time data is read. (Decrements by 1 when MBW = 0 or by 2 when MBW = 1) When all data has been read from one FIFO buffer, this module clears the DTLN bits to 0. In the double-buffer structure, if data reading has been completed in one FIFO buffer before completion in another FIFO buffer, the DTLN bits indicate the receive data length for the latter FIFO buffer when reading has been completed in the former FIFO buffer. When reading these bits during FIFO buffer reading while RCNT = 1, note that these bits are updated within 150 ns after a read cycle for the FIFO port.

23C.2.10 Interrupts Enable Register 0 (INTENB0)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VBSE	RSME	SOFE	DVSE	CTRE	BEMPE	NRDYE	BRDYE	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	VBSE	0	R/W	VBUS Interrupt Enable Enables or disables a USB interrupt request when a VBINT interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled
14	RSME	0	R/W	Resume Interrupt Enable Enables or disables a USB interrupt request when an RESM interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled
13	SOFE	0	R/W	Frame Number Update Interrupt Enable Enables or disables a USB interrupt request when an SOF interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled
12	DVSE	0	R/W	Device State Transition Interrupt Enable Enables or disables a USB interrupt request when a DVST interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled
11	CTRE	0	R/W	Control Transfer Stage Transition Interrupt Enable Enables or disables a USB interrupt request when a CTRT interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled

Bit	Bit Name	Initial Value	R/W	Description
10	BEMPE	0	R/W	Buffer Empty Interrupt Enable Enables or disables a USB interrupt request when an BEMP interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled
9	NRDYE	0	R/W	Buffer Not Ready Response Interrupts Enable Enables or disables a USB interrupt request when an NRDY interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled
8	BRDYE	0	R/W	Buffer Ready Interrupts Enable Enables or disables a USB interrupt request when an BRDY interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled
7 to 0	—	Undefined	R	Reserved The write value should always be 0.

23C.2.11 BRDY Interrupt Enable Register (BRDYENB)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PIPEBRDYE									
Initial value:	—	—	—	—	—	—	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	Undefined	R	Reserved The write value should always be 0.
9 to 0	PIPEBRDYE	H'000	R/W	BRDY interrupt Enable for Each Pipe Specify whether to set the BRDY bit to 1 when a BRDY interrupt is detected in each pipe. 0: Interrupt output disabled 1: Interrupt output enabled Note: Each bit number corresponds to a pipe number. When this module detects a BRDY interrupt for the pipe corresponding to the PIPEBRDYE bit set to 1 by software, this module sets the corresponding PIPEBRDY bit in BRDYSTS to 1, sets the BRDY bit in INTSTS0 to 1, and requests an interrupt. While at least one of the PIPEBRDY bits in BRDYSTS is set to 1, if the corresponding PIPEBRDYE bit is changed from 0 to 1 by software, this module requests an interrupt.

23C.2.12 NRDY Interrupt Enable Register (NRDYENB)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PIPENRDYE									
Initial value:	—	—	—	—	—	—	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	Undefined	R	Reserved The write value should always be 0.
9 to 0	PIPENRDYE	H'000	R/W	NRDY Interrupt Enable for Each Pipe 0: Interrupt output disabled 1: Interrupt output enabled Note: Each bit number corresponds to a pipe number. When this module detects an NRDY interrupt for the pipe corresponding to the PIPENRDYE bit set to 1 by software, this module sets the corresponding PIPENRDY bit in NRDYSTS to 1, sets the NRDY bit in INTSTS0 to 1, and requests an interrupt. While at least one of the PIPENRDY bits in NRDYSTS is set to 1, if the corresponding PIPENRDYE bit is changed from 0 to 1 by software, this module requests an interrupt.

23C.2.13 BEMP Interrupt Enable Register (BEMPENB)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PIPEBEMPE									
Initial value:	—	—	—	—	—	—	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	Undefined	R	Reserved The write value should always be 0.
9 to 0	PIPEBEMPE	H'000	R/W	BEMP Interrupt Enable for Each Pipe 0: Interrupt output disabled 1: Interrupt output enabled Note: Each bit number corresponds to a pipe number. When this module detects a BEMP interrupt for the pipe corresponding to the PIPEBEMPE bit set to 1 by software, this module sets the corresponding PIPEBEMP bit in BEMPSTS to 1, sets the BEMP bit in INTSTS0 to 1, and requests an interrupt. While at least one of the PIPEBEMP bits in BEMPSTS is set to 1, if the corresponding PIPEBEMPE bit is changed from 0 to 1 by software, this module requests an interrupt.

23C.2.14 SOF Pin Configuration Register (SOFCFG)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	BRDYM	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	0	0	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 7	—	Undefined	R	Reserved The write value should always be 0.
6	BRDYM	0	R/W	BRDY Interrupt Status Clear Timing Setting Specifies the timing for clearing the BRDY interrupt status. 0: Software clears the status. 1: This module clears the status when data has been read from the FIFO buffer or data has been written to the FIFO buffer.
5	—	0	R/W	Reserved When writing, write back the value read out from this bit immediately beforehand. Note: The initial value immediately after a power-on reset is 0. Be sure to set this bit to 1 during the initialization routine for this module.
4 to 0	—	Undefined	R	Reserved The write value should always be 0.

23C.2.15 Interrupt Status Register 0 (INTSTS0)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VBINT	RESM	SOFR	DVST	CTRT	BEMP	NRDY	BRDY	VBSTS	DVSQ			VALID	CTSQ		
Initial value:	0	0	0	0	0	0	0	0	—	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	VBINT	0	R/W	VBUS Change Detection Interrupt Status Indicates the VBUS change detection interrupt status. 0: VBUS interrupts not generated 1: VBUS interrupts generated This module sets this bit to 1 when a change in the value input to the VBUS pin (high to low or low to high) is detected. The module indicates the VBUS pin input value in the VBSTS bit. When a VBINT interrupt is generated, read the VBSTS bits several times by software to remove the chattering effect until the same value is read repeatedly from the bit.
14	RESM	0	R/W	Resume Interrupt Status Indicates the resume detection interrupt status. 0: Resume interrupts not generated 1: Resume interrupts generated This module sets this bit to 1 on detecting the falling edge of the signal on the DP pin in the suspended state (DVSQ = B'1xx).

Bit	Bit Name	Initial Value	R/W	Description
13	SOFR	0	R/W	Frame Number Refresh Interrupt Status Indicates the frame number refresh interrupt status. 0: SOF interrupts not generated 1: SOF interrupts generated This module sets this bit to 1 under the following condition. This module sets this bit to 1 on updating the frame number. (This interrupt is detected every 1 ms.) This module can detect an SOFR interrupt through the internal interpolation function even when a damaged SOF packet is received from the USB host.
12	DVST	0*[1]	R/W	Device State Transition Interrupt Status Indicates the device state transition interrupt status. 0: Device state transition interrupts not generated 1: Device state transition interrupts generated This module updates the DVSQ value and sets this bit to 1 on detecting a change in the device state. When this interrupt is generated, clear the status before this module detects the next device state transition.

Bit	Bit Name	Initial Value	R/W	Description
11	CTRT	0	R/W	Control Transfer Stage Transition Interrupt Status*2*4 Indicates the control transfer status. 0: Control transfer stage transition interrupts not generated 1: Control transfer stage transition interrupts generated This module updates the CTSQ value and sets this bit to 1 on detecting a change in the control transfer stage. When this interrupt is generated, clear the status before this module detects the next control transfer stage transition.
10	BEMP	0	R	Buffer Empty Interrupt Status Indicates the BEMP interrupt status. 0: BEMP interrupts not generated 1: BEMP interrupts generated This module sets this bit to 1 when at least one PIPEBEMP bit in BEMPSTS is set to 1 among the PIPEBEMP bits corresponding to the PIPEBEMPE bits in BEMPENB to which 1 has been set (when this module detects the BEMP interrupt status in at least one pipe among the pipes for which software enables the BEMP interrupt output). For the conditions for PIPEBEMP status assertion, refer to PIPEBEMP register. This module clears this bit to 0 when software writes 0 to all the PIPEBEMP bits corresponding to the PIPEBEMPE bits to which 0 has been set. This bit cannot be cleared to 0 even if software writes 0 to this bit.

Bit	Bit Name	Initial Value	R/W	Description
9	NRDY	0	R	Buffer Not Ready Interrupt Status Indicates the NRDY interrupt status. 0: NRDY interrupts not generated 1: NRDY interrupts generated This module sets this bit to 1 when at least one PIPENRDY bit in BNRDYSTS is set to 1 among the PIPENRDY bits corresponding to the PIPENRDYE bits in NRDYENB to which 1 has been set (when this module detects the NRDY interrupt status in at least one pipe among the pipes for which software enables the NRDY interrupt output). For the conditions for PIPENRDY status assertion, refer to PIPENDRY register. This module clears this bit to 0 when software writes 0 to all the PIPENRDY bits corresponding to the PIPENRDYE bits to which 0 has been set. This bit cannot be cleared to 0 even if software writes 0 to this bit.
8	BRDY	0	R	Buffer Ready Interrupt Status Indicates the BRDY interrupt status. 0: BRDY interrupts not generated 1: BRDY interrupts generated This module sets this bit to 1 when at least one PIPEBRDY bit in BRDYSTS is set to 1 among the PIPEBRDY bits corresponding to the PIPEBRDYE bits in BRDYENB to which 1 has been set (when this module detects the BRDY interrupt status in at least one pipe among the pipes for which software enables the BRDY interrupt output). For the conditions for PIPEBRDY status assertion, refer to PIPEBRDY register. This module clears this bit to 0 when software writes 0 to all the PIPEBRDY bits corresponding to the PIPEBRDYE bits to which 0 has been set. This bit cannot be cleared to 0 even if software writes 0 to this bit.

Bit	Bit Name	Initial Value	R/W	Description
7	VBSTS	Undefined	R	VBUS Input Status Indicates the VBUS pin input status. Set the OVC1 bit in the USBPCTRL0 register to 1 to enable this bit. 0: The VBUS pin is low level. 1: The VBUS pin is high level.
6 to 4	DVSQ	000*2	R	Device State Indicate the device state. 000: Powered state 001: Default state 010: Address state 011: Configured state 1xx: Suspended state
3	VALID	0	R/W	USB Request Reception Indicates whether USB request reception has been detected. 0: Not detected 1: Setup packet reception
2 to 0	CTSQ	000	R	Control Transfer Stage Indicate the control transfer stage. 000: Idle or setup stage 001: Control read data stage 010: Control read status stage 011: Control write data stage 100: Control write status stage 101: Control write (no data) status stage 110: Control transfer sequence error 111: Reserved

Note: To clear the status indicated by the VBINT, RESM, SOFR, DVST, or CTRT bit, write 0 only to the bit to be cleared and write 1 to the other bits. Do not write 0 to a status bit that indicates 0.

This module detects any status change indicated by the VBINT and RESM bits of this register even while the clock is stopped and notifies an interrupt if the interrupt is enabled. Clear the status by software only after the clock is enabled.

1. This bit is initialized to 1 by a USB bus reset.
2. These bits are initialized to 001 by a USB bus reset.

23C.2.16 BRDY Interrupt Status Register (BRDYSTS)

Upon detecting a BRDY interrupt for a pipe, this module sets the corresponding PIPEBRDY bit in BRDYSTS to 1. At this time, if the corresponding bit in BRDYENB has been set to 1 by software, this module sets the BRDY bit in INTSTS0 to 1 and requests an interrupt.

The set conditions and clear method of the BRDY interrupt depend on the settings of the BRDYM bit and BFRE bit for each pipe.

When BRDYM = 0 and BFRE=0:

With this setting, the BRDY interrupt shows that the FIFO port has become ready for access. This module generates an internal BRDY interrupt request trigger under any of the following conditions and sets the PIPEBRDY bit corresponding to the trigger-generated pipe to 1.

1. For a pipe set to transmission
   a. When the DIR bit is changed from 0 to 1 by software
   b. When this module completes packet transmission in the target pipe while the FIFO assigned to the pipe is not ready for write access by the CPU (the BSTS bit is read as 0).

      In the continuous transfer mode, a request trigger is generated when data of a single FIFO buffer has been transmitted.
   c. In the double-buffer structure, when one FIFO buffer is empty when writing to the other FIFO buffer is completed

      Even if transmission from a buffer is completed during writing to the other buffer, no request trigger is generated until the current writing is completed.
   d. In a pipe whose transfer type is set to isochronous transfer, when the buffer is flushed by hardware
   e. When the FIFO buffer is shifted from the write-not-ready state to write-ready state by writing 1 to the ACLRM bit

   For the DCP (that is, in the data transmission in the control transfer), no request trigger is generated.

2. For a pipe set to reception
   a. When packet reception is completed successfully and the FIFO buffer becomes ready to be read while the FIFO buffer assigned to the pipe is not ready for read access by the CPU (the BSTS bit is read as 0)

      No request trigger is generated for a transaction in which a data PID mismatch is detected.

      In the continuous transfer mode, no request is generated when the data size is MaxPacketSize and the buffer still has empty space.

      When a short packet is received, a request trigger is generated even if the FIFO buffer has empty space.

      When the transaction counter is used, a request trigger is generated when the specified number of packets have been received. In this case, a request trigger is generated even if the FIFO buffer has empty space.
   b. In the double-buffer structure, when one FIFO buffer is ready to be read when reading from the other FIFO buffer is completed

      Even if reception in a buffer is completed during reading from the other buffer, no request trigger is generated until the current reading is completed.

This interrupt does not occur in communication in the status stage of the control transfer.

The PIPEBRDY interrupt state for a pipe can be cleared to 0 by writing 0 to the PIPEBRDY bit corresponding to the pipe. At this time, write 1 to the bits corresponding to the other pipes.
Be sure to clear this interrupt state before accessing the corresponding FIFO buffer.

When BRDYM = 0 and BFRE = 1:

With this setting, when all data for a transfer has been read from a pipe, this module detects a BRDY interrupt and sets the corresponding PIPEBRDY bit to 1.

This module detects reception of the last data for a transfer under either of the following conditions.

1. When a short packet (including a zero-length packet) has been received
2. When the transaction counter (TRNCNT bits) is used and packets have been received for the number specified in the TRNCNT bits

When either of the above conditions is satisfied and reading of the data is completed, this module detects that all data for a transfer has been read.

If a zero-length packet is received while the FIFO buffer is empty, this module detects that all data for a transfer has been read when the zero-length packet data is toggled to the CPU side. In this case, before starting the next transfer, write 1 to the BCLR bit in FIFOCTR by software.

With this setting, this module does not detect BRDY interrupts in the transmit pipe.

The PIPEBRDY interrupt state for a pipe can be cleared to 0 by writing 0 to the PIPEBRDY bit corresponding to the pipe. At this time, write 1 to the bits corresponding to the other pipes.

When using this mode, do not change the BFRE bit setting until the processing for the transfer is completed. To change the BFRE bit setting during processing, clear all FIFO buffers for the corresponding pipe through the ACLRM bit setting.

When BRDYM = 1 and BFRE = 0:

With this setting, each bit value changes according to the BSTS bit status for each pipe; that is, this module sets the BRDY interrupt status to 1 or 0 according to the FIFO buffer state.

1. For a pipe set to transmission

   The PIPEBRDY bit is set to 1 when the FIFO port is ready to be written to and is cleared to 0 when the port becomes not ready to be written to.

   Note that the BRDY interrupt signal is not asserted when the transmit pipe for the DCP is ready to be written to.

2. For a pipe set to reception

   The PIPEBRDY bit is set to 1 when the FIFO port is ready to be read and is cleared to 0 when reading of all data is completed (the port becomes not ready to be read).

   If a zero-length packet is received while the FIFO buffer is empty, the corresponding PIPEBRDY bit indicates 1 and the BRDY interrupt signal continues to be asserted until BCLR = 1 is set by software.

With this setting, the PIPEBRDY bits cannot be cleared to 0 by software.

When setting BRDYM = 1, be sure to clear all BFRE bits (all pipes) to 0.
When setting BRDYM = 1, be sure to set the INTL bit to 1 (level-trigger).

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PIPEBRDY									
Initial value:	—	—	—	—	—	—	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	Undefined	R	Reserved The write value should always be 0.
9 to 0	PIPEBRDY	H'000	R/W	BRDY Interrupt Status for individual PIPE Indicate the BRDY interrupt status in each pipe. 0: Interrupts not generated 1: Interrupts generated

Notes: 1. Each bit number corresponds to a pipe number.
2. When BRDYM is 0, to clear the status indicated by the bits in this register, write 0 only to the bits to be cleared; write 1 to the other bits.
3. When BRDYM is 0, clearing this bit should be done before accessing the FIFO.

23C.2.17 NRDY Interrupt Status Register (NRDYSTS)

Upon generating an NRDY interrupt for the pipe set to PID = BUF by software, this module sets the corresponding PIPENRDY bit in NRDYSTS to 1. At this time, if the corresponding bit in NRDYENB has been set to 1 by software, this module sets the NRDY bit in INTSTS0 to 1 and requests an interrupt.

This module generates an internal NRDY interrupt request for a pipe under any of the following conditions.

No interrupt request is generated in the status data stage of the control transfer.

1. For a pipe set to transmission
   a. When an IN token is received while the FIFO buffer has no transmit data

      Upon receiving an IN token, this module generates an NRDY interrupt request and sets the PIPENRDY bit to 1.

      In a pipe whose transfer type is set to isochronous transfer, this module transmits a zero-length packet and sets the OVRN bit to 1.

2. For a pipe set to reception
   a. When an OUT token is received while the FIFO buffer has no empty space

      In a pipe whose transfer type is set to isochronous transfer, when an OUT token is received, this module generates an NRDY interrupt request, sets the PIPENRDY bit to 1, and sets the OVRN bit to 1.

      In a pipe whose transfer type is not the isochronous transfer, this module generates an NRDY interrupt request and sets the PIPENRDY bit to 1 upon transmitting the NAK Handshake after receiving the data subsequent to the OUT token.

      Note that no NRDY interrupt request is generated for retransmission (when a DATA-PID mismatch is detected). No NRDY interrupt request is generated when an error is found in the data packet.

   b. When a PING token is received while the FIFO buffer has no empty space

      Upon receiving a PING token, this module generates an NRDY interrupt request and sets the PIPENRDY bit to 1.

   c. In a pipe whose transfer type is set to isochronous transfer, when reception is not completed successfully in the interval frame

      This module generates an NRDY interrupt request and sets the PIPENRDY bit to 1 with the SOF reception timing.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PIPENRDY									
Initial value:	—	—	—	—	—	—	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	Undefined	R	Reserved The write value should always be 0.
9 to 0	PIPENRDY	H'000	R/W	NRDY Interrupt Status for Each Pipe Indicate the NRDY interrupt status in each pipe. 0: Interrupts not generated 1: Interrupts generated

Notes: 1. Each bit number corresponds to a pipe number.
2. To clear the status indicated by each bit of this register, write 0 only to the bit to be cleared and write 1 to the other bits.

23C.2.18 BEMP Interrupt Status Register (BEMPSTS)

Upon detecting a BEMP interrupt in the pipe set to PID = BUF by software, this module sets the corresponding PIPEBEMP bit in BEMPSTS to 1. At this time, if the corresponding bit in BEMPENB has been set to 1 by software, this module sets the BEMP bit in INTSTS0 to 1 and requests an interrupt.

This module generates an internal BEMP interrupt request under any of the following conditions.

1. For a pipe set to transmission, when the FIFO buffer for the pipe is empty when transmission is completed (including transmission of a zero-length packet)

   In the single-buffer structure, this module generates an internal BEMP interrupt request for a pipe that is not DCP at the same time with the BRDY interrupt.

   Note that no internal BEMP interrupt is generated in the following cases.

   a. In the double-buffer structure, when software (DMAC) has started writing to the FIFO buffer on the CPU side before data transmission for one FIFO buffer is completed
   b. When the buffer is cleared (emptied) by writing 1 to the ACLRM or BCLR bit
   c. In the IN transfer (zero-length packet transmission) in the status stage of the control transfer

2. For a pipe set to reception

   When an amount of data larger than the MaxPacketSize setting is received successfully

   In this case, this module generates a BEMP interrupt request, sets the corresponding PIPEBEMP bit to 1, discards the received data, and sets the PID bits for the corresponding pipe to STALL (B'11).

   At this time, this module sends a STALL response.

   Note that no internal BEMP interrupt is generated in the following cases.

   a. When a CRC error or bit staffing error is found in the received data
   b. In SETUP transaction execution

Writing 0 to a PIPEBEMP bit clears the corresponding status.
Writing 1 to a PIPEBEMP bit is ignored.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PIPEBEMP									
Initial value:	—	—	—	—	—	—	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	Undefined	R	Reserved The write value should always be 0.
9 to 0	PIPEBEMP	H'000	R/W	BEMP Interrupts for Each Pipe Indicate the BEMP interrupt status in each pipe. 0: Interrupts not generated 1: Interrupts generated

Notes: 1. Each bit number corresponds to a pipe number.
2. To clear the status indicated by each bit of this register, write 0 only to the bit to be cleared and write 1 to the other bits.

23C.2.19 Frame Number Register (FRMNUM)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OVRN	CRCE	—	—	—	FRNM										
Initial value:	0	0	—	—	—	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	OVRN*	0	R/W	Overrun/Underrun Detection Status Indicates whether an overrun/underrun error has been detected in the pipe during isochronous transfer. 0: No error 1: An error occurred When an overrun/underrun error is detected, this module generates an NRDY interrupt request. For details, see section 23C.2.17, NRDY Interrupt Status Register (NRDYSTS). Software can clear this bit to 0 by writing 0 to the bit. Here, 1 should be written to the other bits in this register. This module sets this bit to 1 on any of the following conditions. • For the isochronous transfer pipe in the transmitting direction, the time to issue an IN token comes before all the transmit data has been written to the FIFO buffer. • For the isochronous transfer pipe in the receiving direction, the time to issue an OUT token comes when no FIFO buffer planes are empty.

Bit	Bit Name	Initial Value	R/W	Description
14	CRCE	0	R/W	CRC Error Detection Status Indicates whether a CRC error was detected in the pipe set to isochronous transfer. 0: No error 1: An error occurred This bit is set to 1 when this module detects a CRC error or a bit staffing error in a pipe whose transfer type is set to isochronous transfer. Software can clear this bit to 0 by writing 0 to the bit. Here, 1 should be written to the other bits in this register. On detecting a CRC error, this module does not generate the internal NRDY interrupt request.
13 to 11	—	Undefined	R	Reserved The write value should always be 0.
10 to 0	FRNM	H'000	R	Frame Number Indicate the latest frame number. This module updates these bits to the latest frame number with the SOF output timing at 1-ms intervals or when an SOF is received. When reading these bits by software, repeat reading until the same value is read at least twice.

Note: The OVRN bit should be used for debugging. When designing a system, control the timing so that neither overrun nor underrun occurs.

23C.2.20 µFrame Number Register (UFRMNUM)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	UFRNM		
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 3	—	Undefined	R	Reserved The write value should always be 0.
2 to 0	UFRNM	000	R	µFrame Indicate the µframe number In high-speed communication, these bits indicate the µframe number. When communication is not in the high-speed mode, these bits are set to H'0. When reading these bits by software, repeat reading until the same value is read at least twice.

23C.2.21 USB Address Register (USBADDR)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	USBADDR						
Initial value:	—	—	—	—	—	—	—	—	—	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 7	—	Undefined	R	Reserved The write value should always be 0.
6 to 0	USBADDR	H'00*	R	USB Address Indicate the USB address assigned by the host. Upon completing the SetAddress request processing successfully, this module indicates the received USB address in these bits. This module indicates H'00 in these bits upon detecting a USB reset.

Note: These bits are initialized to H'00 by a USB bus reset.

23C.2.22 USB Request Type Register (USBREQ)

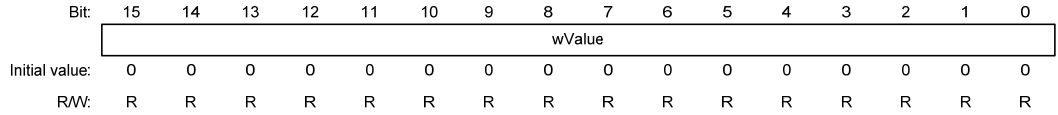
The USB request registers store setup requests for control transfers; that is, the values of the USB requests that have been received.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	bRequest								bmRequestType							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	bRequest	H'00*	R	Request These bits store the USB request bRequest value. Indicates the USB request data value received during the setup transaction. Writing to these bits is invalid.
7 to 0	bmRequestType	H'00*	R	Request Type These bits store the USB request bmRequestType value. Indicates the USB request type value received during the setup transaction. Writing to these bits is invalid.

Note: These bits are initialized to H'00 by a USB bus reset.

23C.2.23 USB Request Value Register (USBVAL)

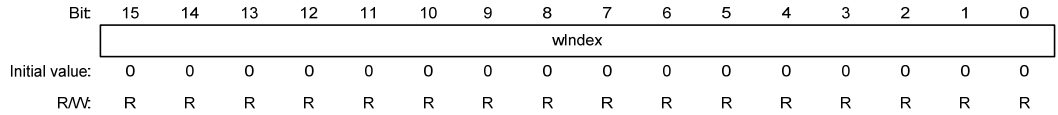


Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	wValue															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	wValue	H'0000*	R	Value These bits are used to write or read the value of USB request wValue. Bits 7 to 0 configure a lower byte. These bits indicate the value of USB request wValue that this module has received in the SETUP transaction. Writing to these bits by software is invalid.

Note: These bits are initialized to H'0000 by a USB bus reset.

23C.2.24 USB Request Index Register (USBINDX)



Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	wIndex															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	wIndex	H'0000*	R/W	Index These bits are used to write or read the value of USB request wIndex. Bits 7 to 0 configure a lower byte. These bits indicate the value of USB request wIndex that this module has received in the SETUP transaction. Writing to these bits by software is invalid.

Note: These bits are initialized to H'0000 by a USB bus reset.

23C.2.25 USB Request Length Register (USBLENG)

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	wLength	H'0000*	R	Length These bits are used to write or read the value of USB request wLength. Bits 7 to 0 configure a lower byte. These bits indicate the value of USB request wLength that this module has received in the SETUP transaction. Writing to these bits by software is invalid.

Note: These bits are initialized to H'0000 by a USB bus reset.

23C.2.26 DCP Maximum Packet Size Register (DCPMAXP)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	MXPS						
Initial value:	—	—	—	—	—	—	—	—	—	1	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 7	—	Undefined	R	Reserved The write value should always be 0.
6 to 0	MXPS	H'40	R/W	Maximum Packet Size Specifies the maximum data payload (maximum packet size) for the DCP. These bits are initialized to H'40 (64 bytes). These bits should be set to the value based on the USB Specification. These bits should be set while PID is NAK and before the pipe is selected by the CURPIPE bits. Before modifying these bits after modifying the PID bits for the corresponding pipe from BUF to NAK, check that PBUSY is 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary. While MXPS is 0, do not write to the FIFO buffer or do not set PID to BUF.

23C.2.27 DCP Control Register (DCPCTR)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSTS	—	—	—	—	—	—	SQCLR	SQSET	SQMON	PBUSY	—	—	CCPL	PID	
Initial value:	0	—	—	—	—	—	—	0	0	1	0	—	—	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	BSTS	0	R	Buffer Status Indicates whether DCP FIFO buffer access is enabled or disabled. 0: Buffer access is disabled. 1: Buffer access is enabled. The meaning of the BSTS bit depends on the ISEL bit setting as follows. • When ISEL = 0, BSTS indicates whether the received data can be read from the buffer. • When ISEL = 1, BSTS indicates whether the data to be transmitted can be written to the buffer.
14 to 9	—	Undefined	R	Reserved The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
8	SQCLR	0	R/W	Toggle Bit Clear Specifies DATA0 as the expected value of the sequence toggle bit for the next transaction during the DCP transfer. 0: Invalid 1: Specifies DATA0. This bit always indicates 0. Do not set the SQCLR and SQSET bits to 1 simultaneously. Set this bit to 1 while PID is NAK. Before setting this bit to 1 after modifying the PID bits for the corresponding pipe from BUF to NAK, check that PBUSY is 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary.
7	SQSET	0	R/W	Toggle Bit Set Specifies DATA1 as the expected value of the sequence toggle bit for the next transaction during the DCP transfer. 0: Invalid 1: Specifies DATA1. This bit always indicates 0. Do not set the SQCLR and SQSET bits to 1 simultaneously. Set this bit to 1 while PID is NAK. Before setting this bit to 1 after modifying the PID bits for the corresponding pipe from BUF to NAK, check that PBUSY is 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary.

Bit	Bit Name	Initial Value	R/W	Description
6	SQMON	1	R	Sequence Toggle Bit Monitor Indicates the expected value of the sequence toggle bit for the next transaction during the DCP transfer. 0: DATA0 1: DATA1 This module allows this bit to toggle upon normal completion of the transaction. However, this bit is not allowed to toggle when a DATA-PID disagreement occurs during the transfer in the receiving direction. This module sets this bit to 1 (specifies DATA1 as the expected value) upon normal reception of the setup packet. This module does not reference to this bit during the IN/OUT transaction of the status stage, and does not allow this bit to toggle upon normal completion.
5	SPBUSY	0	R	Pipe Busy Indicates whether DCP is used or not for the transaction when USB changes the PID bits from BUF to NAK. 0: Has not finished the transition to NAK 1: Has finished the transition to NAK This module modifies this bit from 0 to 1 upon start of the USB transaction for the corresponding pipe, and modifies the bit from 1 to 0 upon completion of one transaction. Reading this bit after software has set PID to NAK allows checking that modification of the pipe settings is possible.

Bit	Bit Name	Initial Value	R/W	Description
4, 3	—	Undefined	R	Reserved The write value should always be 0.
2	CCPL	0*[1]	R/W	Control Transfer End Enable When the function controller function is selected, setting this bit to 1 enables the status stage of the control transfer to be completed. This bit read as 0, writing 0 to this bit is invalid. 0: Invalid 1: Completion of control transfer is enabled. When software sets this bit to 1 while the corresponding PID bits are set to BUF, this module completes the control transfer stage. Specifically, during control read transfer, this module transmits the ACK handshake in response to the OUT transaction from the USB host, and outputs the zero-length packet in response to the IN transaction from the USB host during control write or no-data control transfer. However, on detecting the SET_ADDRESS request, this module operates in auto response mode from the setup stage up to the status stage completion irrespective of the setting of this bit. This module modifies this bit from 1 to 0 on receiving the new setup packet. Software cannot write 1 to this bit while VALID is 1.

Bit	Bit Name	Initial Value	R/W	Description
1, 0	PID	00*2	R/W	Response PID Controls the response type of this module during control transfer. 00: NAK response 01: BUF response (depending on the buffer state) 10: STALL response 11: STALL response Change the setting of these bits from NAK to BUF by software while executing the data stage or status stage of control transfers. This module modifies the setting of these bits as follows. • This module modifies PID to NAK (B'00) on receiving the setup packet. Here, this module sets VALID to 1. Software cannot modify the setting of PID until software sets VALID to 0. • This module sets PID to STALL (B'11) on receiving the data of the size exceeding the maximum packet size when software has set PID to BUF. • This module sets PID to STALL (B'1x) on detecting the control transfer sequence error. • This module sets PID to NAK on detecting the USB bus reset. This module does not reference to the setting of the PID bits while the SET_ADDRESS request is processed (auto processing).

Notes: 1. This bit is initialized to 0 by a USB bus reset.
2. These bits are initialized to 00 by a USB bus reset.

23C.2.28 Pipe Window Select Register (PIPESEL)

PIPE1 to PIPE 9 should be set using PIPESEL, PIPECFG, PIPEBUF, PIPEMAXP, PIPEPERI, PIPEnCTR, PIPEnTRE, and PIPEnTRN. After selecting the pipe using PIPESEL, functions of the pipe should be set using PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI. PIPEnCTR, PIPEnTRE, and PIPEnTRN can be set regardless of the pipe selection in PIPESEL.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	PIPESEL			
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 4	—	Undefined	R	Reserved The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	PIPESEL	0000	R/W	Pipe Window Select Selects the pipe corresponding to the PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI registers which data is written to or read from. 0000: No pipe selected 0001: PIPE1 0010: PIPE2 0011: PIPE3 0100: PIPE4 0101: PIPE5 0110: PIPE6 0111: PIPE7 1000: PIPE8 1001: PIPE9 When a value between B'0001 and B'1001 is set in these bits by software, the information and settings for the corresponding pipe can be read from the PIPECFG, PIPEBUF, and PIPEMAXP registers. After the pipe is specified by these bits, the values that are set to PIPECFG, PIPEBUF, and PIPEMAXP by software will be reflected in the transfer mode of the corresponding pipe. When PIPESEL = B'0000, 0 is read from all of the bits in PIPECFG, PIPEBUF, and PIPEMAXP registers. Writing to these bits is invalid.

23C.2.29 Pipe Configuration Register (PIPECFG)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TYPE		—	—	—	BFRE	DBLB	CNTMD	SHTNAK	—	—	DIR	EPNUM			
Initial value:	0	0	—	—	—	0	0	0	0	—	—	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15, 14	TYPE	00	R/W	Transfer Type Specify the USB transfer type for the pipe selected by the PIPESEL bits (corresponding pipe) 00: Pipe not used 01: Bulk transfer 10: Interrupt transfer 11: Isochronous transfer The selected pipe and the transfer types that can be set by these bits are shown in table 23C.7. Before setting PID to BUF for the selected pipe (before starting USB communication using the selected pipe), be sure to set these bits to the value other than B'00. Modify these bits while the PID bits for the selected pipe are set to NAK. Before modifying these bits after modifying the PID bits for the selected pipe from BUF to NAK, check that PBUSY is 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary.

Bit	Bit Name	Initial Value	R/W	Description
13 to 11	—	Undefined	R	Reserved The write value should always be 0.
10	BFRE	0	R/W	BRDY Interrupt Operation Specification Specifies the timing for this module to notify generation of a BRDY interrupt for the corresponding pipe. 0: BRDY interrupt notification upon transmitting or receiving data 1: BRDY interrupt notification upon completion of reading data This bit is valid when PIPE1 to PIPE5 are selected. When software has set this bit to 1 and the selected pipe is in the receiving direction, this module detects the transfer completion and generates the BRDY interrupt on having read the corresponding packet. When the BRDY interrupt is generated with the above conditions, software needs to write 1 to BCLR. The FIFO buffer assigned to the selected pipe is not enabled for reception until 1 is written to BCLR. When software has set this bit to 1 and the selected pipe is in the transmitting direction, this module does not generate the BRDY interrupt. For details, refer to section 23C.2.16, BRDY Interrupt Status Register (BRDYSTS). Modify these bits while PID is NAK and before the pipe is selected by the CURPIPE bits. To modify these bits after completing USB communication using the selected pipe, write 1 and then 0 to ACLRM continuously through software to clear the FIFO buffer assigned to the selected pipe while the PID and CURPIPE bits are in the above-described state. Before modifying these bits after modifying the PID bits for the selected pipe from BUF to NAK, check that PBUSY is 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary.

Bit	Bit Name	Initial Value	R/W	Description
9	DBLB	0	R/W	Double Buffer Mode Selects either single or double buffer mode for the FIFO buffer used by the corresponding pipe. 0: Single buffer 1: Double buffer This bit is valid when PIPE1 to PIPE5 are selected. When software has set this bit to 1, this module assigns two planes of the FIFO buffer size specified by the BUFSIZE bits in PIPEBUF to the selected pipe. Specifically, the following expression determines the FIFO buffer size assigned to the selected pipe by this module. (BUFSIZE + 1) × 64 × (DBLB + 1) [bytes] Modify these bits while PID is NAK and before the pipe is selected by the CURPIPE bits. To modify these bits after completing USB communication using the selected pipe, write 1 and then 0 to ACLRM continuously through software to clear the FIFO buffer assigned to the selected pipe while the PID and CURPIPE bits are in the above-described state. Before modifying these bits after modifying the PID bits for the selected pipe from BUF to NAK, check that PBUSY is 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary.

Bit	Bit Name	Initial Value	R/W	Description
8	CNTMD	0	R/W	Continuous Transfer Mode Specifies whether to use the corresponding pipe in continuous transfer mode. 0: Non-continuous transfer mode 1: Continuous transfer mode This bit is valid when PIPE1 to PIPE5 are selected by the PIPESEL bits and bulk transfer is selected. This module judges whether transmitting to/receiving from the FIFO buffer allocated for the selected pipe has completed or not using this bit setting, as shown in table 23C.8. Modify these bits while PID is NAK and before the pipe is selected by the CURPIPE bits. To modify these bits after completing USB communication using the selected pipe, write 1 and then 0 to ACLRM continuously through software to clear the FIFO buffer assigned to the selected pipe while the PID and CURPIPE bits are in the above-described state. Before modifying these bits after modifying the PID bits for the selected pipe from BUF to NAK, check that PBUSY is 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary.

Bit	Bit Name	Initial Value	R/W	Description
7	SHTNAK	0	R/W	Pipe Disabled at End of Transfer Specifies whether to modify PID to NAK upon the end of transfer when the selected pipe is in the receiving direction. 0: Pipe continued at the end of transfer 1: Pipe disabled at the end of transfer This bit is valid when the selected pipe is PIPE1 to PIPE5 in the receiving direction. When software has set this bit to 1 for the selected pipe in the receiving direction, this module modifies the PID bits corresponding to the selected pipe to NAK on determining the end of the transfer. This module determines that the transfer has ended on any of the following conditions. • A short packet (including a zero-length packet) is successfully received. • The transaction counter is used and the number of packets specified by the counter is successfully received. Modify these bits while PID is NAK. Before modifying these bits after modifying the PID bits for the selected pipe from BUF to NAK, check that PBUSY is 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary. This bit should be cleared to 0 for the pipe in the transmitting direction.
6, 5	—	Undefined	R	Reserved The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
4	DIR	0	R/W	Transfer Direction Specifies the transfer direction for the corresponding pipe. 0: Receiving direction 1: Sending direction When software has set this bit to 0, this module uses the selected pipe in the receiving direction, and when software has set this bit to 1, this module uses the selected pipe in the transmitting direction. Modify these bits while PID is NAK and before the pipe is selected by the CURPIPE bits. To modify these bits after completing USB communication using the selected pipe, write 1 and then 0 to ACLRM continuously through software to clear the FIFO buffer assigned to the selected pipe while the PID and CURPIPE bits are in the above-described state. Before modifying these bits after modifying the PID bits for the selected pipe from BUF to NAK, check that PBUSY is 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary.
3 to 0	EPNUM	0000	R/W	Endpoint Number Specify the endpoint number for the corresponding pipe. The endpoint number corresponding to the selected pipe should be set in these bits by software. Setting B'0000 means unused pipe. Modify these bits while PID is NAK and before the pipe is selected by the CURPIPE bits. Before modifying these bits after modifying the PID bits for the selected pipe from BUF to NAK, check that PBUSY is 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary. Do not make the settings such that the combination of the set values in the DIR and EPNUM bits should be the same for two or more pipes (EPNUM = B'000 can be set for all the pipes).

Table 23C.7 Selected Pipe and Transfer Types Settable by TYPE Bits

Selected Pipe	TYPE Bit	USB Transfer Type
PIPE1 or PIPE2	B'01 or B'11	Bulk transfer or isochronous transfer
PIPE3 to PIPE5	B'01	Bulk transfer
PIPE6 to PIPE9	B'11	Interrupt transfer

Table 23C.8 Relationship between CNTMD Bit Setting and Method for Determining Completion of FIFO Buffer Transmission/Reception

CNTMD Bit Setting	Method for Determining Reading Enabled State and Transmission Enabled State
0	[Condition for enabling FIFO buffer reading when receiving direction is set (DIR = 0)] When this module has received a single packet
	[Condition for enabling FIFO buffer transmission when transmitting direction is set (DIR = 1)] When either (1) or (2) is satisfied (1) When software (or the DMAC) has written the maximum packet size of data to the FIFO buffer (2) When software (or the DMAC) has written a short packet size of data (including the case of zero bytes) to the FIFO buffer, and has also written 1 to the BVAL bit
1	[Condition for enabling FIFO buffer reading when receiving direction is set (DIR = 0)] (1) When the number of data bytes received in the FIFO buffer allocated for the selected pipe has become equal to the number of allocated bytes ((BUFSIZE+1)*64) (2) When this module has received a short packet other than a zero-length packet (3) When this module has received a zero-length packet while the FIFO buffer allocated for the selected pipe already contains data (4) When the number of received packets becomes equal to the transaction counter number set for the selected pipe by software

CNTMD Bit Setting	Method for Determining Reading Enabled State and Transmission Enabled State
1	[Condition for enabling FIFO buffer transmission when transmitting direction is set (DIR = 1)] When (1), (2), or (3) is satisfied (1) When the number of data bytes written by software (or the DMAC) has become equal to one buffer of the FIFO buffer size allocated for the selected pipe (2) When software (or the DMAC) has written data bytes (including the case of zero bytes) that are less than one buffer of the FIFO buffer size allocated for the selected pipe to the FIFO buffer, and has also written 1 to the BVAL bit (3) When software (or the DMAC) has written data bytes (including the case of zero bytes) that are less than one buffer of the FIFO buffer size allocated for the selected pipe to the FIFO buffer, and has also asserted the DENDx_N signal simultaneously with the last writing

23C.2.30 Pipe Buffer Setting Register (PIPEBUF)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	BUFSIZE					—	—	BUFNMB							
Initial value:	—	0	0	0	0	0	—	—	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	Undefined	R	Reserved The write value should always be 0.
14 to 10	BUFSIZE	H'00	R/W	Buffer Size Specifies the size of the buffer for the pipe selected by the PIPESEL bits (corresponding pipe) in terms of blocks, where one block comprises 64 bytes. H'00: 64 bytes H'01: 128 bytes : : H'0F: 1 Kbyte When software has set the DBLB bit to 1, this module assigns two planes of the FIFO buffer size specified by the BUFSIZE bits to the selected pipe. Specifically, the following expression determines the FIFO buffer size assigned to the selected pipe by this module. (BUFSIZE + 1) * 64 * (DBLB + 1) [bytes] The valid value for these bits depends on the selected pipe. • PIPE1 to PIPE5: Any value from H'00 to H'1F is valid. • PIPE6 to PIPE9: H'00 should be set. When used with CNTMD = 1, set an integer multiple of the maximum packet size to the BUFSIZE bits.
9, 8	—	Undefined	R	Reserved The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	BUFNMB	H'00	R/W	Buffer Number The first block number in the FIFO buffer to be allocated for the corresponding pipe should be set in these bits. (from H'04 to H'87F). The FIFO buffer blocks allocated for the selected pipe by this module are determined as follows: Block number: BUFNMB to block number of (BUFNMB + (BUFSIZE + 1)*(DBLB + 1) – 1) These bits should be set so that the memory size range is not exceeded (0 [H'00] to 8640 [H'87] for 8.5 Kbytes). The following conditions also need to be satisfied. H'01 to H'03 are for DCP only. When the CNTMD bit in DCP is 0 (not connected), H'01 to H'03 can be used for other pipes. H'04 is for PIPE6 only. However, if PIPE6 is not used, H'04 can be used for other pipes. When the selected pipe is PIPE6, writing to these bits is invalid; this module automatically assigns BUFNMB = H'04 for PIPE6. H'05 is for PIPE7 only. However, if PIPE7 is not used, H'05 can be used for other pipes. When the selected pipe is PIPE7, writing to these bits is invalid; this module automatically assigns BUFNMB = H'05 for PIPE7. H'06 is for PIPE8 only. However, if PIPE8 is not used, H'06 can be used for other pipes. When the selected pipe is PIPE8, writing to these bits is invalid; this module automatically assigns BUFNMB = H'06 for PIPE8. H'07 is for PIPE9 only. However, if PIPE9 is not used, H'07 can be used for other pipes. When the selected pipe is PIPE9, writing to these bits is invalid; this module automatically assigns BUFNMB = H'07 for PIPE9.

Notes: 1. The bits in this register should be set only when PID = NAK, and no pipe is set in the CURPIPE bits by software.

2. When changing the bits in this register after the PID bit setting of the selected pipe has been changed from BUF to NAK, confirm PBUSY = 0 before changing the bits. However, if the PID bit setting was changed to NAK by this module, the PBUSY bit value does not have to be confirmed by software.

23C.2.31 Pipe Maximum Packet Size Register (PIPEMAXP)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	MXPS										
Initial value:	—	—	—	—	—	0	0	0	0	0/1	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	Undefined	R	Reserved The write value should always be 0.
10 to 0	MXPS	H'040/ H'000*	R/W	Maximum Packet Size Specifies the maximum data payload (maximum packet size) for the corresponding pipe. The valid value for these bits depends on the pipe as follows. The initial value is H'040 (64 bytes). PIPE1, PIPE2: 1 byte (H'001) to 1,024 bytes (H'400) PIPE3 to PIPE5: 8 bytes (H'008), 16 bytes (H'010), 32 bytes (H'020), 64 bytes (H'040), and 512 bytes (H'200) (Bits 2 to 0 are not provided.) PIPE6 to PIPE9: 1 byte (H'001) to 64 bytes (H'040)

Bit	Bit Name	Initial Value	R/W	Description
10 to 0	MXPS	H'040/ H'000*	R/W	These bits should be set to the appropriate value for each transfer type based on the USB Specification. For split transactions using the isochronous pipe, these bits should be set to 188 bytes or less. These bits should be changed only when PID is NAK and no pipe is set in the CURPIPE bits. Before modifying these bits after modifying the PID bits for the selected pipe from BUF to NAK, check that PBUSY is 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary. While MXPS is 0, do not write to the FIFO buffer or set PID to BUF.

Note: * The initial value of MXPS is H'000 when no pipe is selected with the PIPESEL bits in PIPESEL and H'040 when a pipe is selected with the PIPESEL bit in PIPESEL.

23C.2.32 Pipe Timing Control Register (PIPEPERI)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	IFIS	—	—	—	—	—	—	—	—	—	IITV		
Initial value:	—	—	—	0	—	—	—	—	—	—	—	—	—	0	0	0
R/W:	R	R	R	R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	—	Undefined	R	Reserved The write value should always be 0.
12	IFIS	0	R/W	Isochronous IN Buffer Flush Specifies whether to flush the buffer when the pipe selected by the PIPESEL bits (corresponding pipe) is used for isochronous IN transfers. 0: The buffer is not flushed. 1: The buffer is flushed. The selected pipe is for isochronous IN transfers, this module automatically clears the FIFO buffer when this module fails to receive the IN token from the USB host within the interval set by the IITV bits in terms of (μ) frames. In double buffer mode (DBLB = 1), this module only clears the data in the plane used earlier. This module clears the FIFO buffer on receiving the SOF packet immediately after the (μ) frame in which this module has expected to receive the IN token. Even if the SOF packet is corrupted, this module also clears the FIFO buffer at the right timing to receive the SOF packet by using the internal interpolation.
11 to 3	—	Undefined	R	Reserved The write value should always be 0.
2 to 0	IITV	000	R/W	Interval Error Detection Interval Specifies the interval error detection timing for the selected pipe in terms of frames, which is expressed as n-th power of 2 (n is the value to be set).

Bit	Bit Name	Initial Value	R/W	Description
2 to 0	IITV	000	R/W	These bits should be changed only when PID = NAK, and no pipe is set in the CURPIPE bits. When changing these bits after the PID bit setting of the corresponding pipe has been changed from BUF to NAK, confirm PBUSY = 0 before changing these bits. However, if the PID bit setting was changed to NAK by this module, the PBUSY bit value does not have to be confirmed by software. In the case of setting these bits and performing USB communication, and then changing these bits to a different value, first set the PID bits to NAK, set the ACLRM bit to 1, and initialize the interval timer. These bits are not available for PIPE3 to PIPE5. 0 should be written to the bit locations that correspond to PIPE3 to PIPE5. The IITV bits can be set when isochronous transfer is set as the transfer type for the selected pipe. • When isochronous-OUT transfer is set for the selected pipe When no data packet is received during the (μ) frame for each interval set using the IITV bits, this module generates an NRDY interrupt. An NRDY interrupt is also generated when this module fails to receive data due to an error (CRC error, etc.) occurring in the data packet or the FIFO buffer being full (because software (or the DMAC) was slow in reading data from the FIFO buffer). The timing for generating an NRDY interrupt is when an SOF packet is received. Even when an SOF packet is corrupted, the internal interpolation function enables an NRDY interrupt to be generated at the timing to receive the SOF packet. However, when the IITV bits are set to a value other than 0, an NRDY interrupt is generated when an SOF packet is received at each interval after the interval counter has started.

Bit	Bit Name	Initial Value	R/W	Description
2 to 0	IITV	000	R/W	When the PID bits are set to NAK by software after the interval counter has started, this module does not generate an NRDY interrupt even though an SOF packet is received. The conditions for starting the interval counter differ according to the IITV bit setting. (See figures 23C.2 and 23C.3.) • When isochronous-IN transfer is set for the selected pipe This is used in combination with the IFIS bit being set to 1. When IFIS = 0, a data packet is sent in response to the received token regardless of the IITV bit setting. When IFIS = 1 is set, this module clears the FIFO buffer if an IN-token is not received during the (μ) frame for each interval set using the IITV bits when there is data to be transmitted in the FIFO buffer. The FIFO buffer is also cleared when this module could not receive an IN-token normally because a bus error (CRC error, etc.) has occurred in the IN-token. The timing for clearing the FIFO buffer is when an SOF packet is received. Even when an SOF packet is corrupted, the internal interpolation function enables the FIFO buffer to be cleared at the timing to receive the SOF packet. The conditions for starting the interval counter differ according to the IITV bit setting (similar to isochronous-OUT transfer). The condition for interval counting is one of the following conditions 1, 2, or 3: 1. When a hardware reset is performed to this module (the IITV bit setting is also cleared to 0 in this case) 2. When ACLRM = 1 is set by software 3. When this module has detected a USB reset

(a) When IITV is 0

The interval counter starts from the (μ) frame following the (μ) frame in which the PID bits of the selected pipe were changed to BUF.

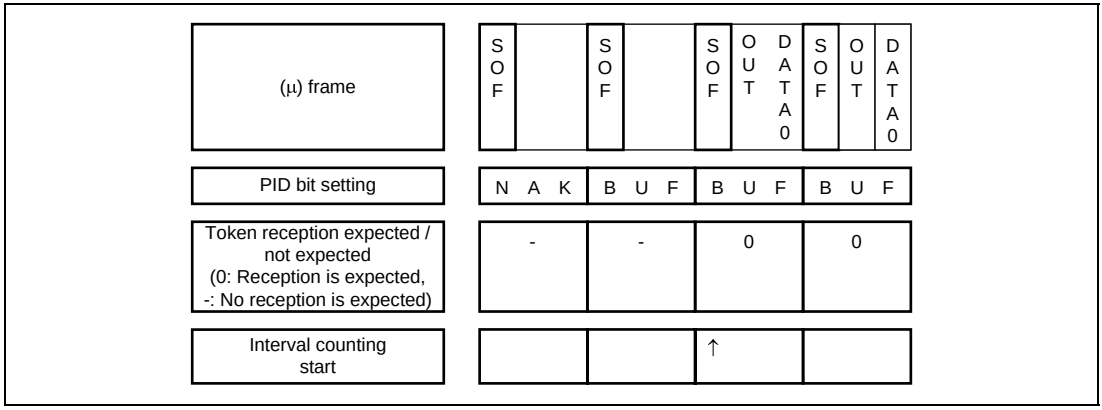


Figure 23C.2 Relationship between (μ) Frame and Token Reception Expectation when IITV = 0

(b) When IITV is other than 0

The interval counter starts from the completion of the first data packet normal reception following the (μ) frame in which the PID bits of the selected pipe were changed to BUF.

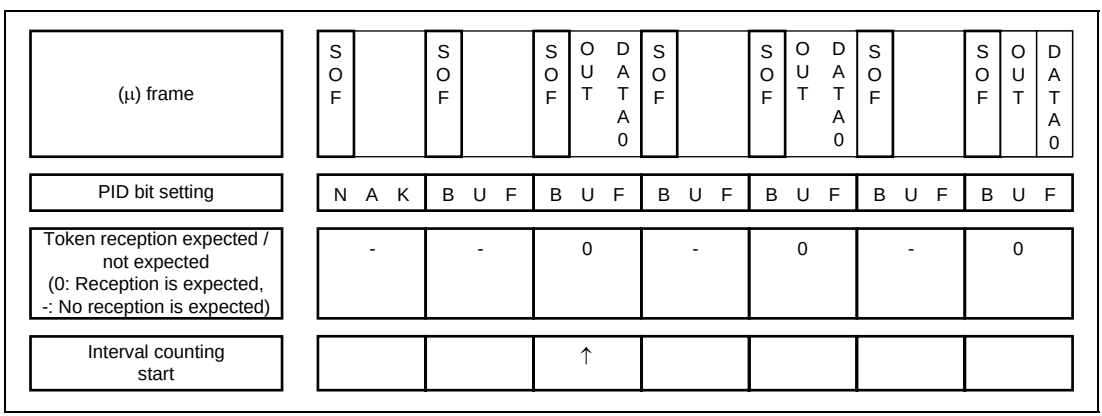


Figure 23C.3 Relationship between (μ) Frame and Token Reception Expectation when IITV ≠ 0

23C.2.33 PIPEn Control Registers (PIPEnCTR) (n = 1 to 5)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSTS	INBUFM	—	—	—	ATREPM	ACLRM	SQCLR	SQSET	SQMON	PBSY	—	—	—	PID	
Initial value:	0	0	—	—	—	0	0	0	0	0	0	—	—	—	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	BSTS	0	R	Buffer Status Whether the FIFO buffer allocated for the corresponding pipe can be accessed by the CPU can be read from this bit. 0: Buffer access is disabled. 1: Buffer access is enabled. The meaning of this bit depends on the settings of the DIR, BFRE, and DCLRM bits as shown in table 23C.9.
14	INBUFM	0	R	Transmission Buffer Monitor Indicates the corresponding FIFO buffer status when the corresponding pipe is in the transmitting direction. 0: There is no data to be transmitted in the FIFO buffer. 1: There is data to be transmitted in the FIFO buffer. When the corresponding pipe is in the transmitting direction (DIR = 1), this module sets this bit to 1 when software (or DMAC) completes writing data to at least one FIFO buffer plane. This module sets this bit to 0 when this module completes transmitting the data from the FIFO buffer plane to which all the data has been written. In double buffer mode (DBLB = 1), this module sets this bit to 0 when this module completes transmitting the data from the two FIFO buffer planes before software (or DMAC) completes writing data to one FIFO buffer plane. This bit indicates the same value as the BSTS bit when the corresponding pipe is in the receiving direction (DIR = 0).

Bit	Bit Name	Initial Value	R/W	Description
13 to 11	—	Undefined	R	Reserved The write value should always be 0.
10	ATREPM	0	R/W	Auto Response Mode Enables or disables auto response of the corresponding pipe. 0: Auto response is disabled 1: Auto response is enabled (Regardless of the FIFO buffer status of the corresponding pipe, a zero-length packet is always sent at transmission, and a NAK response is returned and an NRDY interrupt generated at reception). When bulk transfer is set as the transfer type for the corresponding pipe, this bit can be set to 1. When this bit is set to 1, this module responds to the token from the USB host as shown below. • When bulk-IN transfer (TYPE = B'01 and DIR = 1) is set for the corresponding pipe When both ATREPM = 1 and PID = BUF are set, this module transmits a zero-length packet in response to an IN token. Each time this module receives an ACK response from the USB host (a single transaction consists of IN token reception → zero-length packet transmission → ACK response reception), this module updates (toggles) the sequence toggle bit (DATA-PID). BRDY and BEMP interrupts are not generated.

Bit	Bit Name	Initial Value	R/W	Description
10	ATREPM	0	R/W	— When bulk-OUT transfer (TYPE = B'01 and DIR = 0) is set for the corresponding pipe When both ATREPM = 1 and PID = BUF are set, this module sends a NAK response in response to an OUT token (or PING token) and generates an NRDY interrupt. This bit should be changed only when PID = NAK. When changing this bit after the PID bit setting of the corresponding pipe has been changed from BUF to NAK, confirm PBUSY = 0 before changing this bit. However, if the PID bit setting was changed to NAK by this module, the PBUSY bit value does not have to be confirmed by software. When performing USB communication with this bit set to 1, this bit should be set to 1 only when the FIFO buffer is empty. The FIFO buffer must not be written to during USB communication with this bit set to 1. When isochronous transfer is set as the transfer type for the corresponding pipe, this bit must be cleared to 0.

Bit	Bit Name	Initial Value	R/W	Description
9	ACLRM	0	R/W	Auto Buffer Clear Mode Enables or disables automatic buffer clear mode for the corresponding pipe. 0: Disabled 1: Enabled (all buffers are initialized) To delete the information in the FIFO buffer assigned to the corresponding pipe completely, write 1 and then 0 to this bit continuously. Table 23C.10 shows the information cleared by writing 1 and 0 to this bit continuously and the cases in which clearing the information is necessary. Modify this bit while PID is NAK and before the pipe is selected by the CURPIPE bits. Before modifying this bit after modifying the PID bits for the corresponding pipe from BUF to NAK, check that PBUSY is 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary.
8	SQCLR	0	R/W	Toggle Bit Clear This bit should be set to 1 to clear the expected value (to set DATA0 as the expected value) of the sequence toggle bit for the next transaction of the corresponding pipe. 0: Invalid 1: Specifies DATA0. Setting this bit to 1 through software allows this module to set DATA0 as the expected value of the sequence toggle bit of the corresponding pipe. This module always sets this bit to 0. This bit should be set to 1 only when PID = NAK. When setting this bit to 1 after the PID bit setting of the corresponding pipe has been changed from BUF to NAK, confirm PBUSY = 0 before changing this bit. However, if the PID bit setting was changed to NAK by this module, the PBUSY bit value does not have to be confirmed by software.

Bit	Bit Name	Initial Value	R/W	Description
7	SQSET	0	R/W	Toggle Bit Set This bit should be set to 1 to set DATA1 as the expected value of the sequence toggle bit for the next transaction of the corresponding pipe. 0: Invalid 1: Specifies DATA1. Setting this bit to 1 through software allows this module to set DATA1 as the expected value of the sequence toggle bit of the corresponding pipe. This module always sets this bit to 0. Set the SQSET bit to 1 while PID is NAK. Before modifying this bit after modifying the PID bits for the corresponding pipe from BUF to NAK, check that PBUSY is 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary.
6	SQMON	0	R	Toggle Bit Confirmation Indicates the expected value of the sequence toggle bit for the next transaction of the corresponding pipe. 0: DATA0 1: DATA1 When the corresponding pipe is not for the isochronous transfer, this module allows this bit to toggle upon normal completion of the transaction. However, this bit is not allowed to toggle when a DATA-PID disagreement occurs during the receiving transfer.

Bit	Bit Name	Initial Value	R/W	Description
5	PBUSY	0	R	Pipe Busy Whether the corresponding pipe is currently being used by the USB bus can be read. 0: Corresponding pipe is not being used by the USB bus. 1: Corresponding pipe is being used by the USB bus This module modifies this bit from 0 to 1 upon start of the USB transaction for the corresponding pipe, and modifies the bit from 1 to 0 upon completion of one transaction. Reading this bit after software has set PID to NAK allows checking that modification of the pipe settings is possible.
4 to 2	—	Undefined	R	Reserved The write value should always be 0.
1, 0	PID	00*	R/W	Response PID Specifies the response type for the next transaction of the corresponding pipe. 00: NAK response 01: BUF response (depending on the buffer state) 10: STALL response 11: STALL response The default setting of these bits is NAK. Modify the setting to BUF to use the corresponding pipe for USB transfer. Table 23C.11 shows the basic operation (operation when there are no errors in the transmitted and received packets) of this module depending on the PID bit setting.

Bit	Bit Name	Initial Value	R/W	Description
1, 0	PID	00*	R/W	After modifying the setting of these bits through software from BUF to NAK during USB communication using the corresponding pipe, check that PBUSY is 0 to see if USB communication using the corresponding pipe has actually entered the NAK state. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary. Even though these bits are changed to NAK by software after S-Split of a Split transaction is issued for the corresponding pipe, the transaction continues to be executed until its end. This module modifies the setting of these bits as follows. • When the corresponding pipe is set to the receiving direction and software has set the SHTNAK bit of the corresponding pipe to 1, PID = NAK can be read from these bits as soon as this module recognizes the end of transfer. • When this module receives a data packet whose payload exceeds MaxPacketSize for the corresponding pipe, PID = STALL (B'11) can be read from these bits. • When a USB bus reset has been detected by this module, PID = NAK can be read from these bits. Write B'10 to these bits to change PID = NAK (B'00) to PID = STALL. Write B'11 to these bits to change PID = BUF (B'01) to PID = STALL. Write 10 first and then 00 to these bits to change PID = STALL (B'11) to PID = NAK. To change the STALL state to the BUF state, change these bits to the NAK state first and then change them to the BUF state.

Note: These bits are initialized to 00 by a USB bus reset.

Table 23C.9 Operation of BSTS Bit

DIR Bit Setting Value	BFRE Bit Setting Value	DCLRM Bit Setting Value	Meaning of BSTS Bit
0	0	0	1: The received data can be read from the FIFO buffer. 0: The received data has been completely read from the FIFO buffer.
		1	Setting prohibited
	1	0	1: The received data can be read from the FIFO buffer. 0: Software has set BCLR to 1 after the received data has been completely read from the FIFO buffer.
		1	1: The received data can be read from the FIFO buffer. 0: The received data has been completely read from the FIFO buffer.
1	0	0	1: The transmit data can be written to the FIFO buffer. 0: The transmit data has been completely written to the FIFO buffer.
		1	Setting prohibited
	1	0	Setting prohibited
		1	Setting prohibited

Table 23C.10 Information Cleared by this Module by Setting ACLRM = 1

Information Cleared by ACLRM Bit Manipulation	Cases in which Clearing the Information is Necessary
All the information in the FIFO buffer assigned to the corresponding pipe (all the information in two FIFO buffer planes in double buffer mode)	
The interval count value when the corresponding pipe is for isochronous transfer	When the interval count value is to be reset
Values of the internal flags related to the BFRE bit	When the BFRE setting is modified
FIFO buffer toggle control	When the DBLB setting is modified
Values of the internal flags related to the transaction count	When the transaction count function is forcibly terminated

Table 23C.11 Module Operations with Various PID Bit Settings

PID Bit Setting	Transfer Type (TYPE Bit Setting)	Transfer Direction (DIR Bit Setting)	Operation of This Module
B'00 (NAK)	Bulk (TYPE = B'01) or interrupt (TYPE = B'10)	Does not depend on the setting	The NAK response is returned to the token from the USB host.
	Isochronous (TYPE = B'11)	Does not depend on the setting	No response is returned to the token from the USB host.
B'01 (BUF)	Bulk (TYPE = B'01)	Receiving direction (DIR = 0)	If the FIFO buffer of the corresponding pipe can receive data, the OUT token from the USB host is received and the ACK response is returned. If data cannot be received, the NAK response is returned. If the FIFO buffer of the corresponding pipe can receive data, the PING token from the USB host is received and the ACK response is returned. If data cannot be received, the NAK response is returned.
	Interrupt (TYPE = B'10)	Receiving direction (DIR = 0)	If the FIFO buffer of the corresponding pipe can receive data, the OUT token from the USB host is received and the ACK response is returned. If data cannot be received, the NAK response is returned.
	Bulk (TYPE = B'01) or interrupt (TYPE = B'10)	Transmitting direction (DIR = 1)	If the corresponding FIFO buffer can transmit data, data is transmitted in response to the token from the USB host. If data cannot be transmitted, the NAK response is returned.
	Isochronous (TYPE = B'11)	Receiving direction (DIR = 0)	If the FIFO buffer of the corresponding pipe can receive data, the OUT token from the USB host is received. If data cannot be received, the data is discarded.
		Transmitting direction (DIR = 1)	If the corresponding FIFO buffer can transmit data, data is transmitted in response to the token from the USB host. If data cannot be transmitted, a zero-length packet is transmitted.
B'10 (STALL) or B'11 (STALL)	Bulk (TYPE = B'01) or interrupt (TYPE = B'10)	Does not depend on the setting	The STALL response is returned to the token from the USB host.
	Isochronous (TYPE = B'11)	Does not depend on the setting	No response is returned to the token from the USB host.

23C.2.34 PIPEn Control Registers (PIPEnCTR) (n = 6 to 9)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSTS	—	—	—	—	—	ACLRM	SQCLR	SQSET	SQMON	PBSY	—	—	—	PID	
Initial value:	0	—	—	—	—	—	0	0	0	0	0	—	—	—	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	BSTS	0	R	Buffer Status Indicate whether the FIFO buffer of the corresponding pipe can be accessed from the CPU. 0: Buffer access is disabled. 1: Buffer access is enabled. Refer to section 23C.2.33, PIPEn Control Registers (PIPEnCTR) (n = 1 to 5).
14 to 10	—	Undefined	R	Reserved The write value should always be 0.
9	ACLRM	0	R/W	Auto Buffer Clear Mode*3*4 Enables or disables automatic buffer clear mode for the corresponding pipe. 0: Disabled 1: Enabled (all buffers are initialized) To delete the information in the FIFO buffer assigned to the corresponding pipe completely, write 1 and then 0 to this bit continuously. Table 23C.12 shows the information cleared by writing 1 and 0 to this bit continuously and the cases in which clearing the information is necessary. Modify this bit while PID is NAK and before the pipe is selected by the CURPIPE bits. Before modifying this bit after modifying the PID bits for the corresponding pipe from BUF to NAK, check that PBUSY is 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary.

Bit	Bit Name	Initial Value	R/W	Description
8	SQCLR	0	R/W	Toggle Bit Clear This bit should be set to 1 to clear the expected value (to set DATA0 as the expected value) of the sequence toggle bit for the next transaction of the corresponding pipe. 0: Invalid 1: Specifies DATA0. Refer to section 23C.2.33, PIPEn Control Registers (PIPEnCTR) (n = 1 to 5).
7	SQSET	0	R/W	Toggle Bit Set This bit should be set to 1 to set DATA1 as the expected value of the sequence toggle bit for the next transaction of the corresponding pipe. 0: Invalid 1: Specifies DATA1. Refer to section 23C.2.33, PIPEn Control Registers (PIPEnCTR) (n = 1 to 5).
6	SQMON	0	R	Toggle Bit Confirmation Indicates the expected value of the sequence toggle bit for the next transaction of the corresponding pipe. 0: DATA0 1: DATA1 Refer to section 23C.2.33, PIPEn Control Registers (PIPEnCTR) (n = 1 to 5).
5	PBUSY	0	R	Pipe Busy Indicates whether the relevant pipe is used or not for the USB bus. 0: The corresponding pipe is not used for the USB bus. 1: The corresponding pipe is used for the USB bus. Refer to section 23C.2.33, PIPEn Control Registers (PIPEnCTR) (n = 1 to 5).
4 to 2	—	Undefined	R	Reserved The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1, 0	PID	00*	R/W	Response PID Specify the response type for the next transaction of the corresponding pipe. 00: NAK response 01: BUF response (depending on the buffer state) 10: STALL response 11: STALL response Refer to section 23C.2.33, PIPEn Control Registers (PIPEnCTR) (n = 1 to 5).

Note: These bits are initialized to 00 by a USB bus reset.

Table 23C.12 Information Cleared by this Module by Setting ACLRM = 1

Information Cleared by ACLRM Bit Manipulation	Cases in which Clearing the Information is Necessary
All the information in the FIFO buffer assigned to the corresponding pipe	—
—	When the interval count value is to be reset
Values of the internal flags related to the BFRE bit	When the BFRE setting is modified
Values of the internal flags related to the transaction count	When the transaction count function is forcibly terminated

23C.2.35 PIPEn Transaction Counter Enable Registers (PIPEnTRE) (n = 1 to 5)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TRENB	TRCLR	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	0	0	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	Undefined	R	Reserved The write value should always be 0.
9	TRENB	0	R/W	Transaction Counter Enable Enables or disables the transaction counter. 0: The transaction counter is disabled. 1: The transaction counter is enabled. For the pipe in the receiving direction, setting this bit to 1 after setting the total number of the packets to be received in the TRNCNT bits through software allows this module to control hardware as described below on having received the number of packets equal to the set value in the TRNCNT bits. • In continuous transmission/reception mode (CNTMD = 1), this module switches the FIFO buffer to the CPU side even if the FIFO buffer is not full on completion of reception. • While SHTNAK is 1, this module modifies the PID bits to NAK for the corresponding pipe on having received the number of packets equal to the set value in the TRNCNT bits. • When both DENDE = 1 and PKTMD = 0 are set, the DEND signal is asserted when the last data is to be read after receiving packets for the same number as the TRCNT bit setting. • While BFRE is 1, this module asserts the BRDY interrupt on having received the number of packets equal to the set value in the TRNCNT bits and then reading out the last received data.

Bit	Bit Name	Initial Value	R/W	Description
9	TRENB	0	R/W	For the pipe in the transmitting direction, set this bit to 0. When the transaction counter is not used, set this bit to 0. When the transaction counter is used, set the TRNCNT bits before setting this bit to 1. Set this bit to 1 before receiving the first packet to be counted by the transaction counter.
8	TRCLR	0	R/W	Transaction Counter Clear When this bit is set to 1 by software, this module clears the current counter value of the transaction counter for the corresponding pipe, and then writes 0 to this bit. 0: Invalid 1: The current counter value is cleared.
7 to 0	—	Undefined	R	Reserved The write value should always be 0.

Note: Modify each bit in this register while PID is NAK. Before modifying each bit after modifying the PID bits for the corresponding pipe from BUF to NAK, check that PBUSY is 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary.

23C.2.36 PIPEn Transaction Counter Registers (PIPEnTRN) (n = 1 to 5)

For a pipe that has been set to the receiving direction, when the TRENB bit is set to 1 after the total number of packets that must be received has been set in these bits by software, this module performs the operations listed in section 23C.2.35, PIPEn Transaction Counter Enable Registers (PIPEnTRE) (n = 1 to 5).

The TRNCNT bit value is incremented by 1 when all of the following conditions (a) to (c) are satisfied in the receiving state.

(a) TRENB = 1

(b) (TRNCNT setting ≠ Current counter value + 1) at packet reception

(c) The payload of the received packet matches the MXPS bit setting

The TRNCNT bit value is cleared to 0 when any one of the following conditions (1) to (3) is satisfied.

(1) When the following condition is satisfied
    (TRNCNT setting = Current counter value + 1) at packet reception

(2) When both of the following conditions (a) and (b) are satisfied
    (a) TRENB = 1
    (b) A short packet is received

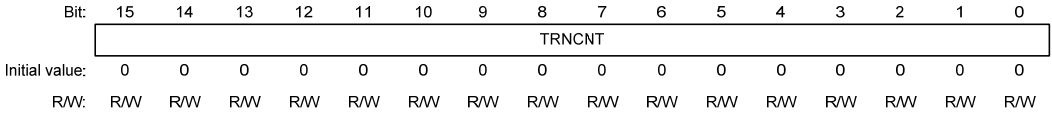
(3) When both of the following conditions (a) and (b) are satisfied
    (a) TRENB = 1
    (b) The TRCLR bit is set to 1 by software

For a pipe that has been set to the transmitting direction, these bits should be cleared to 0.

When the transaction counter function is not used, clear these bits to 0.

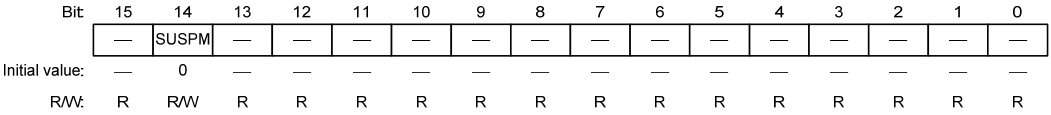
These bits should be changed only when PID = NAK, and TRENB = 0. When changing these bits after the PID bit setting of the corresponding pipe has been changed from BUF to NAK, confirm PBUSY = 0 before changing these bits. However, if the PID bit setting was changed to NAK by this module, the PBUSY bit value does not have to be confirmed by software.

When changing the value of these bits, set TRCLR = 1 prior to the setting of TRENB = 1.



Bit	Bit Name	Initial Value	R/W	Description
15 to 0	TRNCNT	H'0000	R/W	Transaction Counter When written to: Specifies the number of transactions to be transferred through DMA. When read from: Indicates the specified number of transactions if TRENB is 0. Indicates the number of currently counted transaction if TRENB is 1.

23C.2.37 UTMI Suspend Mode Register (SUSPMODE)



Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	SUSPM	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	0	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	—	Undefined	R	Reserved The write value should always be 0.
14	SUSPM	0	R/W	UTMI SuspendM control Controls the SuspendM signal sent to the UTMI. 0: UTMI suspend mode 1: UTMI normal mode The initial value is 0, so the UTMI is in the suspend mode. This bit should be set to 1 when this module is operating. In the UTMI specification, the SuspendM signal is normally used for clock output control. When SuspendM = 0, the clock supply to the LINK is stopped. For details, refer to the respective UTMI Specifications. When this bit is cleared to 0 (UTMI clock is stopped), this module cannot be written to but reading is possible. However, the registers below can be written to even when this bit is 0. SYSCGF0 BUSWAIT SUSPMODE Note that the value written to SYSCFG0 while the UTMI clock is stopped (SUSPM = 0) will be effective after the UTMI clock starts oscillating (SUSPM = 1).
13 to 0	—	Undefined	R	Reserved The write value should always be 0.

23C.3 Operation

23C.3.1 System Control and Oscillation Control

This section describes the register operations that are necessary to the initial settings of this module, and the registers necessary for power consumption control.

(1) Resets

Table 23C.13 lists the types of resets for this module. For the initialized states of the registers following the reset operations, see section 23C.2, List of Registers.

Table 23C.13 Types of Reset

Name	Operation
Hardware reset	Low level input from the $\overline{\text{PRESET}}$ pin
USB bus reset	Automatically detected by this module from the D+ and D– lines

(2) USB Data Bus Resistor Control

This module performs switching of a pull-up resistor for the D+ signal. This signal can be pulled up using the DPRPU bit in SYSCFG0.

When the DPRPU bit in SYSCFG0 is cleared to 0 during communication with the host controller, the pull-up resistor (or the terminal resistor) of the USB data line is disabled, making it possible to notify the host controller of the device disconnection.

23C.3.2 Interrupt Functions

(1) Overview of Interrupt Functions

Table 23C.14 lists the interrupt functions of this module.

Table 23C.14 List of Interrupt Functions

Bit	Interrupt Name	Cause of Interrupt	Related Status
VBINT	VBUS interrupt	When a change in the state of the VBUS input pin has been detected (low to high or high to low)	VBSTS
RESM	Resume interrupt	When a change in the state of the USB bus has been detected in the suspended state (J-state to K-state or J-state to SE0)	—
SOFR	Frame number update interrupt	• When an SOF packet with a different frame number is received	—
DVST	Device state transition interrupt	When a device state transition is detected • A USB bus reset is detected • The suspended state is detected • Set address request is received • Set configuration request is received	DVSQ
CTRT	Control transfer stage transition interrupt	When a stage transition is detected in control transfer • Setup stage is completed • Control write transfer status stage transition • Control read transfer status stage transition • Control transfer end • A control transfer sequence error occurred	CTSQ

Bit	Interrupt Name	Cause of Interrupt	Related Status
BEMP	Buffer empty interrupt	• When transmission of all data items in the buffer memory has been completed and the buffer is empty • When a packet whose size exceeds the maximum packet size has been received	PIPEBEMP
NRDY	Buffer not ready interrupt	• When an IN token, OUT token, or PING token has been received and a NAK response is returned • When an interval error occurred while receiving data in isochronous transfer • When an overrun/underrun occurred during isochronous transfer	PIPENRDY
BRDY	Buffer ready interrupt	When the buffer is ready (reading or writing is enabled)	PIPEBRDY
BCHG	Bus change interrupt	When a change in the state of the USB bus has been detected	—

Figure 23C.4 shows a diagram relating to interrupts of this module.

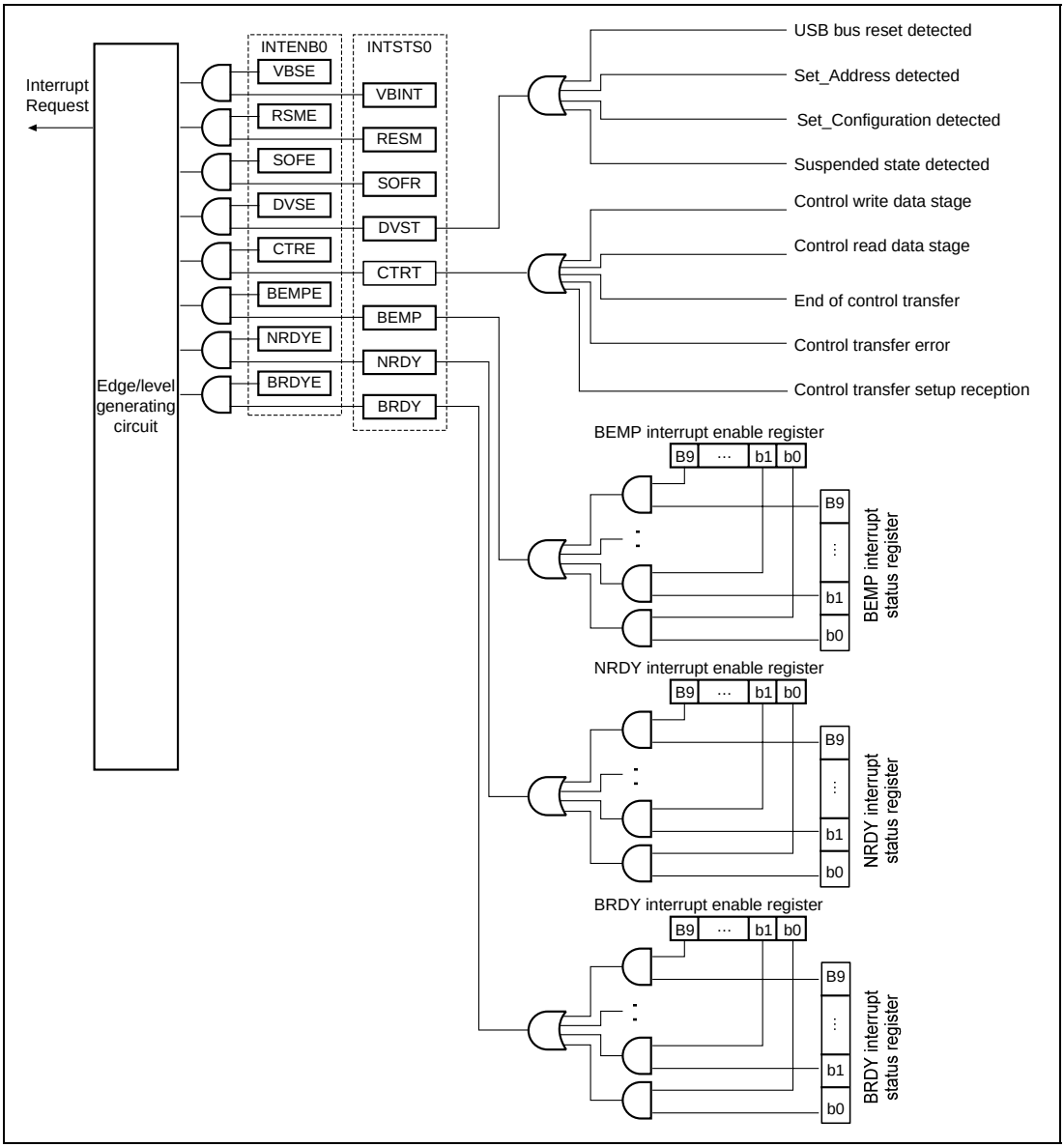


Figure 23C.4 Items Relating to Interrupts

(2) Device State Transition Interrupt (Peripheral Controller Function)

Figure 23C.5 shows a diagram of this module's device state transitions. This module controls device states and generates device state transition interrupts. However, recovery from the suspended state (resume signal detection) is detected by means of the resume interrupt. The device state transition interrupts can be enabled or disabled individually by INTENB0. The device state that made a transition can be confirmed using the DVSQ bit in INTSTS0.

To make a transition to the default state, the device state transition interrupt is generated after the reset handshake protocol has been completed.

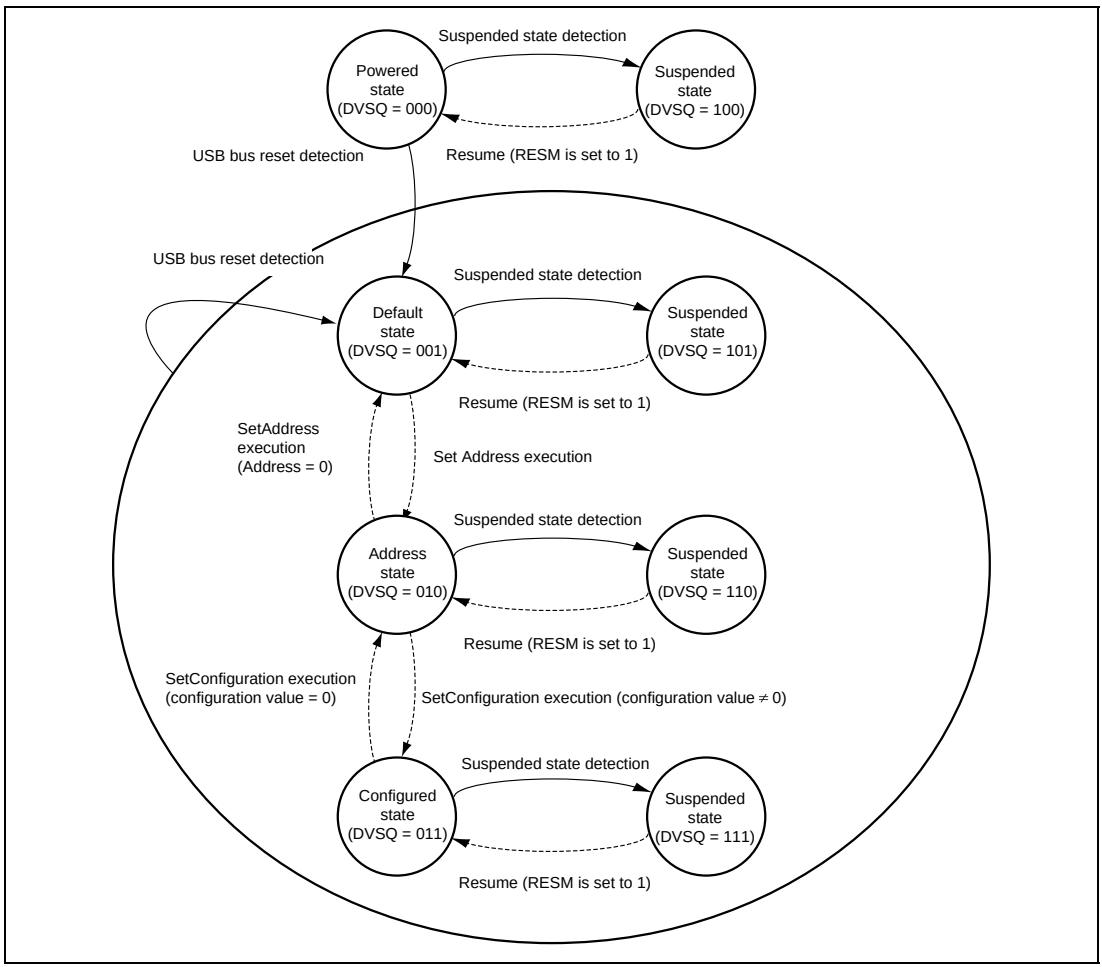


Figure 23C.5 Device State Transitions

(3) Control Transfer Stage Transition Interrupt

Figure 23C.6 shows a diagram of how this module handles the control transfer stage transitions. This module controls the control transfer sequence and generates control transfer stage transition interrupts. Control transfer stage transition interrupts can be enabled or disabled individually by INTENB0. The transfer stage that made a transition can be confirmed using the CTSQ bit in INTSTS0.

The control transfer sequence errors are described below. If an error occurs, the PID bits in DCPCTR are set to B'1x (STALL).

1. During control read transfers
   - At the IN token of the data stage, an OUT or PING token is received when there have been no data transfers at all
   - An IN token is received at the status stage
   - A packet for which the data packet is DATAPID = DATA0 is received at the status stage
2. During control write transfers
   - At the OUT token of the data stage, an IN token is received when there have been no ACK responses at all
   - A packet for which the first data packet is DATAPID = DATA0 is received at the data stage
   - At the status stage, an OUT or PING token is received
3. During control write no-data transfers
   - At the status stage, an OUT or PING token is received

At the data stage of a control write transfer, if the number of received data items exceeds the wLength value of the USB request, it cannot be recognized as a control transfer sequence error. At the status stage of a control read transfer, packets other than zero-length packets are received by an ACK response and the transfer ends normally.

When a CTRT interrupt occurs in response to a sequence error, the CTSQ = B'110 value is retained until CTRT = 0 is written from the user system (the interrupt status is cleared). Therefore, while CTSQ = B'110 is being held, the CTRT interrupt that ends the setup stage will not be generated even if a new USB request is received. (This module retains the setup stage end, and after the interrupt status has been cleared by software, a setup stage end interrupt is generated.)

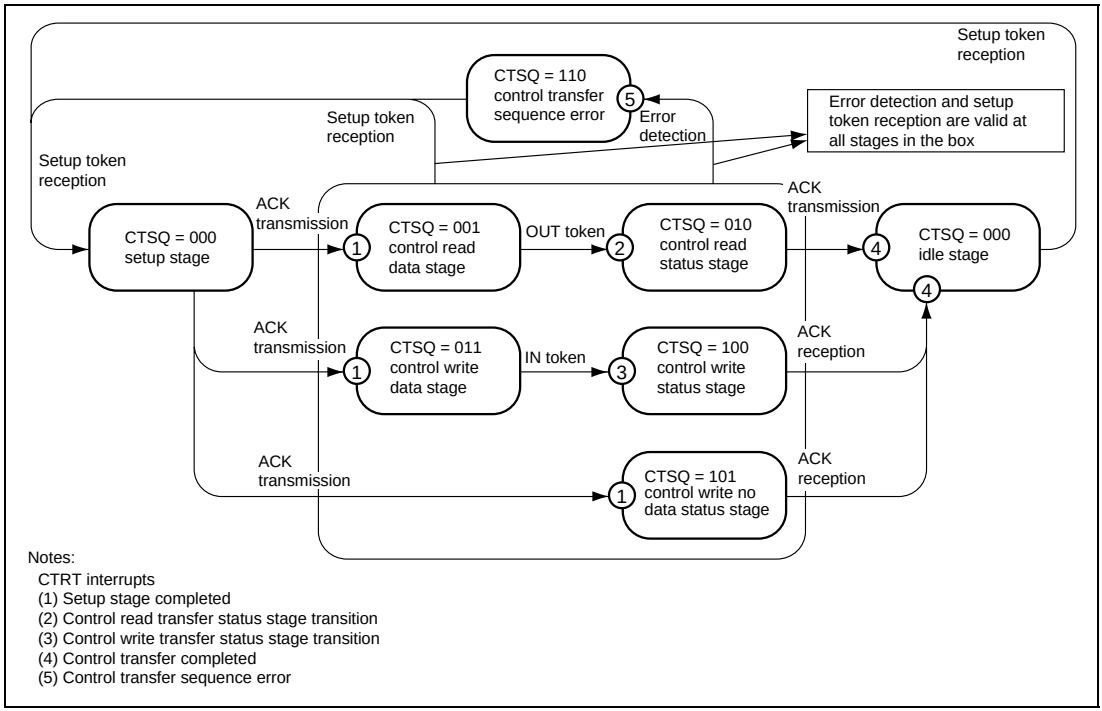


Figure 23C.6 Control Transfer Stage Transitions

23C.3.3 Pipe Control

Table 23C.15 lists the pipe setting items of this module. With USB data transfer, data transmission has to be carried out using the logic pipe called the endpoint. This module has ten pipes that are used for data transfer. Settings should be entered for each of the pipes in conjunction with the specifications of the user system.

Table 23C.15 Pipe Setting Items

Register Name	Bit Name	Setting Contents	Remarks
DCPCFG	TYPE	Specifies the transfer type	PIPE1 to PIPE9: Can be set
PIPECFG	BFRE	Selects the BRDY interrupt mode	PIPE1 to PIPE5: Can be set
	DBLB	Selects a double buffer	PIPE1 to PIPE5: Can be set
	CNTMD	Selects continuous transfer or non-continuous transfer	PIPE1 and PIPE2: Can be set only when bulk transfer has been selected PIPE3 to PIPE5: Can be set
	DIR	Selects the transfer direction	IN or OUT can be set
	EPNUM	Endpoint number	PIPE1 to PIPE9: Can be set Set a value other than B'0000 when the pipe is used.
	SHTNAK	Selects the disabled state for the pipe when transfer ends	PIPE1 and PIPE2: Can be set only when bulk transfer has been selected PIPE3 to PIPE5: Can be set
PIPEBUF	BUFSIZE	Buffer memory size	DCP: Cannot be set (fixed at 256 bytes) PIPE1 to PIPE5: Can be set (a maximum of 1 Kbyte can be specified) PIPE6 to PIPE9: Cannot be set (fixed at 64 bytes)
	BUFNMB	Buffer memory number	DCP: Cannot be set (areas fixed at H'0 to H'3) PIPE1 to PIPE5: Can be set (can be specified in areas H'8 to H'87) PIPE6 to PIPE9: Cannot be set (areas fixed at H'4 to H'7)
DCPMAXP PIPEMAXP	MXPS	Maximum packet size	The setting must conform to the USB specification

Register Name	Bit Name	Setting Contents	Remarks
PIPEPERI	IFIS	Buffer flush	PIPE1 and PIPE2: Can be set only when isochronous transfer has been selected PIPE3 to PIPE9: Cannot be set
	IITV	Interval counter	PIPE1 and PIPE2: Can be set only when isochronous transfer has been selected PIPE3 to PIPE9: Cannot be set
DCPCTR PIPEnCTR	BSTS	Buffer status	The receive and transmit buffer states of the DCP can be switched by the ISEL bit
	INBUFM	IN buffer monitor	Incorporated only in PIPE3 to PIPE5
	ATREPM	Auto response mode	PIPE1 to PIPE5: Can be set
	ACLRM	Auto buffer clear	PIPE1 to PIPE9: Can be set
	SQCLR	Sequence clear	Clears the data toggle bit
	SQSET	Sequence set	Sets the data toggle bit
	SQMON	Sequence monitor	Confirms the data toggle bit
DCPTR	PBUSY	Pipe busy confirmation	
PIPEnCTR	PID	Response PID	
PIPEnTRE	TRENB	Transaction counter enable	PIPE1 to PIPE5: Can be set
	TRCLR	Current transaction counter clear	PIPE1 to PIPE5: Can be set
PIPEnTRN	TRNCNT	Transaction counter	PIPE1 to PIPE5: Can be set

(1) Switching Procedure for Pipe Control Registers

The pipe control register bits shown below can be changed only when USB communication is disabled (PID = NAK). Figure 23C.7 shows the procedure for switching the USB communication enabled state (PID = BUF) in order to change the pipe control register settings.

The registers which must not be set in the USB communication enabled state (PID = BUF) are as follows:

- Bits in DCPCFG and DCPMAXP
- Bits SQCLR and SQSET in DCPCTR
- Bits in PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI
- Bits ATREPM, ACLRM, SQCLR, and SQSET in PIPEnCTR
- Bits in PIPEnTRE and PIPEnTRN

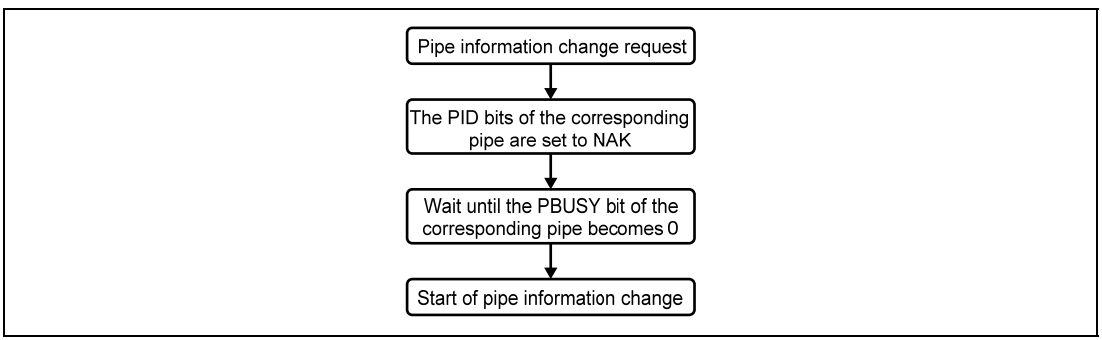


Figure 23C.7 Procedure for Changing Pipe Information When in the USB Communication Enabled State (PID = BUF)

The pipe control register bits shown below can be changed when the corresponding pipe is not set in the CURPIPE bits for any of the FIFO ports (CPU-FIFO, DMA0-FIFO, and DMA1-FIFO).

The registers which must not be set when the corresponding pipe is set in the CURPIPE bits for the FIFO ports are as follows:

- Bits in DCPCFG and DCPMAXP
- Bits in PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI

To change the pipe information, change the CURPIPE setting to a pipe that is not to be changed. Note that after the pipe information has been changed, the FIFO buffer for the DCP should be cleared using the BCLR bit.

23C.3.4 FIFO Buffer

Operations relating to the FIFO buffer in this module are described here.

(1) FIFO Buffer Allocation

Figure 23C.8 shows an example of a memory map for the FIFO buffer in this module. The FIFO buffer is an area shared by this module and the CPU used for controlling the user system. In the FIFO buffer status, there are cases when the access right to the buffer memory is allocated to the user system (CPU side), and cases when it is allocated to this module (SIE side).

The FIFO buffer sets independent areas for each pipe. In the memory areas, 64 bytes comprise one block, and the memory areas are set using the first block number and the number of blocks (specified using the BUFNMB and BUFSIZE bits in PIPEBUF). When continuous transfer mode is selected using the CNTMD bit in PIPEnCFG, the BUFSIZE bit setting must be set to an integer multiple of the maximum packet size. Also, when a double buffer is selected using the DBLB bit in PIPEnCFG, the memory area specified by the BUFSIZE bits in PIPEBUF is allocated for two buffers for the same pipe.

Moreover, three FIFO ports are used for access to the FIFO buffer (reading and writing data). A pipe is assigned to the FIFO port by specifying the pipe number using the CURPIPE bits in C/DnFIFOSEL.

The FIFO buffer statuses of the various pipes can be confirmed using the BSTS bit in DCPCTR and the INBUFM bit in PIPEnCTR. Also, the access right of the FIFO port can be confirmed using the FRDY bit in C/DnFIFOCTR.

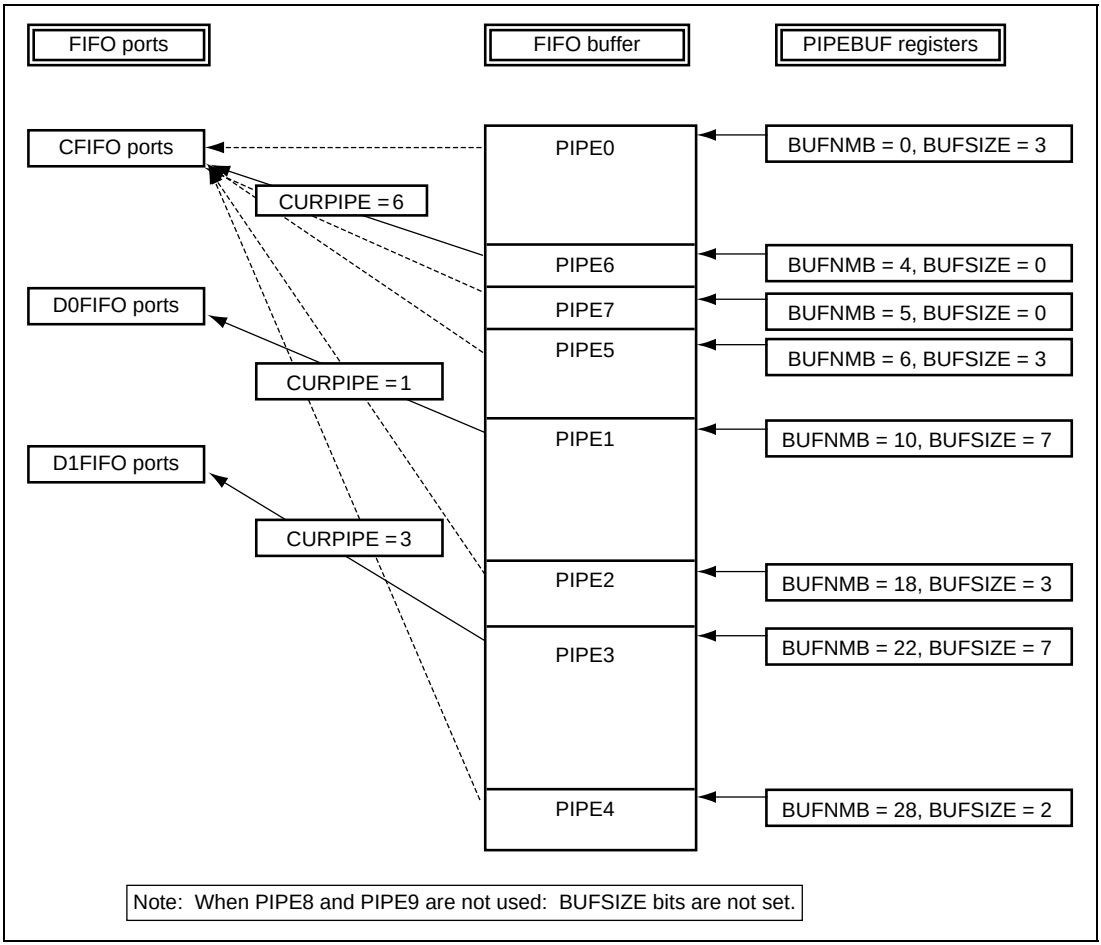


Figure 23C.8 Example of a FIFO Buffer Memory Map

(2) FIFO Buffer Clear

Table 23C.16 shows the FIFO buffer clear by this module. The FIFO buffer can be cleared using the three bits indicated below.

Table 23C.16 List of FIFO Buffer Clearing Methods

Bit Name	BCLR	DCLRM	ACLRM
Register	CFIFOCTR DnFIFOCTR	DnFIFOSEL	PIPExCTR
Function	Clears the FIFO buffer on the CPU side.	In this mode, after the data of the specified pipe has been read, the FIFO buffer is cleared automatically.	This is the auto buffer clear mode, in which all of the received packets are discarded.
Clearing method	Cleared by writing 1.	1: Mode valid 0: Mode invalid	1: Mode valid 0: Mode invalid

23C.3.5 FIFO Port Functions

The functions relating to the FIFO ports are described here. Table 23C.17 shows the settings for the FIFO port functions of this module. In write access, writing data until the buffer is full (or the maximum packet size for non-continuous transfers) automatically enables sending data to the USB bus. To enable sending of data before the buffer is full (or less than the maximum packet size for non-continuous transfers), the BVAL bit in C/DnFIFOCTR must be set to 1 to end the writing (TEND signal for DMA transfers). Also, to send a zero-length packet, the BCLR bit in C/DnFIFOCTR must be used to clear the buffer and then the BVAL bit set to 1 in order to end the writing.

In read access, reception of new packets is automatically enabled if all of the data items have been read. Data cannot be read when a zero-length packet is being received (DTLN = 0), so the BCLR bit in C/DnFIFOCTR must be used to clear the buffer. The length of the data being received can be confirmed using the DTLN bit in C/DnFIFOCTR.

Table 23C.17 FIFO Port Function Settings

Register Name	Bit Name	Function	Note
C/DnFIFOSEL	RCNT	Selects the DTLN read mode	
	REW	Buffer memory rewind (re-read, rewrite)	
	DCLRM	Automatically clears data received for a specified pipe after the data has been read	For DnFIFO only
	DREQE	Asserts the DREQ signal	For DnFIFO only
	MBW	FIFO port access bit width	
	BIGEND	Selects the FIFO port endian	
	ISEL	FIFO port access direction	For DCP only
	CURPIPE	Selects the current pipe	
C/DnFIFOCTR	BVAL	Ends writing to the buffer memory	
	BCLR	Clears the buffer memory on the CPU side	
	FRDY	FIFO port ready monitor	
	DTLN	Confirms the length of received data	

(1) FIFO Port Selection

Table 23C.18 shows the pipes that can be selected with the various FIFO ports. The pipe to be accessed is selected using the CURPIPE bits in C/DnFIFOSEL. After the pipe has been selected, confirm that the written CURPIPE value can be read correctly. (If the previous pipe number is read, this module is still in the middle of changing the pipe.) After this confirmation has been made, FRDY = 1 should be confirmed before accessing the FIFO port.

Also, the bus width to be accessed should be selected using the MBW bit. The buffer memory access direction conforms to the ISEL bit for the DCP, and conforms to the DIR bit in PIPEnCFG for the other pipes.

Table 23C.18 FIFO Port Access Categorized by Pipe

Pipe	Access Method	Port that can be Used
DCP	CPU access	CFIFO port register
PIPE1 to PIPE9	CPU access	CFIFO port register
		DnFIFO port register
	DMA access	DnFIFO port register

(2) DnFIFO Auto Clear Mode (DnFIFO Port Reading Direction)

If 1 is set for the DCLRM bit in DnFIFOSEL, this module automatically clears the buffer memory of the corresponding pipe when reading of the data from the buffer memory has been completed.

Table 23C.19 shows the packet reception and buffer memory clearing processing for each of the various settings. As shown in table 23C.19, the buffer clearing conditions depend on the BFRE bit setting. Using the DCLRM bit eliminates the need for the buffer to be cleared by software even if a situation occurs that necessitates clearing of the buffer. This makes it possible to carry out DMA transfers without involving software.

This function can be set only when reading from the buffer memory.

Table 23C.19 Packet Reception and Buffer Memory Clearing Processing by Software

Register Setting / Buffer Status When Packet is Received	DCLRM = 0		DCLRM = 1	
	BFRE = 0	BFRE = 1	BFRE = 0	BFRE = 1
Buffer full	Doesn't need to be cleared	Doesn't need to be cleared	Doesn't need to be cleared	Doesn't need to be cleared
Zero-length packet reception	Needs to be cleared	Needs to be cleared	Doesn't need to be cleared	Doesn't need to be cleared
Normal short packet reception	Doesn't need to be cleared	Needs to be cleared	Doesn't need to be cleared	Doesn't need to be cleared
Transaction count ended	Doesn't need to be cleared	Needs to be cleared	Doesn't need to be cleared	Doesn't need to be cleared

(3) BRDY Interrupt Timing Selection Function

By setting the BFRE bit in PIPECFG, it is possible to keep the BRDY interrupt from being generated when a data packet consisting of the maximum packet size is received.

When using DMA transfers, this function can be used to generate an interrupt only when the last data item has been received. The last data item refers to the reception of a short packet, or the ending of the transaction counter. When the BFRE bit is set to 1, the BRDY interrupt is generated after the received data has been read. When the DTLN bit in DnFIFOCTR is read, the length of the data received in the last data packet that have been received when BRDY interrupt is generated can be confirmed.

Table 23C.20 shows the timing at which the BRDY interrupts are generated by this module.

Table 23C.20 Timing at which BRDY Interrupts are Generated

Buffer State When Packet is Received \ Register setting	BFRE = 0	BFRE = 1
Buffer full (normal packet received)	When packet is received	Not generated
Zero-length packet received	When packet is received	When packet is received
Normal short packet received	When packet is received	When reading of the received data from the buffer memory has been completed
Transaction count ended	When packet is received	When reading of the received data from the buffer memory has been completed

Note: This function is valid only when reading from the buffer memory. When writing to the buffer memory, the BFRE bit should be fixed at 0.

23C.3.6 Control Transfers (DCP)

Data transfers of the data stage of control transfers are done using the default control pipe (DCP). The DCP buffer memory is a 256-byte single buffer, and is a fixed area that is shared for both control read transfers and control write transfers. The buffer memory can be accessed through only the CFIFO port.

This module always sends an ACK response in response to a normal setup packet for this module. The operations of this module in the setup stage are shown below.

1. When a new SETUP packet is received, this module sets the following bits:
   — Set the VALID bit in INTSTS0 to 1.
   — Set the PID bits in DCPCTR to NAK.
   — Clear the CCPL bit in DCPCTR to 0.
2. When a data packet is received right after the SETUP packet, this module stores the USB request parameters in USBREQ, USBVAL, USBINDX, and USBLENG.

Response processing with respect to the control transfer should always be carried out after first setting VALID = 0. In the VALID = 1 state, PID = BUF cannot be set, and so the data stage cannot be terminated.

Using the function of the VALID bit, this module is able to interrupt the processing of a request currently being processed if a new USB request is received during a control transfer, and can send a response to the newest request.

Also, this module automatically judges the direction bit (bit 8 of bmRequestType) and the request data length (wLength) of the USB request that was received, and then distinguishes between control read transfers, control write transfers, and no-data control transfers, and controls the stage transitions. For an incorrect sequence, the sequence error of the control transfer stage transition interrupt is generated, and software is notified. For information on the stage control of this module, see figure 23C.6.

(1) Data Stage

Data transfers corresponding to USB requests that have been received should be done using the DCP. Before accessing the DCP buffer memory, the access direction should be specified using the ISEL bit in CFIFOSEL.

If the data being transferred is larger than the size of the DCP buffer memory, the data transfer should be carried out using the BRDY interrupt for control write transfers and the BEMP interrupt for control read transfers.

With control write transfers during high-speed operation, the NYET handshake response is returned based on the state of the buffer memory.

(2) Status Stage

Control transfers are ended by setting the CCPL bit to 1 with the PID bits in DCPCTR set to PID = BUF.

After the above settings have been entered, this module automatically executes the status stage in accordance with the data transfer direction determined at the setup stage. The specific procedure is as follows.

1. For control read transfers
   This module sends a zero-length packet and receives an ACK response from the USB host controller.
2. For control write transfers and no-data control transfers
   The zero-length packet is received from the USB host, and this module sends an ACK response.

(3) Control Transfer Auto Response Function

This module automatically responds to a normal SET_ADDRESS request. If any of the following errors occur in the SET_ADDRESS request, a response from software is necessary.

1. Any transfer other than a control write transfer: bmRequestType ≠ H '00
2. If a request error occurs: wIndex ≠ H'00
3. Any transfer other than a no-data control transfer: wLength ≠ H'00
4. If a request error occurs: wValue > H'7F
5. Control transfer of a device state error: DVSQ = B'011 (Configured)

For all requests other than the SET_ADDRESS request, a response from the corresponding software is required.

23C.3.7 Bulk Transfers (PIPE1 to PIPE5)

The buffer memory specifications for bulk transfers (single/double buffer setting or continuous/non-continuous transfer mode setting) can be selected. The maximum size that can be set for the buffer memory is 1 Kbyte. The buffer memory state is controlled by this module, with a response sent automatically for a PING packet/NYET handshake.

23C.3.8 Interrupt Transfers (PIPE6 to PIPE9)

This module carries out interrupt transfers in accordance with the timing controlled by the host controller. For interrupt transfers, PING packets are ignored (no responses are sent), and the ACK, NAK, and STALL responses are returned without an NYET handshake response being made.

This module does not support high-bandwidth transfer for interrupt transfers.

23C.3.9 Isochronous Transfers (PIPE1 and PIPE2)

This module has the following functions pertaining to isochronous transfers.

1. Notification of isochronous transfer error information
2. Interval counter (specified by the IITV bits)
3. Isochronous-IN transfer data setup control (IDLY function)
4. Isochronous-IN transfer buffer flush function (specified by the IFIS bit)
5. SOF pulse output function

This module does not support high-bandwidth transfer for isochronous transfers.

(1) Interval Counter

The isochronous transfer interval can be set using the IITV bits in PIPEPERI. The interval counter enables the functions shown in Table 23C.21.

Table 23C.21 Interval Counter Functions

Transfer Direction	Function	Conditions for Detection
IN	Transfer buffer flush function	When an IN token cannot be normally received in the interval frame during an isochronous-IN transfer
OUT	Notifies that a token is not being received	When an OUT token cannot be normally received in the interval frame during an isochronous-OUT transfer

Interval counting is carried out when an SOF packet is received or for interpolated SOFs, so the isochronism can be maintained even if an SOF packet is corrupted. The frame interval that can be set is 2^{IITV} (μ) frames.

This controller initializes the interval counter under the following conditions.

1. Hardware reset
   The IITV bits are initialized.
2. Buffer memory clear using the ACLRM bit
   The IITV bits are not initialized but the counter value is. Setting the ACLRM bit to 0 starts counting from the value set in the IITV bits.

After the interval counter has been initialized, the interval counter is started under the following conditions 1 or 2 when a packet has been transferred normally.

1. An SOF packet is received following transmission of data in response to an IN token in the PID = BUF state.
2. An SOF packet is received after data following an OUT token is received in the PID = BUF state.

The interval counter is not initialized under the conditions noted below.

1. When the PID bits are set to NAK or STALL

   The interval counter does not stop. This module attempts the transactions at the subsequent interval.
2. At an USB bus reset or when the USB is suspended

   The IITV bits are not initialized. When an SOF packet is received, the interval counter is restarted from the value prior to the reception of the SOF packet.

(2) Setup of Data to be Transmitted using Isochronous Transfer

With isochronous data transmission using this controller, after data has been written to the buffer memory, a data packet can be sent with the next frame in which an SOF packet is detected. This function is called the isochronous transfer transmission data setup function.

If a double buffer is used for the buffer memory, transmission will be enabled for only one of the two buffers even after the writing of data to both buffers has been completed. The enabled buffer memory is the one to which the data writing was completed first. For this reason, even if multiple IN tokens are received, the buffer memory that can be sent is only one packet's worth of data.

When an IN token is received, if the buffer memory is in the transmission enabled state, this module transmits the data as a normal response. If the buffer memory is not in the transmission enabled state, however, a zero-length packet is sent and an underrun error occurs.

Sending of a zero-length packet is displayed in the figure as Null, in a shaded box. Figure 23C.9 shows an example of transmission using the isochronous transfer transmission data setup function with this module, when IITV = 0 (every frame) has been set.

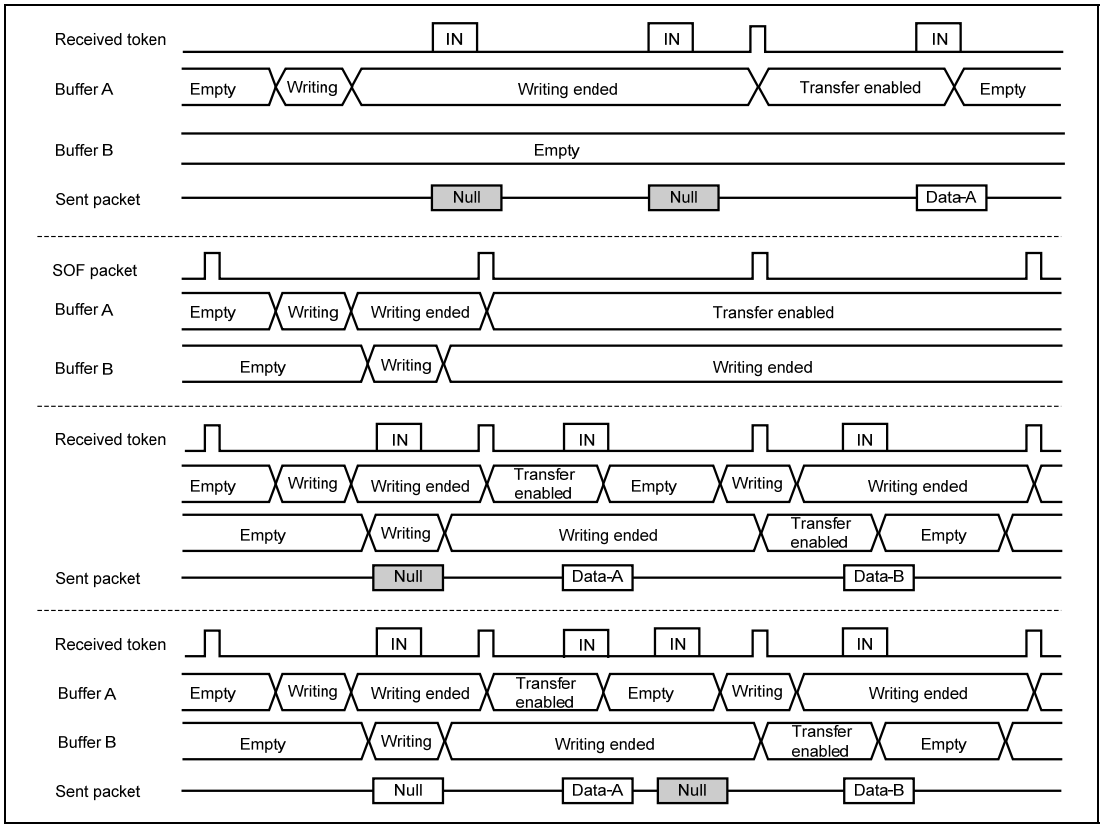


Figure 23C.9 Example of Data Setup Function Operation

(3) Transmission Buffer Flush using Isochronous Transfer

If a (μ) SOF packet of the next frame is received without receiving an IN token in the interval frame during isochronous data transmission, this controller operates as if the IN token had been corrupted, and clears the buffer for which transmission is enabled, putting that buffer in the writing enabled state.

If a double buffer is being used and writing to both buffers has been completed, the buffer memory that was cleared is seen as the data having been sent at the same interval frame, and transmission is enabled for the buffer memory that is not discarded with (μ) SOF packet reception.

The timing for starting operation of the buffer flush function varies depending on the value set for the IITV bits.

- When IITV = 0

  The buffer flush operation starts from the next frame after the pipe becomes valid.
- In any cases other than IITV = 0

  The buffer flush operation is carried out subsequent to the first normal transaction.

Figure 23C.10 shows an example of the buffer flush function of this module. When an unanticipated token is received prior to the interval frame, this module sends the written data or a zero-length packet as an underrun error, according to the data setup status.

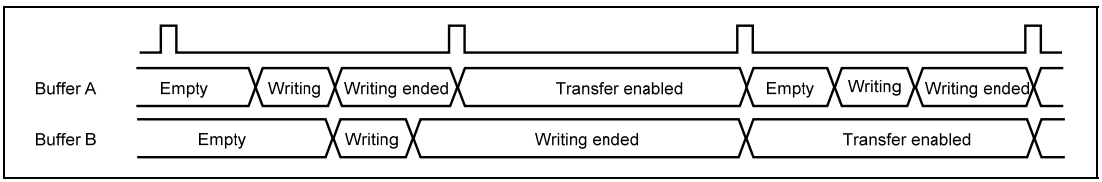


Figure 23C.10 Example of Buffer Flush Function Operation

Figure 23C.11 shows an example of this controller generating an interval error. There are five types of interval errors, as shown below. The interval error is generated at the timing indicated by (1) in the figure, and the buffer flush function is activated.

If an interval error occurs during an IN transfer, the buffer flush function is activated; and if it occurs during an OUT transfer, an NRDY interrupt is generated.

The OVRN bit should be used to distinguish between NRDY interrupts such as received packet errors and overrun errors.

In response to tokens that are shaded in the figure, responses occur based on the buffer memory status.

- IN direction
  - If the buffer is in the transmission enabled state, the data is transferred as a normal response.
  - If the buffer is in the transmission disabled state, a zero-length packet is sent and an underrun error occurs.
- OUT direction
  - If the buffer is in the reception enabled state, the data is received as a normal response.
  - If the buffer is in the reception disabled state, the data is discarded and an overrun error occurs.

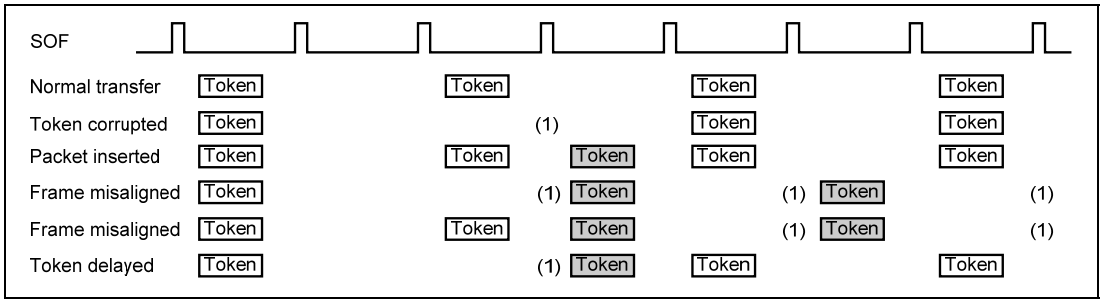


Figure 23C.11 Example of an Interval Error Being Generated when IITV = 1

23C.3.10 SOF Interpolation Function

If data could not be received at intervals of 1 ms (when using full-speed operation) or 125 µs (when using high-speed operation) because an SOF packet was corrupted or missing, this module internally interpolates the SOF. The SOF interpolation operation begins when an SOF packet is received with SYSCFG0.USBE = 1 and SUSPMODE.SUSPM = 1 set. The interpolation function is initialized under the following conditions.

- Hardware reset
- USB bus reset
- Suspended state detected

Also, the SOF interpolation operates under the following specifications.

- The frame interval (125 µs or 1 ms) conforms to the results of the reset handshake protocol.
- The interpolation function is not activated until an SOF packet is received.
- After the first SOF packet is received, either 125 µs or 1 ms is counted with the internal clock of 48 MHz, and interpolation is carried out.
- After the second and subsequent SOF packets are received, interpolation is carried out at the previous reception interval.
- Interpolation is not carried out in the suspended state or while a USB bus reset is being received. (With suspended transitions in high-speed operation, interpolation continues for 3 ms after the last packet is received.)

The SOF interpolation can also operate with the following functions.

- Refreshing of the frame number and the µ-frame number
- SOFR interrupt timing and µSOF lock
- Isochronous transfer interval counter

If an SOF packet is missing during full-speed operation, the FRNM bits in FRMNUM0 are not refreshed. If a µSOF packet is missing during high-speed operation, the UFRNM bits in FRMNUM1 are refreshed. However, if a µSOF packet of µFRNM = B'000 is missing, the FRNM bits are not refreshed. In this case, the FRNM bits are not refreshed even if successive µSOF packets other than µFRNM = B'000 are received normally.

Section 24 Gigabit Ethernet Controller (GETHER)

This LSI has an on-chip Gigabit Ethernet controller (GETHER) supporting the Ethernet or the IEEE802.3 MAC (Media Access Control) layer standard. Connecting a physical-layer LSI (PHY-LSI) complying with this standard enables the GETHER to perform transmission and reception of Ethernet/IEEE802.3 frames. The Ethernet controller in this LSI has one MAC layer interface port, which can be made to perform transmission and reception independently.

The GETHER can transfer the transmitted or received Ethernet frame data to and from the transmit/receive buffer in the memory at high speed using a dedicated direct memory access controller (E-DMAC).

24.1 Features

- MAC (Media Access Control) function
  Constructs/deconstructs data frames (frame format conforming to IEEE802.3, 2000 Edition)
  Supports transfer at 10, 100, and 1000 Mbps
  Supports full-duplex and half-duplex modes
  One channel (GETHER0)
  Flow control conforming to IEEE802.3x
  Supports three PHY interfaces conforming to IEEE802.3
  — GMII (Gigabit Media Independent Interface)
  — MII (Media Independent Interface)
  — RMII (Reduced Media Independent Interface)
  Upward protocol support (checksum) function
- E-DMAC (Direct Memory Access Controller for Ethernet controller) function
  Data transfer between GETHER and external/internal memory
  One channel
  32-byte burst transfer
  Supports single-frame/single-descriptor operation and single-frame/multi-descriptor (multi-buffer) operation
  Transfer data width: 32 bits
  Transmit/receive FIFO (for transmission: 2 Kbytes, for reception: 4 Kbytes)
  Function for calculating the intelligent checksum value

Figure 24.1 shows the configuration of the GETHER.

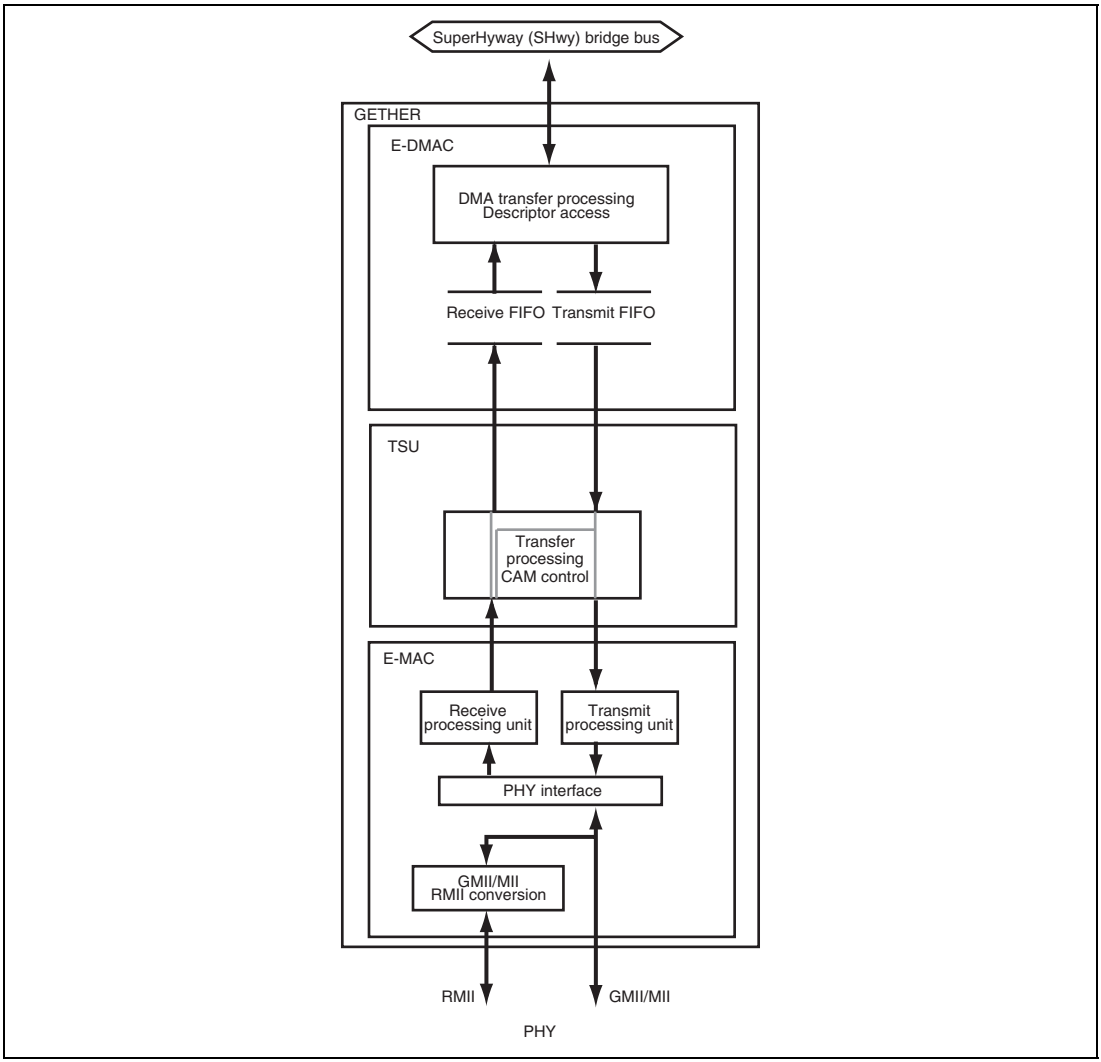


Figure 24.1 Configuration of GETHER

24.2 Input/Output Pins

Table 24.1 lists the pin configuration of the GETHER.

Table 24.1 Pin Configuration

Name	Port	Abbreviation	I/O	Function
Transmit clock	0	ET0_TX-CLK*	Input	ET0_TX-EN, ET0_ETXD3 to ET0_ETXD0, ET0_TX-ER timing reference signal
Transmit enable		ET0_TX-EN*	Output	Indicates that transmit data is ready on ET0_ETXD3 to ET0_ETXD0
MII/GMII transmit data		ET0_ETXD3 to ET0_ETXD0*	Output	4-bit MII transmit data or lower four bits of GMII transmit data
GMII transmit data		GET0_ETXD7 to GET0_ETXD4	Output	Upper four bits of GMII transmit data
Collision detection		ET0_COL*	Input	Collision detection signal
Transmit error		ET0_TX-ER*	Output	Notifies PHY-LSI of error during transmission
Receive clock		ET0_RX-CLK*	Input	ET0_RX-DV, ET0_ERXD3 to ET0_ERXD0, ET0_RX-ER timing reference signal
Receive data valid		ET0_RX-DV*	Input	Indicates that valid receive data is on ET0_ERXD3 to ET0_ERXD0
MII/GMII receive data		ET0_ERXD3 to ET0_ERXD0*	Input	4-bit MII receive data or lower four bits of GMII receive data (MII and GMII)
GMII receive data		GET0_ERXD7 to GET0_ERXD4	Input	Upper four bits of GMII receive data
Receive error		ET0_RX-ER*	Input	Identifies error state occurred during data reception
Carrier detection		ET0_CRS*	Input	Carrier detection signal
Management data clock		ET0_MDC*	Output	Reference clock signal for information transfer via ET0_MDIO
Management data I/O		ET0_MDIO*	I/O	Bidirectional signal for exchange of management information between STA and PHY

Name	Port	Abbreviation	I/O	Function
RMII management data clock	0	RMII0_MDC	Output	Reference clock signal for information transfer via RMII0_MDIO in RMII mode
RMII management data I/O		RMII0_MDIO	I/O	Bidirectional signal for exchange of management information between STA and PHY in RMII mode
RMII management data clock (mirror 0 pin)		RMII0M0_MDC	Output	Reference clock signal for information transfer via RMII0M0_MDIO in RMII mode (mirror 0 pin)
RMII management data I/O (mirror 0 pin)		RMII0M0_MDIO	I/O	Bidirectional signal for exchange of management information between STA and PHY in RMII mode (mirror 0 pin)
RMII management data clock (mirror 1 pin)		RMII0M1_MDC	Output	Reference clock signal for information transfer via RMII0M1_MDIO in RMII mode (mirror 1 pin)
RMII management data I/O (mirror 1 pin)		RMII0M1_MDIO	I/O	Bidirectional signal for exchange of management information between STA and PHY in RMII mode (mirror 1 pin)
Link status		ET0_LINKSTA	Input	Inputs link status from PHY-LSI
Wake-On-LAN		ET0_WOL	Output	Signal indicating reception of Magic Packet
PHY interrupt		ET0_PHY-INT	Input	Interrupt signal from PHY
GMII transmit clock		GET0_GTX-CLK	Output	Transmit signal timing reference signal in GMII mode
RMII carrier detection		RMII0_CRS_DV	Input	Carrier detection signal in RMII mode
RMII receive error		RMII0_RX_ER	Input	Identifies error state occurred during data reception in RMII mode
RMII receive data		RMII0_RXD0	Input	2-bit receive data in RMII mode
RMII receive data		RMII0_RXD1	Input	2-bit receive data in RMII mode
RMII transmit enable		RMII0_TXD_EN	Output	Indicates that transmit data is ready on RMII0_TXD0 and RMII0_TXD1 in RMII mode

Name	Port	Abbreviation	I/O	Function
RMII transmit data	0	RMII0_TXD0	Output	2-bit transmit data in RMII mode
RMII transmit data		RMII0_TXD1	Output	2-bit transmit data in RMII mode
125-MHz reference clock	Common	REF125CK	Input	Transmit clock generation signal in GMII mode
50-MHz reference clock		REF50CK	Input	Transmit clock generation signal in RMII mode

Note: MII signal conforming to IEEE802.3u

24.3 Register Descriptions

Table 24.2 shows the configuration of registers of the GETHER. Table 24.3 shows the state of registers in each processing mode.

Table 24.2 Register Configuration

Name	Abbreviation	R/W	P4 Area Address	Area 7 Address	Access Size
Software reset register	ARSTR	R/W	H'FEE0 1800	H'1EE0 1800	32
E-MAC mode register	ECMR0	R/W	H'FEE0 0500	H'1EE0 0500	32
E-MAC status register	ECSR0	R/W	H'FEE0 0510	H'1EE0 0510	32
E-MAC interrupt permission register	ECSIPR0	R/W	H'FEE0 0518	H'1EE0 0518	32
PHY interface register	PIR0	R/W	H'FEE0 0520	H'1EE0 0520	32
MAC address high register	MAHR0	R/W	H'FEE0 05C0	H'1EE0 05C0	32
MAC address low register	MALR0	R/W	H'FEE0 05C8	H'1EE0 05C8	32
Receive frame length register	RFLR0	R/W	H'FEE0 0508	H'1EE0 0508	32
PHY status register	PSR0	R	H'FEE0 0528	H'1EE0 0528	32
PHY_INT polarity register	PIPR0	R/W	H'FEE0 052C	H'1EE0 052C	32
Transmit retry over counter register	TROCR0	R/W	H'FEE0 0700	H'1EE0 0700	32
Delayed collision detect counter register	CDCR0	R/W	H'FEE0 0708	H'1EE0 0708	32
Lost carrier counter register	LCCR0	R/W	H'FEE0 0710	H'1EE0 0710	32
CRC error frame receive counter register	CEFCR0	R/W	H'FEE0 0740	H'1EE0 0740	32
Frame receive error counter register	FRECR0	R/W	H'FEE0 0748	H'1EE0 0748	32
Too-short frame receive counter register	TSFRCR0	R/W	H'FEE0 0750	H'1EE0 0750	32
Too-long frame receive counter register	TLFRCR0	R/W	H'FEE0 0758	H'1EE0 0758	32
Residual-bit frame receive counter register	RFCR0	R/W	H'FEE0 0760	H'1EE0 0760	32
Carrier extension loss counter register	CERCR0	R/W	H'FEE0 0768	H'1EE0 0768	32

Name	Abbreviation	R/W	P4 Area Address	Area 7 Address	Access Size
Carrier extension error counter register	CEECR0	R/W	H'FEE0 0770	H'1EE0 0770	32
Multicast address frame receive counter register	MAFCR0	R/W	H'FEE0 0778	H'1EE0 0778	32
Automatic PAUSE frame register	APR0	R/W	H'FEE0 0554	H'1EE0 0554	32
Manual PAUSE frame register	MPR0	R/W	H'FEE0 0558	H'1EE0 0558	32
Automatic PAUSE frame retransmit count register	TPAUSER0	R/W	H'FEE0 0564	H'1EE0 0564	32
PAUSE frame transmit counter register	PFTCR0	R	H'FEE0 055C	H'1EE0 055C	32
PAUSE frame receive counter register	PFRCR0	R	H'FEE0 0560	H'1EE0 0560	32
GETHER mode register	GECMR0	R/W	H'FEE0 05B0	H'1EE0 05B0	32
Burst cycle count upper-limit register	BCULR0	R/W	H'FEE0 05B4	H'1EE0 05B4	32
TSU counter reset register	TSU_CTRST	R/W	H'FEE0 1804	H'1EE0 1804	32
Relay function set register (common)	TSU_FWSLC	R/W	H'FEE0 1838	H'1EE0 1838	32
VLANtag set register	TSU_VTAG0	R/W	H'FEE0 1858	H'1EE0 1858	32
CAM entry table busy register	TSU_ADSBSY	R	H'FEE0 1860	H'1EE0 1860	32
CAM entry table enable register	TSU_TEN	R/W	H'FEE0 1864	H'1EE0 1864	32
CAM entry table POST1 register	TSU_POST1	R/W	H'FEE0 1870	H'1EE0 1870	32
CAM entry table POST2 register	TSU_POST2	R/W	H'FEE0 1874	H'1EE0 1874	32
CAM entry table POST3 register	TSU_POST3	R/W	H'FEE0 1878	H'1EE0 1878	32
CAM entry table POST4 register	TSU_POST4	R/W	H'FEE0 187C	H'1EE0 187C	32
CAM entry table 0H register	TSU_ADRH0	R/W	H'FEE0 1900	H'1EE0 1900	32
CAM entry table 1H register	TSU_ADRH1	R/W	H'FEE0 1908	H'1EE0 1908	32
CAM entry table 2H register	TSU_ADRH2	R/W	H'FEE0 1910	H'1EE0 1910	32
CAM entry table 3H register	TSU_ADRH3	R/W	H'FEE0 1918	H'1EE0 1918	32
CAM entry table 4H register	TSU_ADRH4	R/W	H'FEE0 1920	H'1EE0 1920	32
CAM entry table 5H register	TSU_ADRH5	R/W	H'FEE0 1928	H'1EE0 1928	32
CAM entry table 6H register	TSU_ADRH6	R/W	H'FEE0 1930	H'1EE0 1930	32
CAM entry table 7H register	TSU_ADRH7	R/W	H'FEE0 1938	H'1EE0 1938	32
CAM entry table 8H register	TSU_ADRH8	R/W	H'FEE0 1940	H'1EE0 1940	32
CAM entry table 9H register	TSU_ADRH9	R/W	H'FEE0 1948	H'1EE0 1948	32

Name	Abbreviation	R/W	P4 Area Address	Area 7 Address	Access Size
CAM entry table 10H register	TSU_ADRH10	R/W	H'FEE0 1950	H'1EE0 1950	32
CAM entry table 11H register	TSU_ADRH11	R/W	H'FEE0 1958	H'1EE0 1958	32
CAM entry table 12H register	TSU_ADRH12	R/W	H'FEE0 1960	H'1EE0 1960	32
CAM entry table 13H register	TSU_ADRH13	R/W	H'FEE0 1968	H'1EE0 1968	32
CAM entry table 14H register	TSU_ADRH14	R/W	H'FEE0 1970	H'1EE0 1970	32
CAM entry table 15H register	TSU_ADRH15	R/W	H'FEE0 1978	H'1EE0 1978	32
CAM entry table 16H register	TSU_ADRH16	R/W	H'FEE0 1980	H'1EE0 1980	32
CAM entry table 17H register	TSU_ADRH17	R/W	H'FEE0 1988	H'1EE0 1988	32
CAM entry table 18H register	TSU_ADRH18	R/W	H'FEE0 1990	H'1EE0 1990	32
CAM entry table 19H register	TSU_ADRH19	R/W	H'FEE0 1998	H'1EE0 1998	32
CAM entry table 20H register	TSU_ADRH20	R/W	H'FEE0 19A0	H'1EE0 19A0	32
CAM entry table 21H register	TSU_ADRH21	R/W	H'FEE0 19A8	H'1EE0 19A8	32
CAM entry table 22H register	TSU_ADRH22	R/W	H'FEE0 19B0	H'1EE0 19B0	32
CAM entry table 23H register	TSU_ADRH23	R/W	H'FEE0 19B8	H'1EE0 19B8	32
CAM entry table 24H register	TSU_ADRH24	R/W	H'FEE0 19C0	H'1EE0 19C0	32
CAM entry table 25H register	TSU_ADRH25	R/W	H'FEE0 19C8	H'1EE0 19C8	32
CAM entry table 26H register	TSU_ADRH26	R/W	H'FEE0 19D0	H'1EE0 19D0	32
CAM entry table 27H register	TSU_ADRH27	R/W	H'FEE0 19D8	H'1EE0 19D8	32
CAM entry table 28H register	TSU_ADRH28	R/W	H'FEE0 19E0	H'1EE0 19E0	32
CAM entry table 29H register	TSU_ADRH29	R/W	H'FEE0 19E8	H'1EE0 19E8	32
CAM entry table 30H register	TSU_ADRH30	R/W	H'FEE0 19F0	H'1EE0 19F0	32
CAM entry table 31H register	TSU_ADRH31	R/W	H'FEE0 19F8	H'1EE0 19F8	32
CAM entry table 0L register	TSU_ADRL0	R/W	H'FEE0 1904	H'1EE0 1904	32
CAM entry table 1L register	TSU_ADRL1	R/W	H'FEE0 190C	H'1EE0 190C	32
CAM entry table 2L register	TSU_ADRL2	R/W	H'FEE0 1914	H'1EE0 1914	32
CAM entry table 3L register	TSU_ADRL3	R/W	H'FEE0 191C	H'1EE0 191C	32
CAM entry table 4L register	TSU_ADRL4	R/W	H'FEE0 1924	H'1EE0 1924	32
CAM entry table 5L register	TSU_ADRL5	R/W	H'FEE0 192C	H'1EE0 192C	32
CAM entry table 6L register	TSU_ADRL6	R/W	H'FEE0 1934	H'1EE0 1934	32
CAM entry table 7L register	TSU_ADRL7	R/W	H'FEE0 193C	H'1EE0 193C	32
CAM entry table 8L register	TSU_ADRL8	R/W	H'FEE0 1944	H'1EE0 1944	32
CAM entry table 9L register	TSU_ADRL9	R/W	H'FEE0 194C	H'1EE0 194C	32

Name	Abbreviation	R/W	P4 Area Address	Area 7 Address	Access Size
CAM entry table 10L register	TSU_ADRL10	R/W	H'FEE0 1954	H'1EE0 1954	32
CAM entry table 11L register	TSU_ADRL11	R/W	H'FEE0 195C	H'1EE0 195C	32
CAM entry table 12L register	TSU_ADRL12	R/W	H'FEE0 1964	H'1EE0 1964	32
CAM entry table 13L register	TSU_ADRL13	R/W	H'FEE0 196C	H'1EE0 196C	32
CAM entry table 14L register	TSU_ADRL14	R/W	H'FEE0 1974	H'1EE0 1974	32
CAM entry table 15L register	TSU_ADRL15	R/W	H'FEE0 197C	H'1EE0 197C	32
CAM entry table 16L register	TSU_ADRL16	R/W	H'FEE0 1984	H'1EE0 1984	32
CAM entry table 17L register	TSU_ADRL17	R/W	H'FEE0 198C	H'1EE0 198C	32
CAM entry table 18L register	TSU_ADRL18	R/W	H'FEE0 1994	H'1EE0 1994	32
CAM entry table 19L register	TSU_ADRL19	R/W	H'FEE0 199C	H'1EE0 199C	32
CAM entry table 20L register	TSU_ADRL20	R/W	H'FEE0 19A4	H'1EE0 19A4	32
CAM entry table 21L register	TSU_ADRL21	R/W	H'FEE0 19AC	H'1EE0 19AC	32
CAM entry table 22L register	TSU_ADRL22	R/W	H'FEE0 19B4	H'1EE0 19B4	32
CAM entry table 23L register	TSU_ADRL23	R/W	H'FEE0 19BC	H'1EE0 19BC	32
CAM entry table 24L register	TSU_ADRL24	R/W	H'FEE0 19C4	H'1EE0 19C4	32
CAM entry table 25L register	TSU_ADRL25	R/W	H'FEE0 19CC	H'1EE0 19CC	32
CAM entry table 26L register	TSU_ADRL26	R/W	H'FEE0 19D4	H'1EE0 19D4	32
CAM entry table 27L register	TSU_ADRL27	R/W	H'FEE0 19DC	H'1EE0 19DC	32
CAM entry table 28L register	TSU_ADRL28	R/W	H'FEE0 19E4	H'1EE0 19E4	32
CAM entry table 29L register	TSU_ADRL29	R/W	H'FEE0 19EC	H'1EE0 19EC	32
CAM entry table 30L register	TSU_ADRL30	R/W	H'FEE0 19F4	H'1EE0 19F4	32
CAM entry table 31L register	TSU_ADRL31	R/W	H'FEE0 19FC	H'1EE0 19FC	32
Transmit frame counter register (normal transmission only)	TXNLCR0	R	H'FEE0 1880	H'1EE0 1880	32
Transmit frame counter register (normal and erroneous transmission)	TXALCR0	R	H'FEE0 1884	H'1EE0 1884	32
Receive frame counter register (normal reception only)	RXNLCR0	R	H'FEE0 1888	H'1EE0 1888	32
Receive frame counter register (normal and erroneous reception)	RXALCR0	R	H'FEE0 188C	H'1EE0 188C	32
E-DMAC start register	EDSR0	W	H'FEE0 0000	H'1EE0 0000	32
E-DMAC mode register	EDMR0	R/W	H'FEE0 0400	H'1EE0 0400	32

Name	Abbreviation	R/W	P4 Area Address	Area 7 Address	Access Size
E-DMAC transmit request register	EDTRR0	R/W	H'FEE0 0408	H'1EE0 0408	32
E-DMAC receive request register	EDRRR0	R/W	H'FEE0 0410	H'1EE0 0410	32
E-MAC/E-DMAC status register	EESR0	R/W	H'FEE0 0428	H'1EE0 0428	32
E-MAC/E-DMAC status interrupt permission register	EESIPR0	R/W	H'FEE0 0430	H'1EE0 0430	32
Transmit descriptor list start address register	TDLAR0	R/W	H'FEE0 0010	H'1EE0 0010	32
Transmit descriptor fetch address register	TDFAR0	R/W	H'FEE0 0014	H'1EE0 0014	32
Transmit descriptor finished address register	TDFXR0	R/W	H'FEE0 0018	H'1EE0 0018	32
Transmit descriptor final flag register	TDFFR0	R/W	H'FEE0 001C	H'1EE0 001C	32
Receive descriptor list start address register	RDLAR0	R/W	H'FEE0 0030	H'1EE0 0030	32
Receive descriptor fetch address register	RDFAR0	R/W	H'FEE0 0034	H'1EE0 0034	32
Receive descriptor finished address register	RDFXR0	R/W	H'FEE0 0038	H'1EE0 0038	32
Receive descriptor final flag register	RDFFR0	R/W	H'FEE0 003C	H'1EE0 003C	32
Transmit/receive status copy enable register	TRSCER0	R/W	H'FEE0 0438	H'1EE0 0438	32
Receive missed-frame counter register	RMFCR0	R/W	H'FEE0 0440	H'1EE0 0440	32
Transmit FIFO threshold register	TFTR0	R/W	H'FEE0 0448	H'1EE0 0448	32
FIFO depth register	FDR0	R/W	H'FEE0 0450	H'1EE0 0450	32
Receiving method control register	RMCR0	R/W	H'FEE0 0458	H'1EE0 0458	32
Receive data padding insert register	RPADIR0	R/W	H'FEE0 0460	H'1EE0 0460	32
Overflow alert FIFO threshold register	FCFTR0	R/W	H'FEE0 0468	H'1EE0 0468	32
Intelligent checksum mode register	CSMR	R/W	H'FEE0 04E4	H'FEE0 04E4	32
Intelligent checksum skipped bytes monitor register	CSSBM	R	H'FEE0 04E8	H'FEE0 04E8	32
Intelligent checksum monitor register	CSSMR	R	H'FEE0 04EC	H'FEE0 04EC	32
RMII_MII select byte register	RMII_MII	R/W	H'FEE0 0790	H'FEE0 0790	32

Table 24.3 Register States in Each Operating Mode

Name	Power-On Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
ARSTR	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
ECMR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
ECSR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
ECSIPR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
PIR0	H'0000000x	H'0000000x	Retained	Retained	Retained	Initialized
MAHR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
MALR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
RFLR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
PSR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
PIPR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TROCR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
CDCR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
LCCR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
CEFCR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
FRECR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSFRCR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TLFRCR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
RFCR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
CERCR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
CEECR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
MAFCR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
APR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
MPR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
PFTCR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
PFRCR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TPAUSER0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
GECMR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
BCULR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_CTRST	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_FWSLC	H'00000000	H'00000000	Retained	Retained	Retained	Initialized

Name	Power-On Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
TSU_VTAG0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADSBSY	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_TEN	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_POST1	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_POST2	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_POST3	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_POST4	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRH0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRH1	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRH2	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRH3	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRH4	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRH5	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRH6	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRH7	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRH8	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRH9	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRH10	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRH11	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRH12	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRH13	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRH14	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRH15	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRH16	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRH17	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRH18	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRH19	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRH20	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRH21	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRH22	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRH23	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRH24	H'00000000	H'00000000	Retained	Retained	Retained	Initialized

Name	Power-On Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
TSU_ADRH25	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRH26	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRH27	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRH28	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRH29	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRH30	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRH31	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRL0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRL1	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRL2	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRL3	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRL4	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRL5	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRL6	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRL7	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRL8	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRL9	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRL10	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRL11	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRL12	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRL13	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRL14	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRL15	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRL16	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRL17	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRL18	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRL19	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRL20	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRL21	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRL22	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRL23	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRL24	H'00000000	H'00000000	Retained	Retained	Retained	Initialized

Name	Power-On Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
TSU_ADRL25	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRL26	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRL27	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRL28	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRL29	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRL30	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TSU_ADRL31	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TXNLCR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TXALCR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
RXNLCR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
RXALCR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
EDSR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
EDMR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
EDTRR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
EDRRR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
EESR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
EESIPR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TDLAR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TDFAR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TDFXR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TDFFR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
RDLAR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
RDFAR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
RDFXR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
RDFFR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TRSCER0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
RMFCR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
TFTR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
FDR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
RMCR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
RPADIR0	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
FCFTR0	H'001700FF	H'001700FF	Retained	Retained	Retained	Initialized

Name	Power-On Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
CSMR	H'C000001A	H'C000001A	Retained	Retained	Retained	Initialized
CSSBM	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
CSSMR	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
RMII_MII	H'00000000	H'00000000	Retained	Retained	Retained	Initialized

Note: Initialized means the value in the "power-on reset" column.

24.3.1 Software Reset Register (ARSTR)

ARSTR resets all blocks (E-MAC, TSU, and E-DMAC) in the GETHER. By writing 1 to the ARST bit in this register, a software reset is issued to all blocks of the GETHER (for 256 cycles of external bus clock Bck). The ARST bit is always read as 0. While a software reset is issued, register access to all blocks of the GETHER is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ARST
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	ARST	0	R/W	Software Reset When 1 is written to this bit, a software reset is issued to all blocks of the GETHER (for 256 cycles of external bus clock Bck). Writing 0 does not affect this bit. This bit is always read as 0. While a software reset is issued, register access to all blocks of the GETHER is prohibited. The following registers are not initialized by a software reset. TSU_ADRH0 to TSU_ADRH31, TSU_ADRL0 to TSU_ADRL31, TXNLCR0, TXALCR0, RXNLCR0, RXALCR0

24.3.2 E-MAC Mode Register (ECMR)

ECMR is a 32-bit readable/writable register that specifies the operating mode of the GETHER. The settings in this register are normally made in the initialization process following a reset.

The operating mode setting must not be changed while the transmitting and receiving functions are enabled. To switch the operating mode, return the E-MAC and E-DMAC to their initial states by means of the SWRT and SWRR bits in EDMR before making settings again.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	TRCCM	—	—	RCSC	—	DPAD	RZPF	ZPF	PFR	RXF	TXF
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R	R	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	MCT	—	—	—	MPDE	—	—	RE	TE	—	ILB	—	DM	PRM
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R	R	R	R/W	R	R	R/W	R/W	R	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26	TRCCM	0	R/W	Counter Clear Mode Sets the method for clearing the counter register. Refer to the description of each register. 0: Cleared to 0 by writing H'11111111 to the relevant register 1: Cleared to 0 when the relevant register is read
25, 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
23	RCSC	0	R/W	Checksum Calculation Specifies whether to perform automatic calculation (hardware calculation) of the checksum of the receive frame data unit. 0: Checksum is not automatically calculated 1: Checksum is automatically calculated Note that the checksum calculation of a frame with a VLAN tag is not supported. For details, see section 24.6.1, Checksum Calculation of Ethernet Frames.
22	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
21	DPAD	0	R/W	Data Padding 0: Padding is inserted to data less than 60 bytes so it is transmitted as 60-byte data 1: Padding is not inserted to data less than 60 bytes and it is transmitted without changes
20	RZPF	0	R/W	PAUSE Frame Reception with TIME = 0 0: Reception of a PAUSE frame whose TIME parameter value is 0 is disabled 1: Reception of a PAUSE frame whose TIME parameter value is 0 is enabled

Bit	Bit Name	Initial Value	R/W	Description
19	ZPF	0	R/W	PAUSE Frame Usage with TIME = 0 Enable/Lost Carrier Error Detection Enable. PAUSE Frame Usage with TIME = 0 Enable (In full-duplex mode) 0: Control of a PAUSE frame whose TIME parameter value is 0 is disabled. The next frame is not transmitted until the time specified by the Timer value has elapsed. If a PAUSE frame whose time specified by the Timer value is 0 is received, that PAUSE frame is discarded. 1: Control of a PAUSE frame whose TIME parameter value is 0 is enabled. When the data size in the receive FIFO becomes smaller than the FCFTR setting before the time specified by the Timer value elapses, an automatic PAUSE frame with a Timer value of 0 is transmitted. On receiving a PAUSE frame with a Timer value of 0, the transmission wait state is canceled. Lost carrier Error Detection Enable (In half-duplex mode) 0: A lost carrier error is checked during frame transmission. 1: A lost carrier error is not checked during frame transmission Lost carrier error detection can be enabled only when the time period from the EX_TX_EN signal activation (high-active) to the ET0_CRS = 1 detection is 63BT* or less. If the time period from the EX_TX_EN signal activation (high-active) to the ET0_CRS = 1 detection is greater than 63BT*, or if the ET0_CRS signal timing is undefined, this bit should not be cleared to 0. Note*: 1BT = 1 ns (1000 Mbps), 1BT = 10 ns (100 bps), 1BT = 100 ns (10 Mbps)
18	PFR	0	R/W	PAUSE Frame Receive Mode 0: PAUSE frame is not transferred to E-DMAC 1: PAUSE frame is transferred to E-DMAC

Bit	Bit Name	Initial Value	R/W	Description
17	RXF	0	R/W	Operating Mode for Receiving Port Flow Control 0: PAUSE frame detection is disabled 1: Flow control for the receiving port is enabled
16	TXF	0	R/W	Operating Mode for Transmitting Port Flow Control 0: Flow control for the transmitting port is disabled (Automatic PAUSE frame is not transmitted) 1: Flow control for the transmitting port is enabled (Automatic PAUSE frame is transmitted as required)
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13	MCT	0	R/W	Multicast Address Frame Receive Mode 0: Frames other than the multicast address set by the CAM entry table 0 to 31 (H/L) registers are received. However, if the on-chip CAM entry table reference is disabled, all multicast address frames are received. 1: Only the multicast address set by the CAM entry table 0 to 31 (H/L) registers is received.
12 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	MPDE	0	R/W	Magic Packet Detection Enable Enables or disables Magic Packet detection by hardware to allow activation from the Ethernet. 0: Magic Packet detection is not enabled 1: Magic Packet detection is enabled
8, 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
6	RE	0	R/W	Reception Enable If a switch is made from receiving function enabled (RE = 1) to disabled (RE = 0) while a frame is being received, the receiving function will be enabled until reception of the corresponding frame is completed. 0: Receiving function is disabled 1: Receiving function is enabled
5	TE	0	R/W	Transmission Enable If a switch is made from transmitting function enabled (TE = 1) to disabled (TE = 0) while a frame is being transmitted, the transmitting function will be enabled until transmission of the corresponding frame is completed. 0: Transmitting function is disabled 1: Transmitting function is enabled
4	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
3	ILB	0	R/W	Internal Loop Back Mode Specifies loopback mode in the GETHER. 0: Normal data transmission/reception is performed 1: Data loopback is performed inside the E-MAC in the GETHER when DM = 1
2	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
1	DM	0	R/W	Duplex Mode Specifies the GETHER transfer method. 0: Half-duplex transfer is specified 1: Full-duplex transfer is specified

Bit	Bit Name	Initial Value	R/W	Description
0	PRM	0	R/W	Promiscuous Mode Setting this bit enables all Ethernet frames to be received. All Ethernet frames means all receivable frames, irrespective of differences or enabled/disabled status (destination address, broadcast address, multicast bit, etc.). 0: GETHER performs normal operation 1: GETHER performs promiscuous mode operation

Note: All bits, except for TE and RE, should be changed while the transmitting function is disabled (TE = 0) and the receiving function is disabled (RE = 0).

24.3.3 E-MAC Status Register (ECSR)

ECSR is a 32-bit readable/writable register that indicates the status in the E-MAC. This status can be notified to the CPU by interrupts. When 1 is written to the PFROI, LCHNG, MPD, and ICD bits, the corresponding flags can be cleared. Writing 0 does not affect the flag. For bits that generate interrupts, the interrupt can be enabled or disabled by the corresponding bit in ECSIPR. Writing 1 or 0 to the PHYI bit does not change its value.

The interrupts generated due to this status register are indicated in the ECI bit in EESR of the E-DMAC.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	PFROI	PHYI	LCHNG	MPD	ICD
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	PFROI	0	R/W	PAUSE Frame Retransmit Retry Over Indicates whether the retransmit count for retransmitting a PAUSE frame when flow control is enabled has exceeded the retransmit upper-limit set in the automatic PAUSE frame retransmit count register (TPAUSER). 0: PAUSE frame retransmit count has not exceeded the upper limit 1: PAUSE frame retransmit count has exceeded the upper limit

Bit	Bit Name	Initial Value	R/W	Description
3	PHYI	0	R	ET0_PHY-INT Interrupt Indicates the state of the ET0_PHY-INT pin input from the PHY-LSI. 0: ET0_PHY-INT pin is not asserted 1: ET0_PHY-INT pin is asserted The signal polarity of the ET0_PHY-INT pin can be set by PIPR.
2	LCHNG	0	R/W	Link Signal Change Indicates that the ET0_LNKSTA signal input from the PHY-LSI has changed from high to low or low to high. However, a change in the signal level may also be detected at the time of selection of the pin for the ET0_LINKSTA function. To check the current Link state, refer to the LMON bit in the PHY status register (PSR). 0: Change in the ET0_LNKSTA signal has not been detected 1: Change in the ET0_LNKSTA signal has been detected (high to low or low to high)
1	MPD	0	R/W	Magic Packet Detection Indicates that a Magic Packet has been detected on the line. 0: Magic Packet has not been detected 1: Magic Packet has been detected
0	ICD	0	R/W	Illegal Carrier Detection Indicates that the PHY-LSI has detected an illegal carrier on the line. If a change in the signal input from the PHY-LSI occurs in a period shorter than the software recognition period, the correct information may not be obtained. Refer to the timing specification for the PHY-LSI used. 0: PHY-LSI has not detected an illegal carrier on the line 1: PHY-LSI has detected an illegal carrier on the line

24.3.4 E-MAC Interrupt Permission Register (ECSIPR)

ECSIPR is a 32-bit readable/writable register that enables or disables the interrupt sources indicated by ECSR. Each bit can disable or enable interrupts corresponding to the bits in ECSR.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	PFROIP	PHYIP	LCHNGIP	MPDIP	ICDIP
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	PFROIP	0	R/W	PAUSE Frame Retransmit Interrupt Enable 0: Interrupt notification by the PFROI bit is disabled 1: Interrupt notification by the PFROI bit is enabled
3	PHYIP	0	R/W	ET0_PHY-INT Pin Interrupt Enable 0: Interrupt notification by the PHYI bit is disabled 1: Interrupt notification by the PHYI bit is enabled
2	LCHNGIP	0	R/W	LINK Signal Change Interrupt Enable 0: Interrupt notification by the LCHNG bit is disabled 1: Interrupt notification by the LCHNG bit is enabled
1	MPDIP	0	R/W	Magic Packet Detect Interrupt Enable 0: Interrupt notification by the MPD bit is disabled 1: Interrupt notification by the MPD bit is enabled
0	ICDIP	0	R/W	Illegal Carrier Detect Interrupt Enable 0: Interrupt notification by the ICD bit is disabled 1: Interrupt notification by the ICD bit is enabled

24.3.5 PHY Interface Register (PIR)

PIR is a 32-bit readable/writable register that provides a means of accessing the PHY-LSI internal registers via the GMII/MII/RMII.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	MDI	MDO	MMD	MDC
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	—	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	MDI	Undefined	R	GMII/MII/RMII Management Data-In Indicates the level of the ET0_MDIO pin.
2	MDO	0	R/W	GMII/MII/RMII Management Data-Out Outputs the value set in this bit from the ET0_MDIO pin when the MMD bit is 1.
1	MMD	0	R/W	GMII/MII/RMII Management Mode Specifies the data read/write direction with respect to the GMII/MII/RMII. 0: Read direction is specified 1: Write direction is specified
0	MDC	0	R/W	GMII/MII/RMII Management Data Clock Outputs the value set in this bit from the ET0_MDC pin and supplies the GMII/MII/RMII with the management data clock. For the method of accessing the GMII/MII/RMII registers, see section 24.5.4, Accessing MII Registers.

24.3.6 MAC Address High Register (MAHR)

MAHR is a 32-bit readable/writable register that specifies the upper 32 bits of the 48-bit MAC address. The settings in this register are normally made in the initialization process after a reset. The MAC address setting must not be changed while the transmitting and receiving functions are enabled. Return the E-MAC and E-DMAC to their initial states by means of the SWRT and SWRR bits in EDMR before making settings again.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MA[47:32]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MA[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	MA[47:16]	All 0	R/W	MAC Address Bits 47 to 16 These bits are used to set the upper 32 bits of the MAC address. If the MAC address is 01-23-45-67-89-AB (hexadecimal), set H'01234567 in this register.

24.3.7 MAC Address Low Register (MALR)

MALR is a 32-bit readable/writable register that specifies the lower 16 bits of the 48-bit MAC address. The settings in this register are normally made in the initialization process after a reset. The MAC address setting must not be changed while the transmitting and receiving functions are enabled. Return the E-MAC and E-DMAC to their initial states by means of the SWRT and SWRR bits in EDMR before making settings again.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MA[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	MA[15:0]	All 0	R/W	MAC Address Bits 15 to 0 These bits are used to set the lower 16 bits of the MAC address. If the MAC address is 01-23-45-67-89-AB (hexadecimal), set H'000089AB in this register.

24.3.8 Receive Frame Length Register (RFLR)

RFLR is a 32-bit readable/writable register that specifies the maximum frame length (in bytes) that can be received by this LSI. The settings in this register must not be changed while the receiving function is enabled.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RFL[17:16]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFL[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17 to 0	RFL[17:0]	All 0	R/W	Receive Frame Length The frame data described here refers to all fields from the destination address up to the CRC data. Frame contents from the destination address up to the data are actually transferred to memory. CRC data is not included in the transfer. When data that exceeds the specified value is received, the part of data that exceeds the specified value is discarded. H'00000 to H'005EE: 1,518 bytes H'005EF: 1,519 bytes H'005F0: 1,520 bytes : : H'007FF: 2,047 bytes H'00800: 2,048 bytes : : H'01000: 4,096 bytes : : H'10000: 65,536 bytes : : H'20000 to H'3FFFF: 131,072 bytes

24.3.9 PHY Status Register (PSR)

PSR is a read-only register that can read interface signals from the PHY-LSI.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LMON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	LMON	0	R	ET0_LNKSTA Pin Status The Link status can be read by connecting the Link signal output from the PHY-LSI to the ET0_LNKSTA pin. For the polarity, refer to the specifications of the PHY-LSI to be connected.

24.3.10 PHY_INT Polarity Register (PIPR)

PIPR is used to set the polarity of the ET0_PHY-INT pin.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PHYIP
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	PHYIP	0	R/W	ET0_PHY-INT Input Pin Polarity 0: ET0_PHY-INT pin is low-active (enters the interrupt state at low) 1: ET0_PHY-INT pin is high-active (enters the interrupt state at high) For the polarity, refer to the specifications of the PHY-LSI to be connected.

24.3.11 Transmit Retry Over Counter Register (TROCR)

TROCR is a 16-bit counter that indicates the number of frames that were unable to be transmitted in 16 transmission attempts including the retransfer. When 16 transmission attempts have failed, this register is incremented by 1. When the value in this register reaches H'0000FFFF, count-up is halted. This register is cleared to 0 when it is read with the TRCCM bit in ECMR set to 1. When the TRCCM bit in ECMR is 0, this register is cleared to 0 by writing H'11111111.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TROC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	TROC[15:0]	All 0	R/W	Transmit Retry Over Count These bits indicate the number of frames that were unable to be transmitted in 16 transmission attempts including the retransfer.

Note: When count-up and count-clearing of this register value occur simultaneously, count-clearing is performed.

24.3.12 Delayed Collision Detect Counter Register (CDCR)

CDCR is a 16-bit counter that indicates the number of all delayed collisions that occurred on the line after the start of data transmission. When the value in this register reaches H'0000FFFF, count-up is halted. This register is cleared to 0 when it is read with the TRCCM bit in ECMR set to 1. When the TRCCM bit in ECMR is 0, this register is cleared to 0 by writing H'11111111.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	COSDC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	COSDC[15:0]	All 0	R/W	Delayed Collision Detect Count These bits indicate the number of all delayed collisions after the start of data transmission.

Note: When count-up and count-clearing of this register value occur simultaneously, count-clearing is performed.

24.3.13 Lost Carrier Counter Register (LCCR)

LCCR is a 16-bit counter that indicates the number of times the carrier was lost during data transmission. When the value in this register reaches H'0000FFFF, count-up is halted. This register is cleared to 0 when it is read with the TRCCM bit in ECMR set to 1. When the TRCCM bit in ECMR is 0, this register is cleared to 0 by writing H'11111111.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LCC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	LCC[15:0]	All 0	R/W	Lost Carrier Count These bits indicate the number of times the carrier was lost during data transmission.

Note: When count-up and count-clearing of this register value occur simultaneously, count-clearing is performed.

24.3.14 CRC Error Frame Receive Counter Register (CEFCR)

CEFCR is a 16-bit counter that indicates the number of times a frame with a CRC error was received. When the value in this register reaches H'0000FFFF, count-up is halted. This register is cleared to 0 when it is read with the TRCCM bit in ECMR set to 1. When the TRCCM bit in ECMR is 0, this register is cleared to 0 by writing H'11111111.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CEFC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	CEFC[15:0]	All 0	R/W	CRC Error Frame Count These bits indicate the number of CRC error frames received.

Note: When count-up and count-clearing of this register value occur simultaneously, count-clearing is performed.

24.3.15 Frame Receive Error Counter Register (FRECR)

FRECR is a 16-bit counter that indicates the number of frames for which a receive error was generated by the ET0_RX-ER pin input from the PHY-LSI. FRECR is incremented each time the ET0_RX-ER pin becomes active. When the value in this register reaches H'0000FFFF, count-up is halted. This register is cleared to 0 when it is read with the TRCCM bit in ECMR set to 1. When the TRCCM bit in ECMR is 0, this register is cleared to 0 by writing H'11111111.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FREC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	FREC[15:0]	All 0	R/W	Frame Receive Error Count These bits indicate the number of errors during frame reception.

Note: When count-up and count-clearing of this register value occur simultaneously, count-clearing is performed.

24.3.16 Too-Short Frame Receive Counter Register (TSFRCR)

TSFRCR is a 16-bit counter that indicates the number of frames received with a length fewer than 64 bytes. When the value in this register reaches H'0000FFFF, count-up is halted. This register is cleared to 0 when it is read with the TRCCM bit in ECMR set to 1. When the TRCCM bit in ECMR is 0, this register is cleared to 0 by writing H'11111111.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSFC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	TSFC[15:0]	All 0	R/W	Too-Short Frame Receive Count These bits indicate the number of frames received with a length of less than 64 bytes.

Note: When count-up and count-clearing of this register value occur simultaneously, count-clearing is performed.

24.3.17 Too-Long Frame Receive Counter Register (TLFRCR)

TLFRCR is a 16-bit counter that indicates the number of frames received with a length exceeding the value specified by the receive frame length register (RFLR). When the value in this register reaches H'0000FFFF, count-up is halted. This register is not incremented when a frame containing residual bits is received. In this case, the reception of the frame is indicated in the residual-bit frame receive counter register (RFCR). This register is cleared to 0 when it is read with the TRCCM bit in ECMR set to 1. When the TRCCM bit in ECMR is 0, this register is cleared to 0 by writing H'11111111.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TLFC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	TLFC[15:0]	All 0	R/W	Too-Long Frame Receive Count These bits indicate the number of frames received with a length exceeding the value in RFLR.

Note: When count-up and count-clearing of this register value occur simultaneously, count-clearing is performed.

24.3.18 Residual-Bit Frame Receive Counter Register (RFCR)

RFCR is a 16-bit counter that indicates the number of frames received containing residual bits (less than an 8-bit unit). When the value in this register reaches H'0000FFFF, count-up is halted. This register is cleared to 0 when it is read with the TRCCM bit in ECMR set to 1. When the TRCCM bit in ECMR is 0, this register is cleared to 0 by writing H'11111111.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	RFC[15:0]	All 0	R/W	Residual-Bit Frame Receive Count These bits indicate the number of frames received containing residual bits.

Note: When count-up and count-clearing of this register value occur simultaneously, count-clearing is performed.

24.3.19 Carrier Extension Loss Counter Register (CERCR)

CERCR is a 16-bit counter that indicates the number of frames received with the carrier extension lost. When the value in this register reaches H'0000FFFF, count-up is halted. This register is cleared to 0 when it is read with the TRCCM bit in ECMR set to 1. When the TRCCM bit in ECMR is 0, this register is cleared to 0 by writing H'11111111.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CERC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	CERC[15:0]	All 0	R/W	Carrier Extension Loss Frame Receive Count These bits indicate the number of frames received with the carrier extension lost.

Note: When count-up and count-clearing of this register value occur simultaneously, count-clearing is performed.

24.3.20 Carrier Extension Error Counter Register (CEECR)

CEECR is a 16-bit counter that indicates the number of frames received with an illegal carrier extension. When the value in this register reaches H'0000FFFF, count-up is halted. This register is cleared to 0 when it is read with the TRCCM bit in ECMR set to 1. When the TRCCM bit in ECMR is 0, this register is cleared to 0 by writing H'11111111.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CEEC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	CEEC[15:0]	All 0	R/W	Carrier Extension Error Count These bits indicate the number of frames received with an illegal carrier extension.

Note: When count-up and count-clearing of this register value occur simultaneously, count-clearing is performed.

24.3.21 Multicast Address Frame Receive Counter Register (MAFCR)

MAFCR is a 16-bit counter that indicates the number of frames received with a specified multicast address. When the value in this register reaches H'0000FFFF, count-up is halted. This register is cleared to 0 when it is read with the TRCCM bit in ECMR set to 1. When the TRCCM bit in ECMR is 0, this register is cleared to 0 by writing H'11111111.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MAFC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	MAFC[15:0]	All 0	R/W	Multicast Address Frame Count These bits indicate the number of multicast frames received.

Note: When count-up and count-clearing of this register value occur simultaneously, count-clearing is performed.

24.3.22 Automatic PAUSE Frame Register (APR)

APR is used to set the TIME parameter value of an automatic PAUSE frame. When an automatic PAUSE frame is transmitted, the value set in this register is used as the TIME parameter of the PAUSE frame.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AP[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	AP[15:0]	All 0	R/W	Automatic PAUSE These bits set the TIME parameter value of an automatic PAUSE frame. One bit is equivalent to 512 bit-time. When flow control is enabled in transmission (PAUSE frame transmission) (TXF bit in ECMR = 1), set a value other than H'0000 in these bits. H'0000: — H'0001: 512 × 1 bit-time H'0002: 512 × 2 bit-time : : H'FFFF: 512 × 65,535 bit-time Note: The bit-time becomes as follows according to the transfer speed. 1000 Mbps: 1 bit-time = 1 ns 100 Mbps: 1 bit-time = 10 ns 10 Mbps: 1 bit-time = 100 ns

24.3.23 Manual PAUSE Frame Register (MPR)

MPR is used to set the TIME parameter value of a manual PAUSE frame. When a manual PAUSE frame is transmitted, the value set in this register is used as the TIME parameter of the PAUSE frame.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MP[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	MP[15:0]	All 0	R/W	Manual PAUSE These bits set the TIME parameter value of a manual PAUSE frame. One bit is equivalent to 512 bit-time. H'0000: — H'0001: 512 × 1 bit-time H'0002: 512 × 2 bit-time : : H'FFFF: 512 × 65,535 bit-time Note: The bit-time becomes as follows according to the transfer speed. 1000 Mbps: 1 bit-time = 1 ns 100 Mbps: 1 bit-time = 10 ns 10 Mbps: 1 bit-time = 100 ns

24.3.24 Automatic PAUSE Frame Retransmit Count Register (TPAUSER)

TPAUSER is used to set the upper limit for the number of times to retransmit an automatic PAUSE frame. The settings in this register must not be changed while the transmitting function is enabled.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TPAUSE[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	TPAUSE[15:0]	All 0	R/W	Upper Limit for Automatic PAUSE Frame Retransmission H'0000: Retransmit count is unlimited H'0001: Retransmit count is 1 : : H'FFFF: Retransmit count is 65,535

24.3.25 PAUSE Frame Transmit Counter Register (PFTCR)

PFTCR is a 16-bit counter that indicates the number of times a PAUSE frame is transmitted. This register is cleared to 0 when it is read.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PFTXC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	PFTXC[15:0]	All 0	R	PAUSE Frame Transmit Count These bits indicate the total number of automatic PAUSE frames and manual PAUSE frames transmitted.

Note: When count-up and count-clearing of this register value occur simultaneously, count-clearing is performed.

24.3.26 PAUSE Frame Receive Counter Register (PFRCR)

PFRCR is a 16-bit counter that indicates the number of times a PAUSE frame is received. This register is cleared to 0 when it is read.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PFRXC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	PFRXC[15:0]	All 0	R	PAUSE Frame Receive Count These bits indicate the number of PAUSE frames received when flow control is enabled in reception (RXF bit in ECMR = 1).

Note: When count-up and count-clearing of this register value occur simultaneously, count-clearing is performed.

24.3.27 GETHER Mode Register (GECMR)

GECMR is used to set the operating mode of the GETHER.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	SPEED [1]	BSE	SPEED [0]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	SPEED[1]	0	R/W	Transfer Speed Sets the transfer speed in combination with the SPEED[0] bit. Refer to the SPEED[0] bit.
1	BSE	0	R/W	Burst Transfer Enable 0: Burst transfer is not performed 1: Burst transfer is performed when the transfer speed is 1 Gbps in half-duplex transfer (DM bit in ECMR = 0).
0	SPEED[0]	0	R/W	Transfer Speed The transfer speed is specified by a combination of the SPEED[1] and SPEED[0] bits. SPEED[1:0] 00: 10-Mbps transfer 01: 1-Gbps transfer 10: 100-Mbps transfer 11: Setting prohibited

24.3.28 Burst Cycle Count Upper-Limit Register (BCULR)

BCULR sets the upper limit for the number of burst cycles.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	BSTLMT[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	BSTLMT[11:0]	All 0	R/W	Burst Cycle Upper-Limit These bits set the upper limit for burst cycles. Burst transfer is finished when the burst timer exceeds the value set in this register. If the burst timer exceeds the value set in this register while a frame is being transferred, burst transfer is continued until transfer of the corresponding frame is completed. H'000 to H'100: Burst cycle count is 256 cycles H'101: Burst cycle count is 257 cycles : : H'FFE: Burst cycle count is 4,094 cycles H'FFF: Burst cycle count is 4,095 cycles Note: 1 cycle = 32 ns

24.3.29 TSU Counter Reset Register (TSU_CTRST)

TSU_CTRST clears the transmit, receive, and relay frame counters to 0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	CTRST	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	CTRST	0	R/W	TSU Counter Reset When 1 is written to this bit, the values of registers TXNLCR0, TXALCR0, RXNLCR0, and RXALCR0 are cleared to 0. Writing 0 does not affect this bit. This bit is always read as 0.
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

24.3.30 Relay Function Set Register (Common) (TSU_FWSLC)

When the CAM is used, the referred area in the CAM entry table (partially or wholly) can be specified by the TSU_POST1 to TSU_POST4 registers. TSU_FWSLC enables settings by the TSU_POST1 to TSU_POST4 registers.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	POSTENU	POSTENL	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13	POSTENU	0	R/W	Enables the settings of the POST field of CAM entry tables 0 to 15 (settings by the TSU_POST1 and TSU_POST2 registers). 0: Disables the settings of the POST field. 1: Enables the settings of the POST field. (The CAM entry table reference conditions follow the POST field settings.)
12	POSTENL	0	R/W	Enables the settings of the POST field of CAM entry tables 16 to 31 (settings by the TSU_POST3 and TSU_POST4 registers). 0: Disables the settings of the POST field. 1: Enables the settings of the POST field. (The CAM entry table reference conditions follow the POST field settings.)
11 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

24.3.31 VLANtag Set Register (TSU_VTAG0)

TSU_VTAG0 enables or disables the frame receive/discard evaluation function based on the VLAN number, and also sets the VLAN number.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	VTAG0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	VID0[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	VTAG0	0	R/W	Port 0 VLANtag Evaluation Function 0: Disables receive/discard evaluation for frames based on the VLAN number 1: Enables receive/discard evaluation for frames based on the VLAN number
30 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	VID0[11:0]	All 0	R/W	V-LAN ID Setting (VID) These bits set the VLAN number received by receive frames.

24.3.32 CAM Entry Table Busy Register (TSU_ADSBSY)

When CAM entry table registers (TSU_ADRH0 to TSU_ADRH31 and TSU_ADRL0 to TSU_ADRL31) are set by register writing, the ADSBSY bit in this register is set to 1 (when the process of reflecting the contents of the CAM entry table registers in the CAM controller is completed inside the TSU, the ADSBSY bit is automatically restored to 0).

Access to TSU_ADRH0 to TSU_ADRH31 and TSU_ADRL0 to TSU_ADRL31 is prohibited, while the ADSBSY bit in this register is set to 1. This register is a read-only status register, which must not be written to.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ADS BSY
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	ADSBSY	0	R	CAM Entry Table Setting Busy When TSU_ADRH0 to TSU_ADRH31 and TSU_ADRL0 to TSU_ADRL31 are set by register writing, this bit is set to 1. When the process of reflecting the contents of the CAM entry table registers in the CAM controller is completed inside the TSU, this bit is automatically restored to 0. Access to TSU_ADRH0 to TSU_ADRH31 and TSU_ADRL0 to TSU_ADRL31 is prohibited, while this bit is set to 1. Writing to this register is also prohibited.

24.3.33 CAM Entry Table Enable Register (TSU_TEN)

TSU_TEN enables or disables the settings of TSU_ADRH0 to TSU_ADRH31 and TSU_ADRL0 to TSU_ADRL31.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TEN0	TEN1	TEN2	TEN3	TEN4	TEN5	TEN6	TEN7	TEN8	TEN9	TEN10	TEN11	TEN12	TEN13	TEN14	TEN15
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TEN16	TEN17	TEN18	TEN19	TEN20	TEN21	TEN22	TEN23	TEN24	TEN25	TEN26	TEN27	TEN28	TEN29	TEN30	TEN31
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	TEN0	0	R/W	CAM Entry Table 0 (TSU_ADRH0 and TSU_ADRL0) Setting 0: Disabled 1: Enabled
30	TEN1	0	R/W	CAM Entry Table 1 (TSU_ADRH1 and TSU_ADRL1) Setting 0: Disabled 1: Enabled
29	TEN2	0	R/W	CAM Entry Table 2 (TSU_ADRH2 and TSU_ADRL2) Setting 0: Disabled 1: Enabled
28	TEN3	0	R/W	CAM Entry Table 3 (TSU_ADRH3 and TSU_ADRL3) Setting 0: Disabled 1: Enabled
27	TEN4	0	R/W	CAM Entry Table 4 (TSU_ADRH4 and TSU_ADRL4) Setting 0: Disabled 1: Enabled

Bit	Bit Name	Initial Value	R/W	Description
26	TEN5	0	R/W	CAM Entry Table 5 (TSU_ADRH5 and TSU_ADRL5) Setting 0: Disabled 1: Enabled
25	TEN6	0	R/W	CAM Entry Table 6 (TSU_ADRH6 and TSU_ADRL6) Setting 0: Disabled 1: Enabled
24	TEN7	0	R/W	CAM Entry Table 7 (TSU_ADRH7 and TSU_ADRL7) Setting 0: Disabled 1: Enabled
23	TEN8	0	R/W	CAM Entry Table 8 (TSU_ADRH8 and TSU_ADRL8) Setting 0: Disabled 1: Enabled
22	TEN9	0	R/W	CAM Entry Table 9 (TSU_ADRH9 and TSU_ADRL9) Setting 0: Disabled 1: Enabled
21	TEN10	0	R/W	CAM Entry Table 10 (TSU_ADRH10 and TSU_ADRL10) Setting 0: Disabled 1: Enabled
20	TEN11	0	R/W	CAM Entry Table 11 (TSU_ADRH11 and TSU_ADRL11) Setting 0: Disabled 1: Enabled
19	TEN12	0	R/W	CAM Entry Table 12 (TSU_ADRH12 and TSU_ADRL12) Setting 0: Disabled 1: Enabled

Bit	Bit Name	Initial Value	R/W	Description
18	TEN13	0	R/W	CAM Entry Table 13 (TSU_ADRH13 and TSU_ADRL13) Setting 0: Disabled 1: Enabled
17	TEN14	0	R/W	CAM Entry Table 14 (TSU_ADRH14 and TSU_ADRL14) Setting 0: Disabled 1: Enabled
16	TEN15	0	R/W	CAM Entry Table 15 (TSU_ADRH15 and TSU_ADRL15) Setting 0: Disabled 1: Enabled
15	TEN16	0	R/W	CAM Entry Table 16 (TSU_ADRH16 and TSU_ADRL16) Setting 0: Disabled 1: Enabled
14	TEN17	0	R/W	CAM Entry Table 17 (TSU_ADRH17 and TSU_ADRL17) Setting 0: Disabled 1: Enabled
13	TEN18	0	R/W	CAM Entry Table 18 (TSU_ADRH18 and TSU_ADRL18) Setting 0: Disabled 1: Enabled
12	TEN19	0	R/W	CAM Entry Table 19 (TSU_ADRH19 and TSU_ADRL19) Setting 0: Disabled 1: Enabled
11	TEN20	0	R/W	CAM Entry Table 20 (TSU_ADRH20 and TSU_ADRL20) Setting 0: Disabled 1: Enabled

Bit	Bit Name	Initial Value	R/W	Description
10	TEN21	0	R/W	CAM Entry Table 21 (TSU_ADRH21 and TSU_ADRL21) Setting 0: Disabled 1: Enabled
9	TEN22	0	R/W	CAM Entry Table 22 (TSU_ADRH22 and TSU_ADRL22) Setting 0: Disabled 1: Enabled
8	TEN23	0	R/W	CAM Entry Table 23 (TSU_ADRH23 and TSU_ADRL23) Setting 0: Disabled 1: Enabled
7	TEN24	0	R/W	CAM Entry Table 24 (TSU_ADRH24 and TSU_ADRL24) Setting 0: Disabled 1: Enabled
6	TEN25	0	R/W	CAM Entry Table 25 (TSU_ADRH25 and TSU_ADRL25) Setting 0: Disabled 1: Enabled
5	TEN26	0	R/W	CAM Entry Table 26 (TSU_ADRH26 and TSU_ADRL26) Setting 0: Disabled 1: Enabled
4	TEN27	0	R/W	CAM Entry Table 27 (TSU_ADRH27 and TSU_ADRL27) Setting 0: Disabled 1: Enabled
3	TEN28	0	R/W	CAM Entry Table 28 (TSU_ADRH28 and TSU_ADRL28) Setting 0: Disabled 1: Enabled

Bit	Bit Name	Initial Value	R/W	Description
2	TEN29	0	R/W	CAM Entry Table 29 (TSU_ADRH29 and TSU_ADRL29) Setting 0: Disabled 1: Enabled
1	TEN30	0	R/W	CAM Entry Table 30 (TSU_ADRH30 and TSU_ADRL30) Setting 0: Disabled 1: Enabled
0	TEN31	0	R/W	CAM Entry Table 31 (TSU_ADRH31 and TSU_ADRL31) Setting 0: Disabled 1: Enabled

24.3.34 CAM Entry Table POST1 Register (TSU_POST1)

When using the CAM, the conditions for referring to each CAM entry table can be specified independently by using the TSU_POST1 to TSU_POST4 registers. TSU_POST1 specifies the conditions for referring to TSU_ADRH0 to TSU_ADRH7 and TSU_ADRL0 to TSU_ADRL7. The settings of this register are valid only when the POSTENU bit in TSU_FWSLC is set to 1.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	POST0	—	—	—	POST1	—	—	—	POST2	—	—	—	POST3	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	POST4	—	—	—	POST5	—	—	—	POST6	—	—	—	POST7	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	POST0	0	R/W	Sets the condition for referring to CAM entry table 0. By setting this bit to 1, the condition can be selected. POST0: CAM entry table 0 is referred to in reception
30 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27	POST1	0	R/W	Sets the condition for referring to CAM entry table 1. By setting this bit to 1, the condition can be selected. POST1: CAM entry table 1 is referred to in reception
26 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23	POST2	0	R/W	Sets the condition for referring to CAM entry table 2. By setting this bit to 1, the condition can be selected. POST2: CAM entry table 2 is referred to in reception
22 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
19	POST3	0	R/W	Sets the condition for referring to CAM entry table 3. By setting this bit to 1, the condition can be selected. POST3: CAM entry table 3 is referred to in reception
18 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15	POST4	0	R/W	Sets the condition for referring to CAM entry table 4. By setting this bit to 1, the condition can be selected. POST4: CAM entry table 4 is referred to in reception
14 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11	POST5	0	R/W	Sets the condition for referring to CAM entry table 5. By setting this bit to 1, the condition can be selected. POST5: CAM entry table 5 is referred to in reception
10 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	POST6	0	R/W	Sets the condition for referring to CAM entry table 6. By setting this bit to 1, the condition can be selected. POST6: CAM entry table 6 is referred to in reception
6 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	POST7	0	R/W	Sets the condition for referring to CAM entry table 7. By setting this bit to 1, the condition can be selected. POST7: CAM entry table 7 is referred to in reception
2 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

24.3.35 CAM Entry Table POST2 Register (TSU_POST2)

When using the CAM, the conditions for referring to each CAM entry table can be specified independently by using the TSU_POST1 to TSU_POST4 registers. TSU_POST2 specifies the conditions for referring to TSU_ADRH8 to TSU_ADRH15 and TSU_ADRL8 to TSU_ADRL15. The settings of this register are valid only when the POSTENU bit in TSU_FWSLC is set to 1.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	POST8	—	—	—	POST9	—	—	—	POST10	—	—	—	POST11	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R

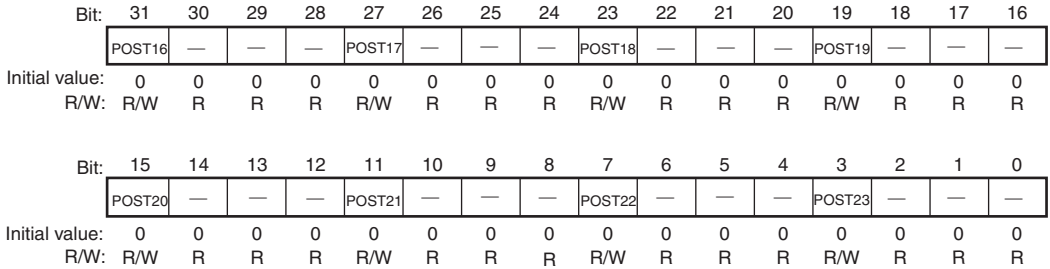
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	POST12	—	—	—	POST13	—	—	—	POST14	—	—	—	POST15	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	POST8	0	R/W	Sets the condition for referring to CAM entry table 8. By setting this bit to 1, the condition can be selected. POST8: CAM entry table 8 is referred to in reception
30 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27	POST9	0	R/W	Sets the condition for referring to CAM entry table 9. By setting this bit to 1, the condition can be selected. POST9: CAM entry table 9 is referred to in reception
26 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23	POST10	0	R/W	Sets the condition for referring to CAM entry table 10. By setting this bit to 1, the condition can be selected. POST10: CAM entry table 10 is referred to in reception
22 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
19	POST11	0	R/W	Sets the condition for referring to CAM entry table 11. By setting this bit to 1, the condition can be selected. POST11: CAM entry table 11 is referred to in reception
18 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15	POST12	0	R/W	Sets the condition for referring to CAM entry table 12. By setting this bit to 1, the condition can be selected. POST12: CAM entry table 12 is referred to in reception
14 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11	POST13	0	R/W	Sets the condition for referring to CAM entry table 13. By setting this bit to 1, the condition can be selected. POST13: CAM entry table 13 is referred to in reception
10 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	POST14	0	R/W	Sets the condition for referring to CAM entry table 14. By setting this bit to 1, the condition can be selected. POST14: CAM entry table 14 is referred to in reception
6 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	POST15	0	R/W	Sets the condition for referring to CAM entry table 15. By setting this bit to 1, the condition can be selected. POST15: CAM entry table 15 is referred to in reception
2 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

24.3.36 CAM Entry Table POST3 Register (TSU_POST3)

When using the CAM, the conditions for referring to each CAM entry table can be specified independently by using the TSU_POST1 to TSU_POST4 registers. TSU_POST3 specifies the conditions for referring to TSU_ADRH16 to TSU_ADRH23 and TSU_ADRL16 to TSU_ADRL23. The settings of this register are valid only when the POSTENL bit in TSU_FWSLC is set to 1.



Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	POST16	—	—	—	POST17	—	—	—	POST18	—	—	—	POST19	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	POST20	—	—	—	POST21	—	—	—	POST22	—	—	—	POST23	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	POST16	0	R/W	Sets the condition for referring to CAM entry table 16. By setting this bit to 1, the condition can be selected. POST16: CAM entry table 16 is referred to in reception
30 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27	POST17	0	R/W	Sets the condition for referring to CAM entry table 17. By setting this bit to 1, the condition can be selected. POST17: CAM entry table 17 is referred to in reception
26 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23	POST18	0	R/W	Sets the condition for referring to CAM entry table 18. By setting this bit to 1, the condition can be selected. POST18: CAM entry table 18 is referred to in reception
22 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
19	POST19	0	R/W	Sets the condition for referring to CAM entry table 19. By setting this bit to 1, the condition can be selected. POST19: CAM entry table 19 is referred to in reception
18 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15	POST20	0	R/W	Sets the condition for referring to CAM entry table 20. By setting this bit to 1, the condition can be selected. POST20: CAM entry table 20 is referred to in reception
14 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11	POST21	0	R/W	Sets the condition for referring to CAM entry table 21. By setting this bit to 1, the condition can be selected. POST21: CAM entry table 21 is referred to in reception
10 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	POST22	0	R/W	Sets the condition for referring to CAM entry table 22. By setting this bit to 1, the condition can be selected. POST22: CAM entry table 22 is referred to in reception
6 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	POST23	0	R/W	Sets the condition for referring to CAM entry table 23. By setting this bit to 1, the condition can be selected. POST23: CAM entry table 23 is referred to in reception
2 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

24.3.37 CAM Entry Table POST4 Register (TSU_POST4)

When using the CAM, the conditions for referring to each CAM entry table can be specified independently by using the TSU_POST1 to TSU_POST4 registers. TSU_POST4 specifies the conditions for referring to TSU_ADRH24 to TSU_ADRH31 and TSU_ADRL24 to TSU_ADRL31. The settings of this register are valid only when the POSTENL bit in TSU_FWSLC is set to 1.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	POST24	—	—	—	POST25	—	—	—	POST26	—	—	—	POST27	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	POST28	—	—	—	POST29	—	—	—	POST30	—	—	—	POST31	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	POST24	0	R/W	Sets the condition for referring to CAM entry table 24. By setting this bit to 1, the condition can be selected. POST24: CAM entry table 24 is referred to in reception
30 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27	POST25	0	R/W	Sets the condition for referring to CAM entry table 25. By setting this bit to 1, the condition can be selected. POST25: CAM entry table 25 is referred to in reception
26 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23	POST26	0	R/W	Sets the condition for referring to CAM entry table 26. By setting this bit to 1, the condition can be selected. POST26: CAM entry table 26 is referred to in reception
22 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
19	POST27	0	R/W	Sets the condition for referring to CAM entry table 27. By setting this bit to 1, the condition can be selected. POST27: CAM entry table 27 is referred to in reception
18 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15	POST28	0	R/W	Sets the condition for referring to CAM entry table 28. By setting this bit to 1, the condition can be selected. POST28: CAM entry table 28 is referred to in reception
14 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11	POST29	0	R/W	Sets the condition for referring to CAM entry table 29. By setting this bit to 1, the condition can be selected. POST29: CAM entry table 29 is referred to in reception
10 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	POST30	0	R/W	Sets the condition for referring to CAM entry table 30. By setting this bit to 1, the condition can be selected. POST30: CAM entry table 30 is referred to in reception
6 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	POST31	0	R/W	Sets the condition for referring to CAM entry table 31. By setting this bit to 1, the condition can be selected. POST31: CAM entry table 31 is referred to in reception
2 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

24.3.38 CAM Entry Table 0H to 31H Registers (TSU_ADRH0 to TSU_ADRH31)

TSU_ADRH0 to TSU_ADRH31 are entry tables referred to by the CAM in reception and relay. Each of these registers sets the upper 32 bits of the 48-bit MAC address. Maximum 32 entries of MAC addresses can be registered.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ADRHn[31:16] (n = 0 to 31)															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADRHn[15:0] (n = 0 to 31)															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ADRHn[31:0] (n: 0 to 31)	All 0	R/W	MAC Address Bits These bits set the upper 32 bits of the MAC address. When the MAC address is 01-23-45-67-89-AB (displayed in hexadecimal), set H'01234567 in this register.

Note: Set the CAM entry tables following the procedure below.

1. Check that the ADSBSY bit in TSU_ADSBSY is cleared to 0.
2. Set the upper 32 bits of the MAC addresses by TSU_ADRH0 to TSU_ADRH31.
3. Set the lower 16 bits of the MAC addresses by TSU_ADRL0 to TSU_ADRL31.

24.3.39 CAM Entry Table 0L to 31L Registers (TSU_ADRL0 to TSU_ADRL31)

TSU_ADRL0 to TSU_ADRL31 are entry tables referred to by the CAM in reception and relay. Each of these registers sets the lower 16 bits of the 48-bit MAC address. Maximum 32 entries of MAC addresses can be registered.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADRLn[15:0] (n = 0 to 31)															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	ADRLn[15:0] (n: 0 to 31)	All 0	R/W	MAC Address Bits These bits set the lower 16 bits of the MAC address. When the MAC address is 01-23-45-67-89-AB (displayed in hexadecimal), set H'000089AB in this register.

Note: Set the CAM entry tables following the procedure below.

1. Check that the ADSBSY bit in TSU_ADSBSY is cleared to 0.
2. Set the upper 32 bits of the MAC addresses by TSU_ADRH0 to TSU_ADRH31.
3. Set the lower 16 bits of the MAC addresses by TSU_ADRL0 to TSU_ADRL31.

24.3.40 Transmit Frame Counter Register (Normal Transmission Only) (TXNLCR0)

TXNLCR0 is a 32-bit counter indicating the number of frames successfully transmitted in the E-MAC. When the value in this register reaches H'FFFFFFFF, count-up is halted. The counter is cleared to 0 by reading from this register. This register cannot be written to.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NTC0[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NTC0[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	NTC0[31:0]	All 0	R	Transmit Frame Counter Bits These bits indicate the number of frames successfully transmitted.

Note: When count-up and count-clearing of this register value occur simultaneously, count-clearing is performed.

24.3.41 Transmit Frame Counter Register (Normal and Erroneous Transmission) (TXALCR0)

TXALCR0 is a 32-bit counter indicating the number of frames transmitted in the E-MAC, including the number of frames erroneously transmitted. When the value in this register reaches H'FFFFFFFF, count-up is halted. The counter is cleared to 0 by reading from this register. This register cannot be written to.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TC0[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TC0[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TC0[31:0]	All 0	R	Transmit Frame Counter Bits These bits indicate the number of frames successfully transmitted and erroneously transmitted.

Note: When count-up and count-clearing of this register value occur simultaneously, count-clearing is performed.

24.3.42 Receive Frame Counter Register (Normal Reception Only) (RXNLCR0)

RXNLCR0 is a 32-bit counter indicating the number of frames successfully received in the E-MAC. When the value in this register reaches H'FFFFFFFF, count-up is halted. The counter is cleared to 0 by reading from this register. This register cannot be written to.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NRC0[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NRC0[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	NRC0[31:0]	All 0	R	Receive Frame Counter Bits These bits indicate the number of frames successfully received.

Note: When count-up and count-clearing of this register value occur simultaneously, count-clearing is performed.

24.3.43 Receive Frame Counter Register (Normal and Erroneous Reception) (RXALCR0)

RXALCR0 is a 32-bit counter indicating the number of frames received in the E-MAC, including the number of frames erroneously received. When the value in this register reaches H'FFFFFFFF, count-up is halted. The counter is cleared to 0 by reading from this register. This register cannot be written to.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RC0[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RC0[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	RC0[31:0]	All 0	R	Receive Frame Counter Bits These bits indicate the number of frames successfully received and erroneously received.

Note: When count-up and count-clearing of this register value occur simultaneously, count-clearing is performed.

24.3.44 E-DMAC Start Register (EDSR)

EDSR specifies activation of the transmitting unit and receiving unit of the E-DMAC. This register can only be written to, and the read values are invalid.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ENT	ENR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	ENT	0	W	E-DMAC Transmitting Unit Start 0: Stops the E-DMAC transmitting unit 1: Starts the E-DMAC transmitting unit
0	ENR	0	W	E-DMAC Receiving Unit Start 0: Stops the E-DMAC receiving unit 1: Starts the E-DMAC receiving unit

24.3.45 E-DMAC Mode Register (EDMR)

EDMR is a 32-bit readable/writable register that specifies E-DMAC resetting and the transmit/receive descriptor length. This register is to be set before the transmitting or receiving function is enabled (before the TR bit in EDTRR or the RR bit in EDRRR is set to 1). However, the SWRR and SWRT bits can be written to even after the transmitting or receiving function is enabled. If a software reset is executed with this register during data transmission, abnormal data may be transmitted on the line. Execute a software reset with this register before specifying the transmit/receive descriptor length or modifying the settings of TDLAR, RDLAR, and so forth, the setting of ECMR (E-MAC mode register), and the settings of registers related to the E-DMAC and E-MAC operation.

To execute a software reset with this register, 1 must be written to both the SWRT and SWRR bits simultaneously. Writing 1 to the SWRT and SWRR bits initializes the E-MAC registers and E-DMAC registers, except for TDLAR, RDLAR, and RMFCR of the E-DMAC. The TSU registers (registers whose names are prefixed with TSU_) are not initialized. The SWRT and SWRR bits in EDMR0 initializes the registers related to the E-DMAC and E-MAC. Note that during the period a software reset is issued (for 64 cycles of the internal bus clock Bck), accesses to all Ethernet-related registers are prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	DE	DL[1:0]		—	—	SWRT	SWRR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6	DE	0	R/W	Transmit/Receive Frame Endian Sets the endian mode for DMA transfer of frame data between the transmit/receive FIFO and transmit/receive buffer. 0: Big endian (longword access) 1: Little endian (longword access)
5, 4	DL[1:0]	00	R/W	Transmit/Receive Descriptor Length These bits specify the descriptor length. (See section 24.4.1, Descriptors and Descriptor List.) 00: 16 bytes 01: 32 bytes 10: 64 bytes 11: Setting prohibited
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	SWRT	0	R/W	Software Reset of Transmit FIFO Controller [Writing] 0: Disabled 1: Software reset started [Reading] 0: Software reset not executed (or completed) 1: Software reset being executed
0	SWRR	0	R/W	Software Reset of Receive FIFO Controller [Writing] 0: Disabled 1: Software reset started [Reading] 0: Software reset not executed (or completed) 1: Software reset being executed

24.3.46 E-DMAC Transmit Request Register (EDTRR)

EDTRR is a 32-bit readable/writable register that issues transmit directives to the E-DMAC. After writing 11 to bits TR[1:0] in this register, the E-DMAC reads the transmit descriptor at the address specified by TDLAR. If the TACT bit of this transmit descriptor is set to 1 (valid), transmit DMA transfer by the E-DMAC starts. When DMA transfer based on the first transmit descriptor is completed, the E-DMAC reads the next transmit descriptor. If the TACT bit of that transmit descriptor is set to 1 (valid), the E-DMAC continues transmit DMA operation. If the TACT bit of a transmit descriptor is cleared to 0 (invalid), the E-DMAC clears bits TR[1:0] and stops transmit DMA operation.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TR[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	TR[1:0]	00	R/W	Transmit Request 00, 01, 10: Transmission-halted state If 00, 01, or 10 is written to these bits, the E-DMAC stops DMA transfer of the currently processed transmit descriptor, reads the next transmit descriptor, and then clears these bits. (Write-back is completed for the valid transmit descriptors that have been detected up till then.) The E-DMAC clears these bits when transmit descriptor empty occurs, or transmission of a transmit descriptor has completed. (Write-back is completed for the valid transmit descriptors that have been detected up till then.) 11: Transmit DMA operation by E-DMAC After writing 11 to these bits, the E-DMAC starts reading a transmit descriptor.

24.3.47 E-DMAC Receive Request Register (EDRRR)

EDRRR is a 32-bit readable/writable register that issues receive directives to the E-DMAC. After writing 1 to the RR bit in this register, the E-DMAC reads the receive descriptor at the address specified by RDLAR. If the RACT bit of this receive descriptor is set to 1 (valid), and the receive FIFO holds a receive frame, the E-DMAC starts receive DMA transfer. When DMA transfer based on the first receive descriptor is completed, the E-DMAC reads the next receive descriptor. If the RACT bit of that receive descriptor is set to 1 (valid), the E-DMAC continues receive DMA operation. However, if the receive FIFO holds no receive data, the E-DMAC places receive DMA operation in the standby state. If the RACT bit of the receive descriptor is cleared to 0 (invalid), the E-DMAC clears the RR bit and stops receive DMAC operation.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	RR	0	R/W	Receive Request 0: Receiving function is disabled* If 0 is written to this bit, the E-DMAC stops receive operation after DMA transfer of one frame has completed and then clears this bit. The E-DMAC clears this bit when receive descriptor empty occurs. 1: Receive descriptor is read, and the E-DMAC is ready to receive.

Note: * If the receiving function is disabled during frame reception, write-back is not performed successfully to the receive descriptor. Following pointers to read a receive descriptor become abnormal and the E-DMAC cannot operate successfully. In this case, to make E-DMAC reception enabled again, execute a software reset by the SWRT and SWRR bits in EDMR0. To disable the E-DMAC receiving function without executing a software reset, specify the RE bit in ECMR0. Next, after the E-DMAC has completed the reception and write-back to the receive descriptor has been confirmed, disable the receiving function using this register.

24.3.48 Transmit Descriptor List Start Address Register (TDLAR)

TDLAR is a 32-bit readable/writable register that specifies the start address of the transmit descriptor list. Descriptors have a boundary configuration in accordance with the descriptor length indicated by the DL bits in EDMR. This register must not be modified during transmission. Modifications to this register should only be made in the transmission-halted state specified by bits TR[1:0] (= 00) in the E-DMAC transmit request register (EDTRR).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TDLA[31:15]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TDLA[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TDLA[31:0]	All 0	R/W	Transmit Descriptor Start Address The lower bits are set according to the specified descriptor length. 16-byte boundary: TDLA[3:0] = 0000 32-byte boundary: TDLA[4:0] = 00000 64-byte boundary: TDLA[5:0] = 000000

24.3.49 Receive Descriptor List Start Address Register (RDLAR)

RDLAR is a 32-bit readable/writable register that specifies the start address of the receive descriptor list. Descriptors have a boundary configuration in accordance with the descriptor length indicated by the DL bits in EDMR. This register must not be modified during reception. Modifications to this register should only be made while reception is disabled by the RR bit (= 0) in the E-DMAC receive request register (EDRRR).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RDLA[31:15]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RDLA[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	RDLA[31:0]	All 0	R/W	Receive Descriptor Start Address The lower bits are set according to the specified descriptor length. 16-byte boundary: RDLA[3:0] = 0000 32-byte boundary: RDLA[4:0] = 00000 64-byte boundary: RDLA[5:0] = 000000

24.3.50 E-MAC/E-DMAC Status Register (EESR)

EESR is a 32-bit readable/writable register that shows communications status information on the E-DMAC in combination with the E-MAC. The information in this register is reported in the form of interrupt sources. Individual bits are cleared by writing 1 (however, bit 22 (ECI) is a read-only bit that is not cleared by writing 1) and are not affected by writing 0. Each interrupt source can also be masked by means of the corresponding bit in the E-MAC/E-DMAC status interrupt permission register (EESIPR).

The interrupt generated by this status register is GEINT0. For interrupt priorities, see section 7.3.2, Interrupt Sources, in section 7, INTC/INTC2.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TWB[1:0]		TC[1]	TUC	ROC	TABT	RABT	RFCOF	—	ECI	TC[0]	TDE	TFUF	FR	RDE	RFE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	DLC	CD	TRO	RMAF	CEEF	CELF	RRF	RTLF	RTSF	PRE	CERF
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	TWB[1:0]	00	R/W	Write-Back Complete Indicates that write-back from the E-DMAC to the corresponding descriptor after frame transmission has completed. This operation is enabled only when the TWBI bit in the transmit descriptor that includes the end of the transmit frame is set to 1. 00: Write-back has not completed, or no transmission directive 11: Write-back has completed Others: Setting disabled

Bit	Bit Name	Initial Value	R/W	Description
29	TC[1]	0	R/W	Frame Transmission Complete Indicates, in combination with the TC[0] bit, that all the data specified by the transmit descriptor has been transmitted from the E-MAC. This bit is set to 1 on assuming the completion of transmission. This is when transmission of one frame is completed and the transmit descriptor valid bit (TACT) of the next transmit descriptor not being set in single-frame/single-descriptor operation or when the last data of a frame has been transmitted and the transmit descriptor valid bit (TACT) of the next descriptor not being set in multi-buffer frame processing based on single-frame/multi-descriptor operation. After frame transmission has completed, the E-DMAC writes the transmission status back to the relevant descriptor. TC[1:0] 00: Transmission has not completed, or no transmission directive 11: Transmission has completed Others: Setting disabled
28	TUC	0	R/W	Transmit Underflow Frame Write-Back Complete 0: Write-back has not completed for the frame causing transmit underflow 1: Write-back has completed for the frame causing transmit underflow
27	ROC	0	R/W	Receive Overflow Frame Write-Back Complete 0: Write-back has not completed for the frame causing receive overflow 1: Write-back has completed for the frame causing receive overflow
26	TABT	0	R/W	Transmit Abort Detect Indicates that the E-MAC aborts transmitting a frame because of failures during frame transmission. 0: Frame transmission has not been aborted or no transmission directive 1: Frame transmission has been aborted

Bit	Bit Name	Initial Value	R/W	Description
25	RABT	0	R/W	Receive Abort Detect Indicates that the E-MAC aborts receiving a frame because of failures during frame reception. 0: Frame reception has not been aborted or no reception directive 1: Frame reception has been aborted
24	RFCOF	0	R/W	Receive Frame Counter Overflow Indicates that the frame counter in the receive FIFO has overflowed. 0: Receive frame counter has not overflowed 1: Receive frame counter has overflowed
23	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
22	ECI	0	R	E-MAC Status Register Source This bit is a read-only bit. When the source of an ECSR interrupt is cleared, this bit is also cleared. 0: E-MAC status interrupt source has not been detected 1: E-MAC status interrupt source has been detected
21	TC[0]	0	R/W	Frame Transmission Complete Indicates, in combination with the TC[1] bit, that all the data specified by the transmit descriptor has been transmitted from the E-MAC. For details, see the description of the TC[1] bit.

Bit	Bit Name	Initial Value	R/W	Description
20	TDE	0	R/W	Transmit Descriptor Empty Indicates that the transmit descriptor valid bit (TACT) of a transmit descriptor read by the E-DMAC is not set if the previous descriptor does not represent the end of a frame in multi-buffer frame processing based on single-frame/multi-descriptor operation. As a result, an incomplete frame may be sent. 0: Transmit descriptor active bit TACT = 1 detected 1: Transmit descriptor active bit TACT = 0 detected When transmit descriptor empty (TDE = 1) occurs, execute a software reset and initiate transmission. In this case, transmission starts from the address that is stored in the transmit descriptor list start address register (TDLAR).
19	TFUF	0	R/W	Transmit FIFO Underflow Indicates that an underflow has occurred in the transmit FIFO during frame transmission. Incomplete data is sent onto the line. 0: Underflow has not occurred 1: Underflow has occurred
18	FR	0	R/W	Frame Reception Indicates that a frame has been received and the receive descriptor has been updated. This bit is set to 1 each time a frame is received. 0: Frame has not been received 1: Frame has been received
17	RDE	0	R/W	Receive Descriptor Empty Indicates that the RACT bit of a receive descriptor read by the E-DMAC for receive DMA operation is cleared to 0 (invalid). When receive descriptor empty (RDE = 1) occurs, reception can be resumed by setting the RACT bit (cleared to 0) of the receive descriptor to 1 and then writing 1 to the RR bit in EDRRR. 0: Receive descriptor active bit RACT = 1 detected 1: Receive descriptor active bit RACT = 0 detected

Bit	Bit Name	Initial Value	R/W	Description
16	RFOF	0	R/W	Receive FIFO Overflow Indicates that the receive FIFO has overflowed during frame reception. 0: Overflow has not occurred 1: Overflow has occurred
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10	DLC	0	R/W	Detect Loss of Carrier Indicates that loss of the carrier has been detected during frame transmission. 0: Loss of carrier has not been detected 1: Loss of carrier has been detected
9	CD	0	R/W	Delayed Collision Detect Indicates that a delayed collision has been detected during frame transmission. 0: Delayed collision has not been detected 1: Delayed collision has been detected
8	TRO	0	R/W	Transmit Retry Over Indicates that a retry-over condition has occurred during frame transmission. Total 16 transmission retries including 15 retries based on the back-off algorithm have failed after the E-MAC transmission starts. 0: Transmit retry-over condition not detected 1: Transmit retry-over condition detected
7	RMAF	0	R/W	Receive Multicast Address Frame 0: Multicast address frame has not been received 1: Multicast address frame has been received
6	CEEF	0	R/W	Carrier Extension Error Indicates that a carrier extension error has occurred during frame reception in 1-Gigabit/half-duplex transfer. 0: Carrier extension error has not occurred 1: Carrier extension error has occurred

Bit	Bit Name	Initial Value	R/W	Description
5	CELF	0	R/W	Carrier Extension Loss Indicates that the carrier extension has been lost in 1-Gigabit/half-duplex transfer. This means that the sum of a frame and carrier extension is smaller than SLOT_TIME (4096 bits). 0: Carrier extension loss has not occurred 1: Carrier extension loss has occurred
4	RRF	0	R/W	Receive Residual-Bit Frame 0: Residual-bit frame has not been received 1: Residual-bit frame has been received
3	RTLF	0	R/W	Receive Too-Long Frame Indicates that a frame whose byte size exceeds the upper limit for the receive frame length set by RFLR has been received. 0: Too-long frame has not been received 1: Too-long frame has been received
2	RTSF	0	R/W	Receive Too-Short Frame Indicates that a frame of fewer than 64 bytes has been received. 0: Too-short frame has not been received 1: Too-short frame has been received
1	PRE	0	R/W	PHY-LSI Receive Error 0: PHY-LSI receive error has not been detected 1: PHY-LSI receive error has been detected
0	CERF	0	R/W	CRC Error on Received Frame 0: CRC error has not been detected 1: CRC error has been detected

24.3.51 E-MAC/E-DMAC Status Interrupt Permission Register (EESIPR)

EESIPR is a 32-bit readable/writable register that enables interrupts corresponding to individual bits in the E-MAC/E-DMAC status register (EESR). An interrupt is enabled by writing 1 to the corresponding bit.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TWB1IP	TWB0IP	TC1IP	TUCIP	ROCIP	TABTIP	RABTIP	RFCOFIP	—	ECIIP	TC0IP	TDEIP	TFUFIP	FRIP	RDEIP	RFEIP
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	DLCIP	CDIP	TROIP	RMAFIP	CEEFIP	CELFIP	RRFIP	RTLFIP	RTSFIP	PREIP	CERFIP
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	TWB1IP	0	R/W	Write-Back Complete Interrupt Enable 0: Write-back complete interrupt is disabled 1: Write-back complete interrupt is enabled
30	TWB0IP	0	R/W	Write-Back Complete Interrupt Enable 0: Write-back complete interrupt is disabled 1: Write-back complete interrupt is enabled
29	TC1IP	0	R/W	Frame Transmission Complete Interrupt Enable 0: Frame transmission complete interrupt is disabled 1: Frame transmission complete interrupt is enabled
28	TUCIP	0	R/W	Transmit Underflow Frame Write-Back Complete Interrupt Enable 0: Transmit underflow frame write-back complete interrupt is disabled 1: Transmit underflow frame write-back complete interrupt is enabled

Bit	Bit Name	Initial Value	R/W	Description
27	ROCIP	0	R/W	Receive Overflow Frame Write-Back Complete Interrupt Enable 0: Receive overflow frame write-back complete interrupt is disabled 1: Receive overflow frame write-back complete interrupt is enabled
26	TABTIP	0	R/W	Transmit Abort Detect Interrupt Enable 0: Transmit abort detect interrupt is disabled 1: Transmit abort detect interrupt is enabled
25	RABTIP	0	R/W	Receive Abort Detect Interrupt Enable 0: Receive abort detect interrupt is disabled 1: Receive abort detect interrupt is enabled
24	RFCOFIP	0	R/W	Receive Frame Counter Overflow Interrupt Enable 0: Receive frame counter overflow interrupt is disabled 1: Receive frame counter overflow interrupt is enabled
23	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
22	ECIIP	0	R/W	E-MAC Status Register Source Interrupt Enable 0: E-MAC status interrupt is disabled 1: E-MAC status interrupt is enabled
21	TC0IP	0	R/W	Frame Transmission Complete Interrupt Enable 0: Frame transmission complete interrupt is disabled 1: Frame transmission complete interrupt is enabled
20	TDEIP	0	R/W	Transmit Descriptor Empty Interrupt Enable 0: Transmit descriptor empty interrupt is disabled 1: Transmit descriptor empty interrupt is enabled
19	TFUFIP	0	R/W	Transmit FIFO Underflow Interrupt Enable 0: Underflow interrupt is disabled 1: Underflow interrupt is enabled
18	FRIP	0	R/W	Frame Reception Interrupt Enable 0: Frame reception interrupt is disabled 1: Frame reception interrupt is enabled

Bit	Bit Name	Initial Value	R/W	Description
17	RDEIP	0	R/W	Receive Descriptor Empty Interrupt Enable 0: Receive descriptor empty interrupt is disabled 1: Receive descriptor empty interrupt is enabled
16	RFOFIP	0	R/W	Receive FIFO Overflow Interrupt Enable 0: Overflow interrupt is disabled 1: Overflow interrupt is enabled
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10	DLCIP	0	R/W	Detect Loss of Carrier Interrupt Enable 0: Detect loss of carrier interrupt is disabled 1: Detect loss of carrier interrupt is enabled
9	CDIP	0	R/W	Delayed Collision Detect Interrupt Enable 0: Delayed collision detect interrupt is disabled 1: Delayed collision detect interrupt is enabled
8	TROIP	0	R/W	Transmit Retry Over Interrupt Enable 0: Transmit retry over interrupt is disabled 1: Transmit retry over interrupt is enabled
7	RMAFIP	0	R/W	Receive Multicast Address Frame Interrupt Enable 0: Receive multicast address frame interrupt is disabled 1: Receive multicast address frame interrupt is enabled
6	CEEFIP	0	R/W	Carrier Extension Error Interrupt Enable 0: Carrier extension error interrupt is disabled 1: Carrier extension error interrupt is enabled
5	CELFIP	0	R/W	Carrier Extension Loss Interrupt Enable 0: Carrier extension loss interrupt is disabled 1: Carrier extension loss interrupt is enabled
4	RRFIP	0	R/W	Receive Residual-Bit Frame Interrupt Enable 0: Receive residual-bit frame interrupt is disabled 1: Receive residual-bit frame interrupt is enabled

Bit	Bit Name	Initial Value	R/W	Description
3	RTLFIP	0	R/W	Receive Too-Long Frame Interrupt Enable 0: Receive too-long frame interrupt is disabled 1: Receive too-long frame interrupt is enabled
2	RTSFIP	0	R/W	Receive Too-Short Frame Interrupt Enable 0: Receive too-short frame interrupt is disabled 1: Receive too-short frame interrupt is enabled
1	PREIP	0	R/W	PHY-LSI Receive Error Interrupt Enable 0: PHY-LSI receive error interrupt is disabled 1: PHY-LSI receive error interrupt is enabled
0	CERFIP	0	R/W	CRC Error on Received Frame Interrupt Enable 0: CRC error interrupt is disabled 1: CRC error interrupt is enabled

24.3.52 Transmit/Receive Status Copy Enable Register (TRSCER)

TRSCER specifies whether the information for the transmit and receive state reported by bits 26, 25, and 10 to 0 in the E-MAC/E-DMAC status register (EESR) is to be reflected in the TFE or RFE bit of the corresponding descriptor. The bits in this register correspond to bits 26, 25, and 10 to 0 in EESR. When a bit is cleared to 0, the transmit status (bits 26 and 10 to 8 in EESR) is reflected in the TFE bit of the transmit descriptor, and the receive status (bits 25 and 7 to 0 in EESR) is reflected in the RFE bit of the receive descriptor. In this case, the state of a status bit set to 1 is reflected as the TFE or RFE bit set to 1. When a bit is set to 1, the occurrence of the corresponding source is not reflected in the descriptor. After this LSI is reset, all bits are cleared to 0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TABT CE	RABT CE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	DLC CE	CD CE	TRO CE	RMAF CE	CEEF CE	CELF CE	RRF CE	RTLF CE	RTSF CE	PRE CE	CERF CE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17	TABTCE	0	R/W	TABT Bit Copy Directive 0: Reflects the TABT bit status in the TFE bit of the transmit descriptor 1: Occurrence of the corresponding source is not reflected in the TFE bit of the transmit descriptor
16	RABTCE	0	R/W	RABT Bit Copy Directive 0: Reflects the RABT bit status in the RFE bit of the receive descriptor 1: Occurrence of the corresponding source is not reflected in the RFE bit of the receive descriptor

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10	DLCCE	0	R/W	DLC Bit Copy Directive 0: Reflects the DLC bit status in the TFE bit of the transmit descriptor 1: Occurrence of the corresponding source is not reflected in the TFE bit of the transmit descriptor
9	CDCE	0	R/W	CD Bit Copy Directive 0: Reflects the CD bit status in the TFE bit of the transmit descriptor 1: Occurrence of the corresponding source is not reflected in the TFE bit of the transmit descriptor
8	TROCE	0	R/W	TRO Bit Copy Directive 0: Reflects the TRO bit status in the TFE bit of the transmit descriptor 1: Occurrence of the corresponding source is not reflected in the TFE bit of the transmit descriptor
7	RMAFCE	0	R/W	RMAF Bit Copy Directive 0: Reflects the RMAF bit status in the RFE bit of the receive descriptor 1: Occurrence of the corresponding source is not reflected in the RFE bit of the receive descriptor
6	CEEFCE	0	R/W	CEEF Bit Copy Directive 0: Reflects the CEEF bit status in the RFE bit of the receive descriptor 1: Occurrence of the corresponding source is not reflected in the RFE bit of the receive descriptor
5	CELFCE	0	R/W	CELF Bit Copy Directive 0: Reflects the CELF bit status in the RFE bit of the receive descriptor 1: Occurrence of the corresponding source is not reflected in the RFE bit of the receive descriptor

Bit	Bit Name	Initial Value	R/W	Description
4	RRFCE	0	R/W	RRF Bit Copy Directive 0: Reflects the RRF bit status in the RFE bit of the receive descriptor 1: Occurrence of the corresponding source is not reflected in the RFE bit of the receive descriptor
3	RTLFCE	0	R/W	RTLF Bit Copy Directive 0: Reflects the RTLF bit status in the RFE bit of the receive descriptor 1: Occurrence of the corresponding source is not reflected in the RFE bit of the receive descriptor
2	RTSFCE	0	R/W	RTSF Bit Copy Directive 0: Reflects the RTSF bit status in the RFE bit of the receive descriptor 1: Occurrence of the corresponding source is not reflected in the RFE bit of the receive descriptor
1	PRECE	0	R/W	PRE Bit Copy Directive 0: Reflects the PRE bit status in the RFE bit of the receive descriptor 1: Occurrence of the corresponding source is not reflected in the RFE bit of the receive descriptor
0	CERFCE	0	R/W	CERF Bit Copy Directive 0: Reflects the CERF bit status in the RFE bit of the receive descriptor 1: Occurrence of the corresponding source is not reflected in the RFE bit of the receive descriptor

24.3.53 Receive Missed-Frame Counter Register (RMFCR)

RMFCR is a 16-bit counter that indicates the number of frames that could not be saved in the receive buffer and so were discarded during reception. When the receive FIFO overflows, the receive frames in the FIFO are discarded. The number of frames discarded at this time is counted. When the value in this register reaches H'FFFF, count-up is halted. Clear the counter by writing H'0000 in this register.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MCF[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	MFC[15:0]	All 0	R/W	Missed-Frame Counter These bits indicate the number of frames that are discarded and not transferred to the receive buffer during reception.

24.3.54 Transmit FIFO Threshold Register (TFTR)

TFTR is a 32-bit readable/writable register that specifies the transmit FIFO threshold at which the first transmission is started. The actual threshold is 4 times the set value. The E-MAC starts transmission when the amount of data in the transmit FIFO exceeds the number of bytes specified by this register, when the transmit FIFO is full, or when one frame of data write is performed. This register must not be written to during transmission (bits TR[1:0] in EDTRR = 11).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	TFT[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	TFT[10:0]	All 0	R/W	Transmit FIFO Threshold A value in 32-byte units and smaller than the FIFO size specified by FDR must be set as the transmit FIFO threshold. H'000: Store and forward modes H'008: 32 bytes H'010: 64 bytes H'018: 128 bytes : : H'07F: 508 bytes H'080: 512 bytes : : H'0FF: 1,020 bytes H'100: 1,024 bytes : : H'1FF: 2,044 bytes H'200: 2,048 bytes

Note: When starting transmission before one frame of data write has completed, take care no underflow occurs.

24.3.55 FIFO Depth Register (FDR)

FDR is a 32-bit readable/writable register that specifies the sizes of the transmit and receive FIFOs.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	TFD[2:0]			—	—	—	RFD[4:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 8	TFD[2:0]	All 0	R/W	Transmit FIFO Size Specifies 256 bytes to 2 Kbytes in 256-byte units as the size of the transmit FIFO whose maximum size is 2 Kbytes. Make the setting for a size between 256 and 2048 bytes. The setting must not be changed after transmission/reception has started. H'00 : 256 bytes (initial value) H'01 : 512 bytes : : H'07 : 2048 bytes
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
4 to 0	RFD[4:0]	All 0	R/W	Receive FIFO Size Specifies 256 bytes to 4 Kbytes in 256-byte units as the size of the receive FIFO whose maximum size is 4 Kbytes. Make the setting for a size between 256 and 4096 bytes. The setting must not be changed after transmission/reception has started. H'00 : 256 bytes (initial value) H'01 : 512 bytes : : H'0F : 4096 bytes

24.3.56 Receiving Method Control Register (RMCR)

RMCR is a 32-bit readable/writable register that specifies the control method for the RE bit in ECMR while a frame is received. This register must be set during the receiving-halted state.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RNC
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
0	RNC	0	R/W	Receive Enable Control Sets whether to continue frame reception. 0: Upon completion of reception of one frame, the E-DMAC writes the receive status to the descriptor and clears the RR bit in EDRRR to 0. 1: Upon completion of reception of one frame, the E-DMAC writes (writes back) the receive status to the descriptor. In addition, the E-DMAC reads the next descriptor and prepares for reception of the next frame.

24.3.57 Receive Descriptor Fetch Address Register (RDFAR)

RDFAR stores the descriptor start address that is required when the E-DMAC fetches descriptor information from the receive descriptor. Which receive descriptor information is used for processing by the E-DMAC can be recognized by monitoring addresses displayed in this register. The address from which the E-DMAC is actually fetching a descriptor may be different from the value read from this register. In the initial setting, set the address of the receive descriptor at which receive processing is to be started.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RDFA[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RDFA[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	RDFA[31:0]	All 0	R/W	Receive Descriptor Fetch Address Writing to these bits during the reception is prohibited.

24.3.58 Receive Descriptor Finished Address Register (RDFXR)

RDFXR stores the start address of the receive descriptor for which the E-DMAC has just completed the write-back processing. Up to which receive descriptor has been processed by the E-DMAC can be recognized by monitoring addresses displayed in this register. In the initial setting, set the address of the descriptor immediately before the descriptor that is pointed to by the address in RDFAR.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RDFX[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RDFX[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	RDFX[31:0]	All 0	R/W	Receive Descriptor Finished Address Writing to these bits during the reception is prohibited.

24.3.59 Receive Descriptor Final Flag Register (RDFFR)

RDFFR indicates whether the receive descriptor for which the E-DMAC has just completed the write-back processing and whose start address is stored in RDFXR is at the end of the receive descriptor queue (descriptor list).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RDLF
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	RDLF	0	R/W	Receive Descriptor Queue Last Flag Indicates whether the receive descriptor for which the E-DMAC has just completed the write-back processing and whose start address is stored in RDFXR is at the end of the receive descriptor queue (descriptor list). 0: Not the last descriptor in the receive descriptor queue 1: Last descriptor in the receive descriptor queue

24.3.60 Transmit Descriptor Fetch Address Register (TDFAR)

TDFAR stores the descriptor start address that is required when the E-DMAC fetches descriptor information from the transmit descriptor. Which transmit descriptor information is used for processing by the E-DMAC can be recognized by monitoring addresses displayed in this register. The address from which the E-DMAC is actually fetching a descriptor may be different from the value read from this register. In the initial setting, set the address of the transmit descriptor at which transmit processing is to be started.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TDFA[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TDFA[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TDFA[31:0]	All 0	R/W	Transmit Descriptor Fetch Address Writing to these bits during transmission is prohibited.

24.3.61 Transmit Descriptor Finished Address Register (TDFXR)

TDFXR stores the start address of the transmit descriptor for which the E-DMAC has just completed the write-back processing. Up to which transmit descriptor has been processed by the E-DMAC can be recognized by monitoring addresses displayed in this register. In the initial setting, set the address of the transmit descriptor immediately before the descriptor that is pointed to by the address in TDFAR.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TDFX[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TDFX[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TDFX[31:0]	All 0	R/W	Transmit Descriptor Finished Address Writing to these bits during transmission is prohibited.

24.3.62 Transmit Descriptor Final Flag Register (TDFFR)

TDFFR indicates whether the transmit descriptor for which the E-DMAC has just completed the write-back processing and whose start address is stored in TDFXR is at the end of the transmit descriptor queue (descriptor list).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TDLF
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	TDLF	0	R/W	Transmit Descriptor Queue Last Flag Indicates whether the transmit descriptor for which the E-DMAC has just completed the write-back processing and whose start address is stored in TDFXR is at the end of the transmit descriptor queue (descriptor list). 0: Not the last descriptor in the transmit descriptor queue 1: Last descriptor in the transmit descriptor queue

24.3.63 Overflow Alert FIFO Threshold Register (FCFTR)

FCFTR is a 32-bit readable/writable register that sets the flow control of the E-MAC. The threshold can be set by the size of the receive FIFO data (bits RFD[7:0]) and the number of receive frames (bits RFF[4:0]).

If the same receive FIFO size as set by the FIFO depth register (FDR) is set when flow control is turned on according to the RFD setting condition, flow control is turned on with (FIFO data size – 64) bytes. For instance, when the RFD bits in FDR is 7 and the RFD bits in this register is 7, flow control is turned on when (2,048 – 64) bytes of data is stored in the receive FIFO. The value set in the RFD bits in this register should be equal to or less than that set in the RFD bits in FDR.

Flow control is turned on when either of the setting conditions of bits RFF[4:0] and bits RFD[7:0] is satisfied. Flow control is turned off when neither of the conditions is satisfied (release).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	RFF[4:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RFD[7:0]							
Initial value:	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20 to 16	RFF[4:0]	H'17	R/W	Receive FIFO Overflow Alert Signal Output Threshold H'00: When one receive frame has been stored in the receive FIFO H'01: When two receive frames have been stored in the receive FIFO : : H'16: When 23 receive frames have been stored in the receive FIFO H'17: When 24 receive frames have been stored in the receive FIFO

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	RFD[7:0]	H'FF	R/W	Receive FIFO Overflow Alert Signal Output Threshold H'00: When (256 – 32) bytes of data is stored in the receive FIFO H'01: When (512 – 32) bytes of data is stored in the receive FIFO : : H'06: When (1,792 – 32) bytes of data is stored in the receive FIFO H'07: When (2,048 – 64) bytes of data is stored in the receive FIFO

24.3.64 Receive Data Padding Insert Register (RPADIR)

RPADIR is a 32-bit readable/writable register that inserts padding in receive data. To change the settings of this register, execute a software reset by means of the SWRT and SWRR bits in the E-DMAC mode register (EDMR) before making settings again.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	PADS[4:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PADR[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20 to 16	PADS[4:0]	H'00	R/W	Padding Size H'00: No padding insertion H'01: 1-byte insertion : : H'1F: 31-byte insertion
15 to 0	PADR[15:0]	H'0000	R/W	Padding Slot H'0000: Inserts specified size of padding at the first byte H'0001: Inserts specified size of padding at the second byte : : H'FFFF: Inserts specified size of padding at the 64K byte

24.3.65 Intelligent Checksum Mode Register (CSMR)

CSMR is a readable 32-bit register that specifies the intelligent checksum operation mode. This register must be set while reception is halted.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CSEBL	CSMD	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	SB[5:0]					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	CSELB	1	R	Intelligent Checksum Calculation Operation Setting 0: The result of checksum calculation is not written back to the receive descriptor. 1: The result of checksum calculation is written back to the receive descriptor.
30	CSMD	1	R/W	Intelligent Checksum Calculation Mode Setting 0: After having skipped the number of bytes specified in SB[5:0], counting from the beginning of the MAC-layer packet, the checksum is calculated for all subsequent data 1: MAC- or IP-layer packets are detected and checksums are calculated for upper-layer protocol packets such as TCP or UDP.
29 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
5 to 0	SB[5:0]*	011010	R/W	Intelligent Checksum Calculation Skip Bytes These bits specify the number of bytes to be skipped for checksum calculation, counting from the beginning of the data received in the E-DMAC. When padding bytes are to be added, specify the checksum start position to cover the amount or extent of padding. H'00: 0 bytes (meaning checksum calculation is performed from the beginning of the packet.) H'02: 2 bytes : : H'1A: 26 bytes H'3E: 62 bytes

Note: This bit should only be set when CSEL is 1 and CSMD is 0.

24.3.66 Intelligent Checksum Skipped Bytes Monitor Register (CSSBM)

CSSBM is a 32-bit readable register that holds the number of bytes that have been skipped in received packets being handled by the E-DMAC. The number of skipped bytes can be monitored through this register. The amount of data received in E-DMAC may not match the number of skipped bytes.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	SBM[5:0]					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5 to 0	SBM[5:0]	000000	R	Number of Skipped Bytes These bits are read-only. Writing is prohibited. These bits are initialized when the beginning of a packet for reception is detected.

Note: This register is only valid when CSEL is 1 and CSMD is 0.

24.3.67 Intelligent Checksum Monitor Register (CSSMR)

CSSMR is a 32-bit register that holds the checksum value of received packets being handled by the E-DMAC. The checksum value can be monitored through this register. The amount of data received in E-DMAC may not match the number of skipped bytes.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CS[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5 to 0	CS[15:0]	000000	R	Intelligent Checksum Value These bits are read-only. Writing is prohibited. These bits are initialized when the beginning of a packet for reception is detected.

Note: This register is only valid when CSEL = 1 and CSMD = 0.

24.3.68 RMII_MII Select Byte Register (RMII_MII)

RMII_MII is a readable/writable 32-bit register that selects the mode from among the GMII, MII, or RMII.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	rmii_st_1	rmii_st_0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	rmii_st[1:0]	00	R	RMII_MII Selection Specify RMII, MII, or GMII mode. H'0: RMII H'1: MII H'2: GMII H'3: Reserved

24.4 Operation

The GETHER consists of the following three function units:

- DMA transfer controller (E-DMAC): DMA transfer between the transmit/receive buffer in the memory and the transmit/receive FIFO
- MAC controller (E-MAC): Transmission/reception processing between the transmit/receive FIFO and the GMII/MII/RMII
- Transfer Switching Unit (TSU): Transfer processing, and CAM processing

Using its direct memory access (DMA) function, the E-DMAC performs DMA transfer of frame data between a user-specified Ethernet frame transmission/reception data storage destination (accessible memory space: transmit buffer/receive buffer) and the transmit/receive FIFO in the E-DMAC. The user cannot read and write data from and to the transmit/receive FIFO directly via the CPU.

To enable the E-DMAC to perform DMA transfer, information (data) including a transmit/receive data storage address and so forth, referred to as a descriptor, is required. The E-DMAC reads transmit data from the transmit buffer or writes receive data to the receive buffer according to the descriptor information. By arranging multiple descriptors as a descriptor row (list) (to be placed in a readable/writable memory space), multiple Ethernet frames can be transmitted or received continuously.

The E-MAC constructs an Ethernet frame using the data written to the transmit FIFO and transmits the frame to the GMII/MII/RMII. It also performs a CRC check of an Ethernet frame received from the GMII/MII/RMII and deconstructs the frame to write to the receive FIFO. The E-MAC supports three formats MII, GMII and RMII for interface to the PHI-LSI connected externally to this LSI.

The TSU, which is placed between the E-DMAC and E-MAC, references the CAM entry table to select one of the following tasks according to the Ethernet frame destination address (DA) input to the E-MAC.

- Receives data and writes to the receive FIFO.
- Discards data.

Figure 24.2 shows the frame data path and an overview of each setting.

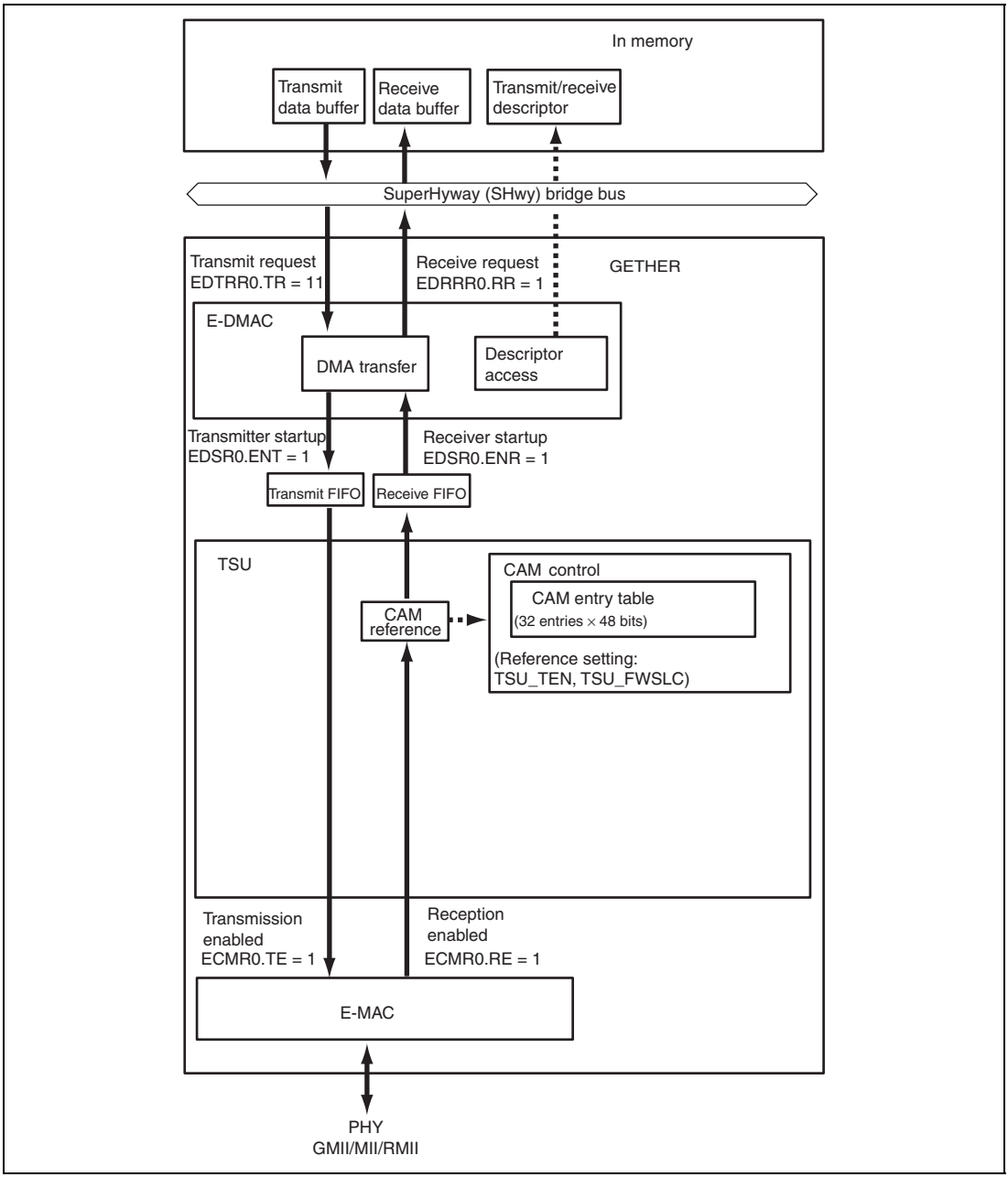


Figure 24.2 GETHER Data Path and Various Settings

24.4.1 Descriptors and Descriptor List

The E-DMAC performs DMA transfer according to the information (data), referred to as a descriptor, written in memory space. There are two types of descriptors: transmit descriptors and receive descriptors. Before a DMA transfer, DMA transfer information including a transmit/receive frame data storage address must be set by software.

The E-DMAC automatically starts reading a transmit/receive descriptor when the TR bits in EDTRR are set to 11 or the RR bit in EDRRR is set to 1, and performs DMA transfer of frame data between the transmit/receive buffer and transmit/receive FIFO according to the information stored in the descriptor. After completion of Ethernet frame transmission/reception, the E-DMAC disables the descriptor valid/invalid bit and reflects the result of transmission/reception in the status bits.

Descriptors are placed in a readable/writable memory space. The address of the start descriptor (descriptor to be read first by the E-DMAC) is set in TDLAR/RDLAR. When multiple descriptors are prepared as a descriptor row (descriptor list), the descriptors are placed in continuous addresses (memory) according to the descriptor length set in the DL0 and DL1 bits in EDMR.

(1) Transmit Descriptor

Figure 24.3 shows the configuration of a transmit descriptor and the relationship with a transmit buffer.

The data of a transmit descriptor consists of TD0, TD1, TD2, and padding data in groups of 32 bits from top to end. The length of padding data is determined according to the descriptor length specified by the DL0 and DL1 bits in EDMR.

TD0 indicates whether the transmit descriptor is valid or invalid, and information about the descriptor configuration and status. TD1 indicates the length of data in a transmit buffer to be transferred (TDL) as specified by the descriptor. TD2 indicates the start address of a transmit buffer that holds data to be transferred (TBA).

Depending on the descriptor specification, one transmit descriptor can specify all transmit data of one frame (single-frame/single-buffer) or multiple descriptors can specify the transmit data of one frame (single-frame/multi-buffer). As an example of single-frame/multi-buffer operation, the data portion that is used in a fixed manner in each Ethernet frame transmission can be referenced by multiple descriptors. For example, multiple descriptors can share the destination address and transmit source address in an Ethernet frame, and the remaining data can be stored in each separate buffer.

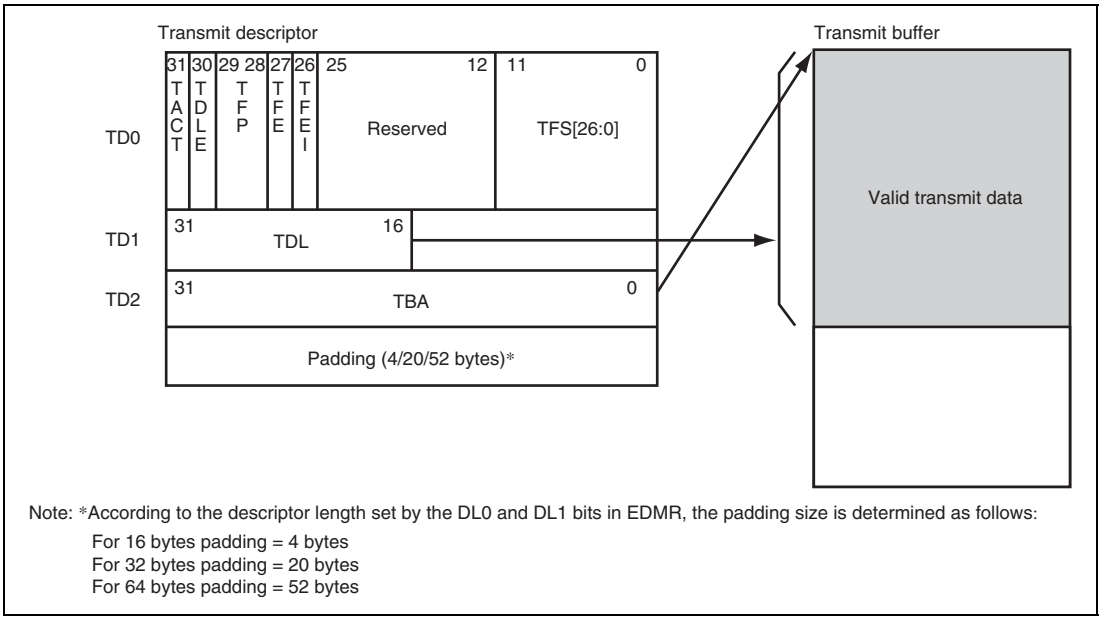


Figure 24.3 Relationship between Transmit Descriptor and Transmit Buffer

(a) Transmit Descriptor 0 (TD0)

Before the TR bits in EDTRR are set to 11, the user sets whether the bits of the descriptor are valid or invalid bit and sets other descriptor configuration. After Ethernet frame transmission, the E-DMAC disables the valid/invalid bits of the descriptor and writes status information. This operation is referred to as write-back.

When using TD0, the user should write desired values to bits 31 to 28 and 26 according to the descriptor configuration. Bits 27 and 25 to 0 should be cleared to 0.

Bit	Bit Name	Initial Value	R/W	Description
31	TACT	0	R/W	Transmit Descriptor Valid/Invalid Indicates whether the corresponding descriptor is valid or invalid. To make this bit valid, store transmit data in a transmit buffer (user-specified transmit data storage destination) beforehand, then write 1 to this bit. The E-DMAC clears this bit to 0 after data transfer. 0: Indicates that this transmit descriptor is invalid Indicates the initial setting state, the state after 0 is written, or (in case the user writes 1 to this bit) that this bit is cleared to 0 because the E-DMAC data transfer processing is completed. If this state is recognized when the E-DMAC reads a descriptor, the E-DMAC clears the TR bit in EDTRR to 0, and halts transfer operation related to transmission by the E-DMAC. 1: Indicates that this transmit descriptor is valid After the user writes 1 to this bit, this bit indicates that data is not transferred yet or data is being transferred. When there is a descriptor row (descriptor list) consisting of multiple continuous descriptors, the E-DMAC can continue operation when this bit of the next descriptor is valid.
30	TDLE	0	R/W	Transmit Descriptor List End Indicates whether the corresponding descriptor is the last descriptor of the descriptor row (descriptor list). 0: Not last descriptor After transfer of the corresponding descriptor, the E-DMAC reads the next one in the list of continuous descriptors. 1: Last descriptor After transfer of the corresponding descriptor, the E-DMAC reads the descriptor placed at the address indicated by TDLAR.

Bit	Bit Name	Initial Value	R/W	Description
29, 28	TFP[1:0]	00	R/W	Transmit Frame Position These bits indicate whether information of this descriptor represents information about the start, middle, or end of the transmit frame. 00: The information of the descriptor represents information about the middle of the frame. 01: The information of the descriptor represents information about the end of the frame. 10: The information of the descriptor represents information about the start of the frame. 11: The information of the descriptor represents all information about the frame (single-frame/single-descriptor (single-buffer)). Reference When one frame is divided for use, the method of specifying this bit for a descriptor row according to the number of divisions is described below. • For single-frame/single-descriptor operation First descriptor: TFP[1:0] = 11 • For single-frame/two-descriptor operation First descriptor: TFP[1:0] = 10 Second descriptor: TFP[1:0] = 01 • For single-frame/three-descriptor operation First descriptor: TFP[1:0] = 10 Second descriptor: TFP[1:0] = 00 Third descriptor: TFP[1:0] = 01 When the number of divisions is large, a descriptor row is configured by adding intermediate descriptors with TFP[1:0] = 00.
27	TFE	0	R/W	Transmit Frame Error Occurrence Indicates that an error occurred in the transmit frame. 0: The TFS11 to TFS0 bits are all 0 1: One of the TFS11 to TFS0 bits is 1 The TFS8 to TFS0 bits can be masked for each factor by using TRSCER. TheTFS11 to TFS9 bits cannot be masked. This bit is set by the E-DMAC write-back operation.

Bit	Bit Name	Initial Value	R/W	Description
26	TWBI	0	R/W	Write-Back Completion Interrupt Notification 0: Does not notify of a write-back completion interrupt 1: After a write-back operation to this descriptor is complete, this bit sets the TWB1 and TWB0 bits in EESR to 11 and notifies the CPU of a write-back completion interrupt. This bit is valid only for the descriptor including the end of transmit frame (TFP = 01 or 11). This bit is cleared to 0 by the E-DMAC write-back operation.
25 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	TFS [11:0]	All 0	R/W	Transmit Frame Status These bits indicate the status of the corresponding frame. A bit below, which is set by the E-DMAC write-back operation, indicates the occurrence of the corresponding event when set to 1. • TFS[11:10]: Reserved (The write value should always be 0.) • TFS[9]: Transmit FIFO underflow (Corresponding to the TUC bit in EESR) • TFS[8]: Detection of transmission abort (Corresponding to the TABT bit in EESR) • TFS[7:3]: Reserved (The write value should always be 0.) • TFS[2]: Detection of carrier loss (Corresponding to the DLC bit in EESR) • TFS[1]: Detection of delay collision (Corresponding to the CD bit in EESR) • TFS[0]: Transmit timeout (Corresponding to the TRO bit in EESR)

(b) Transmit Descriptor 1 (TD1)

TD1 indicates the data length of the transmit buffer used by the corresponding descriptor.

The user should set TD1 before the start of a read by the E-DMAC.

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	TDL [15:0]	All 0	R/W	Transmit Buffer Data Length (in bytes) These bits indicate the data length of the corresponding transmit buffer in bytes. The maximum length is between 64 Kbytes and 32 bytes (H'FFE0).
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

(c) Transmit Descriptor 2 (TD2)

TD2 indicates the start address of the corresponding 32-bit width transmit buffer. An address value should be specified in a longword boundary.

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TBA [31:0]	All 0	R/W	Transmit Buffer Start Address These bits set the start address of the corresponding transmit buffer in a 16-bit boundary.

If descriptors are set below, the E-DMAC does not return to normal operation until a system reset is performed.

- TFP (transmit frame position) is not logically correct
  Example: The TFP bits are set to 11 in a descriptor (descriptor A) and the TFP bits are set to 01 in the next descriptor (descriptor B). This specification means that there is no descriptor indicating the start of the transmit frame specified by descriptor B.
- TBL (transmit buffer length) is set to 0

When one transmit frame is divided into three parts or more with transmit descriptors, the E-DMAC performs the following write-back operation:

- A write-back operation is performed for a transmit descriptor including information for the start of the transmit frame (TFP = 10 or 11) and for a transmit descriptor including information for the end of the frame (TFP = 01 or 11).
- A write-back operation is not performed for a transmit descriptor for the middle of the frame (TFP = 00).

However, TFE (transmit frame error occurrence) or TFS (transmit frame status) is written only to a transmit descriptor including information for the end of the frame (TFP = 01 or 11) by a write-back operation.

Before changing a transmit descriptor with the software, make sure that a write-back operation has been performed (TACT = 0) for the transmit descriptor including information for the end of the frame (TFP = 01 or 11) to avoid overwriting (re-setting) an unprocessed transmit descriptor.

(2) Receive Descriptor

Figure 24.4 shows the relationship between a receive descriptor and receive buffer.

The data of a receive descriptor consists of RD0, RD1, RD2, and padding data in groups of 32 bits from top to end. The length of padding data is determined according to the descriptor length specified by the DL0 and DL1 bits in EDMR.

RD0 indicates whether the receive descriptor is valid or invalid, and information about descriptor configuration and status. RD1 indicates the length of data that can be received in the receive buffer specified by the descriptor (RBL) and the length of the received frame data (RDL). RD2 indicates the start address of the receive buffer for storing receive data (RBA).

Depending on the descriptor specification, one receive descriptor can specify the storing of all receive data of one frame in a receive buffer (single-frame/single-buffer) or multiple descriptors can specify the storing of the receive data of one frame in receive buffers (single-frame/multi-buffer). As an example of single-frame/multi-buffer operation, suppose that a row of multiple descriptors (descriptor list) is prepared, RBL of each descriptor is 500 bytes, and a 1514-byte Ethernet frame is received. In such a case, the received Ethernet frame is transferred sequentially to buffers, 500 bytes for each buffer, starting with the first descriptor. Only the last 14 bytes are transferred to the fourth buffer. When a frame longer than RBL of a descriptor is received, the E-DMAC transfers the remaining data to the receive buffer by using the subsequent descriptors. As an example of efficient single-frame/multi-buffer operation, information items on different processing layers in an Ethernet frame can be separated from each other by using different buffers. For example, the destination address, transmit source address, and type field data in an Ethernet frame can be stored in buffer 1 (set RBL to 14 bytes) and the remaining data can be stored in buffer 2 (set RBL to 1500 bytes). All receive frames, of course, can be stored in a single buffer if multiple descriptors are prepared and RBL of each descriptor is set to more than 1514 bytes (maximum Ethernet frame length).

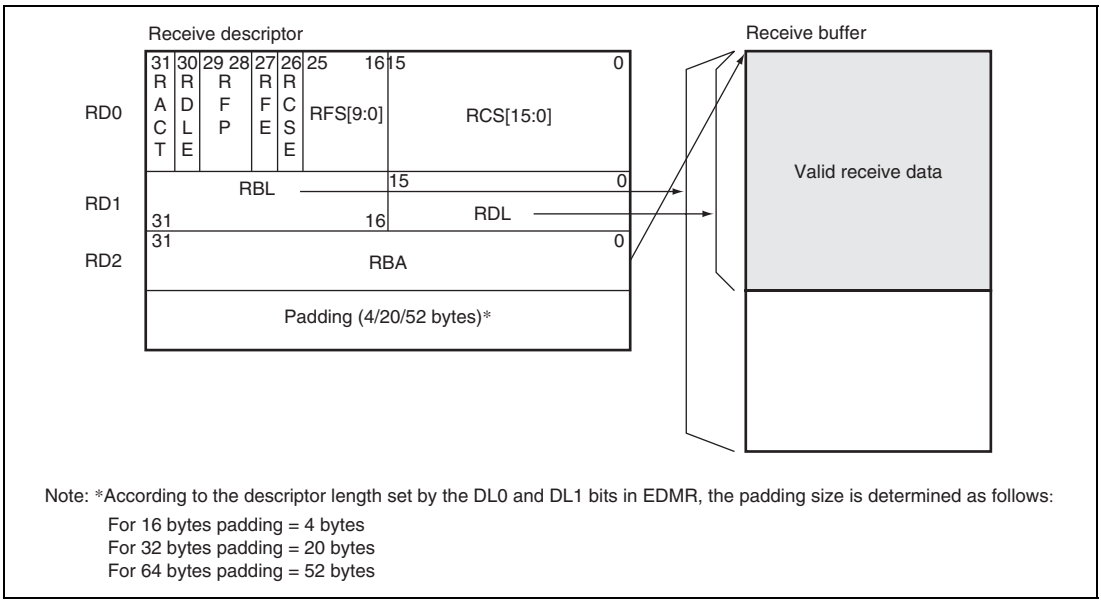


Figure 24.4 Relationship between Receive Descriptor and Receive Buffer

(a) Receive Descriptor 0 (RD0)

The user sets whether the bits of the descriptor are valid or invalid and whether the descriptor represents the end of the descriptor list in RD0 before the RR bit in EDRRR is set to 1 and the

start of a read by the E-DMAC. After receive DMA transfer of an Ethernet frame by the E-DMAC, the E-DMAC disables the valid/invalid bits of the descriptor and writes status information. This operation is referred to as write-back.

When using RD0, the user should write desired values to bits 31 and 30 according to the descriptor configuration. Bits 29 to 0 should be cleared to 0.

Bit	Bit Name	Initial Value	R/W	Description
31	RACT	0	R/W	Receive Descriptor Valid/Invalid Indicates whether this descriptor is valid or invalid. To make this bit valid, prepare a receive buffer (user-specified receive data storage destination) beforehand, then write 1 to this bit. The E-DMAC clears this bit to 0 after data transfer. 0: Indicates that this receive descriptor is invalid Indicates the initial setting state, the state after 0 is written to, or (in case the user writes 1 to this bit) that this bit is cleared to 0 because the E-DMAC data transfer processing is completed If this state is recognized when the E-DMAC reads a descriptor, the E-DMAC clears the RR bit in EDRRR to 0, and halts transfer operation related to reception by the E-DMAC 1: Indicates that this receive descriptor is valid Indicates that data is not transferred yet after the user writes 1 to this bit, or that data is being transferred When there is a descriptor row (descriptor list) consisting of multiple continuous descriptors, the E-DMAC can continue operation when this bit of the next descriptor is valid
30	RDLE	0	R/W	Receive Descriptor List End Indicates whether this descriptor is the last descriptor of the descriptor row (descriptor list). 0: Not last descriptor After transfer of this descriptor, the E-DMAC reads the next one in the list of continuous descriptors 1: Last descriptor After transfer of this descriptor, the E-DMAC reads the descriptor placed at the address indicated by RDLAR

Bit	Bit Name	Initial Value	R/W	Description
29, 28	RFP[1:0]	00	R/W	Receive Frame Position 1, 0 The E-DMAC indicates by write-back operation whether information of the corresponding descriptor represents information about the start, middle, or end of the receive frame. 00: The information of the descriptor represents information about the middle of the frame 01: The information of the descriptor represents information about the end of the frame 10: The information of the descriptor represents information about the start of the frame 11: The information of the descriptor represents all information about the frame (single-frame/single-descriptor (single-buffer)) Reference The relationship between a frame after reception of one frame and a descriptor is described below. • For single-frame/single-descriptor operation First descriptor: RFP[1:0] = 11 • For single-frame/two-descriptor operation First descriptor: RFP[1:0] = 10 Second descriptor: RFP[1:0] = 01 • For single-frame/three-descriptor operation First descriptor: RFP[1:0] = 10 Second descriptor: RFP[1:0] = 00 Third descriptor: RFP[1:0] = 01 When the number of divisions is large, a descriptor row is configured by adding intermediate descriptors with RFP[1:0] = 00.
27	RFE	0	R/W	Receive Frame Error Occurrence Indicates that an error occurred in the receive frame. 0: RFS11 to RFS0 are all 0 1: One of RFS11 to RFS0 is 0 Each of RFS8 to RFS0 can be masked by using TRSCER. RFS11 to RFS9 cannot be masked. This bit is set by the E-DMAC write-back operation.

Bit	Bit Name	Initial Value	R/W	Description
26	RCSE	0	R/W	Receive Packet Checksum Value Evaluation by Intelligent Checksum When CSEBL = 1 and CSMD = 1, the value of this bit is set as shown in table 24.4, according to the receive packet and receive data. The information of this bit is invalid when operation is performed with a setting other than above.
25 to 16	RFS[9:0]	All 0	R/W	Receive Frame Status These bits indicate the error status during frame reception. RFS9: Receive FIFO overflow (corresponding to the RFOF bit in EESR) RFS8: Reserved (write value should be 0) RFS7: Multicast address frame received (corresponding to the RMAF bit in EESR) RFS6: CAM entry unregistered frame received (corresponding to the RUAF bit in EESR) RFS5: Reserved (write value should be 0) RFS4: Residual-bit frame receive error (corresponding to the RRF bit in EESR) RFS3: Long frame receive error (corresponding to the RTLF bit in EESR) RFS2: Short frame receive error (corresponding to the RTSF bit in EESR) RFS1: PHY-LSI receive error (corresponding to the PRE bit in EESR) RFS0: CRC error on receive frame (corresponding to the CERF bit in EESR)
15 to 0	RCS[15:0]	All 0	R/W	Receive Packet Checksum Value in Intelligent Checksum

Table 24.4 RCSE State Determined by Receive Packet Type and Receive Data

Frame Type		Normal Data		Abnormal Data	
IP Version	Option and Extension Header	RCS[15:0]	RCSE	RCS[15:0]	RCSE
IPv4	None	H'FFFF H'0000	0	Undefined	1
	Fragment	Undefined	Undefined	Undefined	Undefined
	Option	H'FFFF H'0000	0	Undefined	1
IPv6	None	H'FFFF H'0000	0	Undefined	1
	Hop-by-hop	H'FFFF H'0000	0	Undefined	1
	Routing	H'FFFF H'0000	0	Undefined	1
	Destination options	H'FFFF H'0000	0	Undefined	1
	AH	H'FFFF H'0000	0	Undefined	1
	Fragment	Undefined	Undefined	Undefined	Undefined
	ESP	H'0000	1	H'0000	1
	MobileIPv6	H'0000	1	H'0000	1
	Others	H'0000	1	H'0000	1
Other than IPv4 or IPv6		H'0000	0	H'0000	0

(b) Receive Descriptor 1 (RD1)

In RD1, the user specifies the data length of a receive buffer usable by the corresponding descriptor. After reception of a frame, RD1 indicates the length of a frame received by the E-DMAC.

The user should set RD1 before the start of a read by the E-DMAC.

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	RBL [15:0]	All 0	R/W	Receive Buffer Data Length (in bytes, to be specified with a 32-byte boundary) These bits set the length of data that can be received by the corresponding receive buffer with an integral multiple of 32 bytes. The maximum receive buffer data length is between 64 Kbytes and 32 bytes (H'FFE0).
15 to 0	RDL [15:0]	All 0	R	Receive Data Length These bits indicate the data length of a receive frame stored in the receive buffer. Receive data transferred to the receive buffer does not include CRC data (4 bytes) placed at the end of a frame. As a receive frame length, the number of bytes (valid data bytes) not including CRC data are reported. In single-frame/multi-buffer (descriptor) operation, only the receive data length of the last descriptor is valid. The receive data length of an intermediate descriptor has no meaning. The maximum frame length that can be received is: When padding function is invalid: 64 Kbytes between 1 byte (H'FFFF) When padding function is valid: 64 Kbytes between 32 bytes (H'FFE0)

(c) Receive Descriptor 2 (RD2)

RD2 indicates the start address of the corresponding receive buffer. Set the start address of a receive buffer with a 32-byte boundary.

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	RBA [31:0]	All 0	R/W	Receive Buffer Start Address These bits set the start address of the corresponding receive buffer with a 32-byte boundary.

The E-DMAC performs DMA transfer for a receive frame from the address specified by RBA (receive buffer address) to the receive buffer in 32-byte units. RBL (receive buffer length) must be set to be an integral multiple of 32 bytes.

If data to be transferred is less than 32 bytes, invalid data will be written to.

[Example]

When the receive frame length is 170 bytes and the required receive buffer capacity is 192 bytes (32 bytes × 6), the sixth DMA-transfer causes invalid data to be written to the receive buffer (In the 32-byte DMA data, the former 10 bytes are valid and the latter 22 bytes are invalid).

Padding of the value 0 can be inserted into only one position in the receive frame by setting RPADIR. The padding size can be selected from 1 byte to 31 bytes in byte units. When padding is inserted into a receive frame, a receive buffer area equal to the total of "receive frame length and padding size" is required. RPADIR setting is valid for all receive frames.

RFE (receive frame error occurrence), PV (padding insertion), RFS (receive frame status) and RFS (receive frame status) are only set in the receive descriptor including information for the end of the frame (TFP = 01 or 11) by a write-back operation.

Before re-setting a receive descriptor with the software, completion of a write-back operation for the receive descriptor (RACT = 0) must be confirmed to avoid rewriting to (and re-setting) an unprocessed receive descriptor.

(3) Descriptor and Transmit/Receive Buffer

(a) Transmission

Each transmit descriptor specifies one transmit buffer. The E-DMAC transfers a transmit frame stored in a transmit buffer specified by a transmit descriptor to the transmit FIFO. Multiple transmit frames stored in transmit buffers specified by multiple descriptors can be connected into one transmit frame and transferred to the transmit FIFO.

Figure 24.5 shows the relationship between the transmit descriptors and transmit buffers.

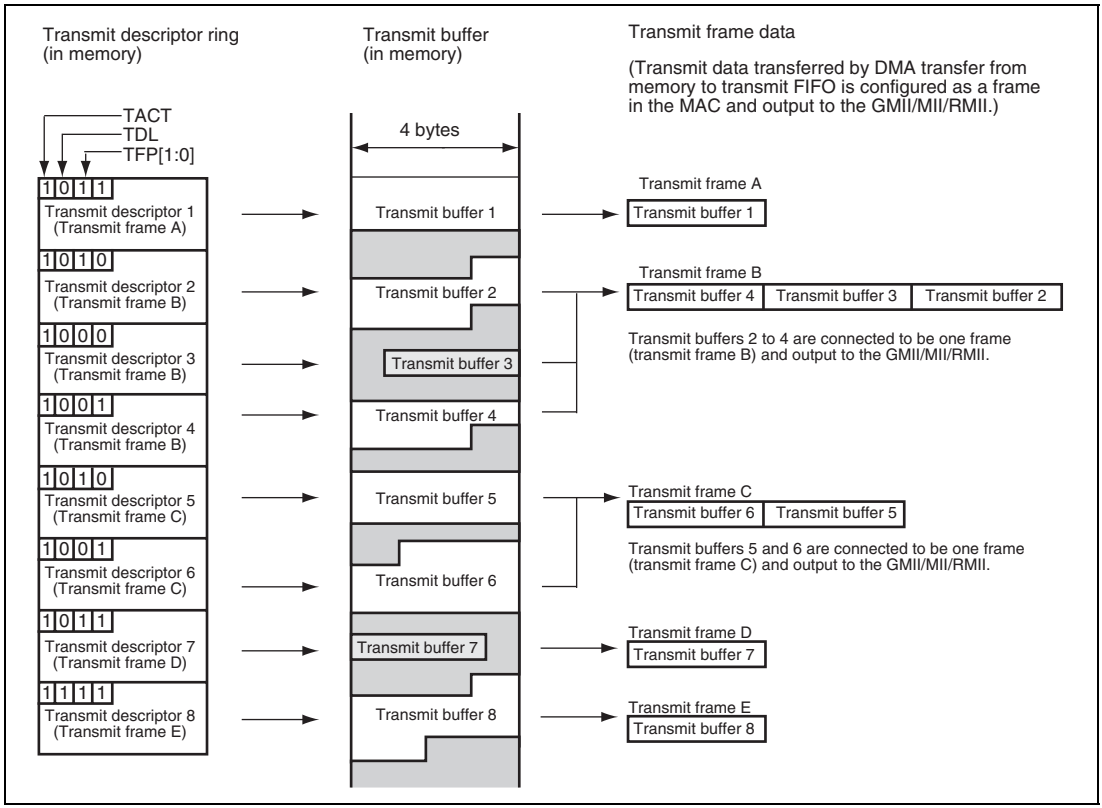


Figure 24.5 Relationship between Transmit Descriptor and Transmit Buffer

(b) Reception

Each receive descriptor specifies one receive buffer. The E-DMAC receives a receive frame from the receive FIFO and stores it in a receive buffer specified by a receive descriptor. If the receive frame size exceeds the receive buffer size, the remaining data of the receive frame can be stored in a different receive buffer specified by a different receive descriptor. Thus, one receive frame can be stored in multiple receive buffers.

Figure 24.6 shows the relationship between the receive descriptors and receive buffers.

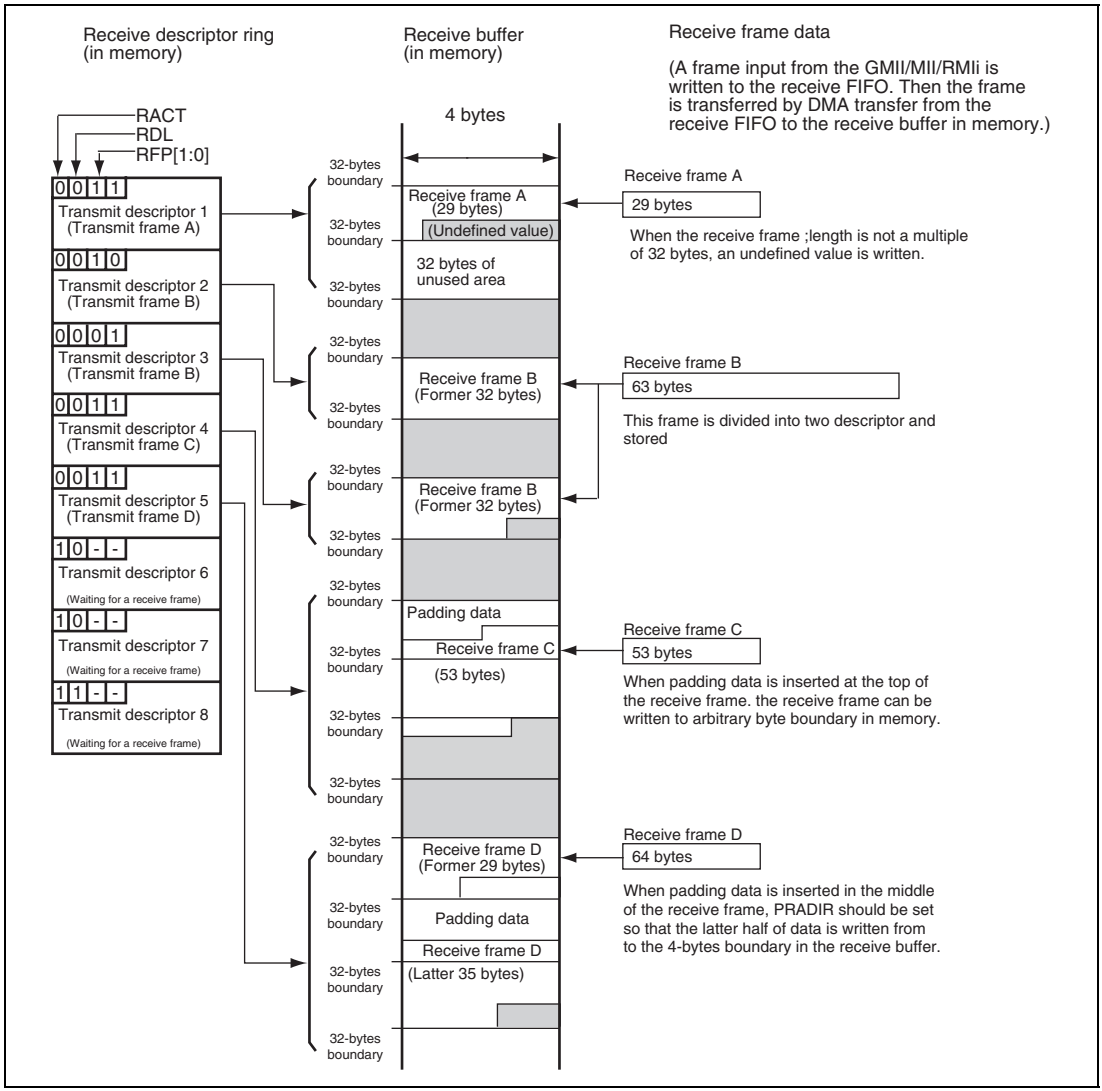


Figure 24.6 Relationship between Receive Descriptor and Receive Buffer

(4) Descriptor Pointer

The E-DMAC controls the transmit and receive descriptor addresses in memory and the processing priority by using the following registers.

1. Registers related to a transmit descriptor
- TDLAR: Address of the start descriptor in a list of transmit descriptors.
- TDFAR: Address of the transmit descriptor to be processed
- TDFXR: Address of the transmit descriptor that finished processing (set by a write-back operation) last
- TDFFR (DL bit): Indicates whether the TDLE value of the transmit descriptor specified by TDFXR is 1 or not.

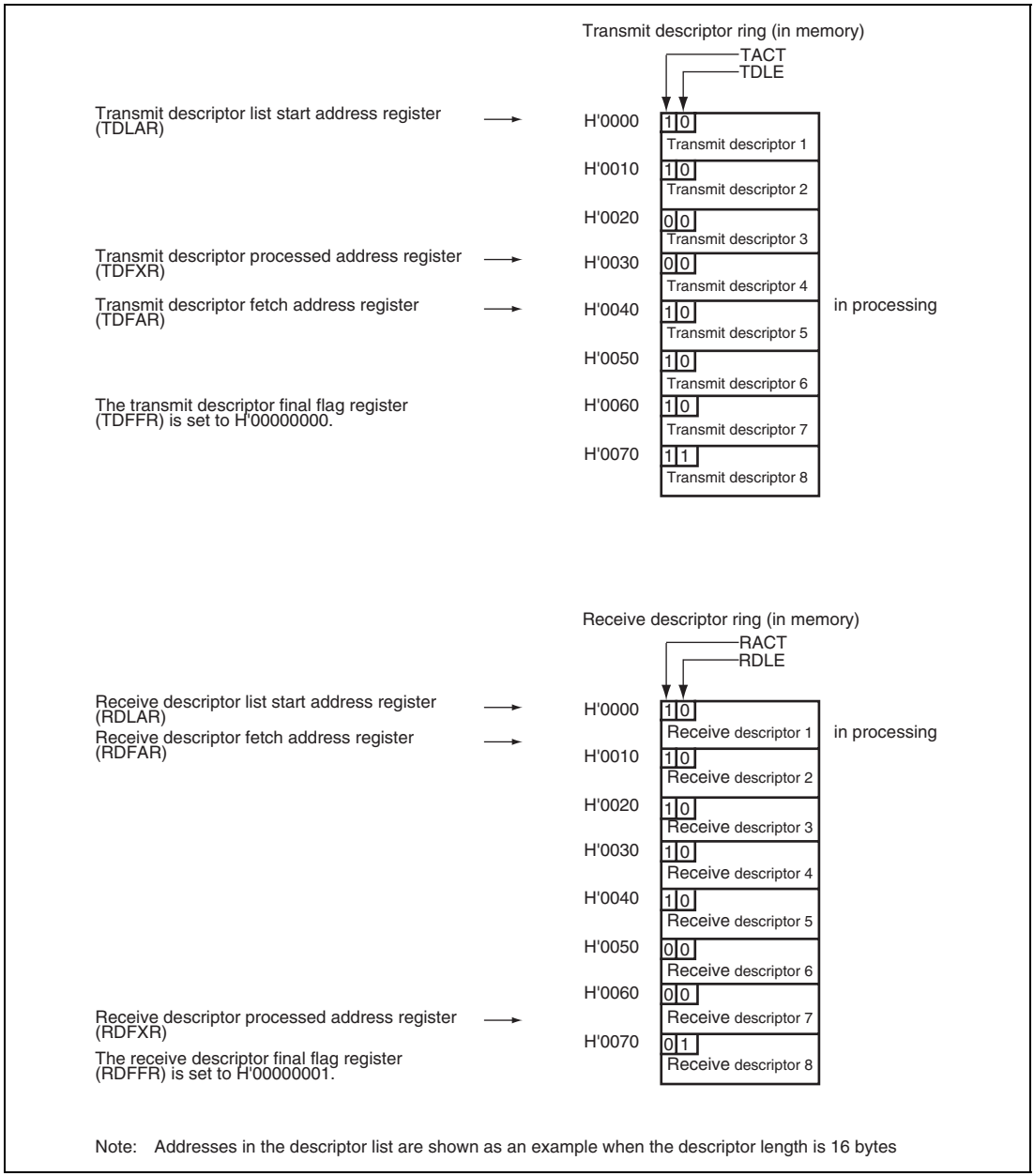
2. Registers related to receive descriptor:
- RDLAR: Address of the start descriptor in a list of receive descriptors.
- RDFAR: Address of the receive descriptor to be processed
- RDFXR: Address of the receive descriptor that finished processing (set by a write-back operation) last
- RDFFR (DL bit): Indicates whether the RDLE value of the receive descriptor specified by RDFXR is 1 or not.

Transmit descriptors and receive descriptors have a ring structure. When the TDLE (RDLE) value of the processed transmit (receive) descriptor is 0, the next descriptor will be processed. The next descriptor is the transmit (receive) descriptor at the address obtained by adding the processed transmit (receive) descriptor address to the descriptor length specified by the DL bits in EDMR. When the TDLE (RDLE) value of the processed transmit (receive) descriptor is 1, the transmit descriptor indicated by TDLAR (RDLAR) will be processed next. Figure 24.7 shows the relationship between the transmit/receive descriptor ring and read pointer.

The transmit descriptor list must be large enough to point to five or more transmit frames. If four or less transmit frames are pointed to in a list, E-DMAC operation is not guaranteed. Accordingly, do not set that all the transmit descriptors in a ring are used by four or less descriptors. The receive descriptor list does not have this restriction. For example, one receive frame can use all receive descriptors in a list.

In the initial setting, the start address of a descriptor list must be set to TDLAR (RDLAR) and TDFAR (RDFAR), and the end descriptor address of the descriptor list to TDFXR (RDFXR) by the software.

The E-DMAC updates TDFAR (RDFAR), TDFXR (RDFXR), and the DL bit in TDFFR (DL bit in RDFFR) each time a descriptor is processed.



Note: Addresses in the descriptor list are shown as an example when the descriptor length is 16 bytes

Figure 24.7 Relationship between Transmit/Receive Descriptor and Descriptor Pointing Registers

24.4.2 Transmission

(1) Transmission Procedure and Processing Flow

When 11 is written to the TR bits in EDTRR with the TE bit in ECMR set to 1 and there is empty space of 32 bytes or more in the transmit FIFO, the E-DMAC reads the descriptor following the previously used descriptor from the transmit descriptor list (or the descriptor indicated by TDLAR at the initial startup).

If the TACT bit of the read descriptor is set to 1 (valid), the E-DMAC sequentially reads transmit frame data from the transmit buffer start address specified by TD2 and transfers the data to the transmit FIFO. The E-DMAC configures a transmit frame and starts transmission to the GMII/MII/RMII. After DMA transfer of data equivalent to the buffer length specified in the descriptor, the following processing is carried out according to the TFP value.

- TFP = 10 (start of a frame)
  Descriptor write-back (writing 0 to the TACT bit) is performed after completion of DMA transfer.
- TFP = 01 or 11 (end of a frame)
  Descriptor write-back (writing 0 to the TACT bit and writing status) is performed after completion of frame transmission.
- TFP = 00 (frame continued)
  Descriptor write-back is not performed. The TACT bit retains the value 1.

As long as the TACT bit of a read descriptor is set to 1 (valid), the reading of E-DMAC descriptors and the transmission of frames continue.

When a descriptor with the TACT bit cleared to 0 (invalid) is read, the E-DMAC performs the following processing and completes transmit processing.

- Clears the TR bits in EDTRR to 00.
- Writes the TC bits in EESR to 11 and generates an interrupt to the CPU.

The E-DMAC can store up to four frames of data in the transmit FIFO.

When the following conditions are satisfied, the E-MAC transmit processing section reads transmit data from the transmit FIFO to configure a frame and transmits the frame to the GMII/MII/RMII

- The amount of data in the transmit FIFO exceeds the number of bytes specified by TFTR.
- One or more frame of data is stored in the transmit FIFO.
- The transmit FIFO has no space (full of transmit wait data for the GMII/MII/RMII).

Figure 24.8 shows an example of transmission flow.

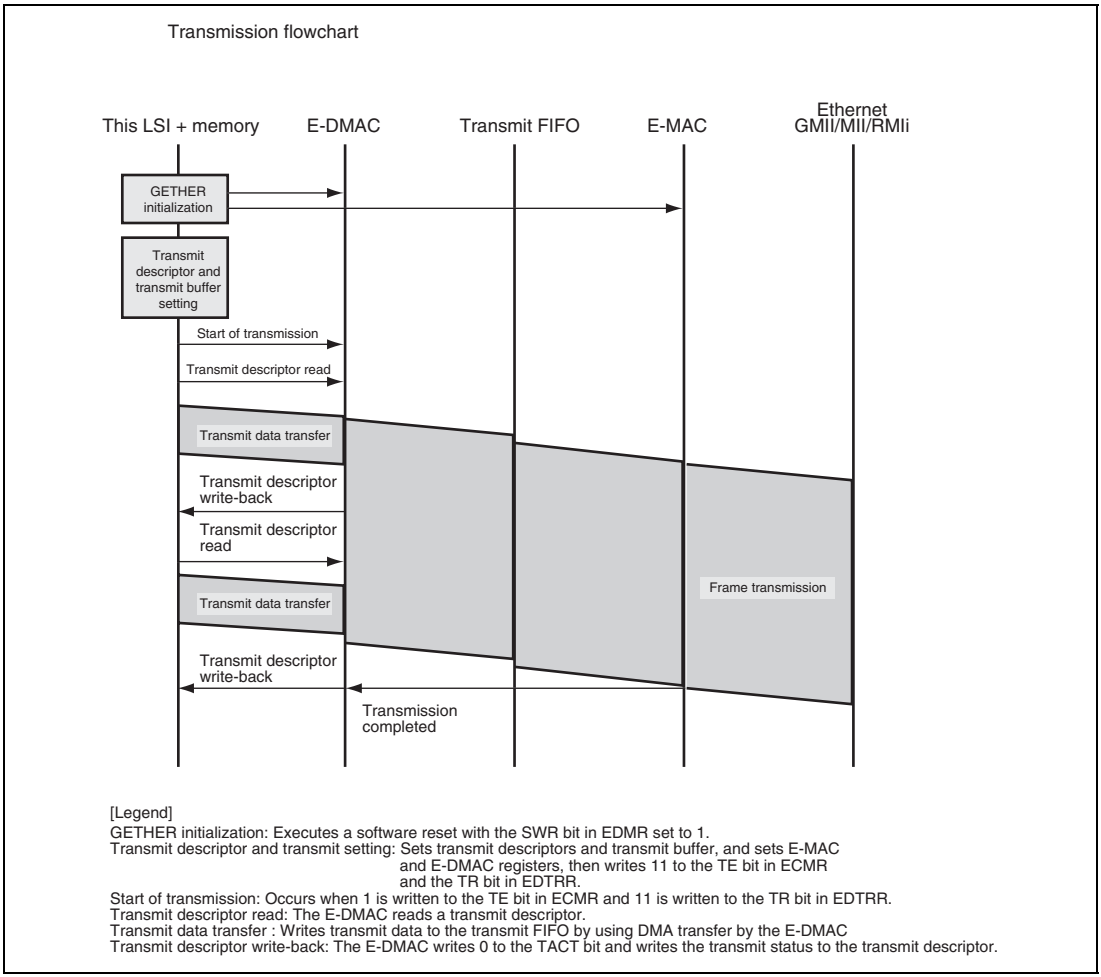
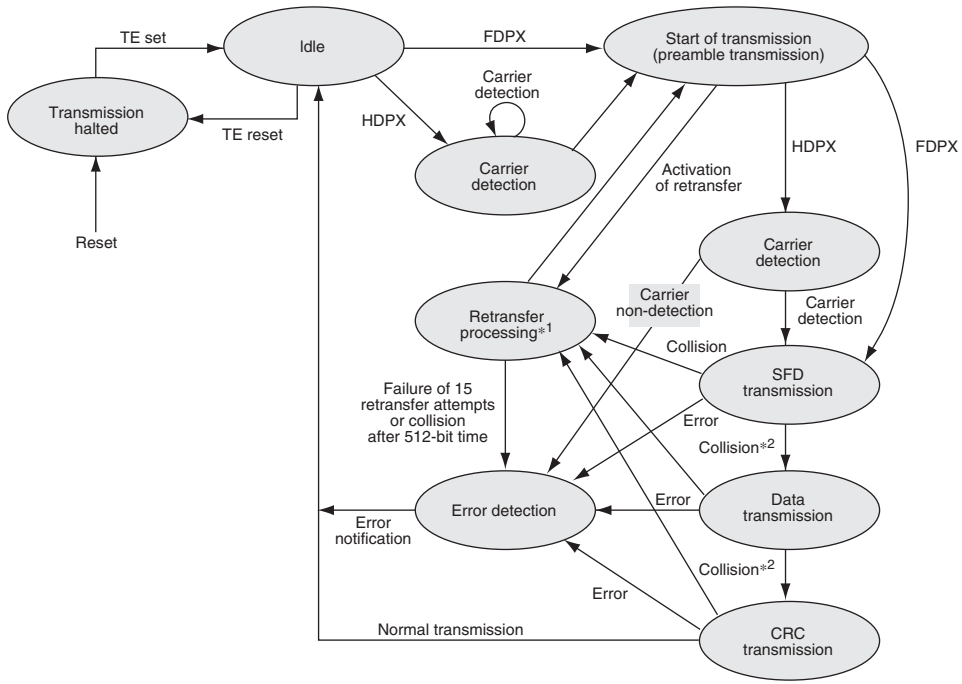


Figure 24.8 Sample Transmission Flowchart (Single-Frame/Two-Description)

Figure 24.9 shows the status change of the E-MAC transmitter.

1. When the TE bit in ECMR is set, the transmitter enters the transmit idle state.
2. When a transmit request is issued by the transmit E-DMAC, the E-MAC sends the preamble to GMII/MII/RMII after a transmission delay caused by the carrier detection and frame interval time. If full-duplex transfer is selected, which does not require carrier detection, the preamble is sent as soon as a transmit request is issued by the E-DMAC.
3. The transmitter sends the SFD, data, and CRC sequentially. At the end of transmission, the transmit E-DMAC generates a transmission complete interrupt (TC). If a collision or the carrier-not-detected state occurs during data transmission, these are reported as interrupt sources.
4. After waiting for the frame interval time, the transmitter enters the idle state, and if there is more transmit data, continues transmitting.

Transmission flowchart



Legend
FDPX: Full-duplex
HDPX: Half-duplex

Notes:
1. Transmission retry processing includes both jam transmission that depends on collision detection and the adjustment of transmission intervals based on the back-off algorithm.
2. Transmission is retried only when data of 512 bits or less (including the preamble and SFD) is transmitted. When a collision is detected during the transmission of data greater than 512 bits, only jam is transmitted and transmission based on the back-off algorithm is not retried.

Figure 24.9 E-MAC Transmitter State Transitions

(2) Transmission Error Processing

(a) Transmission Abort

If a transmission error is detected during frame transmission from the transmit FIFO to the GMII/MII/RMII, transmission of the frame data is aborted. At this time, if DMA transfer of the appropriate frame from the transmit buffer to the transmit FIFO has not been completed, the DMA transfer is also aborted.

Following a write-back operation to the transmit descriptor related to the transmit frame aborted by a transmission error, 1 is written to the TABT bit in EESR and an interrupt is issued to the CPU. The subsequent transmit descriptors will be processed normally.

(b) Transmit FIFO Underflow

If the transmit FIFO is empty (transmit FIFO underflow) during frame transmission from the transmit FIFO to the GMII/MII/RMII, the E-MAC forcibly aborts transmission of the frame to the GMII/MII/RMII. At this time, the frame that the E-MAC receives from the E-DMAC is cut off halfway. Then, the E-MAC performs the following operation:

- Writes the TFUF bit in EESR to 1 and generates an interrupt to the CPU.
- Performs a write-back operation to the transmit descriptor corresponding to the transmit frame.
- Following the write-back operation, writes the TUC bit in EESR and generates an interrupt to the CPU.

The subsequent transmit descriptors operate normally.

The E-MAC waits to start frame transmission from the transmit FIFO to the GMII/MII/RMII until the data that was stored in the transmit FIFO exceeds the number of the bytes specified by TFTR. Through the effective use of TFTR, the transmit FIFO underflow counts can be controlled.

(c) Transmit Descriptor Empty

When the TFP bits of the descriptor previously processed are set to 00 or 10 and the TACT bit of the read transmit descriptor is set to 0 (invalid), a transmit descriptor empty state is determined and 1 is written to the TDE bit in EESR, and then an interrupt is issued to the CPU.

When a transmit descriptor state is empty, start transmission processing after a software reset.

24.4.3 Reception

(1) Reception Procedure and Processing Flow

The E-MAC receiver separates the frame from the GMII/MII/RMII into preamble, SFD, data and CRC, and transfers the fields from DA (destination address) to the data to the receive FIFO. Up to 24 frames can be written in the receive FIFO. Figure 24.10 shows the status change of the E-MAC receiver.

1. When the RE bit in ECMR is set to 1, the receiver enters the receive idle state.
2. When an SFD (start frame delimiter) is detected after a receive packet preamble, the receiver starts receive processing. A frame with an invalid pattern is discarded.
3. In normal mode, if the destination of the frame address is this LSI, the receiver starts data reception when broadcast or multicast transmission is specified. In promiscuous mode, data reception starts regardless of the frame type.
4. Following data reception from the GMII/MII/RMII, the receiver carries out a CRC check. The result is indicated as a status bit in the descriptor after the frame data has been written to the receive FIFO. Reports an error status in the case of an abnormality.

After one frame has been received, if the RE bit in ECMR is set to 1, the receiver prepares to receive the next frame.

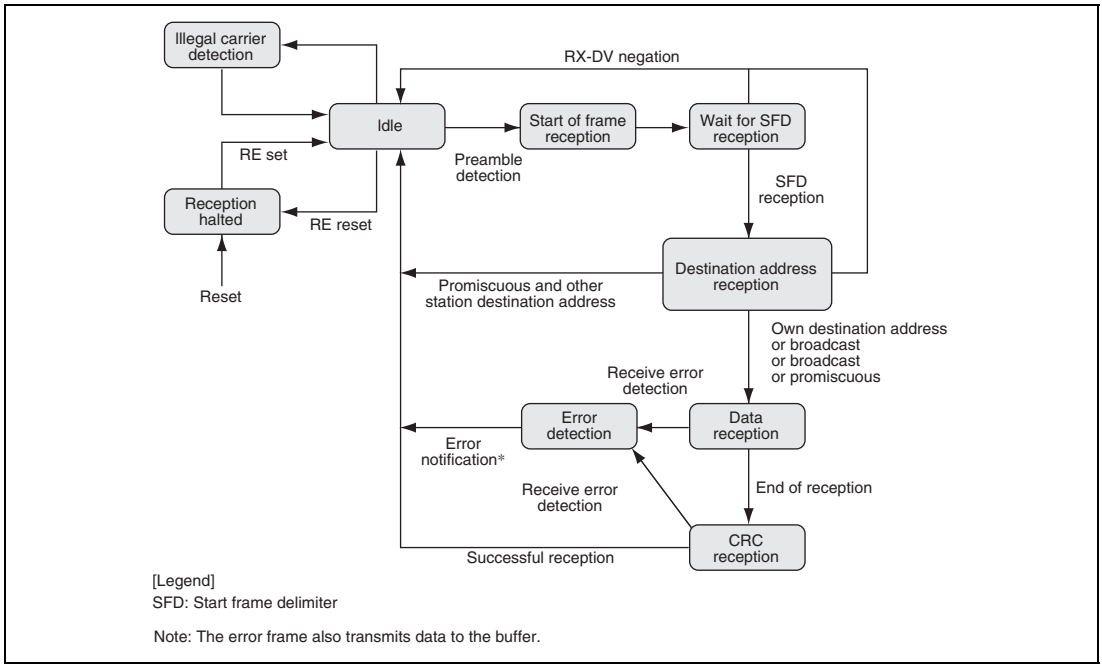


Figure 24.10 E-MAC Receiver State Transitions

CAM evaluation can be referenced during frame processing in reception (for details on the CAM function, refer to section 24.4.5, CAM Function).

When 1 is written to the RR bit in EDRRR while the RE bit in ECMR is set to 1, the E-DMAC reads the descriptor following the previously used descriptor from the receive descriptor list (or the descriptor indicated by RDLAR at the initial startup) then enters the receive wait state. If 32 bytes or more of data or the last byte of the receive frame is stored in the receive FIFO, the E-DMAC transfers receive FIFO data to the receive buffer specified by RD2 according to the receive descriptor with the RACT bit set to 1 (valid).

If the data length of a received frame is longer than the buffer length specified by RD1, the E-DMAC performs a write-back operation to the descriptor (set RFP to 10 or 00) when the buffer is full, then reads the next descriptor. The E-DMAC then continues to transfer data to the receive buffer specified by the new RD2.

When the following conditions are satisfied, a write-back operation is performed for the descriptor (RFP = 11 or 01), 11 is written to the FR bits in EESR, and an interrupt is issued to the CPU.

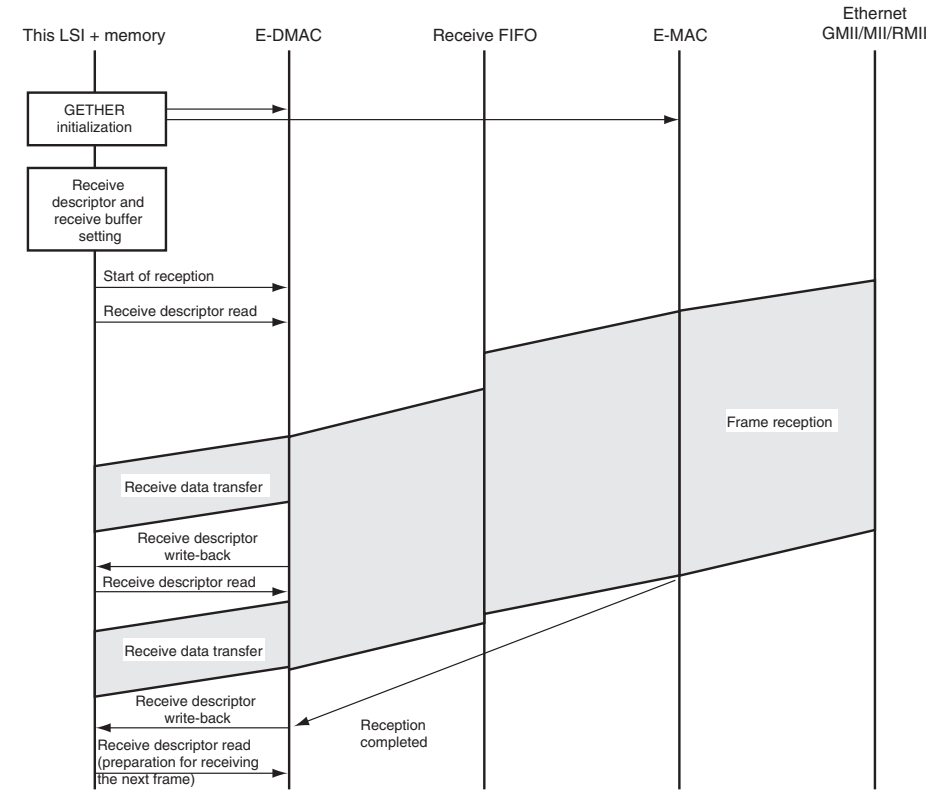
- The receive buffer has been full during DMA transfer.
- DMA transfer to the receive buffer of the last byte of the receive frame has been completed.

After the reception processing of the frame, the next descriptor reading standby state begins. At this time, if 32 bytes or more of data or the last byte of the receive frame is stored in the receive FIFO, the next receive descriptor process is performed continuously.

When the TACT bit of the read receive descriptor is 0 (invalid), the receive descriptor empty state is determined and the RDE bit in EESR is written to 1, and then an interrupt is issued to the CPU.

To receive frames continuously, set the RNC bit in RMCR to 1. The initial value is 0.

Figure 24.11 shows an example of reception flow.



[Legend]

GETHER initialization: Executes a software reset with the SWR bit in EDMR set to 1.

Receive descriptor and receive buffer setting: Sets receive descriptors and receive buffers, and sets E-MAC and E-DMAC registers, then writes 1 to the RE bit in ECMR and the RR bit in EDRRR.

Start of Reception: Occurs when 1 is written to the RE bit in ECMR and the RR bit in EDRRR.

Receive descriptor read: The E-DMAC reads a receive descriptor.

Receive data transfer: Writes receive data from the receive FIFO to the receive buffer by using DMA transfer by the E-DMAC.

Receive descriptor write-back: The E-DMAC writes 0 to the RACT bit and writes the receive status to the receive descriptor

Figure 24.11 Sample Reception Flowchart (Single-Frame/Two-Descriptor)

(2) Reception Error Processing

(a) Reception Error

When a reception error occurs, the FR and RABT bits in EESR are set to 1 and an interrupt is issued to the CPU after a write-back operation for the receive descriptor related to the reception error frame.

If a reception error occurs when the length of the frame received from the GMII/MII/RMII is less than 32 bytes, DMA transfer to the receive buffer for the frame is not performed. At this time, the receive frame is discarded in the E-DMAC (flush function). However, if padding is inserted in the receive frame by RPADIR, the flush function is performed when the frame length including the padding bytes is less than 32 bytes.

(b) Receive FIFO Overflow

In any of the following cases, the E-MAC cannot receive frames from the GMII/MII/RMII because it has no space to store receive frames, and all the receive frames that have been transferred to the E-MAC will be discarded in the E-MAC (receive FIFO overflow).

- Receive FIFO is full of data waiting for DMA transfer (the receive FIFO has no space).
- The number of receive frames waiting for DMA transfer is 24 in total (the receive frame information managing area has no empty space; up to 24 frames can be managed).

If an overflow occurs due to the former case, the RFE bit in EESR is set to 1 and an interrupt is generated to the CPU. If an overflow occurs due to the latter case, the RFCOF bit in EESR is set to 1 and an interrupt is generated to the CPU. Each time a receive frame is discarded due to an overflow, RMFCR is incremented. However, RMFCR is not incremented for a receive frame that is cut off due to insufficient receive FIFO space. If a receive frame is cut off due to insufficient receive FIFO space (the frame is partially stored in the receive FIFO), the E-DMAC performs the following operation:

- Performs DMA transfers for the cut-off frame stored in the receive FIFO to the receive buffer.
- After the DMA transfer, performs a write-back operation on the receive descriptor.
- After the write-back operation, sets the ROC bit in EESR to 1 and generates an interrupt to the CPU.

When the receive FIFO is full of data waiting for DMA transfer, frame reception from the GMII/MII/RMII can be resumed if DMA transfer is performed from the receive FIFO to the receive buffer and 32 bytes or more of empty space is generated in the receive FIFO. When the number of receive frames waiting for DMA transfer is 24 in total, frame reception from the GMII/MII/RMII can be resumed if one or more frame has been DMA transferred from the receive FIFO to the receive buffer. For restarting frame reception from the GMII/MII/RMII, when the E-DMAC resumes frame reception from the GMII/MII/RMII, it only accepts from the start of the frame.

(c) Flow Control

When the amount of receive data or the number of receive frames in the receive FIFO leads to one of the following conditions, the E-DMAC notifies the E-MAC to control E-MAC writing to the receive FIFO.

- When the space used in the receive FIFO exceeds the data amount specified by FCFTR
- When the number of receive frames in the receive FIFO exceeds the value specified by FCFTR

The threshold of the receive data amount can be set in a range from 256 to 65536 bytes in 256-byte units.

The threshold of receive frames can be set in a range from 1 to 24 frames (by the frame) in frame units.

(d) Receive Descriptor Empty

When the RACT bit of the read descriptor is 0 (invalid), the receive descriptor empty state is determined and DMA transfer is stopped. Then the following operation is performed.

- Writes the RR bit in EDRRR to 0
- Sets the RDE bit in EESR to 1 and generates an interrupt to the CPU.

To resume the DMA transfer to the receive buffer, the interrupt source needs to be cleared by software, the receive descriptor needs to be re-set and the RR bit in EDRRR should be set to 1.

Even if receive descriptor is empty, frame reception from the GMII/MII/RMII to the receive FIFO is continued if there is empty space left in the receive FIFO and receive frame information management area. Therefore, even if a receive descriptor empty state is determined, the DMA transfer can be performed without discarding the frames received from the GMII/MII/RMII if DMA transfer to the receive buffer can be resumed before an overflow occurs.

24.4.4 Relay

(1) Relay Procedure and Processing Flow

In the GETHER, when relay is enabled, frames input from the E-MAC are sent to both relay FIFO and receive FIFO in the TSU, and determined independently whether to receive or not by the receive system and whether to relay or not by the relay system. For multicast frames and frames whose destinations are other than this LSI, the CAM evaluation in frame relay processing can be referenced (for details on the CAM function, refer to section 24.4.5, CAM Function).

24.4.5 CAM Function

Frames input to the E-MAC are grouped into the following four types; unicast for this LSI, broadcast, multicast, and unicast to other destinations. The MAC addresses of unicast for this LSI and broadcast are fixed, and determined only by register settings. Consequently, only multicast and unicast to other destinations determine whether to receive or not by using the CAM (unicast frames whose destination MAC addresses match this LSI are called unicast frames to this LSI, and those that do not are called unicast frames to other destinations).

Furthermore, the evaluation of reception of unicast to other destinations and multicast frames by using CAM are performed by referencing the registered MAC addresses of the CAM entry table in the TSU. By using this function, receive FIFO overflow can be prevented caused by accumulation of frame data not required for reception, and CPU processing for determining reception can be reduced.

When the corresponding bit is set to 1, the POST table uses the CAM evaluation results for determining reception.

The on-chip CAM has entry tables which can register the MAC address of 32 entries, the details of which can be set by TSU_ADRH0 to TSU_ADRH31 and TSU_ADRL0 to TSU_ADRL31. The setting to enable/disable referencing of the on-chip CAM entry table is performed by the CAM entry table enable setting register which sets whether to perform CAM evaluation or not, and the CAM entry table POST register for setting whether to use the CAM determination results for determining reception. When on-chip CAM entry table referencing during reception is enabled, the destination address in the frame and MAC address registered in the CAM entry table are compared, and it is determined whether to transfer the frames input to the E-MAC to E-DMAC (have E-DMAC receive the frames) or discard the frames. When relaying and CAM entry table referencing during relay are both enabled, whether to transfer or discard multicast frames and frames for destinations other than this LSI can be determined by comparing the destination address in the frame and MAC address registered in the CAM entry table. Table 24.5 shows the processing method of frames (receive or discard) in reception from E-MAC to E-DMAC.

Table 24.5 Receive Frame Processing

CAM Entry Table Referencing Results	Types of Frame	Normal Mode		Promiscuous Mode	
		MCT = 0	MCT = 1	MCT = 0	MCT = 1
CAM hit (when addresses match)	Frame to this LSI	Discarded		Discarded	
	Broadcast frame	Discarded		Discarded	
	Multicast frame	Discarded	Received	Discarded	Received
	Frames having destinations other than this LSI	Received		Discarded	
CAM mishit (when addresses do not match)	Frames to this LSI	Received		Received	
	Broadcast frame	Received		Received	
	Multicast frame	Received	Discarded	Received	Discarded
	Frames having destinations other than this LSI	Discarded		Received	

[Legend]

MCT (Bit 13 in ECMR): Multicast receive mode (0: Receive when CAM mishit/1: Receive when CAM hit)

24.4.6 Transmit/Receive Processing of Multi-Buffer Frame (Single-Frame/Multi-Descriptor)

(1) Multi-Buffer Frame Transmit Processing

If an error occurs during multi-buffer frame transmission, the processing shown in figure 24.12 is carried out by the E-DMAC.

In the figure where the transmit descriptor is shown as inactive (TACT bit = 0), buffer data has already been transmitted successfully, and where the transmit descriptor is shown as active (TACT bit = 1), buffer data has not been transmitted. If a frame transmit error occurs in the first descriptor part where the transmit descriptor is active (TACT bit = 1), transmission is halted, and the TACT bit is cleared to 0, immediately. The next descriptor is then read, and the position within the transmit frame is determined on the basis of bits TFP1 and TFP0 (continuing [B′00] or end [B′01]). In the case of a continuing descriptor, the TACT bit is cleared to 0, and the next descriptor is read immediately. If the descriptor is the final descriptor, not only is the TACT bit cleared to 0, but write-back is also performed to the TFE and TFS bits at the same time. Data in the buffer is not transmitted between the occurrence of an error and write-back to the final descriptor. If error interrupts are enabled in EESIPR, an interrupt is generated immediately after the final descriptor write-back.

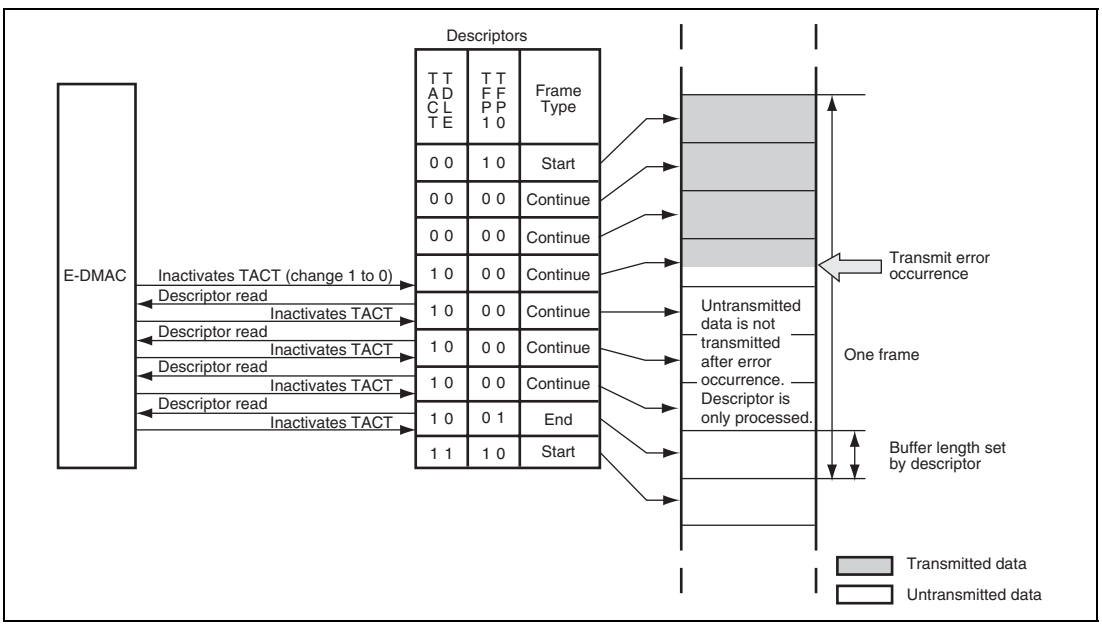


Figure 24.12 E-DMAC Operation after Transmit Error

(2) Receive Processing in the Case of Multi-Buffer Frame

If an error occurs during reception in the case of a multi-buffer frame where a receive frame is divided for storage in multiple buffers, the E-DMAC performs the processing shown in figure 24.13.

In the figure, the invalid receive descriptors (with the RACT bit cleared to 0) represent the successful reception of data to be stored in buffers, and the valid receive descriptors (with the RACT bit set to 1) represent unreceived buffers. If a frame receive error occurs with a descriptor shown in the figure, the status is written back to the corresponding descriptor.

If error interrupts are enabled in EESIPR, an interrupt is generated immediately after the write-back. If there is a new frame receive request, reception is continued from the buffer after that in which the error occurred.

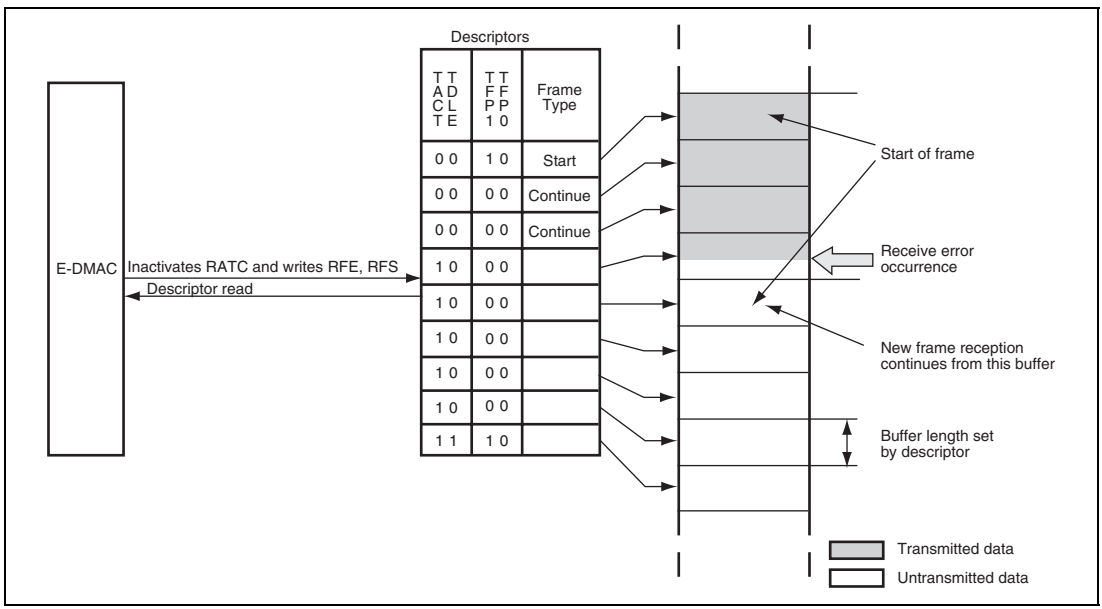


Figure 24.13 E-DMAC Operation after Receive Error

24.4.7 Padding Insertion in Receive Data

In the E-DMAC, one to three bytes of padding can be inserted in any byte position of receive data to improve software handling capability. By using this function, for instance, inserting 2-byte padding after the MAC header (14 bytes) of Ethernet frame enables data following the MAC header to set in 4-byte boundary.

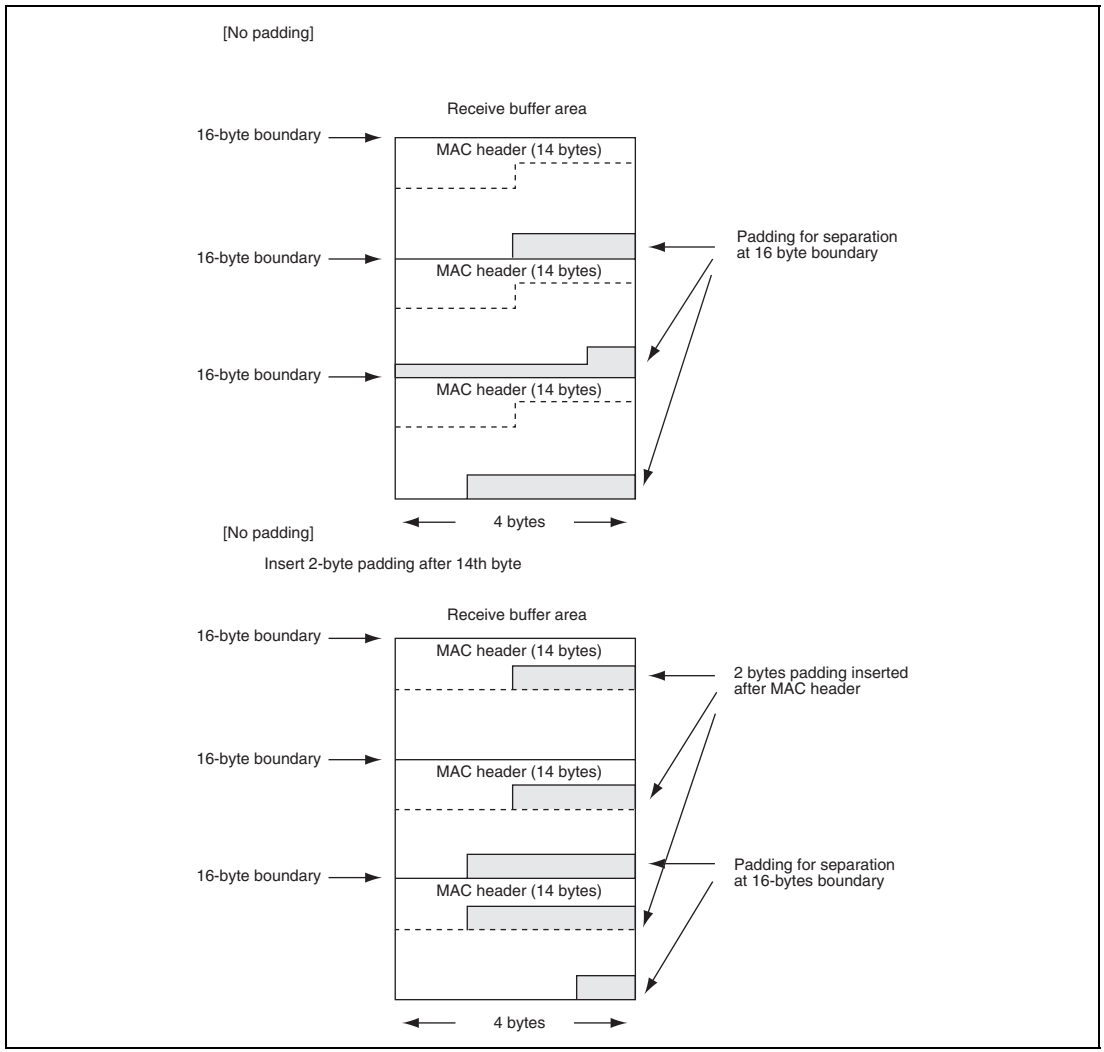


Figure 24.14 Padding Insertion in Receive Data

24.4.8 Interrupt Processing

(1) Interrupt Sources

The GETHER issues one type of interrupt to the CPU: receive/transmit interrupts (GEINT0).

GEINT0 interrupts are generated in correspondence with the transmit/receive operation. When an interrupt source is generated, it is set in EESR0 and an interrupt is issued to the CPU. For some interrupt sources, the EESR0 setting and an interrupt to the CPU are performed after a write-back operation to a descriptor is completed, not immediately after the interrupt source is detected. Interrupt sources other than the E-MAC status register source (ECI bit) are cleared by writing a 1 to the corresponding source bit. The E-MAC status register source (ECI bit) is cleared by writing a 1 to the corresponding source bit in ECSR. Interrupt source bits retain the values until they are cleared. GEINT0 interrupt source is allowed to issue interrupts by setting the corresponding bit in EESIPR0. Each E-MAC state register source (ECI bit) is allowed to issue an interrupt by setting the corresponding bit in ECSIPR. In the initial value, interrupts are disabled.

Table 24.6 shows these three interrupts, interrupt sources, interrupt status registers and bits set at interrupt occurrence and interrupt generation timing.

Table 24.6 List of GETHER Interrupts

Interrupt	Interrupt Source	Register and Bit	Interrupt Generation Timing
Transmit/ receive interrupt for port 0 (GEINT0)	Write-back completed	EESR0.TWB	After write-back
	Transmit underflow frame write-back completed	EESR0.TUC	After write-back
	Receive underflow frame write-back completed	EESR0.ROC	After write-back
	Transmission abort detection	EESR0.TABT	After write-back
	Reception abort detection	EESR0.RABT	After write-back
	Receive frame counter overflow	EESR0.RFCOF	When the interrupt source is detected
	E-MAC status register source	EESR0.ECI	When the interrupt source is detected
	Frame transmission completed	EESR0.TUC	After write-back
	Transmit descriptor empty	EESR0.TDE	When the interrupt source is detected
	Transmit FIFO underflow	EESR0.TFUF	When the interrupt source is detected
	Frame reception	EESR0.FR	After write-back
	Receive descriptor empty	EESR0.RDE	When the interrupt source is detected
	Receive FIFO overflow	EESR0.RFOF	When the interrupt source is detected
	Detect Loss of Carrier	EESR0.DLC	When the interrupt source is detected
	Delayed Collision Detect	EESR0.CD	When the interrupt source is detected
	Transmit Retry Over	EESR0.TRO	When the interrupt source is detected

Interrupt	Interrupt Source	Register and Bit	Interrupt Generated Timing
Transmit/ receive interrupt for port 0 (GEINT0)	Receive Multicast Address Frame	EESR0.RMAF	After write-back
	Carrier Extension Error	EESR0.CEEF	After write-back
	Carrier Extension Loss	EESR0.CELF	After write-back
	Receive Residual-Bit Frame	EESR0.RRF	After write-back
	Receive Too-Long Frame	EESR0.RTLF	After write-back
	Receive Too-Short Frame	EESR0.RTSF	After write-back
	PHY-LSI Receive Error	EESR0.PRE	After write-back
	CRC Error on Received Frame	EESR0.CERF	After write-back

24.4.9 Activation Procedure

The GETHER should be activated by the following procedure:

(1) Reset

1. Perform a power-on reset.
2. Start the E-DMAC transmitter and receiver (activation of descriptor engine).
   — Set ENT to 1 and ENR to 1 in EDSR.
3. Perform a software reset.
   — Set SWRR to 1 and SWRT to 1 in EDMR simultaneously.
4. Initialize the descriptor entry table.
5. Confirm cancellation of the software reset.
   — Check that the SWRR and SWRT bits in EDMR are cleared to 0.

(2) Pin and Operating Mode Settings

1. Pin setting
   — Refer to the section of the PFC (Pin Function Controller).
2. Operating mode setting
   — Set GECMR: Transfer speed, etc.
   — Set RMII_MII: Interface selection

(3) Registration of Descriptor Ring

The address of a descriptor ring configured in memory is registered in the descriptor entry table.

1. Transmit descriptor setting
   — Set TDLAR.
   — Set TDFAR.
   — Set TDFXR.
   — Set TDFFR. When the descriptor indicated by TDFXR is the last descriptor in the descriptor list, set H'00000001.

2. Receive descriptor setting
   — Set RDLAR.
   — Set RDFAR.
   — Set RDFXR.
   — Set RDFFR. When the descriptor indicated by RDFXR is the last descriptor in the descriptor list, set H'00000001.

(4) Register Settings

The following registers should be set as necessary.

1. E-DMAC related registers
   — Set EDMR: Operating mode, etc.
   — Set EESIPR: Interrupt masks
   — Set TRSCER: Error masks
   — Set TFTR: Transmit FIFO threshold
   — Set FDR: External FIFO size
   — Set RMCR: Reset method for reception activation
   — Set RPADIR: Padding insertion into receive data
   — Set FCFTR: Receive BSY output threshold
2. E-MAC related registers
   — Set ECMR setting: Transmission/reception specifications
   — Set ECSIPR setting: Interrupt masks
   — Set MAHR: MAC address
   — Set MALR: MAC address
   — Set RFLR: Maximum receive frame length
   — Set PIPR: ET0_PHY_INT pin polarity
   — Set APR: TIME parameter value of an automatic pause frame
   — Set MPR: TIME parameter value of a manual PAUSE frame
   — Set TPAUSER: Upper limit of automatic PAUSE frame retransmission
   — Set BCULR: Upper limit of burst cycles

(5) Activation

1. Start the E-DMAC transmission/reception function
   — Set the TR bits in EDTRR to 11.
   — Set the RR bit in EDRRR to 1.
2. Start the E-MAC transmission/reception function
   — Set the TE and RE bits in ECMR to 1.

24.4.10 Flow Control

The GETHER supports flow control functions conforming to IEEE802.3x for full-duplex operation. The flow control can be applied to both receive and transmit operations. When transmitting PAUSE frames, flow control can be performed by the following two procedures :

(1) Automatic PAUSE Frame Transmission

For receive frames, PAUSE frames are automatically transmitted when the number of data written to the receive FIFO reaches the value set in FCFTR. The TIME parameter included in the PAUSE frame is set by APR. The automatic PAUSE frame transmission is repeated until the number of data in the receive FIFO becomes less than the value set in FCFTR as the receive data is read from the FIFO. Using TPAUSER, the upper limit of retransmission counts of the PAUSE frames can also be set in the range from 1 to 65535. In this case, PAUSE frame transmission is repeated until the number of receive FIFO data becomes less than the FCFTR value, or the number of transmits reaches the value set by TPAUSER. The transmission counter is cleared to 0 when the next PAUSE frame is transmitted after the number of data in the receive FIFO becomes less than the FCFTR value.

The automatic PAUSE frame transmission is enabled when the TXF bit in ECMR is 1.

(2) Manual PAUSE Frame Transmission

PAUSE frames are transmitted by directives from the software. When writing the Timer value to MPR, manual PAUSE frame transmission is started. With this method, PAUSE frame transmission is carried out only once.

(3) PAUSE Frame Reception

The next frame is not transmitted until the time indicated by the Timer value elapses after receiving a PAUSE frame. However, the transmission of the current frame is continued. A received PAUSE frame is valid only when the RXF bit in ECMR is set to 1. The number of times of PAUSE frame receptions is counted.

(4) 0-Time PAUSE Frame Control

Flow control is performed using a PAUSE frame with the TIME parameter value set to 0. The PAUSE frame with the TIME parameter set to 0 can be enabled or disabled by the ZPF bit in ECMR.

- When PAUSE frame control with the TIME parameter value set to 0 is enabled

  A PAUSE frame with the TIME parameter value set to 0 is transmitted when the number of data in the receive FIFO is less than the FCFTR value before the time indicated by the TIME parameter value has not elapsed. When a PAUSE frame with the time indicated by the TIME parameter value set to 0 is received, the transmit standby state is canceled.
- When PAUSE frame control with the TIME parameter value set to 0 is disabled

  A PAUSE frame with the TIME parameter value set to 0 is not transmitted. When a PAUSE frame with the TIME parameter value set to 0 is received, the PAUSE frame is discarded.

24.4.11 Magic Packet Detection

The GETHER has a Magic Packet detection function. This function provides a Wake-On-LAN (WOL) facility that starts each peripheral device connected to a LAN from the host device or other source. This enables to construct a system in which a peripheral device receives a Magic Packet sent from the host device or other source, and starts itself. When the Magic Packet is detected, data is stored in the FIFO by the broadcast packet that has received data previously and the E-MAC is notified of the receiving status. To return to normal operation from the interrupt processing, the E-MAC, TSU and E-DMAC must be initialized by using ARST bit in ARSTR.

With a Magic Packet, reception is performed regardless of the destination address. As a result, this function is valid, and the ET0_WOL pin enabled, only in the case of a match with the destination address specified by the format in the Magic Packet. Further information on Magic Packets can be found in the technical documentation published by AMD Corporation.

The procedure for using the WOL function with this LSI is as follows.

1. Disable interrupt source output by means of the various interrupt enable/mask registers.
2. Set the MPDE bit in ECMR.
3. Set the MPDIP bit in ECSIPR to the enable setting.
4. Set the ECIIP bit in EESIPR.
5. If necessary, set the CPU operating mode to sleep mode.
6. When a Magic Packet is detected, an interrupt is sent to the CPU. The ET0_WOL pin notifies peripheral LSIs that the Magic Packet has been detected.

Note: For Magic Packet detection in deep standby mode, refer to section 9, Operating Modes and Power-Down Modes.

24.4.12 Intelligent Checksum Calculation Function

This function accelerates checksum calculation on received packets, and provides the following two modes.

- MAC/IP packet analyzing intelligent checksum calculation mode
- All-data intelligent checksum calculation mode with bytes to be skipped specified

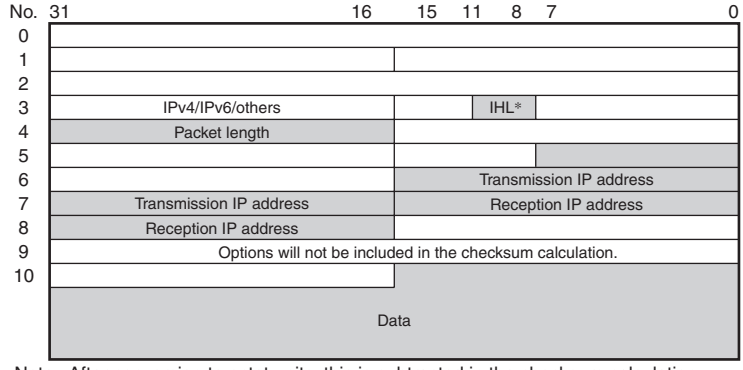
(1) MAC/IP Packet Analyzing Intelligent Checksum Calculation Mode (CSEBL = 1 and CSMD = 1 in CSMR)

In this mode, the checksum of received packets indicated in the table is calculated. However, if a MAC packet payload includes padding data in fields other than those for the IP packet itself because there is too little data for a full packet, it is not included in the checksum.

IPver	Items
IPv4	Option present
	Option not present
	Fragment*1
IPv6	Extension header not present
	Hop-by-hop options extension header length
	Routing extension header length
	Fragment extension header length*1
	Destination options header length
	AH extension header length
	ESP extension header length*2
	Extension header length for mobile IPv6*2

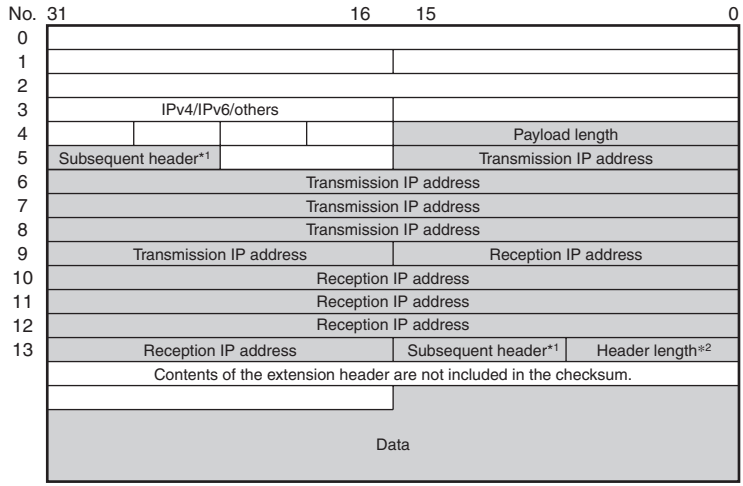
Notes: 1. This packet is to be checksumed, however, the RCS[15:0] bits and RCSE bit in RD0 are to be undefined even if the data is successfully received.
2. The RD0.RCSE bit is set to 1 without calculating the value of the RD0.RCS[15:0] bits.

The shaded regions of the following figure indicate the parts of an IPv4 packet which are used to obtain the checksum.



Note: After conversion to octet units, this is subtracted in the checksum calculation.
In the calculation: {8'h00, protocol no.[7:0]}

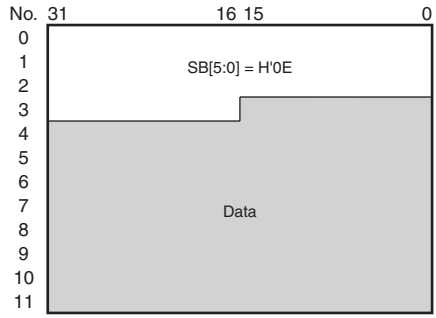
The shaded regions of the following figure indicate the parts of an IPv6 packet which are used to obtain the checksum.



Notes: *1 Only included in the checksum when the header is for the TCP or UDP. Calculation coverage is extended to {8'h00, protocol No.[7:0]} when the checksum is taken.
*2 After conversion to octet units, this is subtracted in the checksum calculation.

(2) All-Data Intelligent Checksum Calculation Mode with Bytes to be Skipped Specified (CSELB = 1 and CSMD = 0 in CSMR)

After having skipped the number of bytes specified in the SB[5:0] bits in CSMR, counting from the beginning of the packet, the checksum is calculated for all subsequent data (e.g. 14 bytes may be skipped).



24.5 Connection to PHY-LSI

24.5.1 MII Frame Transmission/Reception Timing

Each MII frame transmission/reception timing is shown in figures 24.15 to 24.20.

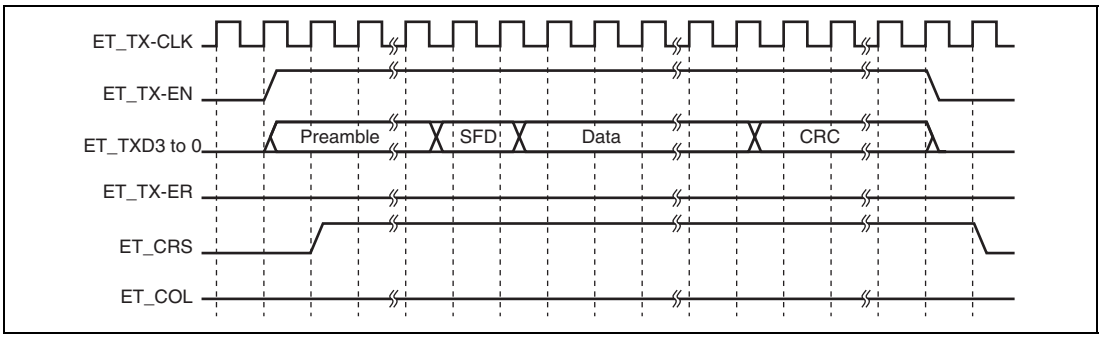


Figure 24.15 MII Frame Transmit Timing (Normal Transmission)

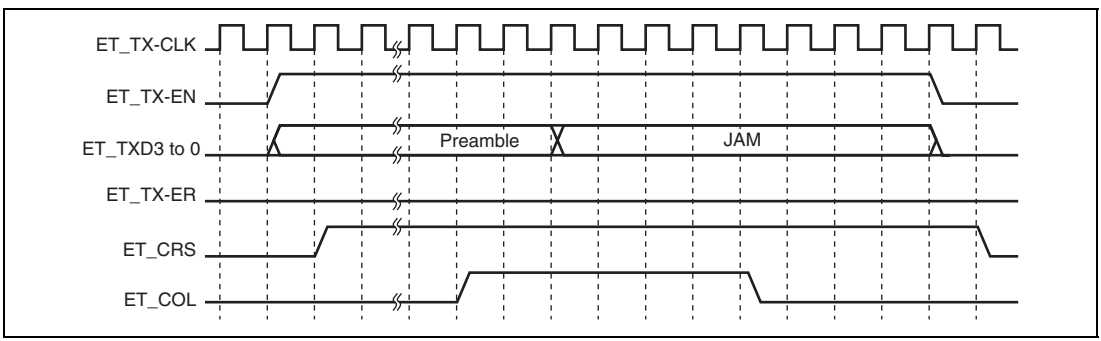


Figure 24.16 MII Frame Transmit Timing (Collision)

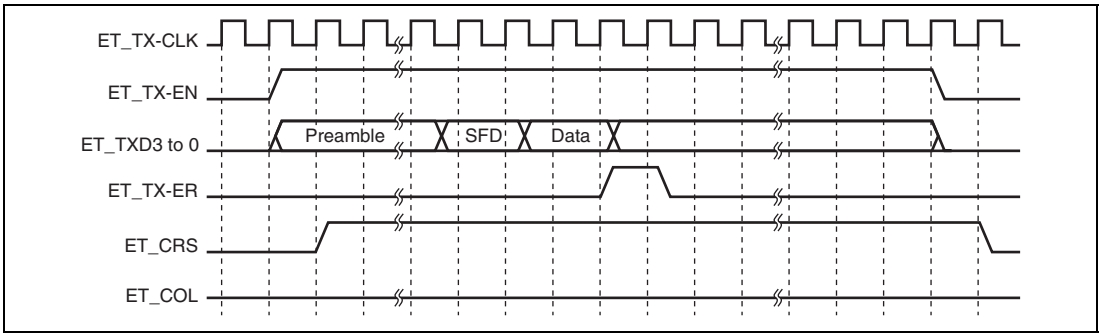


Figure 24.17 MII Frame Transmit Timing (Transmit Error)

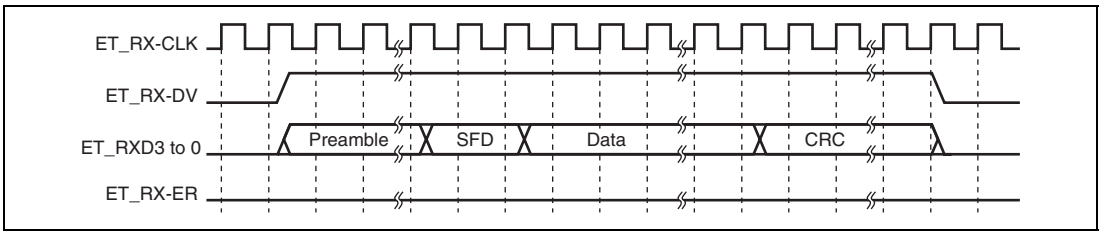


Figure 24.18 MII Frame Receive Timing (Normal Reception)

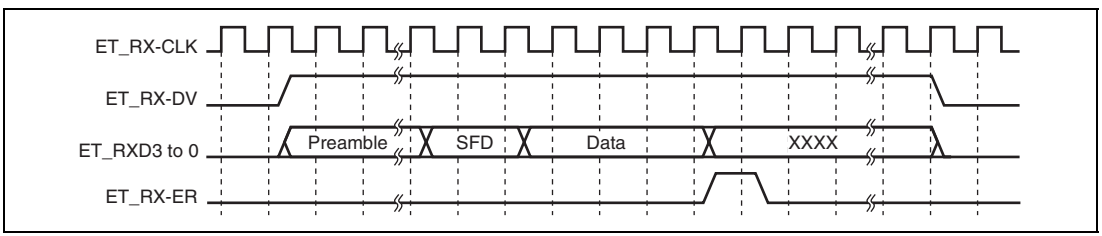


Figure 24.19 MII Frame Receive Timing (Reception Error (1))

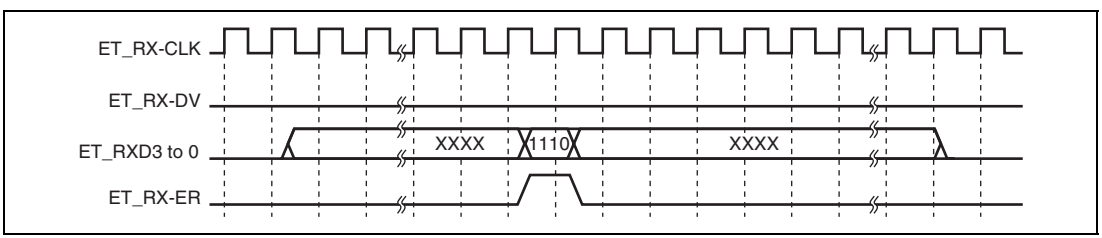


Figure 24.20 MII Fame Receive Timing (Reception Error (2))

24.5.2 GMII/MII Frame Reception Timing

Each GMII/MII frame reception timing is shown in figures 24.21 to 24.26.

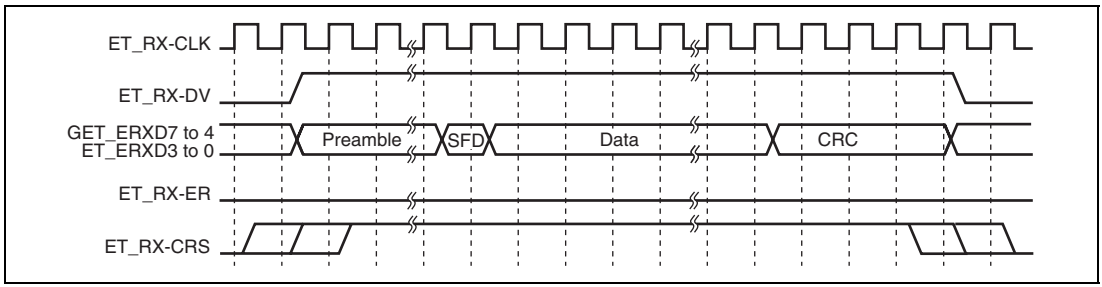


Figure 24.21 GMII/MII Fame Receive Timing (Normal Reception)

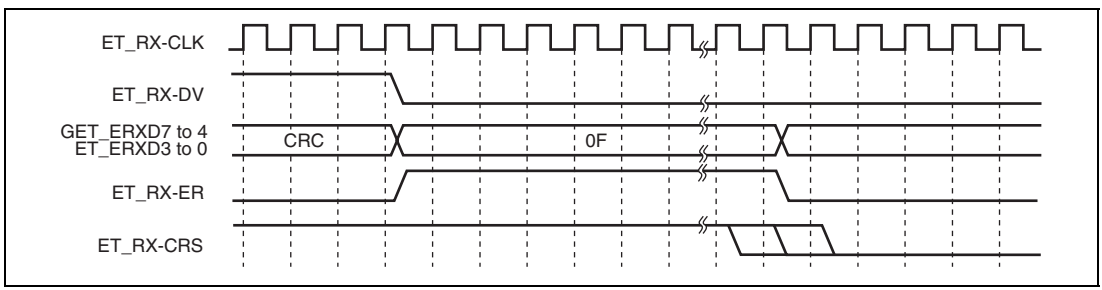


Figure 24.22 GMII/MII Fame Receive Timing (with Carrier Extension)

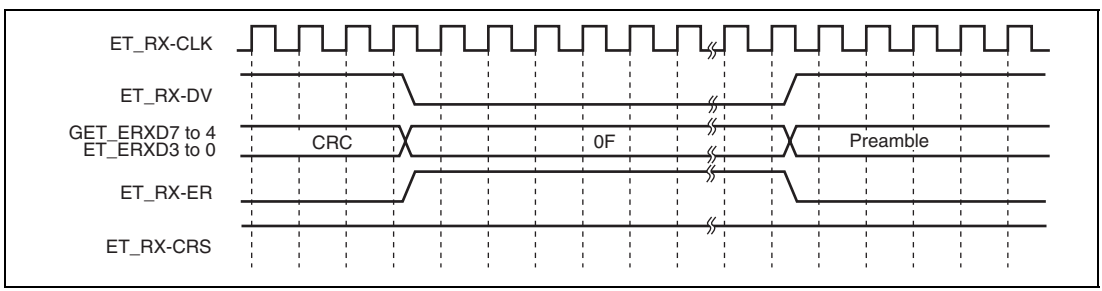


Figure 24.23 GMII/MII Fame Receive Timing (Burst Reception)

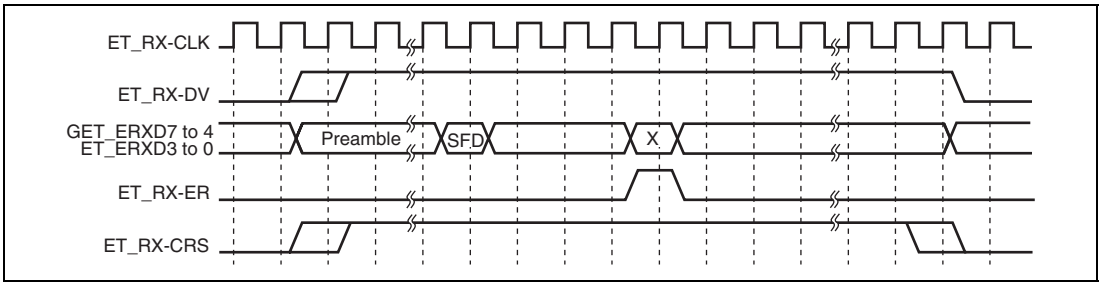


Figure 24.24 GMII/MII Fame Receive Timing (Reception Error)

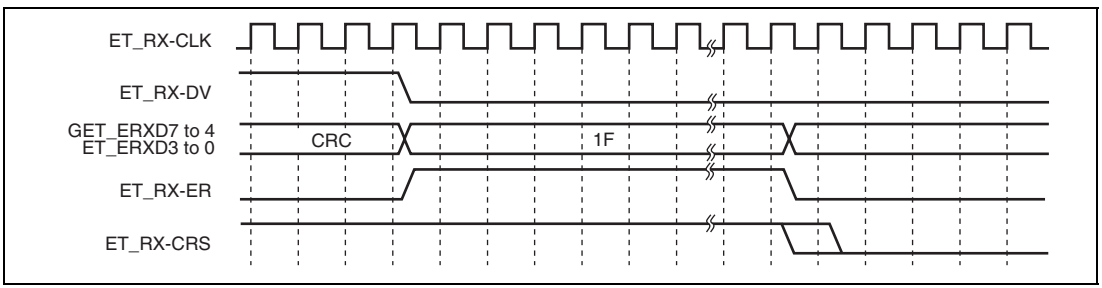


Figure 24.25 GMII/MII Fame Receive Timing (Error with Carrier Extension)

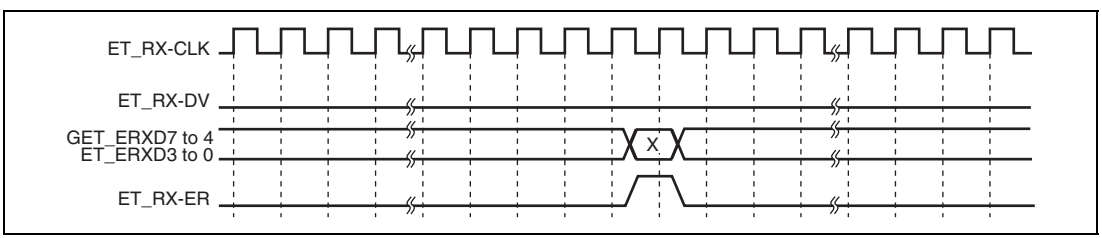


Figure 24.26 GMII/MII Fame Receive Timing (False Carrier Indication)

24.5.3 RMII Frame Transmission/Reception Timing

Each RMII frame transmission/reception timing is shown in figures 24.27 to 24.29.

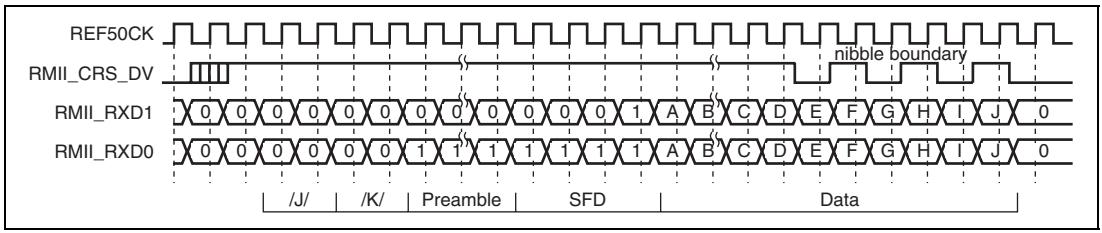


Figure 24.27 RMII Fame Receive Timing (Normal 100-Mbps Reception)

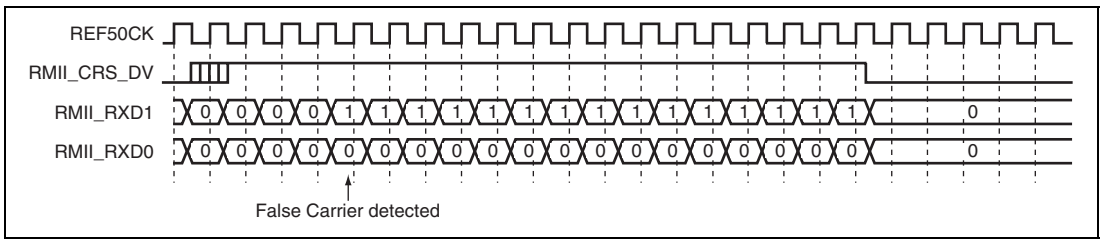


Figure 24.28 RMII Fame Receive Timing (100-Mbps Reception with Illegal Carrier Detected)

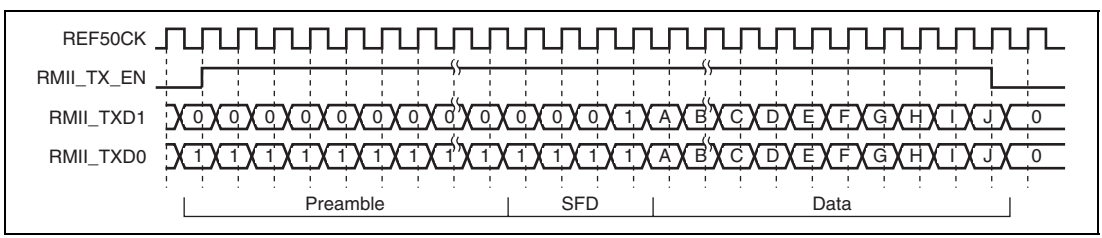


Figure 24.29 RMII Fame Transmit Timing (Normal 100-Mbps Transmission)

24.5.4 Accessing MII Registers

MII registers in the PHY-LSI are accessed via PIR in this LSI. PIR is used as a serial interface conforming to the MII frame format specified in IEEE802.3u.

(1) MII Management Frame Format

Figure 24.30 shows the format of an MII management frame. To access an MII register, a management frame is implemented by the program in accordance with the procedures shown in MII Register Access Procedure.

Access Type	MII Management Frame							
Item	PRE	ST	OP	PHYAD	REGAD	TA	DATA	IDLE
Number of bits	32	2	2	5	5	2	16	
Read	1..1	01	10	00001	RRRRR	Z0	D..D	
Write	1..1	01	01	00001	RRRRR	10	D..D	X

[Legend]

PRE: 32 consecutive 1s
ST: Write of 01 indicating start of frame
OP: Write of code indicating access type
PHYAD: Write of 0001 if the PHY-LSI address is 1 (sequential write starting with the MSB).
This bit changes depending on the PHY-LSI address.
REGAD: Write of 000q if the register address is 1 (sequential write starting with the MSB).
This bit changes depending on the PHY-LSI register address.
TA: Time for switching data transmission source on MII interface
(a) Write: 10 written
(b) Read: Bus release (notation: Z0) perdormed
DATA: 16-bit data. Sequential write or read from MSB
(a) Write: 16-bit data write
(b) Read: 16-bit data read
IDLE: Wait time until next MII management format input
(a) Write: Independent bus release (notation: X) performad
(d) Read: Bus already released in TA: control unnecessary

Figure 24.30 MII Management Frame Format

(2) MII Register Access Procedure

The program accesses MII registers via PIR. Access is implemented by a combination of 1-bit-unit data write, 1-bit-unit data read, bus release, and independent bus release. Figures 24.31 (1) to 24.31 (4) show the MII register access timing. The timing will differ depending on the PHY-LSI type.

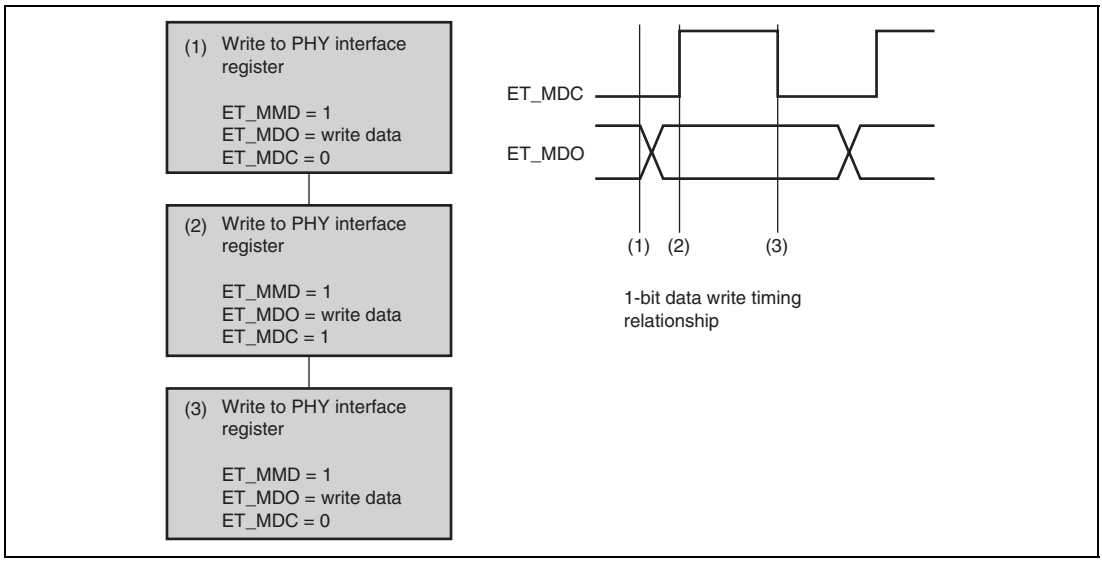


Figure 24.31 (1) 1-Bit Data Write Flowchart

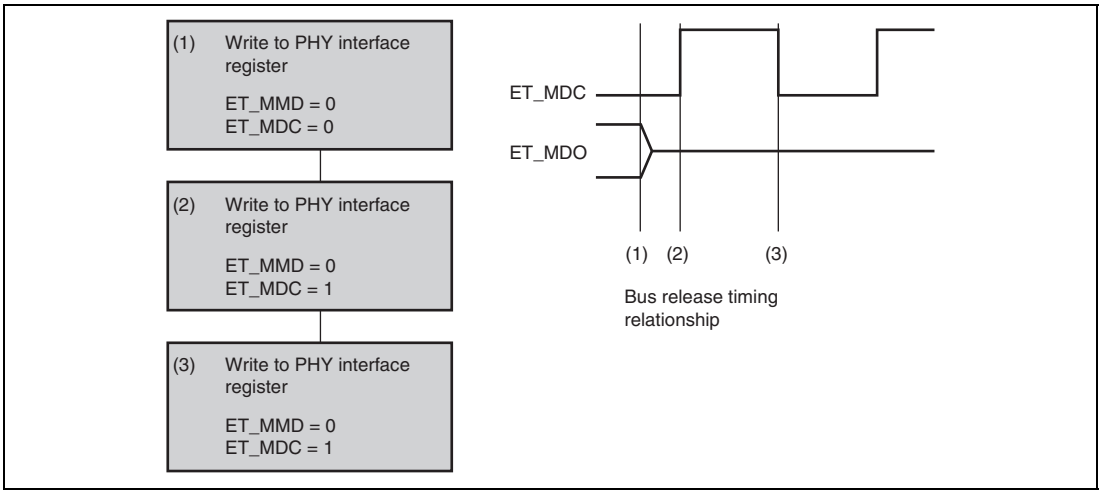


Figure 24.31 (2) Bus Release Flowchart (TA in Read in Figure 24.30)

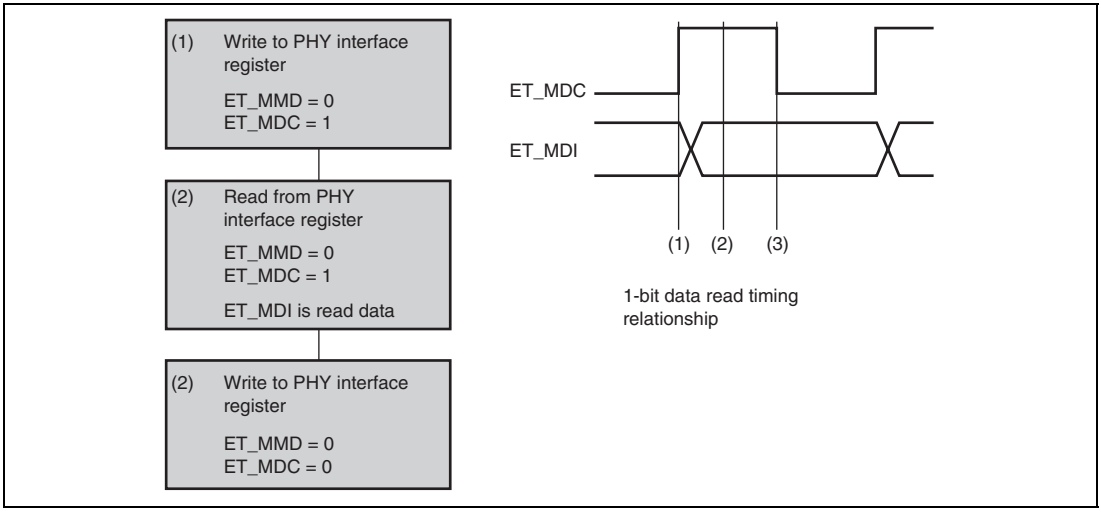


Figure 24.31 (3) 1-Bit Data Read Flowchart

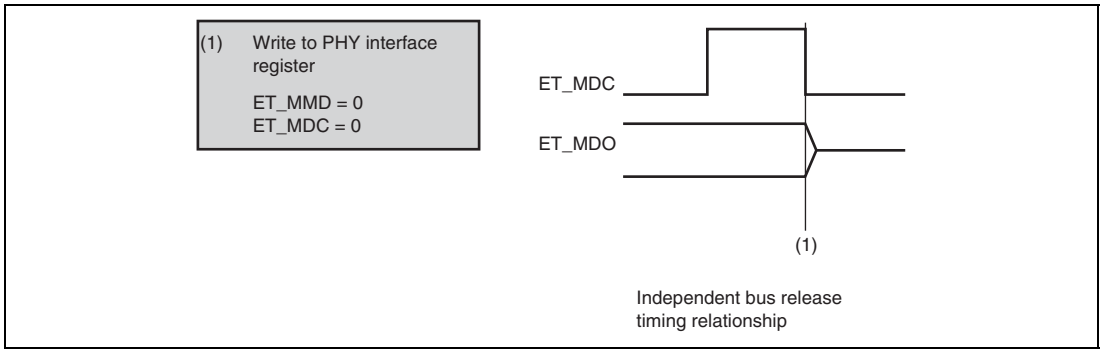


Figure 24.31 (4) Independent Bus Release Flowchart (IDLE in Write in Figure 24.30)

24.5.5 MII-RMII Interface Conversion

This LSI supports an RMII interface. The RMII signals are generated by converting the MII signals in the MII-RMII conversion circuit.

(1) Clock

REF50CK (50 MHz) from the RMII interface is divided and ET0_TX-CLK/ET0_RX-CLK (25 MHz or 2.5 MHz) is output.

(2) Reception

Waveforms received from the RMII interface are converted to MII waveforms and output (10 Mbps or 100 Mbps). Illegal carrier detection signal received from the RMII interface is converted to MII signal and output. RMII_RX-ER signal received from the RMII interface is converted to MII interface signal and output.

Note: Illegal carrier detection is not generated from preamble detection to reception completion (ET0_RX_DV negation).

(3) Transmission

Transmit waveforms from the MII interface is converted to the RMII interface waveforms and output (10 Mbps or 100 Mbps). The collision signal, ET0_COL, is generated by AND operation of the ET0_CRS and ET0_TX-EN signals.

(4) Full-Duplex/Half-Duplex Selection

In full-duplex transfer mode, the assertion of the COL is suppressed. Figure 24.32 shows a schematic of the conversion circuit.

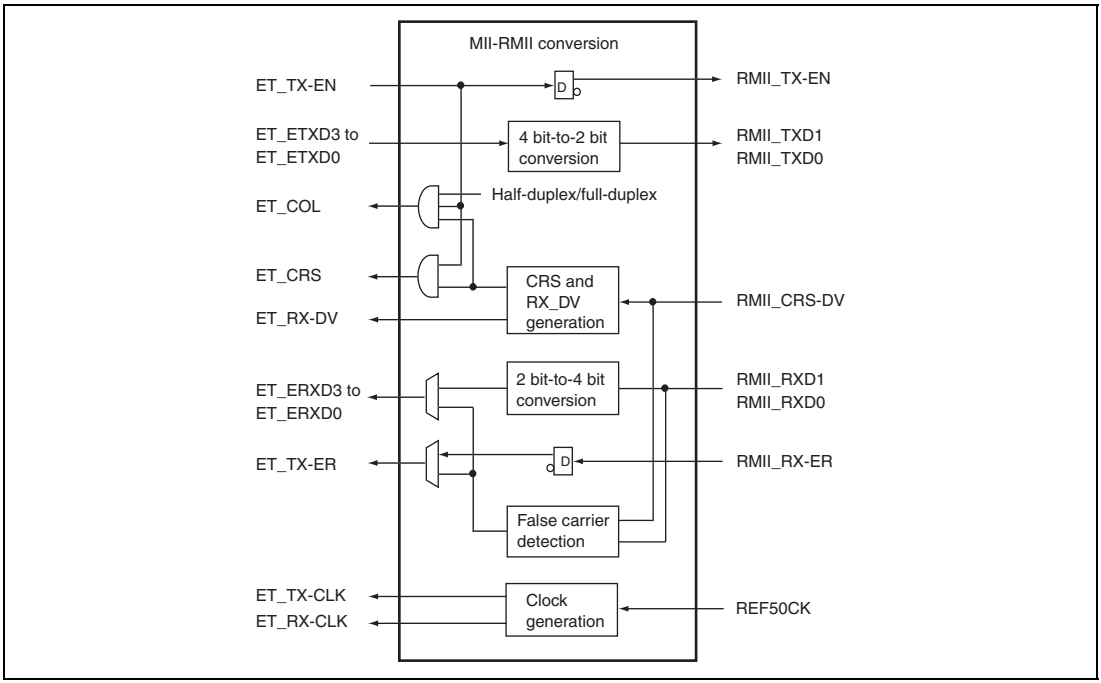


Figure 24.32 MII-RMII Conversion Circuit

24.6 Usage Notes

24.6.1 Checksum Calculation of Ethernet Frames

This LSI is capable of calculating the checksum data of the received frames. Only the data fields of the Ethernet frames are subject to checksum calculation. Specifically, a data field follows the length/type field and is followed by the CRC field. Figure 24.33 shows schematics indicating which parts of the Ethernet frames are calculated. Calculation involves 16-bit addition only; it does not involve bit inversion. Note that when the checksum data is valid, the CRC data (4 bytes) is not transferred as a receive frame, and the checksum data (sum data) is added automatically. Figure 24.34 shows schematics of Ethernet frames to which the checksum data has been added.

Note: Also for the frames with VLANtag inserted, the 15th byte from the top and the following bytes before the CRC field are subject to calculation.

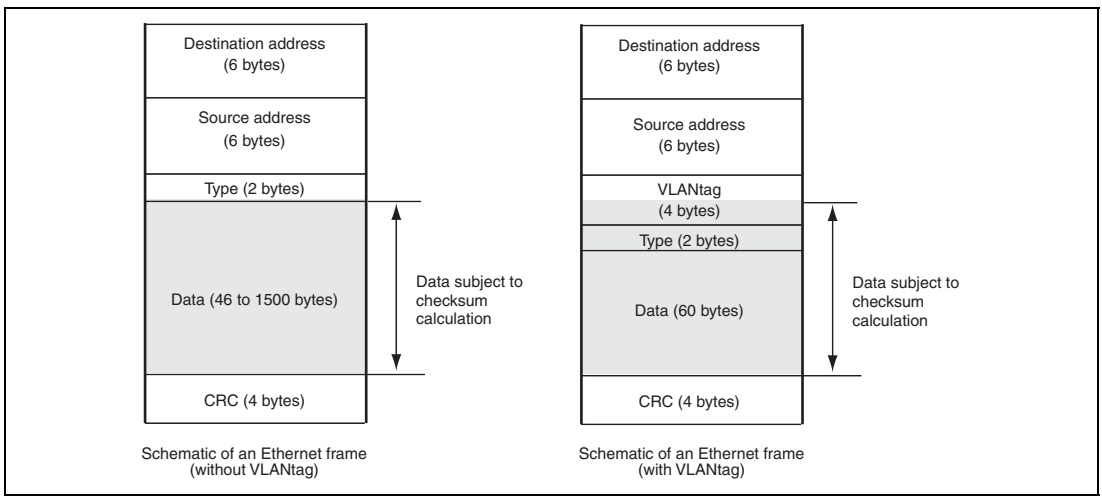


Figure 24.33 Data Subject to Checksum Calculation

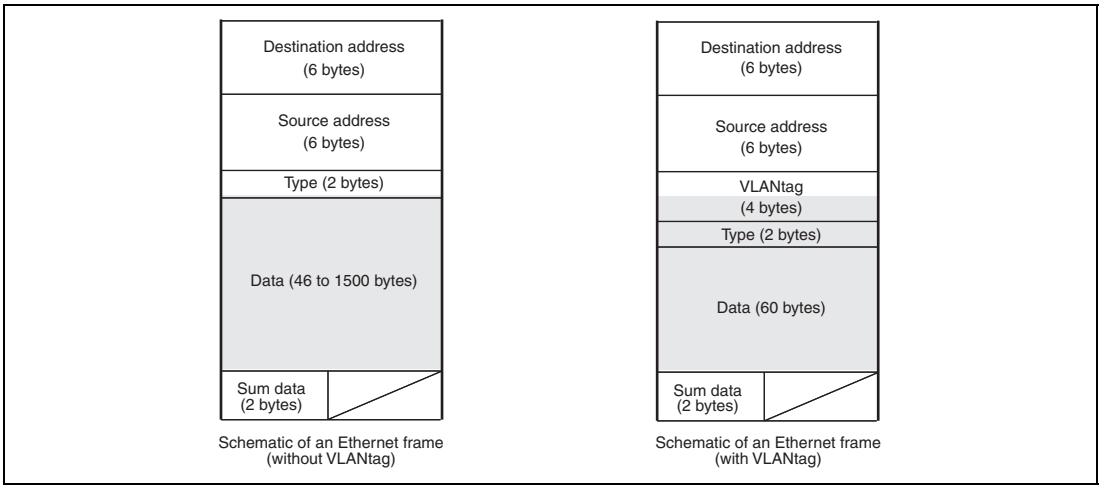


Figure 24.34 Data after Checksum Data Addition

24.6.2 Notes on TSU Use

The TSU of this LSI supports up to 100BASE-T data transfers. Therefore, even when The TSU of this LSI is used with 1000BASE-T, the transfer performance is equal to that with 100BASE-T.

24.6.3 Notes on Using the Intelligent Checksum Function

Checksum calculation using the intelligent checksum function is not affected by padding insertion specified by the receive data padding insert register (RPADIR). This is because checksum calculation is performed when transferring receive data from E-MAC to E-DMAC, while padding of receive data is performed when transferring receive data from E-DMAC to the receive buffer in memory.

24.6.4 Input Signal on the ET0_RX-ER Pin when the RMII interface is Selected

If an error signal received from PHY is not at the active level for more than one cycle of the RMII reference clock at 50 MHz when the RMII interface is selected, the signal is not detected as an error signal.

Section 25 Timer Unit (TMU)

This LSI includes a 32-bit timer unit (TMU) with nine channels (channels 0 to 8).

25.1 Features

TMU has the following features:

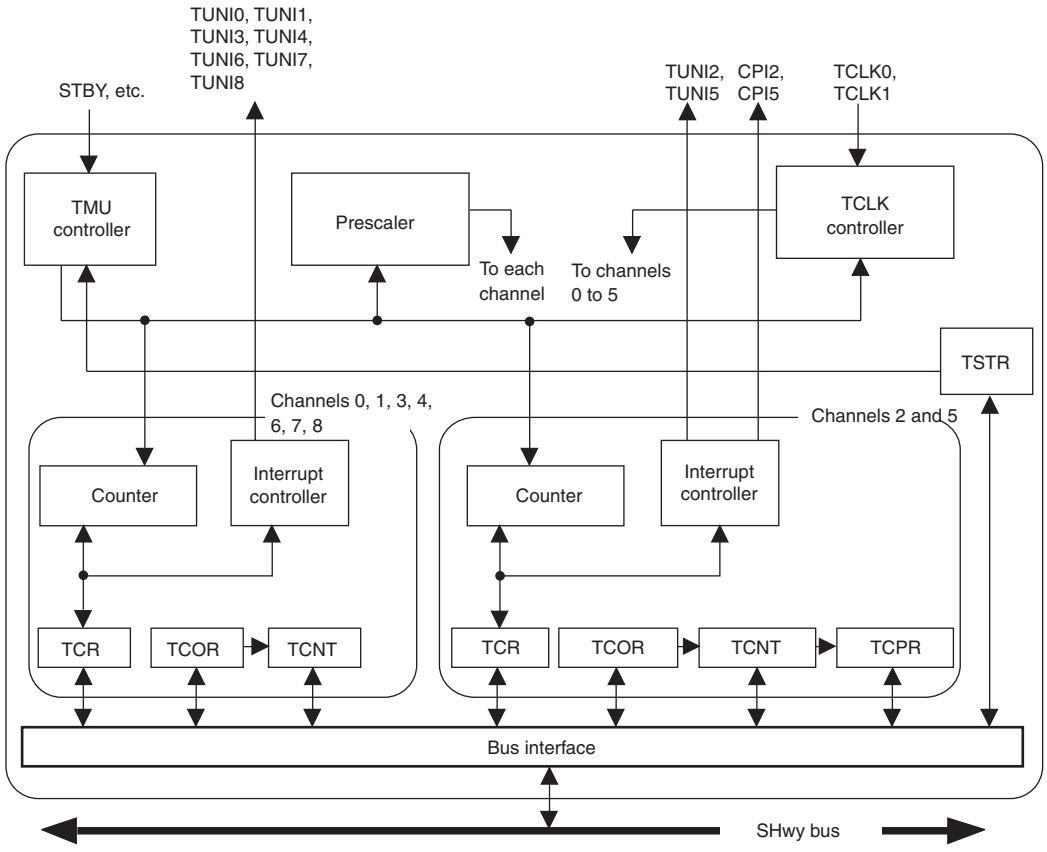
- Auto-reload type 32-bit down counter is provided for each channel
- Input capture function provided: Channels 2 and 5
- Selection of rising edge or falling edge as external clock input edge when external clock is selected or input capture function is used: Channels 0 to 5
- 32-bit timer constant register for auto-reload use, readable/writable at any time, and 32-bit down-counter are provided for each channel
- Selection of six counter input clocks: Channels 0 to 5

  External clock (TCLK) and five peripheral clocks (clkp/4, clkp/16, clkp/64, clkp/256, and clkp/1024) ("clkp" refers to the peripheral clock)
- Selection of five counter input clocks: Channels 6 to 8

  Five peripheral clocks (clkp/4, clkp/16, clkp/64, clkp/256, and clkp/1024) ("clkp" refers to the peripheral clock)
- Two interrupt sources

  One underflow source (each channel) and one input capture source (channels 2 and 5)

Figure 25.1 shows a block diagram of TMU.



[Legend]
TSTR: Timer start register
TCOR: Timer constant register
TCNT: Timer counter
TCR: Timer control register
TCPR: Input capture register 2 (channels 2 and 5 only)

Note: * Internal signals with 1/4, 1/16, or 1/64 of the clkp frequency supplied to the on-chip peripheral module.

Figure 25.1 Block Diagram of TMU

25.2 Input/Output Pins

Table 25.1 shows the pin configuration of the TMU.

Table 25.1 Pin Configuration

Name	Abbreviation	I/O	Function
Clock input 0	TCLK0	Input	External clock input pin for channels 0 to 2/ input capture control input pin for channel 2
Clock input 1	TCLK1	Input	External clock input pin for channels 3 to 5/ input capture control input pin for channel 5

25.3 Register Descriptions

The TMU has the following registers. Do not write to any addresses other than listed below. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than listed below are undefined.

Table 25.2 (1) Register Configuration

Channel	Name	Abbreviation	R/W	P4 Address	Size
Common to 0 to 2	Timer start register 0	TSTR0	R/W	H'FFD80004	8
0	Timer constant register 0	TCOR0	R/W	H'FFD80008	32
	Timer counter 0	TCNT0	R/W	H'FFD8000C	32
	Timer control register 0	TCR0	R/W	H'FFD80010	16
1	Timer constant register 1	TCOR1	R/W	H'FFD80014	32
	Timer counter 1	TCNT1	R/W	H'FFD80018	32
	Timer control register 1	TCR1	R/W	H'FFD8001C	16
2	Timer constant register 2	TCOR2	R/W	H'FFD80020	32
	Timer counter 2	TCNT2	R/W	H'FFD80024	32
	Timer control register 2	TCR2	R/W	H'FFD80028	16
	Input capture register 2	TCPR2	R	H'FFD8002C	32

Channel	Name	Abbreviation	R/W	P4 Address	Size
Common to 3 to 5	Timer start register 1	TSTR1	R/W	H'FFD81004	8
3	Timer constant register 3	TCOR3	R/W	H'FFD81008	32
	Timer counter 3	TCNT3	R/W	H'FFD8100C	32
	Timer control register 3	TCR3	R/W	H'FFD81010	16
4	Timer constant register 4	TCOR4	R/W	H'FFD81014	32
	Timer counter 4	TCNT4	R/W	H'FFD81018	32
	Timer control register 4	TCR4	R/W	H'FFD8101C	16
5	Timer constant register 5	TCOR5	R/W	H'FFD81020	32
	Timer counter 5	TCNT5	R/W	H'FFD81024	32
	Timer control register 5	TCR5	R/W	H'FFD81028	16
	Input capture register 5	TCPR5	R	H'FFD8102C	32
Common to 6 to 8	Timer start register 2	TSTR2	R/W	H'FFD82004	8
6	Timer constant register 6	TCOR6	R/W	H'FFD82008	32
	Timer counter 6	TCNT6	R/W	H'FFD8200C	32
	Timer control register 6	TCR6	R/W	H'FFD82010	16
7	Timer constant register 7	TCOR7	R/W	H'FFD82014	32
	Timer counter 7	TCNT7	R/W	H'FFD82018	32
	Timer control register 7	TCR7	R/W	H'FFD8201C	16
8	Timer constant register 8	TCOR8	R/W	H'FFD82020	32
	Timer counter 8	TCNT8	R/W	H'FFD82024	32
	Timer control register 8	TCR8	R/W	H'FFD82028	16

Table 25.2 (2) Register State in Each Operating Mode

Abbreviation	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
TSTR0	Initialized	Initialized	Retained	Retained	Retained	Initialized
TCOR0	Initialized	Initialized	Retained	Retained	Retained	Initialized
TCNT0	Initialized	Initialized	Retained	Retained	Retained	Initialized
TCR0	Initialized	Initialized	Retained	Retained	Retained	Initialized
TCOR1	Initialized	Initialized	Retained	Retained	Retained	Initialized
TCNT1	Initialized	Initialized	Retained	Retained	Retained	Initialized
TCR1	Initialized	Initialized	Retained	Retained	Retained	Initialized
TCOR2	Initialized	Initialized	Retained	Retained	Retained	Initialized
TCNT2	Initialized	Initialized	Retained	Retained	Retained	Initialized
TCR2	Initialized	Initialized	Retained	Retained	Retained	Initialized
TCPR2	Initialized	Initialized	Retained	Retained	Retained	Initialized
TSTR1	Initialized	Initialized	Retained	Retained	Retained	Initialized
TCOR3	Initialized	Initialized	Retained	Retained	Retained	Initialized
TCNT3	Initialized	Initialized	Retained	Retained	Retained	Initialized
TCR3	Initialized	Initialized	Retained	Retained	Retained	Initialized
TCOR4	Initialized	Initialized	Retained	Retained	Retained	Initialized
TCNT4	Initialized	Initialized	Retained	Retained	Retained	Initialized
TCR4	Initialized	Initialized	Retained	Retained	Retained	Initialized
TCOR5	Initialized	Initialized	Retained	Retained	Retained	Initialized
TCNT5	Initialized	Initialized	Retained	Retained	Retained	Initialized
TCR5	Initialized	Initialized	Retained	Retained	Retained	Initialized
TCPR5	Initialized	Initialized	Retained	Retained	Retained	Initialized
TSTR2	Initialized	Initialized	Retained	Retained	Retained	Initialized
TCOR6	Initialized	Initialized	Retained	Retained	Retained	Initialized
TCNT6	Initialized	Initialized	Retained	Retained	Retained	Initialized
TCR6	Initialized	Initialized	Retained	Retained	Retained	Initialized
TCOR7	Initialized	Initialized	Retained	Retained	Retained	Initialized
TCNT7	Initialized	Initialized	Retained	Retained	Retained	Initialized
TCR7	Initialized	Initialized	Retained	Retained	Retained	Initialized
TCOR8	Initialized	Initialized	Retained	Retained	Retained	Initialized

Abbreviation	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
TCNT8	Initialized	Initialized	Retained	Retained	Retained	Initialized
TCR8	Initialized	Initialized	Retained	Retained	Retained	Initialized

25.3.1 Timer Start Registers (TSTRn) (n = 0 to 2)

TSTR are 8-bit readable/writable registers that select whether to run or halt the TCNT.

- (TSTR0)

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	STR2	STR1	STR0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	STR2	0	R/W	Counter Start 2 Selects whether to run or halt TCNT2. 0: TCNT2 count halted 1: TCNT2 counts
1	STR1	0	R/W	Counter Start 1 Selects whether to run or halt TCNT1. 0: TCNT1 count halted 1: TCNT1 counts
0	STR0	0	R/W	Counter Start 0 Selects whether to run or halt TCNT0. 0: TCNT0 count halted 1: TCNT0 counts

- (TSTR1)

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	STR5	STR4	STR3
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	STR5	0	R/W	Counter Start 5 Selects whether to run or halt TCNT5. 0: TCNT5 count halted 1: TCNT5 counts
1	STR4	0	R/W	Counter Start 4 Selects whether to run or halt TCNT4. 0: TCNT4 count halted 1: TCNT4 counts
0	STR3	0	R/W	Counter Start 3 Selects whether to run or halt TCNT3. 0: TCNT3 count halted 1: TCNT3 counts

- (TSTR2)

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	STR8	STR7	STR6
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	STR8	0	R/W	Counter Start 8 Selects whether to run or halt TCNT8. 0: TCNT8 count halted 1: TCNT8 counts
1	STR7	0	R/W	Counter Start 7 Selects whether to run or halt TCNT7. 0: TCNT7 count halted 1: TCNT7 counts
0	STR6	0	R/W	Counter Start 6 Selects whether to run or halt TCNT6. 0: TCNT6 count halted 1: TCNT6 counts

25.3.2 Timer Constant Registers (TCORn) (n = 0 to 8)

TCOR are 32-bit readable/writable registers. After underflow has been generated according to the result of the TCNT countdown, the value of TCOR is set to TCNT and TCNT continues countdown from the value.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

25.3.3 Timer Counters (TCNTn) (n = 0 to 8)

TCNT are 32-bit readable/writable registers that count down upon the input clock selected using the bits TPSC2 to TPSC0 in TCR.

When a TCNT countdown results in an underflow, the UNF in TCR of corresponding channel is set. At the same time, the value of TCOR is set to TCNT and TCNT continues countdown from that value.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

25.3.4 Timer Control Registers (TCRn) (n = 0 to 8)

TCR are 16-bit readable/writable registers that select a count clock and edge when an external clock is selected, and control an interrupt generation when the flag that indicates the generation of a TCNT is set to 1. TCR of channel 2, 5 control the input capture function and generation of an interrupt during the input capture.

- TCR0, TCR1, TCR3, TCR4, TCR6, TCR7, TCR8

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	UNF	—	—	UNIE	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

- TCR2, TCR5

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	ICPF	UNF	ICPE1	ICPE0	UNIE	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	ICPF*[1]	0	R/W	Input Capture Interrupt Flag Status flag, provided in channels 2 and 5 only, which indicates the occurrence of input capture. 0: No input capture has occurred [Clearing condition] When 0 is written to ICPF 1: Input capture has occurred [Setting condition] When an input capture occurs*[2]

Bit	Bit Name	Initial Value	R/W	Description
8	UNF	0	R/W	Underflow Flag Status flag which indicates the occurrence of a TCNT underflow. 0: TCNT has not underflowed [Clearing condition] When 0 is written to UNF 1: TCNT has underflowed [Setting condition] When TCNT underflows*2
7	ICPE1*1	0	R/W	Input Capture Control
6	ICPE0*1	0	R/W	A function of channels 2 and 5 only: determines whether the input capture function can be used, and when used, whether or not to enable interrupts. Use the CKEG bits to designate use of either the rising or falling edge of the TCLK pin to set the values of TCNT2 and TCNT5 to TCPR2 and TCPR5, respectively. Only when the ICPF bits in TCR2 and TCR5 are 0, the values of TCNT2 and TCNT5 are set to TCPR2 and TCPR5. When the ICPF bit is set to 1, neither TCPR2 nor TCPR5 is set even when input capture is generated. 00: Input capture function is not used. 01: Reserved (setting prohibited) 10: Input capture function is used. Interrupt due to input capture (TICPI2 and TICPI5) is not enabled. 11: Input capture function is used. Interrupt due to input capture (TICPI2 and TICPI5) is enabled.

Bit	Bit Name	Initial Value	R/W	Description
5	UNIE	0	R/W	Underflow Interrupt Control Controls enabling of interrupt generation when the status flag (UNF) indicating TCNT underflow has been set to 1. 0: Interrupt due to underflow (TUNI) is not enabled 1: Interrupt due to underflow (TUNI) is enabled
4	CKEG1	0	R/W	Clock Edge 1, 0
3	CKEG0	0	R/W	Select an input edge of the external clock when the external clock is selected, or when the input capture function is used. 00: Count/capture register set on rising edge 01: Count/capture register set on falling edge 1X: Count/capture register set on both rising and falling edge
2	TPSC2	0	R/W	Timer Prescaler 2 to 0
1	TPSC1	0	R/W	Select the TCNT count clock.
0	TPSC0	0	R/W	000: Count on clkp/4 001: Count on clkp/16 010: Count on clkp/64 011: Count on clkp/256 100: Count on clkp/1024 101: Setting prohibited 110: Setting prohibited 111: Count on external clock (TCLK) (Not usable in channels 6, 7 and 8)

[Legend]

X: Don't care

Notes: 1. Reserved in channels 0, 1, 3, 4, 6, 7, 8 (Initial value is 0. Read only).

2. Writing 1 does not change the value.

25.3.5 Input Capture Registers 2, 5 (TCPR2, TCPR5)

TCPR 2 and 5 are read-only 32-bit registers used for the input capture function provided only in channels 2 and 5. The ICPE and CKEG bits in TCR2 and TCR5 control the input capture function. When an input capture occurs, the value of TCNT2 is copied into TCPR2, and the value of TCNT5 is copied into TCPR5. The values of TCNT2 and TCNT5 are set in TCPR2 and TCPR5, respectively, only when the ICPF bits in TCR2 and TCR5 are 0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

25.4 Operation

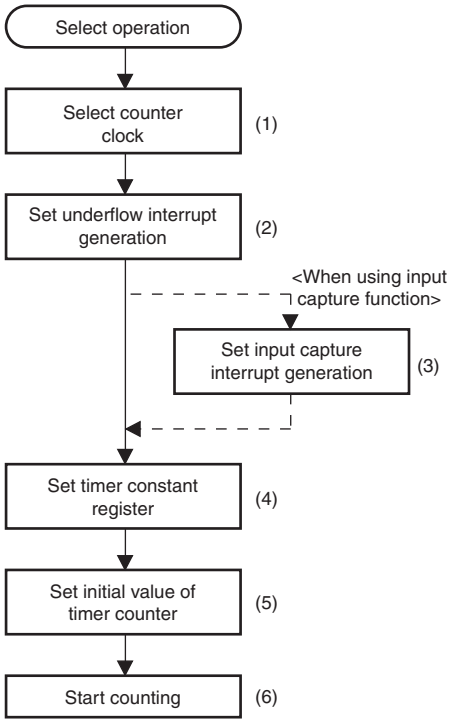
Each channel has a 32-bit timer counter (TCNT) and 32-bit timer constant register (TCOR). TCNT counts down. The auto-reload function enables synchronized counting and external-event counting. Channels 2 and 5 have an input capture function.

25.4.1 Counter Operation

When the bits STR8 to STR0 in TSTR0 to TSTR2 are set to 1, TCNT of corresponding channel starts counting. When TCNT underflows, the UNF flag of corresponding TCR is set. In this case, if the UNIE bit in TCR is set to 1, an interrupt request is sent to the CPU. Also, the value is copied from TCOR to TCNT and the down-count operation is continued (Auto reload function).

(1) Procedure for Setting Count Operation

Figure 25.2 shows an example of the procedure for setting the count operation.



(1) Select the counter clock using TPSC2 to TPSC0 in TCR.
To select the external clock (TCLK), select the edge of the external clock by setting bits CKEG1 and CKEG0 in TCR.
(2) Use the UNIE bit in TCR to set whether to generate an interrupt when TCNT underflows.
(3) When using the input capture function, set the ICPE bit in TCR, including the specification whether the interrupt function is to be used.
(4) Set a value in TCOR.
(5) Set the initial value in TCNT.
(6) To start counting, set the STR bit in TSTR to 1.

Note: When an interrupt has been generated, clear the source flag in the interrupt handler.
If the interrupt is enabled without clearing the flag, another interrupt will be generated.

Figure 25.2 Procedure for Setting Count Operation

(2) Auto-Reload Count Operation

Figure 25.3 shows the TCNT auto-reload operation.

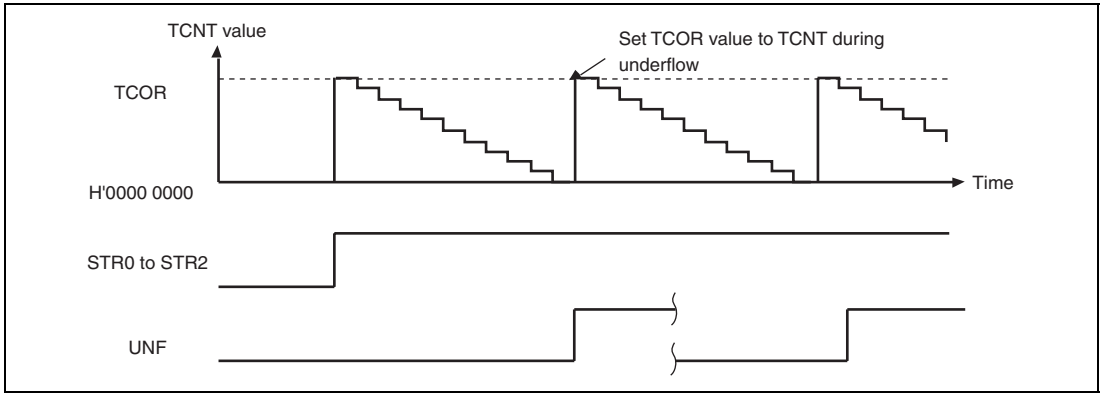


Figure 25.3 TCNT Auto-Reload Operation

(3) TCNT Count Timing

- Internal Clock Operation

  Five clocks (clkp/4, clkp/16, clkp/64, clkp/256, clkp/1024) that are created by dividing peripheral clocks are selected as count clocks by setting bits TPSC2 to TPSC0 in TCR. Figure 25.4 shows the timing.

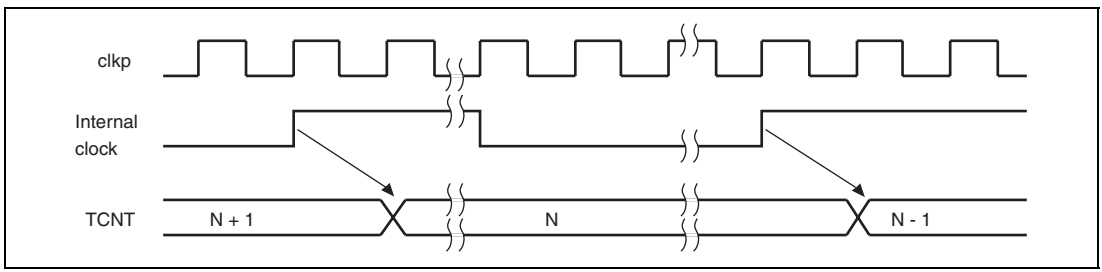


Figure 25.4 Count Timing when Internal Clock is Operating

- External Clock Operation

  Set the bits TPSC2 to TPSC0 in TCR to select the external clock pin (TCLK) as the timer clock. Use the bits CKEG1 and CKEG0 in TCR to select the detection edge. Rise, fall, or both can be selected.

  Figure 25.5 shows the timing for both-edge detection.

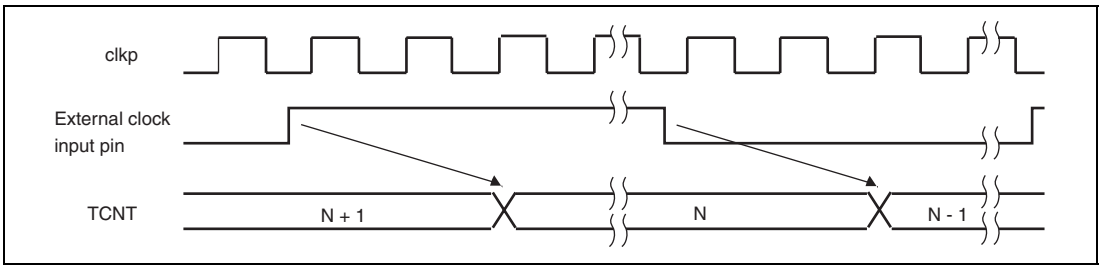


Figure 25.5 Count Timing when External Clock is Operating

25.4.2 Input Capture Function

Channels 2 and 5 have the input capture function. When using the input capture function, follow the procedure shown below.

1. Set the timer operating clock as an internal clock with bits TPSC2 to TPSC0 in TCR.
2. Set use of the input capture function and whether to generate an interrupt on using it with bits ICPE1 and ICPE0 in TCR.
3. Specify either rising edge or falling edge of the TCLK pin to be used to set the value of TCNT to TCPR2 and TCPR5 with bits CKEG1 and CKEG0 in TCR.

Only when an input capture is occurred and the ICPF bits in TCR2 and TCR5 are 0, the values of TCNT2 and TCNT5 are set in TCPR2 and TCPR5, respectively.

Figure25.6 shows the operating timing when the input capture function is used (the rising edge of TCLK is used).

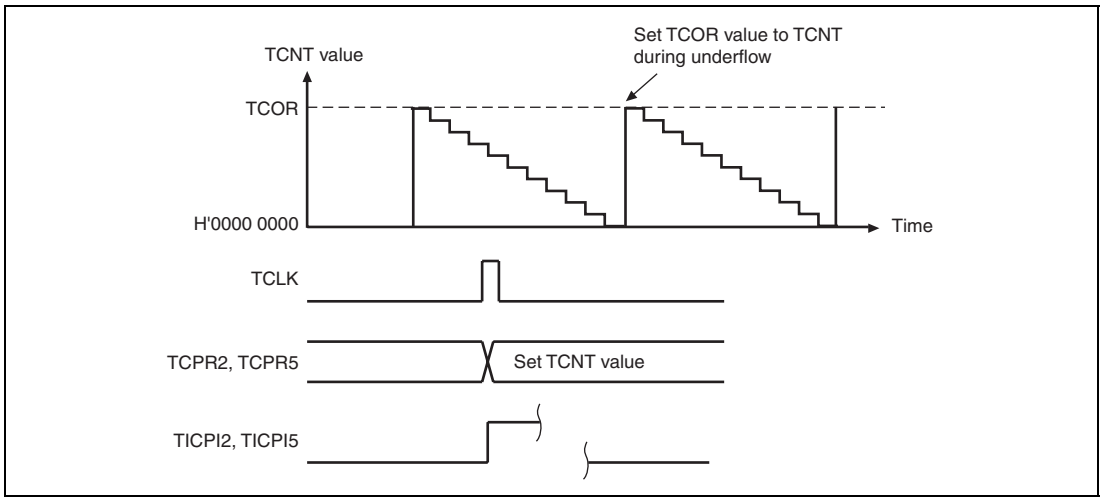


Figure 25.6 Operating Timing when Using Input Capture Function

25.5 Interrupt

The TMU interrupt sources are underflow interrupt or input capture interrupt when the input capture function is used. The underflow interrupt is generated at each channel. The input capture interrupt is generated at channels 2 and 5 only.

An underflow interrupt request is generated (for each channel) when both the UNF bit and the interrupt enable bit for that channel are set to 1.

When the input capture function is used and the input capture request is generated, an interrupt request is generated if the ICPF bit in TCR2 or TCR5 is 1 and the input capture control bits ICPE1 and ICPE0 in TCR2 and TCR5 are 11.

Table 25.3 shows the TMU interrupt sources.

Table 25.3 TMU Interrupt Sources

Channel	Interrupt Sources	Description
0	TUNI0	Underflow interrupt 0
1	TUNI1	Underflow interrupt 1
2	TUNI2	Underflow interrupt 2
	TICPI2	Input capture interrupt 2
3	TUNI3	Underflow interrupt 3
4	TUNI4	Underflow interrupt 4
5	TUNI5	Underflow interrupt 5
	TICPI5	Input capture interrupt 5
6	TUNI6	Underflow interrupt 6
7	TUNI7	Underflow interrupt 7
8	TUNI8	Underflow interrupt 8

25.6 Usage Notes

25.6.1 Writing to Registers

When writing to the TMU registers, clear the start bits (STR8 to STR0) of the corresponding TSTR channel to stop the timer counting.

Writing to TSTR and clearing bits UNF and ICPF in TCR can be executed during counting. To clear flags UNF and ICPF during counting, do not change the values of bits other than those to be cleared.

25.6.2 Reading TCNT Register

Reading from TCNT is performed synchronously with the timer count operation. When timer counting and register read processing are performed simultaneously, the value before TCNT counting down is read.

25.6.3 External Clock Frequency

The frequency of external clock (TCLK) input to each channel should be clkp/4 or less.

25.6.4 Access to Registers During Reset

Since doing so creates a risk of halting the system, do not execute register access during a reset.

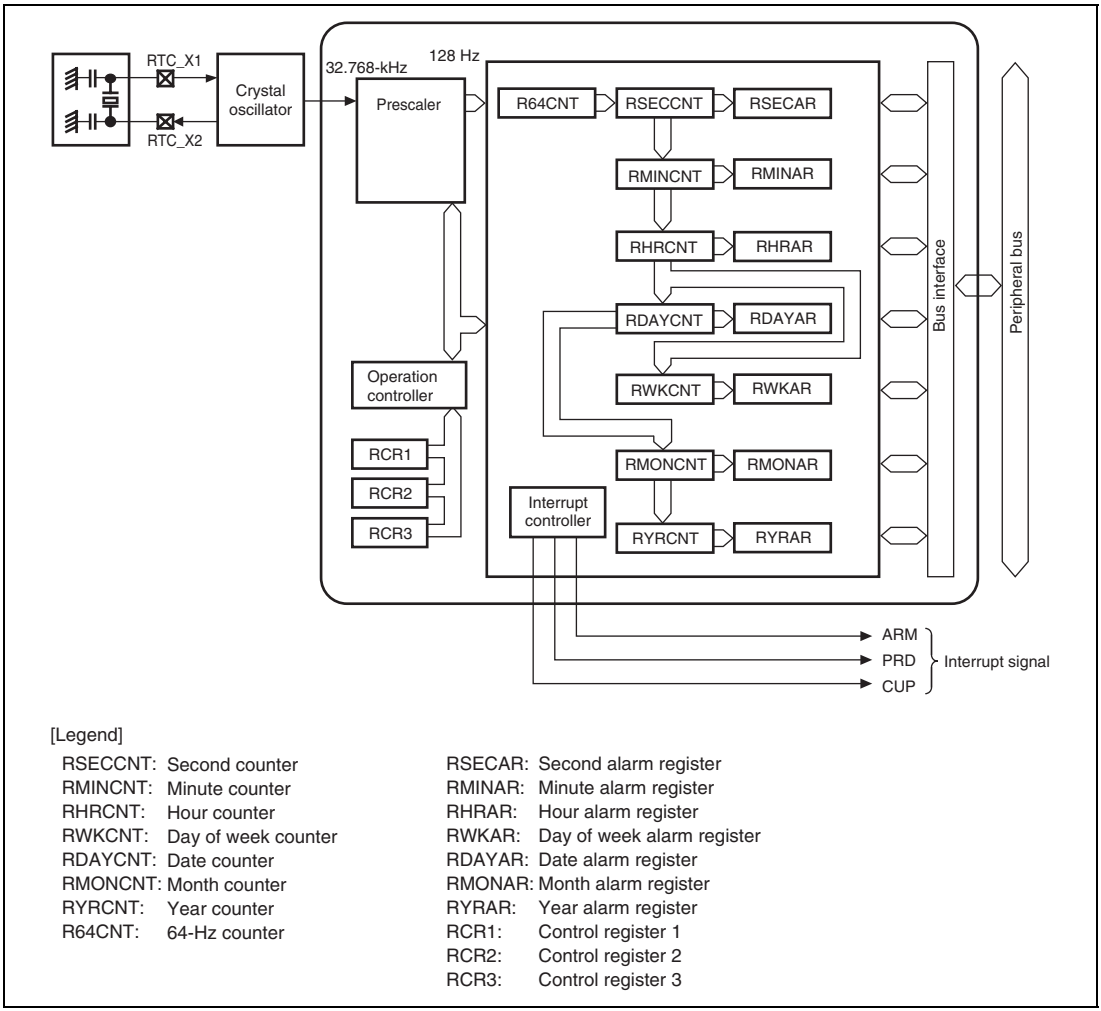
Section 26 Realtime Clock

This LSI has a realtime clock and a 32.768-kHz crystal oscillator.

26.1 Features

- Clock and calendar functions (BCD format): Seconds, minutes, hours, day of the week, date, month, and year.
- 1-Hz to 64-Hz timer (binary format)
  64-Hz counter indicates the state of the divider circuit between 64 Hz and 1 Hz
- Start/stop function
- 30-second adjust function
- Alarm interrupt: Frame comparison of seconds, minutes, hours, day of the week, date, month, and year can be used as conditions for the alarm interrupt
- Periodic interrupts: the interrupt cycle may be 1/64 second, 1/16 second, 1/4 second, 1/2 second, 1 second, or 2 seconds
- Carry interrupt: a carry interrupt indicates when a carry occurs during a counter read
- Automatic leap year adjustment
- Recovery from deep standby mode can be performed by an alarm interrupt.

Figure 26.1 shows the block diagram.



[Legend]

RSECCNT: Second counter
RMINCNT: Minute counter
RHRCNT: Hour counter
RWKCNT: Day of week counter
RDAYCNT: Date counter
RMONCNT: Month counter
RYRCNT: Year counter
R64CNT: 64-Hz counter

RSECAR: Second alarm register
RMINAR: Minute alarm register
RHRAR: Hour alarm register
RWKAR: Day of week alarm register
RDAYAR: Date alarm register
RMONAR: Month alarm register
RYRAR: Year alarm register
RCR1: Control register 1
RCR2: Control register 2
RCR3: Control register 3

Figure 26.1 Block Diagram

26.2 Input/Output Pin

Table 26.1 shows the pin configuration.

Table 26.1 Pin Configuration

Pin Name	Symbol	I/O	Description
Realtime clock resonator crystal pin/ external clock	RTC_X1	Input	Connects 32.768-kHz crystal resonator for this module, and enables to input the external clock to the RTC_X1 pin.
	RTC_X2	Output	

26.3 Register Descriptions

Table 26.2 (1) shows the register configuration.

Table 26.2 (1) Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
64-Hz counter	R64CNT	R	H'xx	H'FFFC5000	8
Second counter	RSECCNT	R/W	H'xx	H'FFFC5002	8
Minute counter	RMINCNT	R/W	H'xx	H'FFFC5004	8
Hour counter	RHRCNT	R/W	H'xx	H'FFFC5006	8
Day of week counter	RWKCNT	R/W	H'xx	H'FFFC5008	8
Date counter	RDAYCNT	R/W	H'xx	H'FFFC500A	8
Month counter	RMONCNT	R/W	H'xx	H'FFFC500C	8
Year counter	RYRCNT	R/W	H'xxxx	H'FFFC500E	16
Second alarm register	RSECAR	R/W	H'xx	H'FFFC5010	8
Minute alarm register	RMINAR	R/W	H'xx	H'FFFC5012	8
Hour alarm register	RHRAR	R/W	H'xx	H'FFFC5014	8
Day of week alarm register	RWKAR	R/W	H'xx	H'FFFC5016	8
Date alarm register	RDAYAR	R/W	H'xx	H'FFFC5018	8
Month alarm register	RMONAR	R/W	H'xx	H'FFFC501A	8
Year alarm register	RYRAR	R/W	H'xxxx	H'FFFC5020	16
Control register 1	RCR1	R/W	H'xx	H'FFFC501C	8
Control register 2	RCR2	R/W	H'09	H'FFFC501E	8
Control register 3	RCR3	R/W	H'x0	H'FFFC5024	8

Table 26.2 (2) Register State in Each Operating Mode

Abbreviation	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
R64CNT	Retained*[1]	Retained*[1]	Retained*[1]	Retained*[1]	Retained*[2]	Retained*[1]
RSECCNT						
RMINCNT						
RHRCNT						
RWKCNT						
RDAYCNT						
RMONCNT						
RYRCNT						
RSECAR	Retained	Retained	Retained	Retained	Retained	Retained
RMINAR						
RHRAR						
RWKAR						
RDAYAR						
RMONAR						
RYRAR						
RCR1	Initialized	Initialized	Retained	Retained	Retained	Initialized
RCR2	Initialized	Initialized*[3]	Retained	Retained	Retained	Initialized
RCR3	Retained	Retained	Retained	Retained	Retained	Retained

Notes: 1. Counting up is resumed.
2. To stop the module, set the RTCEN bit in the control register 2 (RCR2) to 0.
3. The RTCEN and START bits are retained.

26.3.1 64-Hz Counter (R64CNT)

R64CNT indicates the state of the divider circuit between 64 Hz and 1 Hz.

Reading this register, when carry from 128-Hz divider stage is generated, sets the CF bit in the control register 1 (RCR1) to 1 so that the carrying and reading 64 Hz counter are performed at the same time is indicated. In this case, the R64CNT should be read again after writing 0 to the CF bit in RCR1 since the read value is not valid.

After the RESET bit or ADJ bit in the control register 2 (RCR2) is set to 1, the divider circuit is initialized and R64CNT is initialized.

Bit:	7	6	5	4	3	2	1	0
	-	1 Hz	2 Hz	4 Hz	8 Hz	16 Hz	32 Hz	64 Hz
Initial value:	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	1 Hz	Undefined	R	Indicate the state of the divider circuit between 64 Hz and 1 Hz.
5	2 Hz	Undefined	R	
4	4 Hz	Undefined	R	
3	8 Hz	Undefined	R	
2	16 Hz	Undefined	R	
1	32 Hz	Undefined	R	
0	64 Hz	Undefined	R	

26.3.2 Second Counter (RSECCNT)

RSECCNT is used for setting/counting in the BCD-coded second section. The count operation is performed by a carry for each second of the 64-Hz counter.

The assignable range is from 00 through 59 (practically in BCD), otherwise operation errors occur. Carry out write processing after stopping the count operation through the setting of the START bit in RCR2.

Bit:	7	6	5	4	3	2	1	0
	-	10 seconds			1 second			
Initial value:	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6 to 4	10 seconds	Undefined	R/W	Counting Ten's Position of Seconds Counts on 0 to 5 for 60-seconds counting.
3 to 0	1 second	Undefined	R/W	Counting One's Position of Seconds Counts on 0 to 9 once per second. When a carry is generated, 1 is added to the ten's position.

26.3.3 Minute Counter (RMINCNT)

RMINCNT is used for setting/counting in the BCD-coded minute section. The count operation is performed by a carry for each minute of the second counter.

The assignable range is from 00 through 59 (practically in BCD), otherwise operation errors occur. Carry out write processing after stopping the count operation through the setting of the START bit in RCR2.

Bit:	7	6	5	4	3	2	1	0
	-	10 minutes			1 minute			
Initial value:	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	—	0	R	Reserved This bit is always read as 0.The write value should always be 0.
6 to 4	10 minutes	Undefined	R/W	Counting Ten's Position of Minutes Counts on 0 to 5 for 60-minutes counting.
3 to 0	1 minute	Undefined	R/W	Counting One's Position of Minutes Counts on 0 to 9 once per second. When a carry is generated, 1 is added to the ten's position.

26.3.4 Hour Counter (RHRCNT)

RHRCNT is used for setting/counting in the BCD-coded hour section. The count operation is performed by a carry for each 1 hour of the minute counter.

The assignable range is from 00 through 23 (practically in BCD), otherwise operation errors occur. Carry out write processing after stopping the count operation through the setting of the START bit in RCR2.

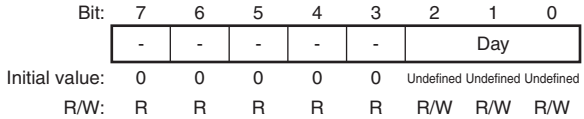
Bit:	7	6	5	4	3	2	1	0
	-	-	10 hours		1 hour			
Initial value:	0	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5, 4	10 hours	Undefined	R/W	Counting Ten's Position of Hours Counts on 0 to 2 for ten's position of hours.
3 to 0	1 hour	Undefined	R/W	Counting One's Position of Hours Counts on 0 to 9 once per hour. When a carry is generated, 1 is added to the ten's position.

26.3.5 Day of Week Counter (RWKCNT)

RWKCNT is used for setting/counting day of week section. The count operation is performed by a carry for each day of the date counter.

The assignable range is from 0 through 6 (practically in BCD), otherwise operation errors occur. Carry out write processing after stopping the count operation through the setting of the START bit in RCR2.



Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	Day		
Initial value:	0	0	0	0	0	Undefined	Undefined	Undefined
R/W:	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	Day	Undefined	R/W	Day-of-Week Counting Day-of-week is indicated with a binary code. 000: Sunday 001: Monday 010: Tuesday 011: Wednesday 100: Thursday 101: Friday 110: Saturday 111: Reserved (setting prohibited)

26.3.6 Date Counter (RDAYCNT)

RDAYCNT is used for setting/counting in the BCD-coded date section. The count operation is performed by a carry for each day of the hour counter.

The assignable range is from 01 through 31 (practically in BCD), otherwise operation errors occur. Carry out write processing after stopping the count operation through the setting of the START bit in RCR2.

The range of date changes with each month and in leap years. Confirm the correct setting. Leap years are recognized by dividing the year counter (RYRCNT) values by 400, 100, and 4 and obtaining a fractional result of 0.

Bit:	7	6	5	4	3	2	1	0
	-	-	10 days		1 day			
Initial value:	0	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5, 4	10 days	Undefined	R/W	Counting Ten's Position of Dates
3 to 0	1 day	Undefined	R/W	Counting One's Position of Dates Counts on 0 to 9 once per date. When a carry is generated, 1 is added to the ten's position.

26.3.7 Month Counter (RMONCNT)

RMONCNT is used for setting/counting in the BCD-coded month section. The count operation is performed by a carry for each month of the date counter.

The assignable range is from 01 through 12 (practically in BCD), otherwise operation errors occur. Carry out write processing after stopping the count operation through the setting of the START bit in RCR2.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	10 months	1 month			
Initial value:	0	0	0	Undefined	Undefined	Undefined	Undefined	Undefined
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	10 months	Undefined	R/W	Counting Ten's Position of Months
3 to 0	1 month	Undefined	R/W	Counting One's Position of Months Counts on 0 to 9 once per month. When a carry is generated, 1 is added to the ten's position.

26.3.8 Year Counter (RYRCNT)

RYRCNT is used for setting/counting in the BCD-coded year section. The count operation is performed by a carry for each year of the month counter.

The assignable range is from 0000 through 9999 (practically in BCD), otherwise operation errors occur. Carry out write processing after stopping the count operation through the setting of the START bit in RCR2.

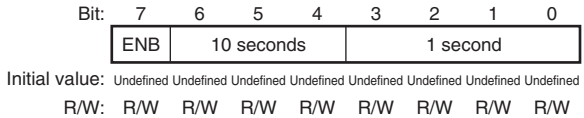
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	1000 years				100 years				10 years				1 year			
Initial value:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	1000 years	Undefined	R/W	Counting Thousand's Position of Years
11 to 8	100 years	Undefined	R/W	Counting Hundred's Position of Years
7 to 4	10 years	Undefined	R/W	Counting Ten's Position of Years
3 to 0	1 year	Undefined	R/W	Counting One's Position of Years

26.3.9 Second Alarm Register (RSECAR)

RSECAR is an alarm register corresponding to the BCD-coded second counter RSECCNT. When the ENB bit is set to 1, a comparison with the RSECCNT value is performed. From among RSECAR/RMINAR/RHRAR/RWKAR/RDAYAR/RMONAR/RCR3, the counter and alarm register comparison is performed only on those with ENB bits set to 1, and if each of those coincides, an alarm flag of RCR1 is set to 1.

The assignable range is from 00 through 59 + ENB bits (practically in BCD), otherwise operation errors occur.

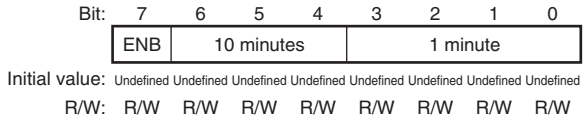


Bit	Bit Name	Initial Value	R/W	Description
7	ENB	Undefined	R/W	When this bit is set to 1, a comparison with the RSECCNT value is performed.
6 to 4	10 seconds	Undefined	R/W	Ten's position of seconds setting value
3 to 0	1 second	Undefined	R/W	One's position of seconds setting value

26.3.10 Minute Alarm Register (RMINAR)

RMINAR is an alarm register corresponding to the BCD-coded minute counter RMINCNT. When the ENB bit is set to 1, a comparison with the RMINCNT value is performed. From among RSECAR/RMINAR/RHRAR/RWKAR/RDAYAR/RMONAR/RCR3, the counter and alarm register comparison is performed only on those with ENB bits set to 1, and if each of those coincides, an alarm flag of RCR1 is set to 1.

The assignable range is from 00 through 59 + ENB bits (practically in BCD), otherwise operation errors occur.



Bit	Bit Name	Initial Value	R/W	Description
7	ENB	Undefined	R/W	When this bit is set to 1, a comparison with the RMINCNT value is performed.
6 to 4	10 minutes	Undefined	R/W	Ten's position of minutes setting value
3 to 0	1 minute	Undefined	R/W	One's position of minutes setting value

26.3.11 Hour Alarm Register (RHRAR)

RHRAR is an alarm register corresponding to the BCD-coded hour counter RHRCNT. When the ENB bit is set to 1, a comparison with the RHRCNT value is performed. From among RSECAR/RMINAR/RHRAR/RWKAR/RDAYAR/RMONAR/RCR3, the counter and alarm register comparison is performed only on those with ENB bits set to 1, and if each of those coincides, an alarm flag of RCR1 is set to 1.

The assignable range is from 00 through 23 + ENB bits (practically in BCD), otherwise operation errors occur.

Bit:	7	6	5	4	3	2	1	0
	ENB	-	10 hours		1 hour			
Initial value:	Undefined	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W:	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	ENB	Undefined	R/W	When this bit is set to 1, a comparison with the RHRCNT value is performed.
6	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
5, 4	10 hours	Undefined	R/W	Ten's position of hours setting value
3 to 0	1 hour	Undefined	R/W	One's position of hours setting value

26.3.12 Day of Week Alarm Register (RWKAR)

RWKAR is an alarm register corresponding to the BCD-coded day of week counter RWKCNT. When the ENB bit is set to 1, a comparison with the RWKCNT value is performed. From among RSECAR/RMINAR/RHRAR/RWKAR/RDAYAR/RMONAR/RCR3, the counter and alarm register comparison is performed only on those with ENB bits set to 1, and if each of those coincides, an alarm flag of RCR1 is set to 1.

The assignable range is from 0 through 6 + ENB bits (practically in BCD), otherwise operation errors occur.

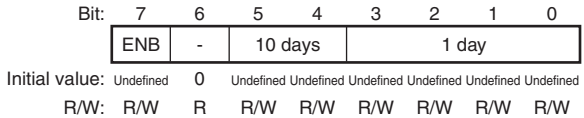
Bit:	7	6	5	4	3	2	1	0
	ENB	-	-	-	-	Day		
Initial value:	Undefined	0	0	0	0	Undefined	Undefined	Undefined
R/W:	R/W	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	ENB	Undefined	R/W	When this bit is set to 1, a comparison with the RWKCNT value is performed.
6 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	Day	Undefined	R/W	Day of Week Setting Value 000: Sunday 001: Monday 010: Tuesday 011: Wednesday 100: Thursday 101: Friday 110: Saturday 111: Reserved (setting prohibited)

26.3.13 Date Alarm Register (RDAYAR)

RDAYAR is an alarm register corresponding to the BCD-coded date counter RDAYCNT. When the ENB bit is set to 1, a comparison with the RDAYCNT value is performed. From among RSECAR/RMINAR/RHRAR/RWKAR/RDAYAR/RMONAR/RCR3, the counter and alarm register comparison is performed only on those with ENB bits set to 1, and if each of those coincides, an alarm flag of RCR1 is set to 1.

The assignable range is from 01 through 31 + ENB bits (practically in BCD), otherwise operation errors occur.



Bit	Bit Name	Initial Value	R/W	Description
7	ENB	Undefined	R/W	When this bit is set to 1, a comparison with the RDAYCNT value is performed.
6	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
5, 4	10 days	Undefined	R/W	Ten's position of dates setting value
3 to 0	1 day	Undefined	R/W	One's position of dates setting value

26.3.14 Month Alarm Register (RMONAR)

RMONAR is an alarm register corresponding to the BCD-coded month counter RMONCNT. When the ENB bit is set to 1, a comparison with the RMONCNT value is performed. From among RSECAR/RMINAR/RHRAR/RWKAR/RDAYAR/RMONAR/RCR3, the counter and alarm register comparison is performed only on those with ENB bits set to 1, and if each of those coincides, an alarm flag of RCR1 is set to 1.

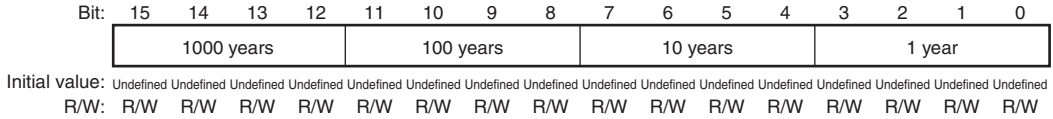
The assignable range is from 01 through 12 + ENB bits (practically in BCD), otherwise operation errors occur.

Bit:	7	6	5	4	3	2	1	0
	ENB	-	-	10 months	1 month			
Initial value:	Undefined	0	0	Undefined	Undefined	Undefined	Undefined	Undefined
R/W:	R/W	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	ENB	Undefined	R/W	When this bit is set to 1, a comparison with the RMONCNT value is performed.
6, 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	10 months	Undefined	R/W	Ten's position of months setting value
3 to 0	1 month	Undefined	R/W	One's position of months setting value

26.3.15 Year Alarm Register (RYRAR)

RYRAR is an alarm register corresponding to the year counter RYRCNT. The assignable range is from 0000 through 9999 (practically in BCD), otherwise operation errors occur.



Bit	Bit Name	Initial Value	R/W	Description
15 to 12	1000 years	Undefined	R/W	Thousand's position of years setting value
11 to 8	100 years	Undefined	R/W	Hundred's position of years setting value
7 to 4	10 years	Undefined	R/W	Ten's position of years setting value
3 to 0	1 year	Undefined	R/W	One's position of years setting value

26.3.16 Control Register 1 (RCR1)

RCR1 is a register that affects carry flags and alarm flags. It also selects whether to generate interrupts for each flag.

The CF flag remains undefined until the divider circuit is reset (the RESET and ADJ bits in RCR2 are set to 1). When using the CF flag, make sure to reset the divider circuit beforehand.

The AF flag remains undefined until the value is set to an alarm register and a counter. When using the AF flag, make sure to set the alarm register and counter beforehand.

Bit:	7	6	5	4	3	2	1	0
	CF	-	-	CIE	AIE	-	-	AF
Initial value:	Undefined	0	0	0	0	0	0	Undefined
R/W:	R/W	R	R	R/W	R/W	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	CF	Undefined	R/W	Carry Flag Status flag that indicates that a carry has occurred. CF is set to 1 when a count-up to 64-Hz occurs at the second counter carry or 64-Hz counter read. A count register value read at this time cannot be guaranteed; another read is required. 0: No carry of 64-Hz counter by second counter or 64-Hz counter [Clearing condition] When 0 is written to CF 1: Carry of 64-Hz counter by second counter or 64 Hz counter [Setting condition] When the second counter or 64-Hz counter is read during a carry occurrence by the 64-Hz counter, or 1 is written to CF.
6, 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
4	CIE	0	R/W	Carry Interrupt Enable Flag When the carry flag (CF) is set to 1, the CIE bit enables interrupts. 0: A carry interrupt is not generated when the CF flag is set to 1 1: A carry interrupt is generated when the CF flag is set to 1
3	AIE	0	R/W	Alarm Interrupt Enable Flag When the alarm flag (AF) is set to 1, the AIE bit allows interrupts. 0: An alarm interrupt is not generated when the AF flag is set to 1 1: An alarm interrupt is generated when the AF flag is set to 1
2, 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	AF	Undefined	R/W	Alarm Flag The AF flag is set when the alarm time, which is set by an alarm register (ENB bit in RSECAR, RMINAR, RHRAR, RWKAR, RDAYAR, RMONAR, or RYRAR is set to 1), and counter match. 0: Alarm register and counter not match [Clearing condition] When 0 is written to AF. 1: Alarm register and counter match* [Setting condition] When alarm register (only a register with ENB bit set to 1) and counter match Note: * Writing 1 holds previous value.

26.3.17 Control Register 2 (RCR2)

RCR2 is a register for periodic interrupt control, 30-second adjustment, divider circuit RESET, and count control.

RCR2 is initialized by a power-on reset or in deep standby mode. Bits other than the RTCEN and START bits are initialized by a manual reset.

Bit:	7	6	5	4	3	2	1	0
	PEF	PES[2:0]			RTCEN	ADJ	RESET	START
Initial value:	0	0	0	0	1	0	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	PEF	0	R/W	Periodic Interrupt Flag Indicates interrupt generation with the period designated by the PES2 to PES0 bits. When set to 1, PEF generates periodic interrupts. 0: Interrupts not generated with the period designated by the bits PES2 to PES0. [Clearing condition] When 0 is written to PEF 1: Interrupts generated with the period designated by the PES2 to PES0 bits. [Setting condition] When an interrupt is generated with the period designated by the bits PES0 to PES2 or when 1 is written to the PEF flag

Bit	Bit Name	Initial Value	R/W	Description
6 to 4	PES[2:0]	000	R/W	Interrupt Enable Flags These bits specify the periodic interrupt. 000: No periodic interrupts generated 001: Setting inhibited 010: Periodic interrupt generated every 1/64 second 011: Periodic interrupt generated every 1/16 second 100: Periodic interrupt generated every 1/4 second 101: Periodic interrupt generated every 1/2 second 110: Periodic interrupt generated every 1 second 111: Periodic interrupt generated every 2 seconds
3	RTCEN	1	R/W	RTC_X1 Clock Control Controls the function of RTC_X1 pin. 0: Halts the on-chip crystal oscillator/disables the external clock input. 1: Runs the on-chip crystal oscillator/enables the external clock input.
2	ADJ	0	R/W	30-Second Adjustment When 1 is written to the ADJ bit, times of 29 seconds or less will be rounded to 00 seconds and 30 seconds or more to 1 minute. The divider circuit (prescaler and R64CNT) will be simultaneously reset. This bit always reads 0. 0: Runs normally. 1: 30-second adjustment.

Bit	Bit Name	Initial Value	R/W	Description
1	RESET	0	R/W	Reset Writing 1 to this bit initializes the divider circuit, the R64CNT register, the alarm register, the RCR3 register, bits CF and AF in RCR1, and bit PEF in RCR2. In this case, the RESET bit is automatically reset to 0 after 1 is written to and the above registers are reset. Thus, there is no need to write 1 to this bit. This bit is always read as 0. 0: Runs normally. 1: Divider circuit is reset.
0	START	1	R/W	Start Halts and restarts the counter (clock). 0: Second/minute/hour/date/day of the week/month/year counter halts. 1: Second/minute/hour/date/day of the week/month/year counter runs normally.

26.3.18 Control Register 3 (RCR3)

When the ENB bit is set to 1, RCR3 performs a comparison with the RYRCNT. From among RSECAR/RMINAR/RHRAR/RWKAR/RDAYAR/RMONAR/RCR3, the counter and alarm register comparison is performed only on those with ENB bits set to 1, and if each of those coincides, an alarm flag of RCR1 is set to 1.

Bit:	7	6	5	4	3	2	1	0
	ENB	-	-	-	-	-	-	-
Initial value:	Undefined	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	ENB	Undefined	R/W	When this bit is set to 1, comparison of the year alarm register (RYRAR) and the year counter (RYRCNT) is performed.
6 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

26.4 Operation

Usage of this module is shown below.

26.4.1 Initial Settings of Registers after Power-On

All the registers should be set after the power is turned on.

26.4.2 Setting Time

Figure 26.2 shows how to set the time when the clock is stopped.

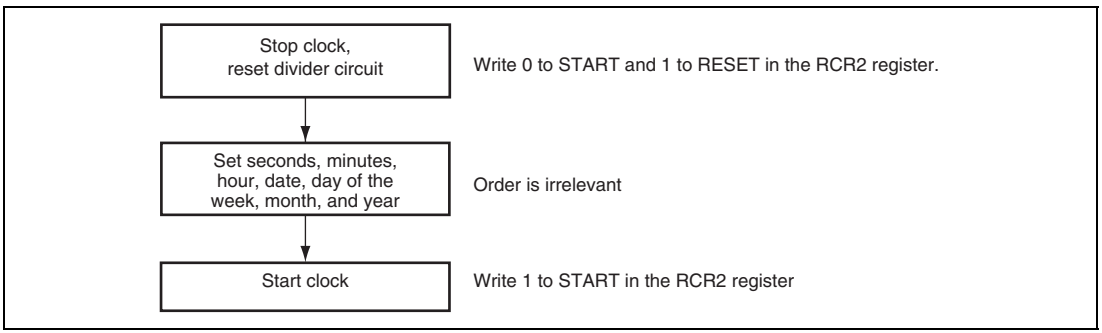


Figure 26.2 Setting Time

26.4.3 Reading Time

Figure 26.3 shows how to read the time.

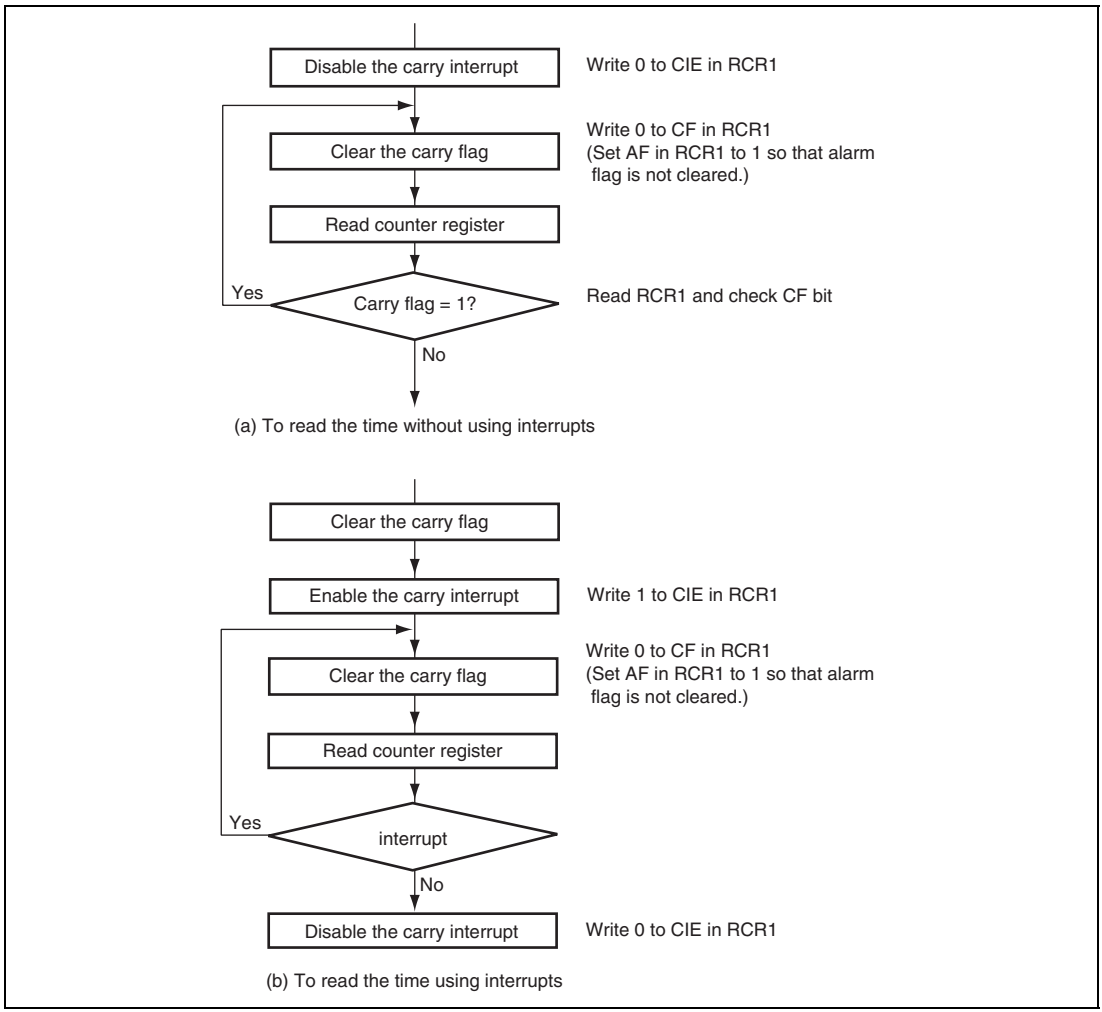


Figure 26.3 Reading Time

If a carry occurs while reading the time, the correct time will not be obtained, so it must be read again. Part (a) in figure 26.3 shows the method of reading the time without using interrupts; part (b) in figure 26.3 shows the method using carry interrupts. To keep programming simple, method (a) should normally be used.

26.4.4 Alarm Function

Figure 26.4 shows how to use the alarm function.

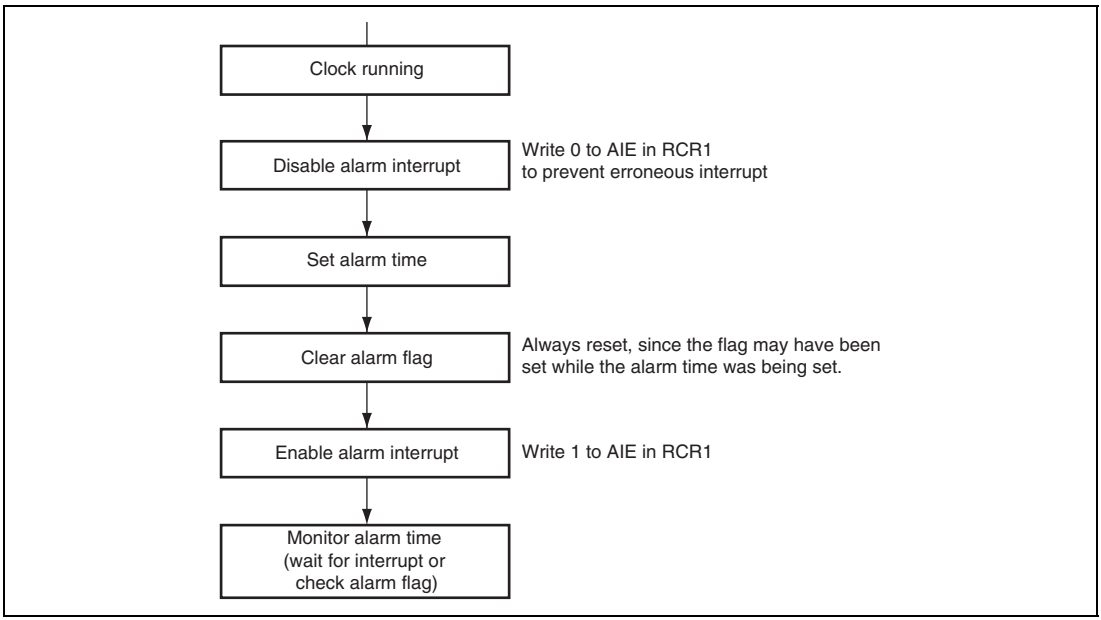


Figure 26.4 Using Alarm Function

Alarms can be generated using seconds, minutes, hours, day of the week, date, month, year, or any combination of these. Set the ENB bit in the register on which the alarm is placed to 1, and then set the alarm time in the lower bits. Clear the ENB bit in the register on which the alarm is not placed to 0.

When the clock and alarm times match, 1 is set in the AF bit in RCR1. Alarm detection can be checked by reading this bit, but normally it is done by interrupt. If 1 is set in the AIE bit in RCR1, an interrupt is generated when an alarm occurs.

The alarm flag is set when the clock and alarm times match. However, the alarm flag can be cleared by writing 0.

26.5 Usage Notes

26.5.1 Register Writing during Count

The following registers cannot be written to during a count (while bit 0 = 1 in RCR2).

RSECCNT, RMINCNT, RHRCNT, RDAYCNT, RWKCNT, RMONCNT, RYRCONT

The count must be stopped before writing to any of the above registers.

26.5.2 Use of Realtime Clock Periodic Interrupts

The method of using the periodic interrupt function is shown in figure 26.5.

A periodic interrupt can be generated periodically at the interval set by bits PES2 to PES0 in RCR2. When the time set by bits PES2 to PES0 has elapsed, the PEF is set to 1.

The PEF is cleared to 0 upon periodic interrupt generation or when bits PES2 to PES0 are set. Periodic interrupt generation can be confirmed by reading this bit, but normally the interrupt function is used.

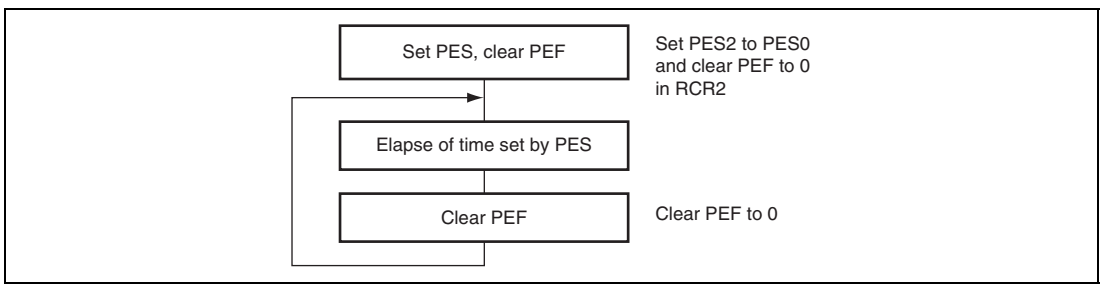


Figure 26.5 Using Periodic Interrupt Function

26.5.3 Transition to Standby Mode after Setting Register

When a transition to standby mode is made after registers in this module are set, sometimes counting is not performed correctly. In case the registers are set, be sure to make a transition to standby mode after performing one dummy read of the register.

26.5.4 Usage Notes when Writing to and Reading the Register

- When reading a counter register such as the second counter after having written to the register, follow the procedure in section 26.4.3, Reading Time.
- After writing to the RCR2 register, perform two dummy reads before reading data. The register contents from before the write are returned by the two dummy reads, and the third read returns the register contents reflecting the write.
- Registers other than the above can be read immediately after a write and the written value is reflected.

Section 27 Multi-Function Timer Pulse Unit 2

This LSI has an on-chip multi-function timer pulse unit 2 that comprises five 16-bit timer channels.

27.1 Features

- Maximum 16 pulse input/output lines
- Selection of eight counter input clocks for each channel
- The following operations can be set:
  - Waveform output at compare match
  - Input capture function
  - Counter clear operation
  - Multiple timer counters (TCNT) can be written to simultaneously
  - Simultaneous clearing by compare match and input capture is possible
  - Register simultaneous input/output is possible by synchronous counter operation
  - A maximum 12-phase PWM output is possible in combination with synchronous operation
- Buffer operation settable for channels 0, 3, and 4
- Phase counting mode settable independently for each of channels 1 and 2
- Cascade connection operation
- 28 interrupt sources
- Automatic transfer of register data
- A/D converter start trigger can be generated
- Module standby mode can be settable
- A total of six-phase waveform output, which includes complementary PWM output, and positive and negative phases of reset PWM output by interlocking operation of channels 3 and 4, is possible.
- AC synchronous motor (brushless DC motor) drive mode using complementary PWM output and reset PWM output is settable by interlocking operation of channels 0, 3, and 4, and the selection of two types of waveform outputs (chopping and level) is possible.
- In complementary PWM mode, interrupts at the crest and trough of the counter value and A/D converter start triggers can be skipped.

Note: Pϕ refers to the HPB P2BUS clock (32 bits/50 MHz) in this section. It is accessible by a 16-bit bus within this module.

Table 27.1 Functions of Multi-Function Timer Pulse Unit 2

Item		Channel 0	Channel 1	Channel 2	Channel 3	Channel 4
Count clock		Pφ/1 Pφ/4 Pφ/16 Pφ/64 TCLKA TCLKB TCLKC TCLKD	Pφ/1 Pφ/4 Pφ/16 Pφ/64 Pφ/256 TCLKA TCLKB	Pφ/1 Pφ/4 Pφ/16 Pφ/64 Pφ/1024 TCLKA TCLKB TCLKC	Pφ/1 Pφ/4 Pφ/16 Pφ/64 Pφ/256 Pφ/1024 TCLKA TCLKB	Pφ/1 Pφ/4 Pφ/16 Pφ/64 Pφ/256 Pφ/1024 TCLKA TCLKB
General registers (TGR)		TGRA_0 TGRB_0 TGRE_0	TGRA_1 TGRB_1	TGRA_2 TGRB_2	TGRA_3 TGRB_3	TGRA_4 TGRB_4
General registers/ buffer registers		TGRC_0 TGRD_0 TGRF_0	—	—	TGRC_3 TGRD_3	TGRC_4 TGRD_4
I/O pins		TIOC0A TIOC0B TIOC0C TIOC0D	TIOC1A TIOC1B	TIOC2A TIOC2B	TIOC3A TIOC3B TIOC3C TIOC3D	TIOC4A TIOC4B TIOC4C TIOC4D
Counter clear function		TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture
Compare match output	0 output	√	√	√	√	√
	1 output	√	√	√	√	√
	Toggle output	√	√	√	√	√
Input capture function		√	√	√	√	√
Synchronous operation		√	√	√	√	√
PWM mode 1		√	√	√	√	√
PWM mode 2		√	√	√	—	—
Complementary PWM mode		—	—	—	√	√
Reset PWM mode		—	—	—	√	√
AC synchronous motor drive mode		√	—	—	√	√

Item	Channel 0	Channel 1	Channel 2	Channel 3	Channel 4
Phase counting mode	—	√	√	—	—
Buffer operation	√	—	—	√	√
Activation of direct memory access controller	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture and TCNT overflow or underflow
A/D converter start trigger	TGRA_0 compare match or input capture TGRE_0 compare match	TGRA_1 compare match or input capture	TGRA_2 compare match or input capture	TGRA_3 compare match or input capture	TGRA_4 compare match or input capture TCNT_4 underflow (trough) in complementary PWM mode
Interrupt sources	7 sources • Compare match or input capture 0A • Compare match or input capture 0B • Compare match or input capture 0C • Compare match or input capture 0D • Compare match 0E • Compare match 0F • Overflow	4 sources • Compare match or input capture 1A • Compare match or input capture 1B • Overflow • Underflow	4 sources • Compare match or input capture 2A • Compare match or input capture 2B • Overflow • Underflow	5 sources • Compare match or input capture 3A • Compare match or input capture 3B • Compare match or input capture 3C • Compare match or input capture 3D • Overflow	5 sources • Compare match or input capture 4A • Compare match or input capture 4B • Compare match or input capture 4C • Compare match or input capture 4D • Overflow or underflow

Item	Channel 0	Channel 1	Channel 2	Channel 3	Channel 4
A/D converter start request delaying function	—	—	—	—	• A/D converter start request at a match between TADCORA_4 and TCNT_4 • A/D converter start request at a match between TADCORB_4 and TCNT_4
Interrupt skipping function	—	—	—	• Skips TGRA_3 compare match interrupts	• Skips TCIV_4 interrupts

[Legend]
√: Available
—: Not available

Figure 27.1 shows a block diagram.

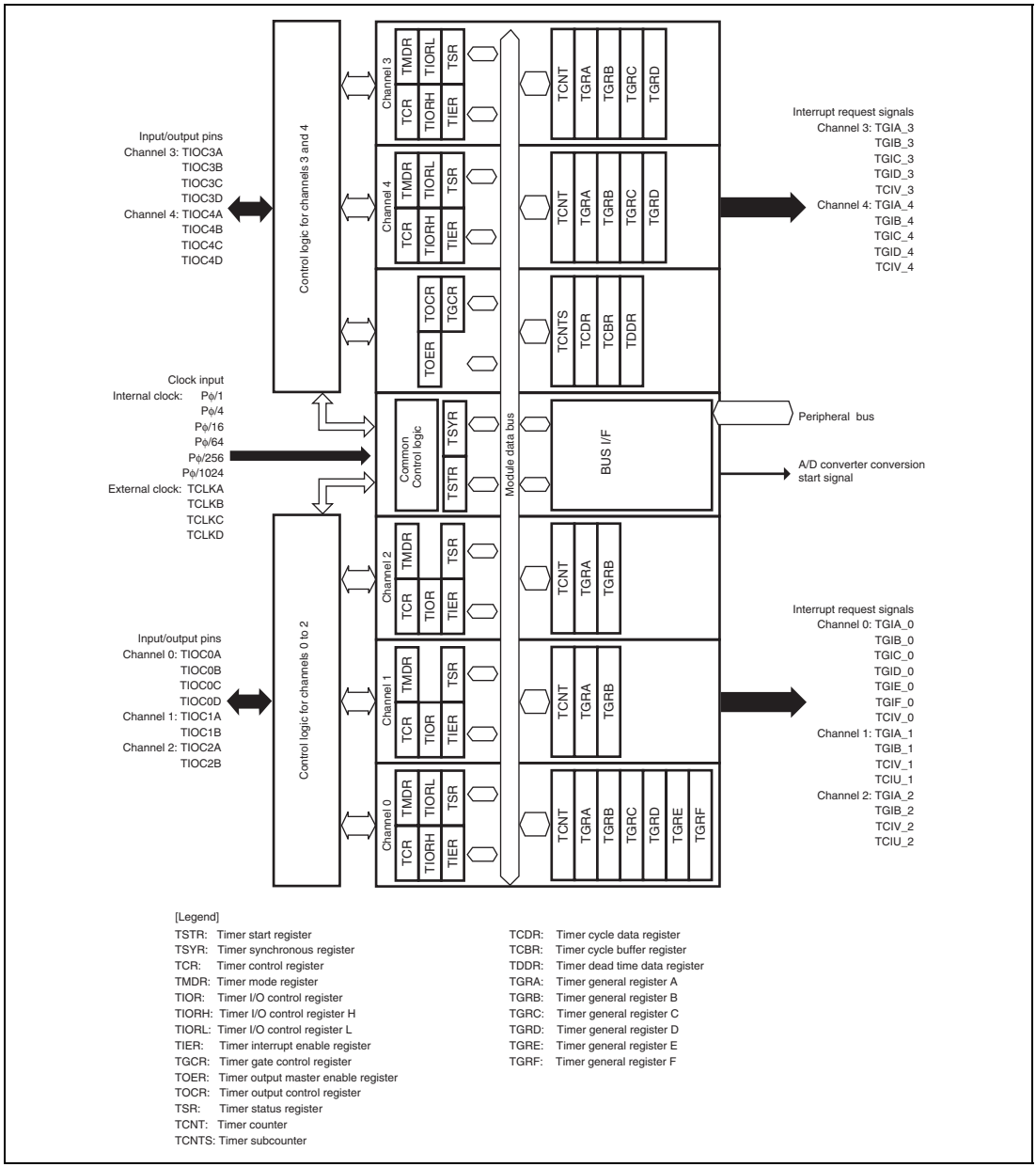


Figure 27.1 Block Diagram

27.2 Input/Output Pins

Table 27.2 shows the pin configuration.

Table 27.2 Pin Configuration

Channel	Pin Name	I/O	Function
Common	TCLKA	Input	External clock A input pin (Channel 1 phase counting mode A phase input)
	TCLKB	Input	External clock B input pin (Channel 1 phase counting mode B phase input)
	TCLKC	Input	External clock C input pin (Channel 2 phase counting mode A phase input)
	TCLKD	Input	External clock D input pin (Channel 2 phase counting mode B phase input)
0	TIOC0A	I/O	TGRA_0 input capture input/output compare output/PWM output pin
	TIOC0B	I/O	TGRB_0 input capture input/output compare output/PWM output pin
	TIOC0C	I/O	TGRC_0 input capture input/output compare output/PWM output pin
	TIOC0D	I/O	TGRD_0 input capture input/output compare output/PWM output pin
1	TIOC1A	I/O	TGRA_1 input capture input/output compare output/PWM output pin
	TIOC1B	I/O	TGRB_1 input capture input/output compare output/PWM output pin
2	TIOC2A	I/O	TGRA_2 input capture input/output compare output/PWM output pin
	TIOC2B	I/O	TGRB_2 input capture input/output compare output/PWM output pin
3	TIOC3A	I/O	TGRA_3 input capture input/output compare output/PWM output pin
	TIOC3B	I/O	TGRB_3 input capture input/output compare output/PWM output pin
	TIOC3C	I/O	TGRC_3 input capture input/output compare output/PWM output pin
	TIOC3D	I/O	TGRD_3 input capture input/output compare output/PWM output pin
4	TIOC4A	I/O	TGRA_4 input capture input/output compare output/PWM output pin
	TIOC4B	I/O	TGRB_4 input capture input/output compare output/PWM output pin
	TIOC4C	I/O	TGRC_4 input capture input/output compare output/PWM output pin
	TIOC4D	I/O	TGRD_4 input capture input/output compare output/PWM output pin

Note: For the pin configuration in complementary PWM mode, see table 27.52 in section 27.4.8, Complementary PWM Mode.

27.3 Register Descriptions

Table 27.3 shows the register configuration. To distinguish registers in each channel, an underscore and the channel number are added as a suffix to the register name; TCR for channel 0 is expressed as TCR_0.

Table 27.3 (1) Register Configuration

Channel	Register Name	Abbreviation	R/W	Initial value	Address	Access Size
0	Timer control register_0	TCR_0	R/W	H'00	H'FFFC6300	8
	Timer mode register_0	TMDR_0	R/W	H'00	H'FFFC6301	8
	Timer I/O control register H_0	TIORH_0	R/W	H'00	H'FFFC6302	8
	Timer I/O control register L_0	TIORL_0	R/W	H'00	H'FFFC6303	8
	Timer interrupt enable register_0	TIER_0	R/W	H'00	H'FFFC6304	8
	Timer status register_0	TSR_0	R/W	H'C0	H'FFFC6305	8
	Timer counter_0	TCNT_0	R/W	H'0000	H'FFFC6306	16
	Timer general register A_0	TGRA_0	R/W	H'FFFF	H'FFFC6308	16
	Timer general register B_0	TGRB_0	R/W	H'FFFF	H'FFFC630A	16
	Timer general register C_0	TGRC_0	R/W	H'FFFF	H'FFFC630C	16
	Timer general register D_0	TGRD_0	R/W	H'FFFF	H'FFFC630E	16
	Timer general register E_0	TGRE_0	R/W	H'FFFF	H'FFFC6320	16
	Timer general register F_0	TGRF_0	R/W	H'FFFF	H'FFFC6322	16
	Timer interrupt enable register 2_0	TIER2_0	R/W	H'00	H'FFFC6324	8
	Timer status register 2_0	TSR2_0	R/W	H'C0	H'FFFC6325	8
	Timer buffer operation transfer mode register_0	TBTM_0	R/W	H'00	H'FFFC6326	8
1	Timer control register_1	TCR_1	R/W	H'00	H'FFFC6380	8
	Timer mode register_1	TMDR_1	R/W	H'00	H'FFFC6381	8
	Timer I/O control register_1	TIOR_1	R/W	H'00	H'FFFC6382	8
	Timer interrupt enable register_1	TIER_1	R/W	H'00	H'FFFC6384	8
	Timer status register_1	TSR_1	R/W	H'C0	H'FFFC6385	8

Channel	Register Name	Abbreviation	R/W	Initial value	Address	Access Size
1	Timer counter_1	TCNT_1	R/W	H'0000	H'FFFC6386	16
	Timer general register A_1	TGRA_1	R/W	H'FFFF	H'FFFC6388	16
	Timer general register B_1	TGRB_1	R/W	H'FFFF	H'FFFC638A	16
	Timer input capture control register	TICCR	R/W	H'00	H'FFFC6390	8
2	Timer control register_2	TCR_2	R/W	H'00	H'FFFC6000	8
	Timer mode register_2	TMDR_2	R/W	H'00	H'FFFC6001	8
	Timer I/O control register_2	TIOR_2	R/W	H'00	H'FFFC6002	8
	Timer interrupt enable register_2	TIER_2	R/W	H'00	H'FFFC6004	8
	Timer status register_2	TSR_2	R/W	H'C0	H'FFFC6005	8
	Timer counter_2	TCNT_2	R/W	H'0000	H'FFFC6006	16
	Timer general register A_2	TGRA_2	R/W	H'FFFF	H'FFFC6008	16
	Timer general register B_2	TGRB_2	R/W	H'FFFF	H'FFFC600A	16
3	Timer control register_3	TCR_3	R/W	H'00	H'FFFC6200	8
	Timer mode register_3	TMDR_3	R/W	H'00	H'FFFC6202	8
	Timer I/O control register H_3	TIORH_3	R/W	H'00	H'FFFC6204	8
	Timer I/O control register L_3	TIORL_3	R/W	H'00	H'FFFC6205	8
	Timer interrupt enable register_3	TIER_3	R/W	H'00	H'FFFC6208	8
	Timer status register_3	TSR_3	R/W	H'C0	H'FFFC622C	8
	Timer counter_3	TCNT_3	R/W	H'0000	H'FFFC6210	16
	Timer general register A_3	TGRA_3	R/W	H'FFFF	H'FFFC6218	16
	Timer general register B_3	TGRB_3	R/W	H'FFFF	H'FFFC621A	16
	Timer general register C_3	TGRC_3	R/W	H'FFFF	H'FFFC6224	16
	Timer general register D_3	TGRD_3	R/W	H'FFFF	H'FFFC6226	16
	Timer buffer operation transfer mode register_3	TBTM_3	R/W	H'00	H'FFFC6238	8
4	Timer control register_4	TCR_4	R/W	H'00	H'FFFC6201	8
	Timer mode register_4	TMDR_4	R/W	H'00	H'FFFC6203	8
	Timer I/O control register H_4	TIORH_4	R/W	H'00	H'FFFC6206	8
	Timer I/O control register L_4	TIORL_4	R/W	H'00	H'FFFC6207	8

Channel	Register Name	Abbreviation	R/W	Initial value	Address	Access Size
4	Timer interrupt enable register_4	TIER_4	R/W	H'00	H'FFFC6209	8
	Timer status register_4	TSR_4	R/W	H'C0	H'FFFC622D	8
	Timer counter_4	TCNT_4	R/W	H'0000	H'FFFC6212	16
	Timer general register A_4	TGRA_4	R/W	H'FFFF	H'FFFC621C	16
	Timer general register B_4	TGRB_4	R/W	H'FFFF	H'FFFC621E	16
	Timer general register C_4	TGRC_4	R/W	H'FFFF	H'FFFC6228	16
	Timer general register D_4	TGRD_4	R/W	H'FFFF	H'FFFC622A	16
	Timer buffer operation transfer mode register_4	TBTM_4	R/W	H'00	H'FFFC6239	8
	Timer A/D converter start request control register	TADCR	R/W	H'0000	H'FFFC6240	16
	Timer A/D converter start request cycle set register A_4	TADCORA_4	R/W	H'FFFF	H'FFFC6244	16
	Timer A/D converter start request cycle set register B_4	TADCORB_4	R/W	H'FFFF	H'FFFC6246	16
	Timer A/D converter start request cycle set buffer register A_4	TADCOBRA_4	R/W	H'FFFF	H'FFFC6248	16
	Timer A/D converter start request cycle set buffer register B_4	TADCOBRB_4	R/W	H'FFFF	H'FFFC624A	16
Common	Timer start register	TSTR	R/W	H'00	H'FFFC6280	8
	Timer synchronous register	TSYR	R/W	H'00	H'FFFC6281	8
	Timer read/write enable register	TRWER	R/W	H'01	H'FFFC6284	8

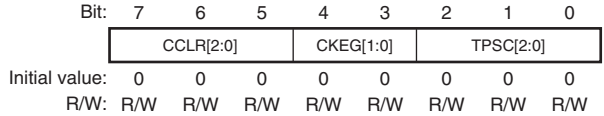
Channel	Register Name	Abbreviation	R/W	Initial value	Address	Access Size
Common to 3 and 4	Timer output master enable register	TOER	R/W	H'C0	H'FFFC620A	8
	Timer output control register 1	TOCR1	R/W	H'00	H'FFFC620E	8
	Timer output control register 2	TOCR2	R/W	H'00	H'FFFC620F	8
	Timer gate control register	TGCR	R/W	H80	H'FFFC620D	8
	Timer cycle data register	TCDR	R/W	H'FFFF	H'FFFC6214	16
	Timer dead time data register	TDDR	R/W	H'FFFF	H'FFFC6216	16
	Timer subcounter	TCNTS	R	H'0000	H'FFFC6220	16
	Timer cycle buffer register	TCBR	R/W	H'FFFF	H'FFFC6222	16
	Timer interrupt skipping set register	TITCR	R/W	H'00	H'FFFC6230	8
	Timer interrupt skipping counter	TITCNT	R	H'00	H'FFFC6231	8
	Timer buffer transfer set register	TBTER	R/W	H'00	H'FFFC6232	8
	Timer dead time enable register	TDER	R/W	H'01	H'FFFC6234	8
	Timer waveform control register	TWCR	R/W	H'00	H'FFFC6260	8
	Timer output level buffer register	TOLBR	R/W	H'00	H'FFFC6236	8

Table 27.3 (2) Register State in Each Operating Mode

Abbreviation	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
All registers	Initialized	Initialized	Retained	Retained	Initialized	Initialized

27.3.1 Timer Control Register (TCR)

The TCR registers are 8-bit readable/writable registers that control the TCNT operation for each channel. This module has a total of five TCR registers, one each for channels 0 to 4. TCR register settings should be conducted only when TCNT operation is stopped.



Bit	Bit Name	Initial Value	R/W	Description
7 to 5	CCLR[2:0]	000	R/W	Counter Clear 0 to 2 These bits select the TCNT counter clearing source. See tables 27.4 and 27.5 for details.
4, 3	CKEG[1:0]	00	R/W	Clock Edge 0 and 1 These bits select the input clock edge. When the input clock is counted using both edges, the input clock period is halved (e.g. Pϕ/4 both edges = Pϕ/2 rising edge). If phase counting mode is used on channels 1 and 2, this setting is ignored and the phase counting mode setting has priority. Internal clock edge selection is valid when the input clock is Pϕ/4 or slower. When Pϕ/1, or the overflow/underflow of another channel is selected for the input clock, although values can be written, counter operation compiles with the initial value. 00: Count at rising edge 01: Count at falling edge 1x: Count at both edges
2 to 0	TPSC[2:0]	000	R/W	Time Prescaler 0 to 2 These bits select the TCNT counter clock. The clock source can be selected independently for each channel. See tables 27.6 to 27.9 for details.

[Legend]

x: Don't care

Table 27.4 CCLR0 to CCLR2 (Channels 0, 3, and 4)

Channel	Bit 7 CCLR2	Bit 6 CCLR1	Bit 5 CCLR0	Description
0, 3, 4	0	0	0	TCNT clearing disabled
			1	TCNT cleared by TGRA compare match/input capture
		1	0	TCNT cleared by TGRB compare match/input capture
			1	TCNT cleared by counter clearing for another channel performing synchronous clearing/ synchronous operation*1
	1	0	0	TCNT clearing disabled
			1	TCNT cleared by TGRC compare match/input capture*2
		1	0	TCNT cleared by TGRD compare match/input capture*2
			1	TCNT cleared by counter clearing for another channel performing synchronous clearing/ synchronous operation*1

Notes: 1. Synchronous operation is set by setting the SYNC bit in TSYR to 1.
2. When TGRC or TGRD is used as a buffer register, TCNT is not cleared because the buffer register setting has priority, and compare match/input capture does not occur.

Table 27.5 CCLR0 to CCLR2 (Channels 1 and 2)

Channel	Bit 7 Reserved*2	Bit 6 CCLR1	Bit 5 CCLR0	Description
1, 2	0	0	0	TCNT clearing disabled
			1	TCNT cleared by TGRA compare match/input capture
		1	0	TCNT cleared by TGRB compare match/input capture
			1	TCNT cleared by counter clearing for another channel performing synchronous clearing/ synchronous operation*1

Notes: 1. Synchronous operation is selected by setting the SYNC bit in TSYR to 1.
2. Bit 7 is reserved in channels 1 and 2. It is always read as 0 and cannot be modified.

Table 27.6 TPSC0 to TPSC2 (Channel 0)

Channel	Bit 2 TPSC2	Bit 1 TPSC1	Bit 0 TPSC0	Description
0	0	0	0	Internal clock: counts on Pϕ/1
			1	Internal clock: counts on Pϕ/4
		1	0	Internal clock: counts on Pϕ/16
			1	Internal clock: counts on Pϕ/64
	1	0	0	External clock: counts on TCLKA pin input
			1	External clock: counts on TCLKB pin input
		1	0	External clock: counts on TCLKC pin input
			1	External clock: counts on TCLKD pin input

Table 27.7 TPSC0 to TPSC2 (Channel 1)

Channel	Bit 2 TPSC2	Bit 1 TPSC1	Bit 0 TPSC0	Description
1	0	0	0	Internal clock: counts on Pϕ/1
			1	Internal clock: counts on Pϕ/4
		1	0	Internal clock: counts on Pϕ/16
			1	Internal clock: counts on Pϕ/64
	1	0	0	External clock: counts on TCLKA pin input
			1	External clock: counts on TCLKB pin input
		1	0	Internal clock: counts on Pϕ/256
			1	Counts on TCNT_2 overflow/underflow

Note: This setting is ignored when channel 1 is in phase counting mode.

Table 27.8 TPSC0 to TPSC2 (Channel 2)

Channel	Bit 2 TPSC2	Bit 1 TPSC1	Bit 0 TPSC0	Description
2	0	0	0	Internal clock: counts on Pϕ/1
			1	Internal clock: counts on Pϕ/4
		1	0	Internal clock: counts on Pϕ/16
			1	Internal clock: counts on Pϕ/64
	1	0	0	External clock: counts on TCLKA pin input
			1	External clock: counts on TCLKB pin input
		1	0	External clock: counts on TCLKC pin input
			1	Internal clock: counts on Pϕ/1024

Note: This setting is ignored when channel 2 is in phase counting mode.

Table 27.9 TPSC0 to TPSC2 (Channels 3 and 4)

Channel	Bit 2 TPSC2	Bit 1 TPSC1	Bit 0 TPSC0	Description
3, 4	0	0	0	Internal clock: counts on Pϕ/1
			1	Internal clock: counts on Pϕ/4
		1	0	Internal clock: counts on Pϕ/16
			1	Internal clock: counts on Pϕ/64
	1	0	0	Internal clock: counts on Pϕ/256
			1	Internal clock: counts on Pϕ/1024
		1	0	External clock: counts on TCLKA pin input
			1	External clock: counts on TCLKB pin input

27.3.2 Timer Mode Register (TMDR)

The TMDR registers are 8-bit readable/writable registers that are used to set the operating mode of each channel. This module has five TMDR registers, one each for channels 0 to 4. TMDR register settings should be changed only when TCNT operation is stopped.

Bit:	7	6	5	4	3	2	1	0
	-	BFE	BFB	BFA	MD[3:0]			
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	BFE	0	R/W	Buffer Operation E Specifies whether TGRE_0 and TGRF_0 are to operate in the normal way or to be used together for buffer operation. TGRF compare match is generated when TGRF is used as the buffer register. In channels 1 to 4, this bit is reserved. It is always read as 0 and the write value should always be 0. 0: TGRE_0 and TGRF_0 operate normally 1: TGRE_0 and TGRF_0 used together for buffer operation

Bit	Bit Name	Initial Value	R/W	Description
5	BFB	0	R/W	Buffer Operation B Specifies whether TGRB is to operate in the normal way, or TGRB and TGRD are to be used together for buffer operation. When TGRD is used as a buffer register, TGRD input capture/output compare is not generated in a mode other than complementary PWM. In channels 1 and 2, which have no TGRD, bit 5 is reserved. It is always read as 0 and cannot be modified. 0: TGRB and TGRD operate normally 1: TGRB and TGRD used together for buffer operation
4	BFA	0	R/W	Buffer Operation A Specifies whether TGRA is to operate in the normal way, or TGRA and TGRC are to be used together for buffer operation. When TGRC is used as a buffer register, TGRC input capture/output compare is not generated in a mode other than complementary PWM. TGRC compare match is generated when in complementary PWM mode. When compare match for channel 4 occurs during the Tb period in complementary PWM mode, TGFC is set. Therefore, set the TGIEC bit in the timer interrupt enable register 4 (TIER_4) to 0. In channels 1 and 2, which have no TGRC, bit 4 is reserved. It is always read as 0 and cannot be modified. 0: TGRA and TGRC operate normally 1: TGRA and TGRC used together for buffer operation
3 to 0	MD[3:0]	0000	R/W	Modes 0 to 3 These bits are used to set the timer operating mode. See table 27.10 for details.

Table 27.10 Setting of Operation Mode by Bits MD0 to MD3

Bit 3 MD3	Bit 2 MD2	Bit 1 MD1	Bit 0 MD0	Description
0	0	0	0	Normal operation
			1	Setting prohibited
		1	0	PWM mode 1
			1	PWM mode 2*[1]
	1	0	0	Phase counting mode 1*[2]
			1	Phase counting mode 2*[2]
		1	0	Phase counting mode 3*[2]
			1	Phase counting mode 4*[2]
1	0	0	0	Reset synchronous PWM mode*[3]
			1	Setting prohibited
		1	X	Setting prohibited
	1	0	0	Setting prohibited
			1	Complementary PWM mode 1 (transmit at crest)*[3]
		1	0	Complementary PWM mode 2 (transmit at trough)*[3]
			1	Complementary PWM mode 2 (transmit at crest and trough)*[3]

[Legend]

X: Don't care

Notes: 1. PWM mode 2 cannot be set for channels 3 and 4.

2. Phase counting mode cannot be set for channels 0, 3, and 4.

3. Reset synchronous PWM mode, complementary PWM mode can only be set for channel 3. When channel 3 is set to reset synchronous PWM mode or complementary PWM mode, the channel 4 settings become ineffective and automatically conform to the channel 3 settings. However, do not set channel 4 to reset synchronous PWM mode or complementary PWM mode. Reset synchronous PWM mode and complementary PWM mode cannot be set for channels 0, 1, and 2.

27.3.3 Timer I/O Control Register (TIOR)

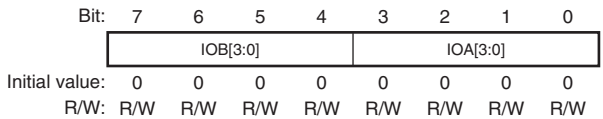
The TIOR registers are 8-bit readable/writable registers that control the TGR registers. This module has a total of eight TIOR registers, two each for channels 0, 3, and 4, one each for channels 1 and 2.

TIOR should be set while TMDR is set in normal operation, PWM mode, or phase counting mode.

The initial output specified by TIOR is valid when the counter is stopped (the CST bit in TSTR is cleared to 0). Note also that, in PWM mode 2, the output at the point at which the counter is cleared to 0 is specified.

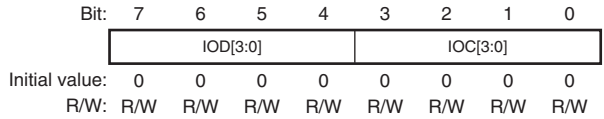
When TGRC or TGRD is designated for buffer operation, this setting is invalid and the register operates as a buffer register.

- TIORH_0, TIOR_1, TIOR_2, TIORH_3, TIORH_4



Bit	Bit Name	Initial Value	R/W	Description
7 to 4	IOB[3:0]	0000	R/W	I/O Control B0 to B3 Specify the function of TGRB. See the following tables. TIORH_0: Table 27.11 TIOR_1: Table 27.13 TIOR_2: Table 27.14 TIORH_3: Table 27.15 TIORH_4: Table 27.17
3 to 0	IOA[3:0]	0000	R/W	I/O Control A0 to A3 Specify the function of TGRA. See the following tables. TIORH_0: Table 27.19 TIOR_1: Table 27.21 TIOR_2: Table 27.22 TIORH_3: Table 27.23 TIORH_4: Table 27.25

- TIORL_0, TIORL_3, TIORL_4



Bit	Bit Name	Initial Value	R/W	Description
7 to 4	IOD[3:0]	0000	R/W	I/O Control D0 to D3 Specify the function of TGRD. See the following tables. TIORL_0: Table 27.12 TIORL_3: Table 27.16 TIORL_4: Table 27.18
3 to 0	IOC[3:0]	0000	R/W	I/O Control C0 to C3 Specify the function of TGRC. See the following tables. TIORL_0: Table 27.20 TIORL_3: Table 27.24 TIORL_4: Table 27.26

Table 27.11 TIORH_0 (Channel 0)

				Description	
Bit 7 IOB3	**Bit 6 IOB2**	**Bit 5 IOB1**	**Bit 4 IOB0**	**TGRB_0 Function**	**TIOC0B Pin Function**
0	0	0	0	Output compare register	Output retained*
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	0	0	0	Input capture register	Input capture at rising edge
			1		Input capture at falling edge
		1	X		Input capture at both edges
	1	X	X		Capture input source is channel 1/count clock Input capture at TCNT_1 count-up/count-down

[Legend]

X: Don't care

Note: * After power-on reset, 0 is output until TIOR is set.

Table 27.12 TIORL_0 (Channel 0)

				Description	
Bit 7 IOD3	**Bit 6 IOD2**	**Bit 5 IOD1**	**Bit 4 IOD0**	**TGRD_0 Function**	**TIOC0D Pin Function**
0	0	0	0	Output compare register*2	Output retained*1
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	0	0	0	Input capture register*2	Input capture at rising edge
			1		Input capture at falling edge
		1	X		Input capture at both edges
	1	X	X		Capture input source is channel 1/count clock Input capture at TCNT_1 count-up/count-down

[Legend]

X: Don't care

Notes: 1. After power-on reset, 0 is output until TIOR is set.

2. When the BFB bit in TMDR_0 is set to 1 and TGRD_0 is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 27.13 TIOR_1 (Channel 1)

				Description	
Bit 7 IOB3	**Bit 6 IOB2**	**Bit 5 IOB1**	**Bit 4 IOB0**	**TGRB_1 Function**	**TIOC1B Pin Function**
0	0	0	0	Output compare register	Output retained*
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	0	0	0	Input capture register	Input capture at rising edge
			1		Input capture at falling edge
		1	X		Input capture at both edges
	1	X	X		Input capture at generation of TGRC_0 compare match/input capture

[Legend]

X: Don't care

Note: * After power-on reset, 0 is output until TIOR is set.

Table 27.14 TIOR_2 (Channel 2)

				Description	
Bit 7 IOB3	**Bit 6 IOB2**	**Bit 5 IOB1**	**Bit 4 IOB0**	**TGRB_2 Function**	**TIOC2B Pin Function**
0	0	0	0	Output compare register	Output retained*
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	X	0	0	Input capture register	Input capture at rising edge
			1		Input capture at falling edge
		1	X		Input capture at both edges

[Legend]

X: Don't care

Note: * After power-on reset, 0 is output until TIOR is set.

Table 27.15 TIORH_3 (Channel 3)

				Description	
Bit 7 IOB3	**Bit 6 IOB2**	**Bit 5 IOB1**	**Bit 4 IOB0**	**TGRB_3 Function**	**TIOC3B Pin Function**
0	0	0	0	Output compare register	Output retained*
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	X	0	0	Input capture register	Input capture at rising edge
			1		Input capture at falling edge
		1	X		Input capture at both edges

[Legend]

X: Don't care

Note: * After power-on reset, 0 is output until TIOR is set.

Table 27.16 TIORL_3 (Channel 3)

				Description	
Bit 7 IOD3	**Bit 6 IOD2**	**Bit 5 IOD1**	**Bit 4 IOD0**	**TGRD_3 Function**	**TIOC3D Pin Function**
0	0	0	0	Output compare register*2	Output retained*1
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	X	0	0	Input capture register*2	Input capture at rising edge
			1		Input capture at falling edge
		1	X		Input capture at both edges

[Legend]

X: Don't care

Notes: 1. After power-on reset, 0 is output until TIOR is set.

2. When the BFB bit in TMDR_3 is set to 1 and TGRD_3 is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 27.17 TIORH_4 (Channel 4)

				Description	
Bit 7 IOB3	**Bit 6 IOB2**	**Bit 5 IOB1**	**Bit 4 IOB0**	**TGRB_4 Function**	**TIOC4B Pin Function**
0	0	0	0	Output compare register	Output retained*
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	X	0	0	Input capture register	Input capture at rising edge
			1		Input capture at falling edge
		1	X		Input capture at both edges

[Legend]

X: Don't care

Note: * After power-on reset, 0 is output until TIOR is set.

Table 27.18 TIORL_4 (Channel 4)

				Description	
Bit 7 IOD3	**Bit 6 IOD2**	**Bit 5 IOD1**	**Bit 4 IOD0**	**TGRD_4 Function**	**TIOC4D Pin Function**
0	0	0	0	Output compare register*2	Output retained*1
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	X	0	0	Input capture register*2	Input capture at rising edge
			1		Input capture at falling edge
		1	X		Input capture at both edges

[Legend]

X: Don't care

Notes: 1. After power-on reset, 0 is output until TIOR is set.

2. When the BFB bit in TMDR_4 is set to 1 and TGRD_4 is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 27.19 TIORH_0 (Channel 0)

				Description	
Bit 3 IOA3	**Bit 2 IOA2**	**Bit 1 IOA1**	**Bit 0 IOA0**	**TGRA_0 Function**	**TIOC0A Pin Function**
0	0	0	0	Output compare register	Output retained*
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	0	0	0	Input capture register	Input capture at rising edge
			1		Input capture at falling edge
		1	X		Input capture at both edges
	1	X	X		Capture input source is channel 1/count clock Input capture at TCNT_1 count-up/count-down

[Legend]

X: Don't care

Note: * After power-on reset, 0 is output until TIOR is set.

Table 27.20 TIORL_0 (Channel 0)

				Description	
Bit 3 IOC3	**Bit 2 IOC2**	**Bit 1 IOC1**	**Bit 0 IOC0**	**TGRC_0 Function**	**TIOC0C Pin Function**
0	0	0	0	Output compare register*2	Output retained*1
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	0	0	0	Input capture register*2	Input capture at rising edge
			1		Input capture at falling edge
		1	X		Input capture at both edges
	1	X	X		Capture input source is channel 1/count clock Input capture at TCNT_1 count-up/count-down

[Legend]

X: Don't care

Notes: 1. After power-on reset, 0 is output until TIOR is set.

2. When the BFA bit in TMDR_0 is set to 1 and TGRC_0 is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 27.21 TIOR_1 (Channel 1)

				Description	
Bit 3 IOA3	**Bit 2 IOA2**	**Bit 1 IOA1**	**Bit 0 IOA0**	**TGRA_1 Function**	**TIOC1A Pin Function**
0	0	0	0	Output compare register	Output retained*
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	0	0	0	Input capture register	Input capture at rising edge
			1		Input capture at falling edge
		1	X		Input capture at both edges
	1	X	X		Input capture at generation of channel 0/TGRA_0 compare match/input capture

[Legend]

X: Don't care

Note: * After power-on reset, 0 is output until TIOR is set.

Table 27.22 TIOR_2 (Channel 2)

				Description	
Bit 3 IOA3	**Bit 2 IOA2**	**Bit 1 IOA1**	**Bit 0 IOA0**	**TGRA_2 Function**	**TIOC2A Pin Function**
0	0	0	0	Output compare register	Output retained*
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	X	0	0	Input capture register	Input capture at rising edge
			1		Input capture at falling edge
		1	X		Input capture at both edges

[Legend]

X: Don't care

Note: * After power-on reset, 0 is output until TIOR is set.

Table 27.23 TIORH_3 (Channel 3)

				Description	
Bit 3 IOA3	**Bit 2 IOA2**	**Bit 1 IOA1**	**Bit 0 IOA0**	**TGRA_3 Function**	**TIOC3A Pin Function**
0	0	0	0	Output compare register	Output retained*
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	X	0	0	Input capture register	Input capture at rising edge
			1		Input capture at falling edge
		1	X		Input capture at both edges

[Legend]

X: Don't care

Note: * After power-on reset, 0 is output until TIOR is set.

Table 27.24 TIORL_3 (Channel 3)

				Description	
Bit 3 IOC3	**Bit 2 IOC2**	**Bit 1 IOC1**	**Bit 0 IOC0**	**TGRC_3 Function**	**TIOC3C Pin Function**
0	0	0	0	Output compare register*2	Output retained*1
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	X	0	0	Input capture register*2	Input capture at rising edge
			1		Input capture at falling edge
		1	X		Input capture at both edges

[Legend]

X: Don't care

Notes: 1. After power-on reset, 0 is output until TIOR is set.

2. When the BFA bit in TMDR_3 is set to 1 and TGRC_3 is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 27.25 TIORH_4 (Channel 4)

				Description	
Bit 3 IOA3	**Bit 2 IOA2**	**Bit 1 IOA1**	**Bit 0 IOA0**	**TGRA_4 Function**	**TIOC4A Pin Function**
0	0	0	0	Output compare register	Output retained*
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	X	0	0	Input capture register	Input capture at rising edge
			1		Input capture at falling edge
		1	X		Input capture at both edges

[Legend]

X: Don't care

Note: * After power-on reset, 0 is output until TIOR is set.

Table 27.26 TIORL_4 (Channel 4)

				Description	
Bit 3 IOC3	**Bit 2 IOC2**	**Bit 1 IOC1**	**Bit 0 IOC0**	**TGRC_4 Function**	**TIOC4C Pin Function**
0	0	0	0	Output compare register*2	Output retained*1
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	X	0	0	Input capture register*2	Input capture at rising edge
			1		Input capture at falling edge
		1	X		Input capture at both edges

[Legend]

X: Don't care

Notes: 1. After power-on reset, 0 is output until TIOR is set.

2. When the BFA bit in TMDR_4 is set to 1 and TGRC_4 is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

27.3.4 Timer Interrupt Enable Register (TIER)

The TIER registers are 8-bit readable/writable registers that control enabling or disabling of interrupt requests for each channel. This module has six TIER registers, two for channel 0 and one each for channels 1 to 4.

- TIER_0, TIER_1, TIER_2, TIER_3, TIER_4

Bit:	7	6	5	4	3	2	1	0
	TTGE	TTGE2	TCIEU	TCIEV	TGIED	TGIEC	TGIEB	TGIEA
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	TTGE	0	R/W	A/D Converter Start Request Enable Enables or disables generation of A/D converter start requests by TGRA input capture/compare match. 0: A/D converter start request generation disabled 1: A/D converter start request generation enabled
6	TTGE2	0	R/W	A/D Converter Start Request Enable 2 Enables or disables generation of A/D converter start requests by TCNT_4 underflow (trough) in complementary PWM mode. In channels 0 to 3, bit 6 is reserved. It is always read as 0 and the write value should always be 0. 0: A/D converter start request generation by TCNT_4 underflow (trough) disabled 1: A/D converter start request generation by TCNT_4 underflow (trough) enabled
5	TCIEU	0	R/W	Underflow Interrupt Enable Enables or disables interrupt requests (TCIU) by the TCFU flag when the TCFU flag in TSR is set to 1 in channels 1 and 2. In channels 0, 3, and 4, bit 5 is reserved. It is always read as 0 and the write value should always be 0. 0: Interrupt requests (TCIU) by TCFU disabled 1: Interrupt requests (TCIU) by TCFU enabled

Bit	Bit Name	Initial Value	R/W	Description
4	TCIEV	0	R/W	Overflow Interrupt Enable Enables or disables interrupt requests (TCIV) by the TCFV flag when the TCFV flag in TSR is set to 1. 0: Interrupt requests (TCIV) by TCFV disabled 1: Interrupt requests (TCIV) by TCFV enabled
3	TGIED	0	R/W	TGR Interrupt Enable D Enables or disables interrupt requests (TGID) by the TGFD bit when the TGFD bit in TSR is set to 1 in channels 0, 3, and 4. In channels 1 and 2, bit 3 is reserved. It is always read as 0 and the write value should always be 0. 0: Interrupt requests (TGID) by TGFD bit disabled 1: Interrupt requests (TGID) by TGFD bit enabled
2	TGIEC	0	R/W	TGR Interrupt Enable C Enables or disables interrupt requests (TGIC) by the TGFC bit when the TGFC bit in TSR is set to 1 in channels 0, 3, and 4. In channels 1 and 2, bit 2 is reserved. It is always read as 0 and the write value should always be 0. 0: Interrupt requests (TGIC) by TGFC bit disabled 1: Interrupt requests (TGIC) by TGFC bit enabled
1	TGIEB	0	R/W	TGR Interrupt Enable B Enables or disables interrupt requests (TGIB) by the TGFB bit when the TGFB bit in TSR is set to 1. 0: Interrupt requests (TGIB) by TGFB bit disabled 1: Interrupt requests (TGIB) by TGFB bit enabled
0	TGIEA	0	R/W	TGR Interrupt Enable A Enables or disables interrupt requests (TGIA) by the TGFA bit when the TGFA bit in TSR is set to 1. 0: Interrupt requests (TGIA) by TGFA bit disabled 1: Interrupt requests (TGIA) by TGFA bit enabled

- TIER2_0

Bit:	7	6	5	4	3	2	1	0
	TTGE2	-	-	-	-	-	TGIEF	TGIEE
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	TTGE2	0	R/W	A/D Converter Start Request Enable 2 Enables or disables generation of A/D converter start requests by compare match between TCNT_0 and TGRE_0. 0: A/D converter start request generation by compare match between TCNT_0 and TGRE_0 disabled 1: A/D converter start request generation by compare match between TCNT_0 and TGRE_0 enabled
6 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	TGIEF	0	R/W	TGR Interrupt Enable F Enables or disables interrupt requests by compare match between TCNT_0 and TGRF_0. 0: Interrupt requests (TGIF) by TGFE bit disabled 1: Interrupt requests (TGIF) by TGFE bit enabled
0	TGIEE	0	R/W	TGR Interrupt Enable E Enables or disables interrupt requests by compare match between TCNT_0 and TGRE_0. 0: Interrupt requests (TGIE) by TGEE bit disabled 1: Interrupt requests (TGIE) by TGEE bit enabled

27.3.5 Timer Status Register (TSR)

The TSR registers are 8-bit readable/writable registers that indicate the status of each channel. This module has six TSR registers, two for channel 0 and one each for channels 1 to 4.

- TSR_0, TSR_1, TSR_2, TSR_3, TSR_4

Bit:	7	6	5	4	3	2	1	0
	TCFD	-	TCFU	TCFV	TGFD	TGFC	TGFB	TGFA
Initial value:	1	1	0	0	0	0	0	0
R/W:	R	R	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1

Note: 1. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

Bit	Bit Name	Initial Value	R/W	Description
7	TCFD	1	R	Count Direction Flag Status flag that shows the direction in which TCNT counts in channels 1 to 4. In channel 0, bit 7 is reserved. It is always read as 1 and the write value should always be 1. 0: TCNT counts down 1: TCNT counts up
6	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
5	TCFU	0	R/(W)*1	Underflow Flag Status flag that indicates that TCNT underflow has occurred when channels 1 and 2 are set to phase counting mode. Only 0 can be written, for flag clearing. In channels 0, 3, and 4, bit 5 is reserved. It is always read as 0 and the write value should always be 0. [Clearing condition] • When 0 is written to TCFU after reading TCFU = 1*2 [Setting condition] • When the TCNT value underflows (changes from H'0000 to H'FFFF)

Bit	Bit Name	Initial Value	R/W	Description
4	TCFV	0	R/(W)*1	Overflow Flag Status flag that indicates that TCNT overflow has occurred. Only 0 can be written, for flag clearing. [Clearing condition] • When 0 is written to TCFV after reading TCFV = 1*2 [Setting condition] • When the TCNT value overflows (changes from H'FFFF to H'0000) In channel 4, when the TCNT_4 value underflows (changes from H'0001 to H'0000) in complementary PWM mode, this flag is also set.
3	TGFD	0	R/(W)*1	Input Capture/Output Compare Flag D Status flag that indicates the occurrence of TGRD input capture or compare match in channels 0, 3, and 4. Only 0 can be written, for flag clearing. In channels 1 and 2, bit 3 is reserved. It is always read as 0 and the write value should always be 0. [Clearing condition] • When 0 is written to TGFD after reading TGFD = 1*2 [Setting conditions] • When TCNT = TGRD and TGRD is functioning as output compare register • When TCNT value is transferred to TGRD by input capture signal and TGRD is functioning as input capture register

Bit	Bit Name	Initial Value	R/W	Description
2	TGFC	0	R/(W)*1	Input Capture/Output Compare Flag C Status flag that indicates the occurrence of TGRC input capture or compare match in channels 0, 3, and 4. Only 0 can be written, for flag clearing. In channels 1 and 2, bit 2 is reserved. It is always read as 0 and the write value should always be 0. [Clearing condition] • When 0 is written to TGFC after reading TGFC = 1*2 [Setting conditions] • When TCNT = TGRC and TGRC is functioning as output compare register • When TCNT value is transferred to TGRC by input capture signal and TGRC is functioning as input capture register
1	TGFB	0	R/(W)*1	Input Capture/Output Compare Flag B Status flag that indicates the occurrence of TGRB input capture or compare match. Only 0 can be written, for flag clearing. [Clearing condition] • When 0 is written to TGFB after reading TGFB = 1*2 [Setting conditions] • When TCNT = TGRB and TGRB is functioning as output compare register • When TCNT value is transferred to TGRB by input capture signal and TGRB is functioning as input capture register

Bit	Bit Name	Initial Value	R/W	Description
0	TGFA	0	R/(W)*1	Input Capture/Output Compare Flag A Status flag that indicates the occurrence of TGRA input capture or compare match. Only 0 can be written, for flag clearing. [Clearing conditions] • When the direct memory access controller is activated by TGIA interrupt • When 0 is written to TGFA after reading TGFA = 1*2 [Setting conditions] • When TCNT = TGRA and TGRA is functioning as output compare register • When TCNT value is transferred to TGRA by input capture signal and TGRA is functioning as input capture register

Notes: 1. Writing 0 to this bit after reading it as 1 clears the flag.
2. If the next flag is set before TGFA is cleared to 0 after reading TGFA = 1, TGFA remains 1 even when 0 is written to. In this case, read TGFA = 1 again to clear TGFA to 0.

- TSR2_0

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	TGFF	TGFE
Initial value:	1	1	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/(W)*1	R/(W)*1

Note: 1. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
5 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	TGFF	0	R/(W)*1	Compare Match Flag F Status flag that indicates the occurrence of compare match between TCNT_0 and TGRF_0. [Clearing condition] • When 0 is written to TGFF after reading TGFF = 1*2 [Setting condition] • When TCNT_0 = TGRF_0 and TGRF_0 is functioning as compare register
0	TGFE	0	R/(W)*1	Compare Match Flag E Status flag that indicates the occurrence of compare match between TCNT_0 and TGRE_0. [Clearing condition] • When 0 is written to TGFE after reading TGFE = 1*2 [Setting condition] • When TCNT_0 = TGRE_0 and TGRE_0 is functioning as compare register

Notes: 1. Writing 0 to this bit after reading it as 1 clears the flag.
2. If the next flag is set before TGFA is cleared to 0 after reading TGFA = 1, TGFA remains 1 even when 0 is written to. In this case, read TGFA = 1 again to clear TGFA to 0.

27.3.6 Timer Buffer Operation Transfer Mode Register (TBTM)

The TBTM registers are 8-bit readable/writable registers that specify the timing for transferring data from the buffer register to the timer general register in PWM mode. This module has three TBTM registers, one each for channels 0, 3, and 4.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	TTSE	TTSB	TTSA
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	TTSE	0	R/W	Timing Select E Specifies the timing for transferring data from TGRF_0 to TGRE_0 when they are used together for buffer operation. In channels 3 and 4, bit 2 is reserved. It is always read as 0 and the write value should always be 0. 0: When compare match E occurs in channel 0 1: When TCNT_0 is cleared
1	TTSB	0	R/W	Timing Select B Specifies the timing for transferring data from TGRD to TGRB in each channel when they are used together for buffer operation. 0: When compare match B occurs in each channel 1: When TCNT is cleared in each channel
0	TTSA	0	R/W	Timing Select A Specifies the timing for transferring data from TGRC to TGRA in each channel when they are used together for buffer operation. 0: When compare match A occurs in each channel 1: When TCNT is cleared in each channel

27.3.7 Timer Input Capture Control Register (TICCR)

TICCR is an 8-bit readable/writable register that specifies input capture conditions when TCNT_1 and TCNT_2 are cascaded. This module has one TICCR in channel 1.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	I2BE	I2AE	I1BE	I1AE
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	I2BE	0	R/W	Input Capture Enable Specifies whether to include the TIOC2B pin in the TGRB_1 input capture conditions. 0: Does not include the TIOC2B pin in the TGRB_1 input capture conditions 1: Includes the TIOC2B pin in the TGRB_1 input capture conditions
2	I2AE	0	R/W	Input Capture Enable Specifies whether to include the TIOC2A pin in the TGRA_1 input capture conditions. 0: Does not include the TIOC2A pin in the TGRA_1 input capture conditions 1: Includes the TIOC2A pin in the TGRA_1 input capture conditions
1	I1BE	0	R/W	Input Capture Enable Specifies whether to include the TIOC1B pin in the TGRB_2 input capture conditions. 0: Does not include the TIOC1B pin in the TGRB_2 input capture conditions 1: Includes the TIOC1B pin in the TGRB_2 input capture conditions

Bit	Bit Name	Initial Value	R/W	Description
0	I1AE	0	R/W	Input Capture Enable Specifies whether to include the TIOC1A pin in the TGRA_2 input capture conditions. 0: Does not include the TIOC1A pin in the TGRA_2 input capture conditions 1: Includes the TIOC1A pin in the TGRA_2 input capture conditions

27.3.8 Timer A/D Converter Start Request Control Register (TADCR)

TADCR is a 16-bit readable/writable register that enables or disables A/D converter start requests and specifies whether to link A/D converter start requests with interrupt skipping operation. This module has one TADCR in channel 4.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BF[1:0]		-	-	-	-	-	-	UT4AE	DT4AE	UT4BE	DT4BE	ITA3AE	ITA4VE	ITB3AE	ITB4VE
Initial value:	0	0	0	0	0	0	0	0	0	0*	0	0*	0*	0*	0*	0*
R/W:	R/W	R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: * Do not set to 1 when complementary PWM mode is not selected.

Bit	Bit Name	Initial Value	R/W	Description
15, 14	BF[1:0]	00	R/W	TADCOBRA_4/TADCOBRB_4 Transfer Timing Select Select the timing for transferring data from TADCOBRA_4 and TADCOBRB_4 to TADCORA_4 and TADCORB_4. For details, see table 27.27.
13 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	UT4AE	0	R/W	Up-Count TRG4AN Enable Enables or disables A/D converter start requests (TRG4AN) during TCNT_4 up-count operation. 0: A/D converter start requests (TRG4AN) disabled during TCNT_4 up-count operation 1: A/D converter start requests (TRG4AN) enabled during TCNT_4 up-count operation
6	DT4AE	0*	R/W	Down-Count TRG4AN Enable Enables or disables A/D converter start requests (TRG4AN) during TCNT_4 down-count operation. 0: A/D converter start requests (TRG4AN) disabled during TCNT_4 down-count operation 1: A/D converter start requests (TRG4AN) enabled during TCNT_4 down-count operation

Bit	Bit Name	Initial Value	R/W	Description
5	UT4BE	0	R/W	Up-Count TRG4BN Enable Enables or disables A/D converter start requests (TRG4BN) during TCNT_4 up-count operation. 0: A/D converter start requests (TRG4BN) disabled during TCNT_4 up-count operation 1: A/D converter start requests (TRG4BN) enabled during TCNT_4 up-count operation
4	DT4BE	0*	R/W	Down-Count TRG4BN Enable Enables or disables A/D converter start requests (TRG4BN) during TCNT_4 down-count operation. 0: A/D converter start requests (TRG4BN) disabled during TCNT_4 down-count operation 1: A/D converter start requests (TRG4BN) enabled during TCNT_4 down-count operation
3	ITA3AE	0*	R/W	TGIA_3 Interrupt Skipping Link Enable Select whether to link A/D converter start requests (TRG4AN) with TGIA_3 interrupt skipping operation. 0: Does not link with TGIA_3 interrupt skipping 1: Links with TGIA_3 interrupt skipping
2	ITA4VE	0*	R/W	TCIV_4 Interrupt Skipping Link Enable Select whether to link A/D converter start requests (TRG4AN) with TCIV_4 interrupt skipping operation. 0: Does not link with TCIV_4 interrupt skipping 1: Links with TCIV_4 interrupt skipping
1	ITB3AE	0*	R/W	TGIA_3 Interrupt Skipping Link Enable Select whether to link A/D converter start requests (TRG4BN) with TGIA_3 interrupt skipping operation. 0: Does not link with TGIA_3 interrupt skipping 1: Links with TGIA_3 interrupt skipping

Bit	Bit Name	Initial Value	R/W	Description
0	ITB4VE	0*	R/W	TCIV_4 Interrupt Skipping Link Enable Select whether to link A/D converter start requests (TRG4BN) with TCIV_4 interrupt skipping operation. 0: Does not link with TCIV_4 interrupt skipping 1: Links with TCIV_4 interrupt skipping

Notes: 1. TADCR must not be accessed in eight bits; it should always be accessed in 16 bits.
2. When interrupt skipping is disabled (the T3AEN and T4VEN bits in the timer interrupt skipping set register (TITCR) are cleared to 0 or the skipping count set bits (3ACOR and 4VCOR) in TITCR are cleared to 0), do not link A/D converter start requests with interrupt skipping operation (clear the ITA3AE, ITA4VE, ITB3AE, and ITB4VE bits in the timer A/D converter start request control register (TADCR) to 0).
3. If link with interrupt skipping is enabled while interrupt skipping is disabled, A/D converter start requests will not be issued.
* Do not set to 1 when complementary PWM mode is not selected.

Table 27.27 Setting of Transfer Timing by Bits BF1 and BF0

Bit 7	Bit 6	
BF1	**BF0**	**Description**
0	0	Does not transfer data from the cycle set buffer register to the cycle set register.
0	1	Transfers data from the cycle set buffer register to the cycle set register at the crest of the TCNT_4 count.*[1]
1	0	Transfers data from the cycle set buffer register to the cycle set register at the trough of the TCNT_4 count.*[2]
1	1	Transfers data from the cycle set buffer register to the cycle set register at the crest and trough of the TCNT_4 count.*[2]

Notes: 1. Data is transferred from the cycle set buffer register to the cycle set register when the crest of the TCNT_4 count is reached in complementary PWM mode, when compare match occurs between TCNT_3 and TGRA_3 in reset-synchronized PWM mode, or when compare match occurs between TCNT_4 and TGRA_4 in PWM mode 1 or normal operation mode.
2. These settings are prohibited when complementary PWM mode is not selected.

27.3.9 Timer A/D Converter Start Request Cycle Set Registers (TADCORA_4 and TADCORB_4)

TADCORA_4 and TADCORB_4 are 16-bit readable/writable registers. When the TCNT_4 count reaches the value in TADCORA_4 or TADCORB_4, a corresponding A/D converter start request will be issued.

TADCORA_4 and TADCORB_4 are initialized to H'FFFF.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: TADCOBRA_4 and TADCOBRB_4 must not be accessed in eight bits; they should always be accessed in 16 bits.

27.3.10 Timer A/D Converter Start Request Cycle Set Buffer Registers (TADCOBRA_4 and TADCOBRB_4)

TADCOBRA_4 and TADCOBRB_4 are 16-bit readable/writable registers. When the crest or trough of the TCNT_4 count is reached, these register values are transferred to TADCORA_4 and TADCORB_4, respectively.

TADCOBRA_4 and TADCOBRB_4 are initialized to H'FFFF.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: TADCORA_4 and TADCORB_4 must not be accessed in eight bits; they should always be accessed in 16 bits.

27.3.11 Timer Counter (TCNT)

The TCNT counters are 16-bit readable/writable counters. This module has five TCNT counters, one each for channels 0 to 4.

The TCNT counters must not be accessed in eight bits; they should always be accessed in 16 bits.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: The TCNT counters must not be accessed in eight bits; they should always be accessed in 16 bits.

27.3.12 Timer General Register (TGR)

The TGR registers are 16-bit readable/writable registers. This module has eighteen TGR registers, six for channel 0, two each for channels 1 and 2, four each for channels 3 and 4.

TGRA, TGRB, TGRC, and TGRD function as either output compare or input capture registers. TGRC and TGRD for channels 0, 3, and 4 can also be designated for operation as buffer registers. TGR buffer register combinations are TGRA and TGRC, and TGRB and TGRD.

TGRE_0 and TGRF_0 function as compare registers. When the TCNT_0 count matches the TGRE_0 value, an A/D converter start request can be issued. TGRF can also be designated for operation as a buffer register. TGR buffer register combination is TGRE and TGRF.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: The TGR registers must not be accessed in eight bits; they should always be accessed in 16 bits.
TGR registers are initialized to H'FFFF.

27.3.13 Timer Start Register (TSTR)

TSTR is an 8-bit readable/writable register that selects operation/stoppage of TCNT for channels 0 to 4.

When setting the operating mode in TMDR or setting the count clock in TCR, first stop the TCNT counter.

Bit:	7	6	5	4	3	2	1	0
	CST4	CST3	-	-	-	CST2	CST1	CST0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	CST4	0	R/W	Counter Start 4 and 3
6	CST3	0	R/W	These bits select operation or stoppage for TCNT. If 0 is written to the CST bit during operation with the TIOC pin designated for output, the counter stops but the TIOC pin output compare output level is retained. If TIOR is written to when the CST bit is cleared to 0, the pin output level will be changed to the set initial output value. 0: TCNT_4 and TCNT_3 count operation is stopped 1: TCNT_4 and TCNT_3 performs count operation
5 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	CST2	0	R/W	Counter Start 2 to 0
1	CST1	0	R/W	These bits select operation or stoppage for TCNT.
0	CST0	0	R/W	If 0 is written to the CST bit during operation with the TIOC pin designated for output, the counter stops but the TIOC pin output compare output level is retained. If TIOR is written to when the CST bit is cleared to 0, the pin output level will be changed to the set initial output value. 0: TCNT_2 to TCNT_0 count operation is stopped 1: TCNT_2 to TCNT_0 performs count operation

27.3.14 Timer Synchronous Register (TSYR)

TSYR is an 8-bit readable/writable register that selects independent operation or synchronous operation for the channel 0 to 4 TCNT counters. A channel performs synchronous operation when the corresponding bit in TSYR is set to 1.

Bit:	7	6	5	4	3	2	1	0
	SYNC4	SYNC3	-	-	-	SYNC2	SYNC1	SYNC0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	SYNC4	0	R/W	Timer Synchronous operation 4 and 3
6	SYNC3	0	R/W	These bits are used to select whether operation is independent of or synchronized with other channels. When synchronous operation is selected, the TCNT synchronous presetting of multiple channels, and synchronous clearing by counter clearing on another channel, are possible. To set synchronous operation, the SYNC bits for at least two channels must be set to 1. To set synchronous clearing, in addition to the SYNC bit , the TCNT clearing source must also be set by means of bits CCLR0 to CCLR2 in TCR. 0: TCNT_4 and TCNT_3 operate independently (TCNT presetting/clearing is unrelated to other channels) 1: TCNT_4 and TCNT_3 performs synchronous operation TCNT synchronous presetting/synchronous clearing is possible
5 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2	SYNC2	0	R/W	Timer Synchronous operation 2 to 0
1	SYNC1	0	R/W	These bits are used to select whether operation is independent of or synchronized with other channels.
0	SYNC0	0	R/W	When synchronous operation is selected, the TCNT synchronous presetting of multiple channels, and synchronous clearing by counter clearing on another channel, are possible. To set synchronous operation, the SYNC bits for at least two channels must be set to 1. To set synchronous clearing, in addition to the SYNC bit, the TCNT clearing source must also be set by means of bits CCLR0 to CCLR2 in TCR. 0: TCNT_2 to TCNT_0 operates independently (TCNT presetting /clearing is unrelated to other channels) 1: TCNT_2 to TCNT_0 performs synchronous operation TCNT synchronous presetting/synchronous clearing is possible

27.3.15 Timer Read/Write Enable Register (TRWER)

TRWER is an 8-bit readable/writable register that enables or disables access to the registers and counters which have write-protection capability against accidental modification in channels 3 and 4.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	RWE
Initial value:	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	RWE	1	R/W	Read/Write Enable Enables or disables access to the registers which have write-protection capability against accidental modification. 0: Disables read/write access to the registers 1: Enables read/write access to the registers [Clearing condition] • When 0 is written to the RWE bit after reading RWE = 1

- Registers and counters having write-protection capability against accidental modification
  22 registers: TCR_3, TCR_4, TMDR_3, TMDR_4, TIORH_3, TIORH_4, TIORL_3, TIORL_4, TIER_3, TIER_4, TGRA_3, TGRA_4, TGRB_3, TGRB_4, TOER, TOCR1, TOCR2, TGCR, TCDR, TDDR, TCNT_3, and TCNT4.

27.3.16 Timer Output Master Enable Register (TOER)

TOER is an 8-bit readable/writable register that enables/disables output settings for output pins TIOC4D, TIOC4C, TIOC3D, TIOC4B, TIOC4A, and TIOC3B. These pins do not output correctly if the TOER bits have not been set. Set TOER of CH3 and CH4 prior to setting TIOR of CH3 and CH4.

Bit:	7	6	5	4	3	2	1	0
	-	-	OE4D	OE4C	OE3D	OE4B	OE4A	OE3B
Initial value:	1	1	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
5	OE4D	0	R/W	Master Enable TIOC4D This bit enables/disables the TIOC4D pin output for this module. 0: Output for this module is disabled (inactive level)* 1: Output for this module is enabled
4	OE4C	0	R/W	Master Enable TIOC4C This bit enables/disables the TIOC4C pin output for this module. 0: Output for this module is disabled (inactive level)* 1: Output for this module is enabled
3	OE3D	0	R/W	Master Enable TIOC3D This bit enables/disables the TIOC3D pin output for this module. 0: Output for this module is disabled (inactive level)* 1: Output for this module is enabled
2	OE4B	0	R/W	Master Enable TIOC4B This bit enables/disables the TIOC4B pin output for this module. 0: Output for this module is disabled (inactive level)* 1: Output for this module is enabled

Bit	Bit Name	Initial Value	R/W	Description
1	OE4A	0	R/W	Master Enable TIOC4A This bit enables/disables the TIOC4A pin output for this module. 0: Output for this module is disabled (inactive level)* 1: Output for this module is enabled
0	OE3B	0	R/W	Master Enable TIOC3B This bit enables/disables the TIOC3B pin output for this module. 0: Output for this module is disabled (inactive level)* 1: Output for this module is enabled

Note: * The inactive level is determined by the settings in timer output control registers 1 and 2 (TOCR1 and TOCR2). For details, refer to section 27.3.17, Timer Output Control Register 1 (TOCR1), and section 27.3.18, Timer Output Control Register 2 (TOCR2). Set these bits to 1 to enable output for this module in other than complementary PWM or reset-synchronized PWM mode. When these bits are set to 0, low level is output.

27.3.17 Timer Output Control Register 1 (TOCR1)

TOCR1 is an 8-bit readable/writable register that enables/disables PWM synchronized toggle output in complementary PWM mode/reset synchronized PWM mode, and controls output level inversion of PWM output.

Bit:	7	6	5	4	3	2	1	0
	-	PSYE	-	-	TOCL	TOCS	OLSN	OLSP
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R	R/(W)*3	R/W	R/W	R/W

Bit	Bit Name	Initial value	R/W	Description
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	PSYE	0	R/W	PWM Synchronous Output Enable This bit selects the enable/disable of toggle output synchronized with the PWM period. 0: Toggle output is disabled 1: Toggle output is enabled
5, 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial value	R/W	Description
3	TOCL	0	R/(W)*3	TOC Register Write Protection*1 This bit selects the enable/disable of write access to the TOCS, OLSN, and OLSP bits in TOCR1. 0: Write access to the TOCS, OLSN, and OLSP bits is enabled 1: Write access to the TOCS, OLSN, and OLSP bits is disabled
2	TOCS	0	R/W	TOC Select This bit selects either the TOCR1 or TOCR2 setting to be used for the output level in complementary PWM mode and reset-synchronized PWM mode. 0: TOCR1 setting is selected 1: TOCR2 setting is selected
1	OLSN	0	R/W	Output Level Select N*2 This bit selects the negative phase output level in reset-synchronized PWM mode/complementary PWM mode. See table 27.28.
0	OLSP	0	R/W	Output Level Select P*2 This bit selects the positive phase output level in reset-synchronized PWM mode/complementary PWM mode. See table 27.29.

Notes: 1. Setting the TOCL bit to 1 prevents accidental modification when the CPU goes out of control.
2. Clearing the TOCS0 bit to 0 makes this bit setting valid.
3. After power-on reset, 1 can be written only once. After 1 has been written, 0 cannot be written.

Table 27.28 Output Level Select Function

Bit 1	Function			
			Compare Match Output	
OLSN	Initial Output	Active Level	Up Count	Down Count
0	High level	Low level	High level	Low level
1	Low level	High level	Low level	High level

Note: The negative phase waveform initial output value changes to active level after elapse of the dead time after count start.

Table 27.29 Output Level Select Function

Bit 0	Function			
			Compare Match Output	
OLSP	Initial Output	Active Level	Up Count	Down Count
0	High level	Low level	Low level	High level
1	Low level	High level	High level	Low level

Figure 27.2 shows an example of complementary PWM mode output (1 phase) when OLSN = 1, OLSP = 1.

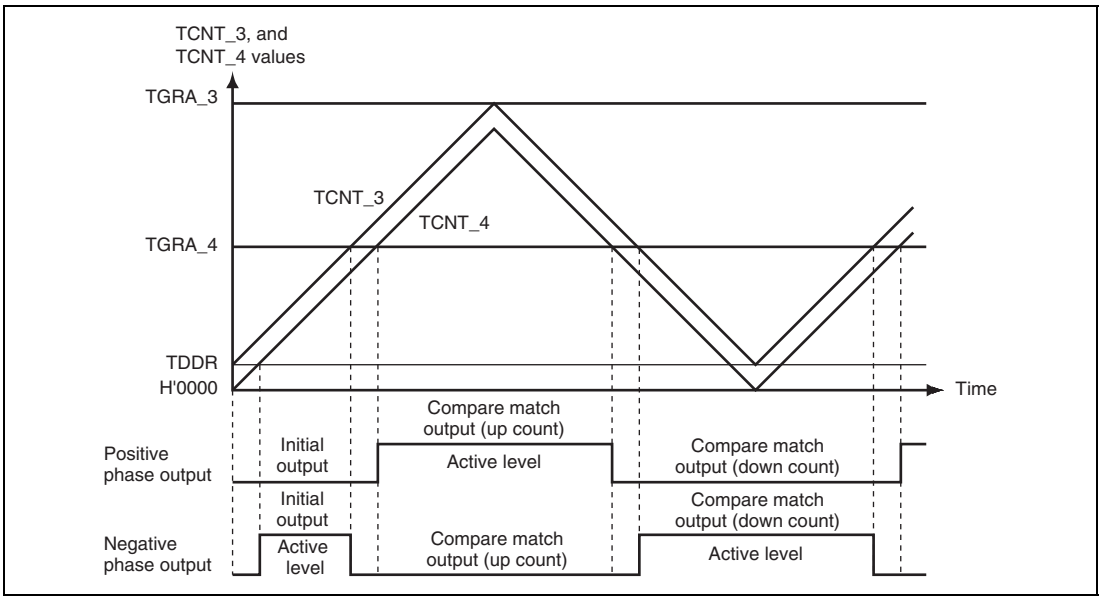


Figure 27.2 Complementary PWM Mode Output Level Example

27.3.18 Timer Output Control Register 2 (TOCR2)

TOCR2 is an 8-bit readable/writable register that controls output level inversion of PWM output in complementary PWM mode and reset-synchronized PWM mode.

Bit:	7	6	5	4	3	2	1	0
	BF[1:0]		OLS3N	OLS3P	OLS2N	OLS2P	OLS1N	OLS1P
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial value	R/W	Description
7, 6	BF[1:0]	00	R/W	TOLBR Buffer Transfer Timing Select These bits select the timing for transferring data from TOLBR to TOCR2. For details, see table 27.30.
5	OLS3N	0	R/W	Output Level Select 3N* This bit selects the output level on TIOC4D in reset-synchronized PWM mode/complementary PWM mode. See table 27.31.
4	OLS3P	0	R/W	Output Level Select 3P* This bit selects the output level on TIOC4B in reset-synchronized PWM mode/complementary PWM mode. See table 27.32.
3	OLS2N	0	R/W	Output Level Select 2N* This bit selects the output level on TIOC4C in reset-synchronized PWM mode/complementary PWM mode. See table 27.33.
2	OLS2P	0	R/W	Output Level Select 2P* This bit selects the output level on TIOC4A in reset-synchronized PWM mode/complementary PWM mode. See table 27.34.
1	OLS1N	0	R/W	Output Level Select 1N* This bit selects the output level on TIOC3D in reset-synchronized PWM mode/complementary PWM mode. See table 27.35.

Bit	Bit Name	Initial value	R/W	Description
0	OLS1P	0	R/W	Output Level Select 1P* This bit selects the output level on TIOC3B in reset-synchronized PWM mode/complementary PWM mode. See table 27.36.

Note: * Setting the TOCS bit in TOCR1 to 1 makes this bit setting valid.

Table 27.30 Setting of Bits BF1 and BF0

Bit 7	Bit 6	Description	
BF1	**BF0**	**Complementary PWM Mode**	**Reset-Synchronized PWM Mode**
0	0	Does not transfer data from the buffer register (TOLBR) to TOCR2.	Does not transfer data from the buffer register (TOLBR) to TOCR2.
0	1	Transfers data from the buffer register (TOLBR) to TOCR2 at the crest of the TCNT_4 count.	Transfers data from the buffer register (TOLBR) to TOCR2 when TCNT_3/TCNT_4 is cleared
1	0	Transfers data from the buffer register (TOLBR) to TOCR2 at the trough of the TCNT_4 count.	Setting prohibited
1	1	Transfers data from the buffer register (TOLBR) to TOCR2 at the crest and trough of the TCNT_4 count.	Setting prohibited

Table 27.31 TIOC4D Output Level Select Function

Bit 5	Function			
			Compare Match Output	
OLS3N	**Initial Output**	**Active Level**	**Up Count**	**Down Count**
0	High level	Low level	High level	Low level
1	Low level	High level	Low level	High level

Note: The negative phase waveform initial output value changes to the active level after elapse of the dead time after count start.

Table 27.32 TIOC4B Output Level Select Function

Bit 4	Function			
			Compare Match Output	
OLS3P	Initial Output	Active Level	Up Count	Down Count
0	High level	Low level	Low level	High level
1	Low level	High level	High level	Low level

Table 27.33 TIOC4C Output Level Select Function

Bit 3	Function			
			Compare Match Output	
OLS2N	Initial Output	Active Level	Up Count	Down Count
0	High level	Low level	High level	Low level
1	Low level	High level	Low level	High level

Note: The negative phase waveform initial output value changes to the active level after elapse of the dead time after count start.

Table 27.34 TIOC4A Output Level Select Function

Bit 2	Function			
			Compare Match Output	
OLS2P	Initial Output	Active Level	Up Count	Down Count
0	High level	Low level	Low level	High level
1	Low level	High level	High level	Low level

Table 27.35 TIOC3D Output Level Select Function

Bit 1	Function			
			Compare Match Output	
OLS1N	Initial Output	Active Level	Up Count	Down Count
0	High level	Low level	High level	Low level
1	Low level	High level	Low level	High level

Note: The negative phase waveform initial output value changes to the active level after elapse of the dead time after count start.

Table 27.36 TIOC4B Output Level Select Function

Bit 0	Function			
			Compare Match Output	
OLS1P	Initial Output	Active Level	Up Count	Down Count
0	High level	Low level	Low level	High level
1	Low level	High level	High level	Low level

27.3.19 Timer Output Level Buffer Register (TOLBR)

TOLBR is an 8-bit readable/writable register that functions as a buffer for TOCR2 and specifies the PWM output level in complementary PWM mode and reset-synchronized PWM mode.

Bit:	7	6	5	4	3	2	1	0
	-	-	OLS3N	OLS3P	OLS2N	OLS2P	OLS1N	OLS1P
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial value	R/W	Description
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	OLS3N	0	R/W	Specifies the buffer value to be transferred to the OLS3N bit in TOCR2.
4	OLS3P	0	R/W	Specifies the buffer value to be transferred to the OLS3P bit in TOCR2.
3	OLS2N	0	R/W	Specifies the buffer value to be transferred to the OLS2N bit in TOCR2.
2	OLS2P	0	R/W	Specifies the buffer value to be transferred to the OLS2P bit in TOCR2.
1	OLS1N	0	R/W	Specifies the buffer value to be transferred to the OLS1N bit in TOCR2.
0	OLS1P	0	R/W	Specifies the buffer value to be transferred to the OLS1P bit in TOCR2.

Figure 27.3 shows an example of the PWM output level setting procedure in buffer operation.

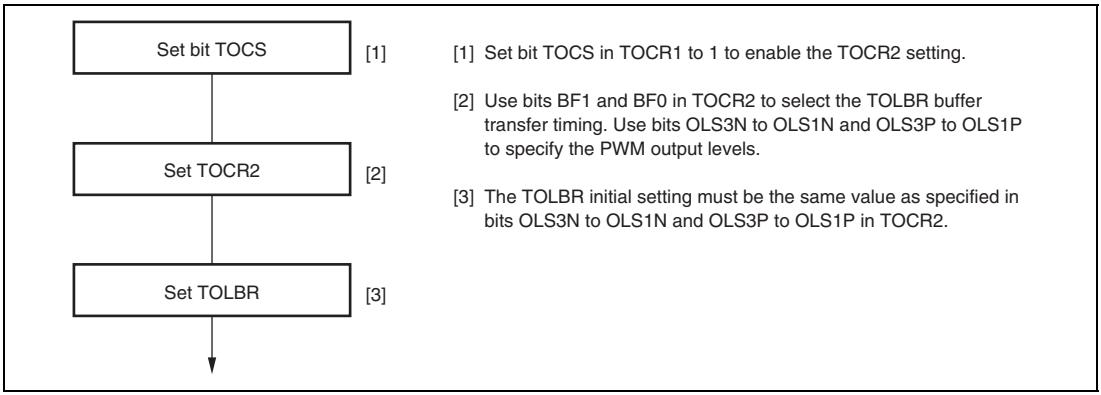


Figure 27.3 PWM Output Level Setting Procedure in Buffer Operation

27.3.20 Timer Gate Control Register (TGCR)

TGCR is an 8-bit readable/writable register that controls the waveform output necessary for brushless DC motor control in reset-synchronized PWM mode/complementary PWM mode. These register settings are ineffective for anything other than complementary PWM mode/reset-synchronized PWM mode.

Bit:	7	6	5	4	3	2	1	0
	-	BDC	N	P	FB	WF	VF	UF
Initial value:	1	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial value	R/W	Description
7	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
6	BDC	0	R/W	Brushless DC Motor This bit selects whether to make the functions of this register (TGCR) effective or ineffective. 0: Ordinary output 1: Functions of this register are made effective

Bit	Bit Name	Initial value	R/W	Description
5	N	0	R/W	Negative Phase Output (N) Control This bit selects whether the level output or the reset-synchronized PWM/complementary PWM output while the reverse pins (TIOC3D, TIOC4C, and TIOC4D) are output. 0: Level output 1: Reset synchronized PWM/complementary PWM output
4	P	0	R/W	Positive Phase Output (P) Control This bit selects whether the level output or the reset-synchronized PWM/complementary PWM output while the positive pin (TIOC3B, TIOC4A, and TIOC4B) are output. 0: Level output 1: Reset synchronized PWM/complementary PWM output
3	FB	0	R/W	External Feedback Signal Enable This bit selects whether the switching of the output of the positive/negative phase is carried out automatically with channel-0 TGRA, TGRB, TGRC input capture signals or by writing 0 or 1 to bits 2 to 0 in TGCR. 0: Output switching is external input (Input sources are channel 0 TGRA, TGRB, TGRC input capture signal) 1: Output switching is carried out by software (setting values of UF, VF, and WF in TGCR).
2	WF	0	R/W	Output Phase Switch 2 to 0 These bits set the positive phase/negative phase output phase on or off state. The setting of these bits is valid only when the FB bit in this register is set to 1. In this case, the setting of bits 2 to 0 is a substitute for external input. See table 27.37.
1	VF	0	R/W	
0	UF	0	R/W	

Table 27.37 Output level Select Function

			Function					
Bit 2	**Bit 1**	**Bit 0**	**TIOC3B**	**TIOC4A**	**TIOC4B**	**TIOC3D**	**TIOC4C**	**TIOC4D**
WF	**VF**	**UF**	**U Phase**	**V Phase**	**W Phase**	**U Phase**	**V Phase**	**W Phase**
0	0	0	OFF	OFF	OFF	OFF	OFF	OFF
		1	ON	OFF	OFF	OFF	OFF	ON
	1	0	OFF	ON	OFF	ON	OFF	OFF
		1	OFF	ON	OFF	OFF	OFF	ON
1	0	0	OFF	OFF	ON	OFF	ON	OFF
		1	ON	OFF	OFF	OFF	ON	OFF
	1	0	OFF	OFF	ON	ON	OFF	OFF
		1	OFF	OFF	OFF	OFF	OFF	OFF

27.3.21 Timer Subcounter (TCNTS)

TCNTS is a 16-bit read-only counter that is used only in complementary PWM mode.

The initial value of TCNTS is H'0000.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Note: Accessing the TCNTS in 8-bit units is prohibited. Always access in 16-bit units.

27.3.22 Timer Dead Time Data Register (TDDR)

TDDR is a 16-bit register, used only in complementary PWM mode that specifies the TCNT_3 and TCNT_4 counter offset values. In complementary PWM mode, when the TCNT_3 and TCNT_4 counters are cleared and then restarted, the TDDR register value is loaded into the TCNT_3 counter and the count operation starts.

The initial value of TDDR is H'FFFF.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: Accessing the TDDR in 8-bit units is prohibited. Always access in 16-bit units.

27.3.23 Timer Cycle Data Register (TCDR)

TCDR is a 16-bit register used only in complementary PWM mode. Set half the PWM carrier sync value as the TCDR register value. This register is constantly compared with the TCNTS counter in complementary PWM mode, and when a match occurs, the TCNTS counter switches direction (decrement to increment).

The initial value of TCDR is H'FFFF.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: Accessing the TCDR in 8-bit units is prohibited. Always access in 16-bit units.

27.3.24 Timer Cycle Buffer Register (TCBR)

TCBR is a 16-bit register used only in complementary PWM mode. It functions as a buffer register for the TCDR register. The TCBR register values are transferred to the TCDR register with the transfer timing set in the TMDR register. The initial value of TCBR is H'FFFF.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: Accessing the TCBR in 8-bit units is prohibited. Always access in 16-bit units.

27.3.25 Timer Interrupt Skipping Set Register (TITCR)

TITCR is an 8-bit readable/writable register that enables or disables interrupt skipping and specifies the interrupt skipping count. This module has one TITCR.

Bit:	7	6	5	4	3	2	1	0
	T3AEN	3ACOR[2:0]			T4VEN	4VCOR[2:0]		
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial value	R/W	Description
7	T3AEN	0	R/W	T3AEN Enables or disables TGIA_3 interrupt skipping. 0: TGIA_3 interrupt skipping disabled 1: TGIA_3 interrupt skipping enabled
6 to 4	3ACOR[2:0]	000	R/W	These bits specify the TGIA_3 interrupt skipping count within the range from 0 to 7.* For details, see table 27.38.
3	T4VEN	0	R/W	T4VEN Enables or disables TCIV_4 interrupt skipping. 0: TCIV_4 interrupt skipping disabled 1: TCIV_4 interrupt skipping enabled

Bit	Bit Name	Initial value	R/W	Description
2 to 0	4VCOR[2:0]	000	R/W	These bits specify the TCIV_4 interrupt skipping count within the range from 0 to 7.* For details, see table 27.39.

Note: * When 0 is specified for the interrupt skipping count, no interrupt skipping will be performed. Before changing the interrupt skipping count, be sure to clear the T3AEN and T4VEN bits to 0 to clear the skipping counter (TICNT).

Table 27.38 Setting of Interrupt Skipping Count by Bits 3ACOR2 to 3ACOR0

Bit 6	Bit 5	Bit 4	
3ACOR2	3ACOR1	3ACOR0	Description
0	0	0	Does not skip TGIA_3 interrupts.
0	0	1	Sets the TGIA_3 interrupt skipping count to 1.
0	1	0	Sets the TGIA_3 interrupt skipping count to 2.
0	1	1	Sets the TGIA_3 interrupt skipping count to 3.
1	0	0	Sets the TGIA_3 interrupt skipping count to 4.
1	0	1	Sets the TGIA_3 interrupt skipping count to 5.
1	1	0	Sets the TGIA_3 interrupt skipping count to 6.
1	1	1	Sets the TGIA_3 interrupt skipping count to 7.

Table 27.39 Setting of Interrupt Skipping Count by Bits 4VCOR2 to 4VCOR0

Bit 2	Bit 1	Bit 0	
4VCOR2	4VCOR1	4VCOR0	Description
0	0	0	Does not skip TCIV_4 interrupts.
0	0	1	Sets the TCIV_4 interrupt skipping count to 1.
0	1	0	Sets the TCIV_4 interrupt skipping count to 2.
0	1	1	Sets the TCIV_4 interrupt skipping count to 3.
1	0	0	Sets the TCIV_4 interrupt skipping count to 4.
1	0	1	Sets the TCIV_4 interrupt skipping count to 5.
1	1	0	Sets the TCIV_4 interrupt skipping count to 6.
1	1	1	Sets the TCIV_4 interrupt skipping count to 7.

27.3.26 Timer Interrupt Skipping Counter (TITCNT)

TITCNT is an 8-bit readable/writable counter. This module has one TITCNT. TITCNT retains its value even after stopping the count operation of TCNT_3 and TCNT_4.

Bit:	7	6	5	4	3	2	1	0
	-	3ACNT[2:0]			-	4VCNT[2:0]		
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	—	0	R	Reserved This bit is always read as 0.
6 to 4	3ACNT[2:0]	000	R	TGIA_3 Interrupt Counter While the T3AEN bit in TITCR is set to 1, the count in these bits is incremented every time a TGIA_3 interrupt occurs. [Clearing conditions] • When the 3ACNT2 to 3ACNT0 value in TITCNT matches the 3ACOR2 to 3ACOR0 value in TITCR • When the T3AEN bit in TITCR is cleared to 0 • When the 3ACOR2 to 3ACOR0 bits in TITCR are cleared to 0
3	—	0	R	Reserved This bit is always read as 0.
2 to 0	4VCNT[2:0]	000	R	TCIV_4 Interrupt Counter While the T4VEN bit in TITCR is set to 1, the count in these bits is incremented every time a TCIV_4 interrupt occurs. [Clearing conditions] • When the 4VCNT2 to 4VCNT0 value in TITCNT matches the 4VCOR2 to 4VCOR2 value in TITCR • When the T4VEN bit in TITCR is cleared to 0 • When the 4VCOR2 to 4VCOR2 bits in TITCR are cleared to 0

Note: To clear the TITCNT, clear the bits T3AEN and T4VEN in TITCR to 0.

27.3.27 Timer Buffer Transfer Set Register (TBTER)

TBTER is an 8-bit readable/writable register that enables or disables transfer from the buffer registers* used in complementary PWM mode to the temporary registers and specifies whether to link the transfer with interrupt skipping operation. This module has one TBTER.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	BTE[1:0]	
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	BTE[1:0]	00	R/W	These bits enable or disable transfer from the buffer registers* used in complementary PWM mode to the temporary registers and specify whether to link the transfer with interrupt skipping operation. For details, see table 27.40.

Note: * Applicable buffer registers:
TGRC_3, TGRD_3, TGRC_4, TGRD_4, and TCBR

Table 27.40 Setting of Bits BTE1 and BTE0

Bit 1	Bit 0	
BTE1	**BTE0**	**Description**
0	0	Enables transfer from the buffer registers to the temporary registers*1 and does not link the transfer with interrupt skipping operation.
0	1	Disables transfer from the buffer registers to the temporary registers.
1	0	Links transfer from the buffer registers to the temporary registers with interrupt skipping operation.*2
1	1	Setting prohibited

Notes: 1. Data is transferred according to the MD3 to MD0 bit setting in TMDR. For details, refer to section 27.4.8, Complementary PWM Mode.

2. When interrupt skipping is disabled (the T3AEN and T4VEN bits are cleared to 0 in the timer interrupt skipping set register (TITCR) or the skipping count set bits (3ACOR and 4VCOR) in TITCR are cleared to 0)), be sure to disable link of buffer transfer with interrupt skipping (clear the BTE1 bit in the timer buffer transfer set register (TBTER) to 0). If link with interrupt skipping is enabled while interrupt skipping is disabled, buffer transfer will not be performed.

27.3.28 Timer Dead Time Enable Register (TDER)

TDER is an 8-bit readable/writable register that controls dead time generation in complementary PWM mode. This module has one TDER in channel 3. TDER must be modified only while TCNT stops.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	TDER
Initial value:	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R/(W)

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	TDER	1	R/(W)	Dead Time Enable Specifies whether to generate dead time. 0: Does not generate dead time 1: Generates dead time* [Clearing condition] • When 0 is written to TDER after reading TDER = 1

Note: * TDDR must be set to 1 or a larger value.

27.3.29 Timer Waveform Control Register (TWCR)

TWCR is an 8-bit readable/writable register that controls the waveform when synchronous counter clearing occurs in TCNT_3 and TCNT_4 in complementary PWM mode and specifies whether to clear the counters at TGRA_3 compare match. The CCE bit and WRE bit in TWCR must be modified only while TCNT stops.

Bit:	7	6	5	4	3	2	1	0
	CCE	-	-	-	-	-	-	WRE
Initial value:	0*	0	0	0	0	0	0	0
R/W:	R/(W)	R	R	R	R	R	R	R/(W)

Note: * Do not set to 1 when complementary PWM mode is not selected.

Bit	Bit Name	Initial Value	R/W	Description
7	CCE	0*	R/(W)	Compare Match Clear Enable Specifies whether to clear counters at TGRA_3 compare match in complementary PWM mode. 0: Does not clear counters at TGRA_3 compare match 1: Clears counters at TGRA_3 compare match [Setting condition] • When 1 is written to CCE after reading CCE = 0
6 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
0	WRE	0	R/(W)	Initial Output Suppression Enable Selects the waveform output when synchronous counter clearing occurs in complementary PWM mode. The initial output is suppressed only when synchronous clearing occurs within the Tb interval at the trough in complementary PWM mode. When synchronous clearing occurs outside this interval, the initial value specified in TOCR is output regardless of the WRE bit setting. The initial value is also output when synchronous clearing occurs in the Tb interval at the trough immediately after TCNT_3 and TCNT_4 start operation. For the Tb interval at the trough in complementary PWM mode, see figure 27.40. 0: Outputs the initial value specified in TOCR 1: Suppresses initial output [Setting condition] • When 1 is written to WRE after reading WRE = 0

Note: * Do not set to 1 when complementary PWM mode is not selected.

27.3.30 Bus Master Interface

The timer counters (TCNT), general registers (TGR), timer subcounter (TCNTS), timer cycle buffer register (TCBR), timer dead time data register (TDDR), timer cycle data register (TCDR), timer A/D converter start request control register (TADCR), timer A/D converter start request cycle set registers (TADCOR), and timer A/D converter start request cycle set buffer registers (TADCOBR) are 16-bit registers. A 16-bit data bus to the bus master enables 16-bit read/writes. 8-bit read/write is not possible. Always access in 16-bit units.

All registers other than the above registers are 8-bit registers. These are connected to the CPU by a 16-bit data bus, so 16-bit read/writes and 8-bit read/writes are both possible.

27.4 Operation

27.4.1 Basic Functions

Each channel has a TCNT and TGR register. TCNT performs up-counting, and is also capable of free-running operation, cycle counting, and external event counting.

Each TGR can be used as an input capture register or output compare register.

Always select functions for external pins of this module using the general I/O ports.

(1) Counter Operation

When one of bits CST0 to CST4 in TSTR is set to 1, the TCNT counter for the corresponding channel begins counting. TCNT can operate as a free-running counter, periodic counter, for example.

(a) Example of Count Operation Setting Procedure

Figure 27.4 shows an example of the count operation setting procedure.

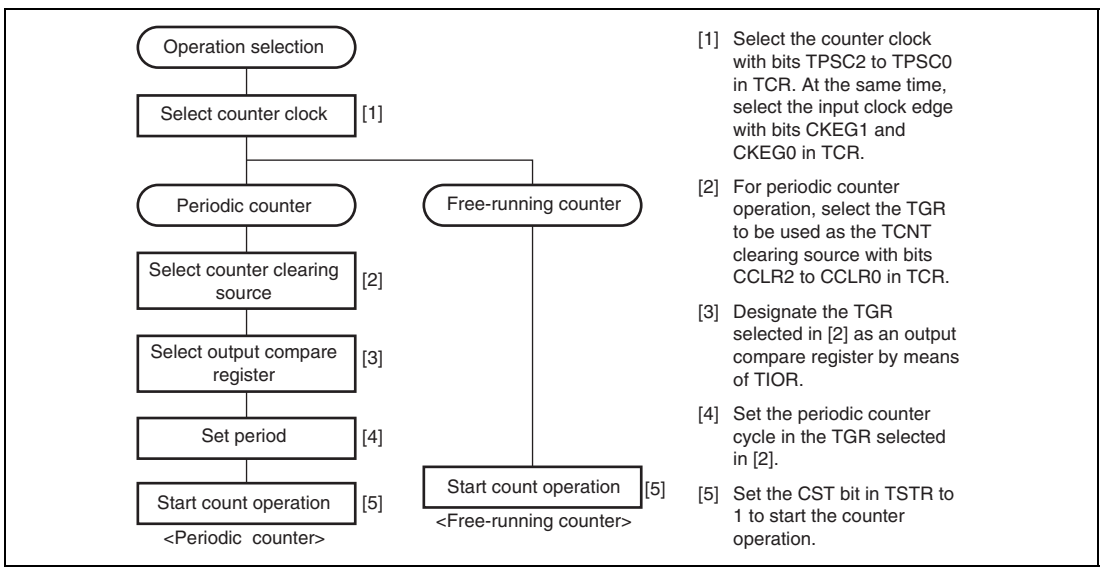


Figure 27.4 Example of Counter Operation Setting Procedure

(b) Free-Running Count Operation and Periodic Count Operation:

Immediately after a reset, the TCNT counters of this module are all designated as free-running counters. When the relevant bit in TSTR is set to 1 the corresponding TCNT counter starts up-count operation as a free-running counter. When TCNT overflows (from H'FFFF to H'0000), the TCFV bit in TSR is set to 1. If the value of the corresponding TCIEV bit in TIER is 1 at this point, this module requests an interrupt. After overflow, TCNT starts counting up again from H'0000.

Figure 27.5 illustrates free-running counter operation.

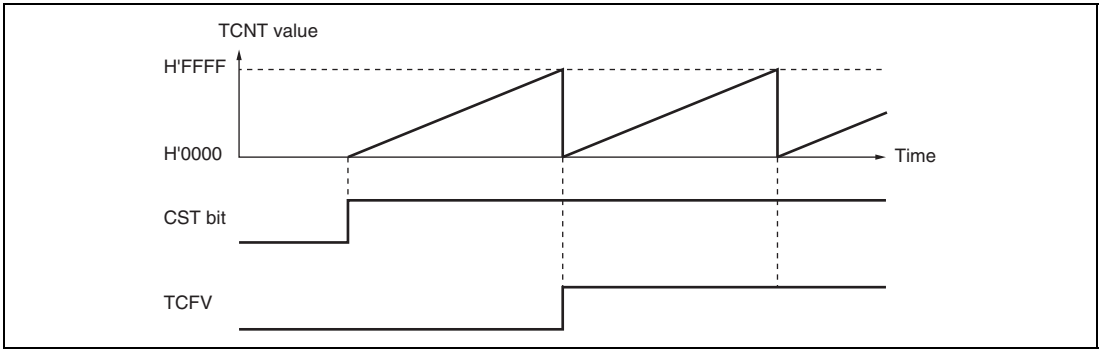


Figure 27.5 Free-Running Counter Operation

When compare match is selected as the TCNT clearing source, the TCNT counter for the relevant channel performs periodic count operation. The TGR register for setting the period is designated as an output compare register, and counter clearing by compare match is selected by means of bits CCLR0 to CCLR2 in TCR. After the settings have been made, TCNT starts up-count operation as a periodic counter when the corresponding bit in TSTR is set to 1. When the count value matches the value in TGR, the TGF bit in TSR is set to 1 and TCNT is cleared to H'0000.

If the value of the corresponding TGIE bit in TIER is 1 at this point, this module requests an interrupt. After a compare match, TCNT starts counting up again from H'0000.

Figure 27.6 illustrates periodic counter operation.

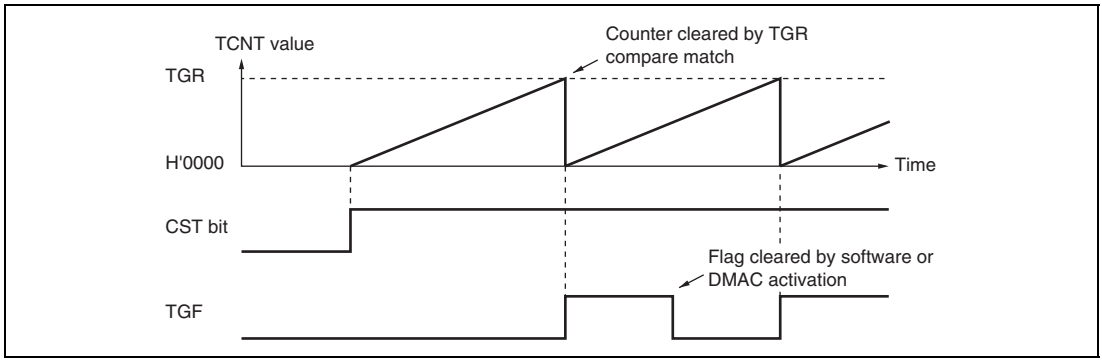


Figure 27.6 Periodic Counter Operation

(2) Waveform Output by Compare Match

This module can perform 0, 1, or toggle output from the corresponding output pin using compare match.

(a) Example of Setting Procedure for Waveform Output by Compare Match

Figure 27.7 shows an example of the setting procedure for waveform output by compare match

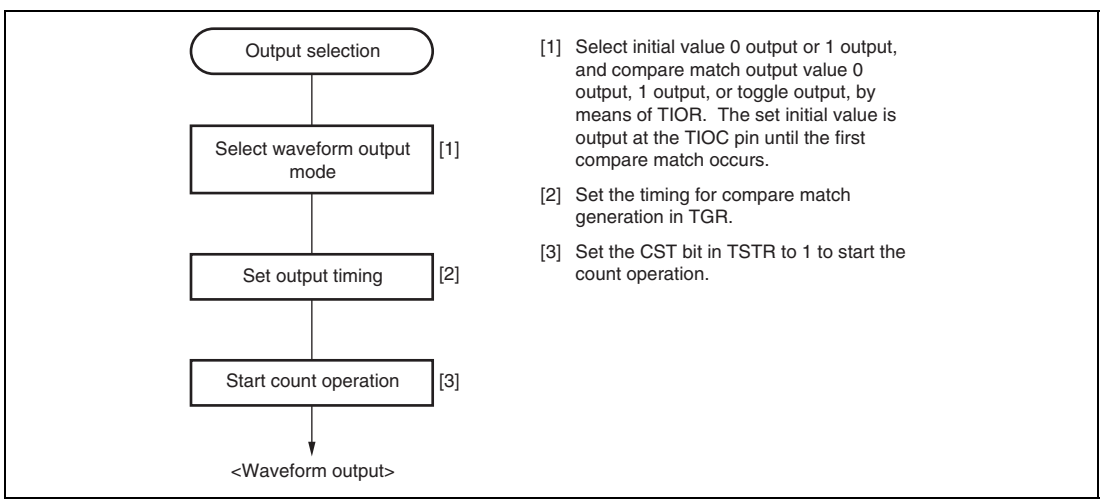


Figure 27.7 Example of Setting Procedure for Waveform Output by Compare Match

(b) Examples of Waveform Output Operation:

Figure 27.8 shows an example of 0 output/1 output.

In this example TCNT has been designated as a free-running counter, and settings have been made such that 1 is output by compare match A, and 0 is output by compare match B. When the set level and the pin level coincide, the pin level does not change.

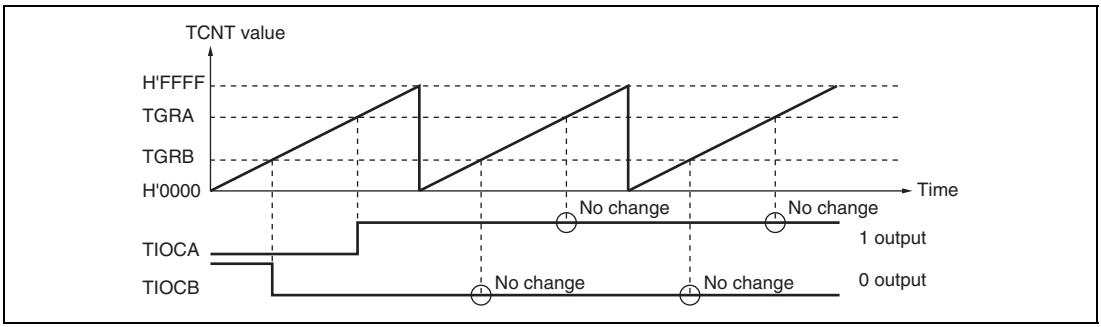


Figure 27.8 Example of 0 Output/1 Output Operation

Figure 27.9 shows an example of toggle output.

In this example, TCNT has been designated as a periodic counter (with counter clearing on compare match B), and settings have been made such that the output is toggled by both compare match A and compare match B.

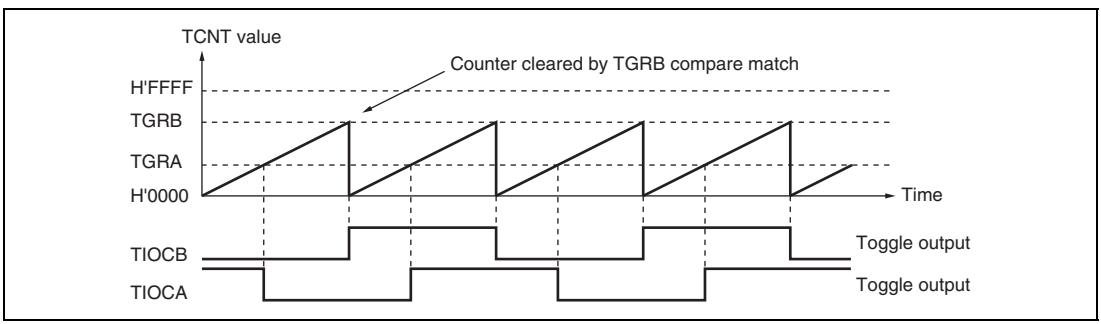


Figure 27.9 Example of Toggle Output Operation

(3) Input Capture Function

The TCNT value can be transferred to TGR on detection of the TIOC pin input edge.

Rising edge, falling edge, or both edges can be selected as the detected edge. For channels 0 and 1, it is also possible to specify another channel's counter input clock or compare match signal as the input capture source.

Note: When another channel's counter input clock is used as the input capture input for channels 0 and 1, Pϕ/1 should not be selected as the counter input clock used for input capture input. Input capture will not be generated if Pϕ/1 is selected.

(a) Example of Input Capture Operation Setting Procedure

Figure 27.10 shows an example of the input capture operation setting procedure.

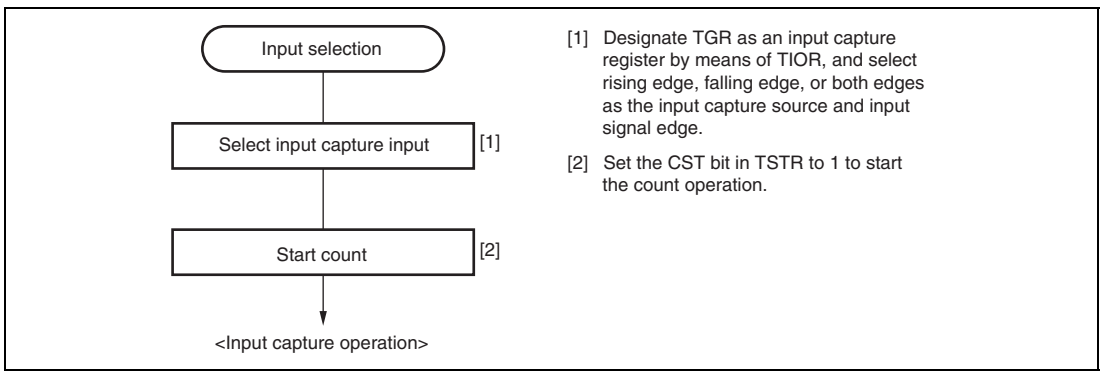


Figure 27.10 Example of Input Capture Operation Setting Procedure

(b) Example of Input Capture Operation

Figure 27.11 shows an example of input capture operation.

In this example both rising and falling edges have been selected as the TIOCA pin input capture input edge, the falling edge has been selected as the TIOCB pin input capture input edge, and counter clearing by TGRB input capture has been designated for TCNT.

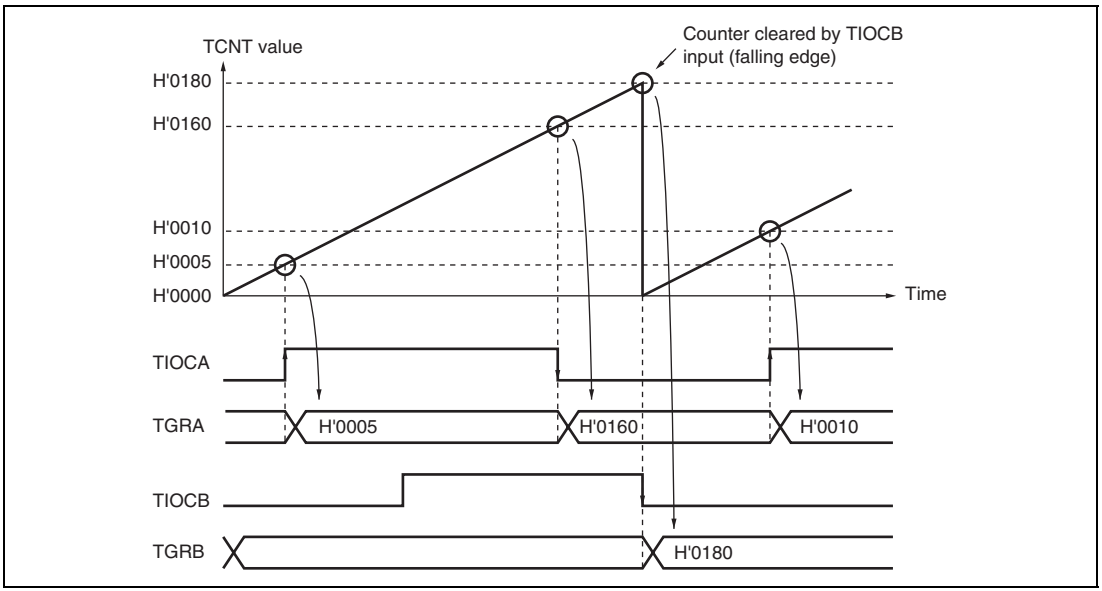


Figure 27.11 Example of Input Capture Operation

27.4.2 Synchronous Operation

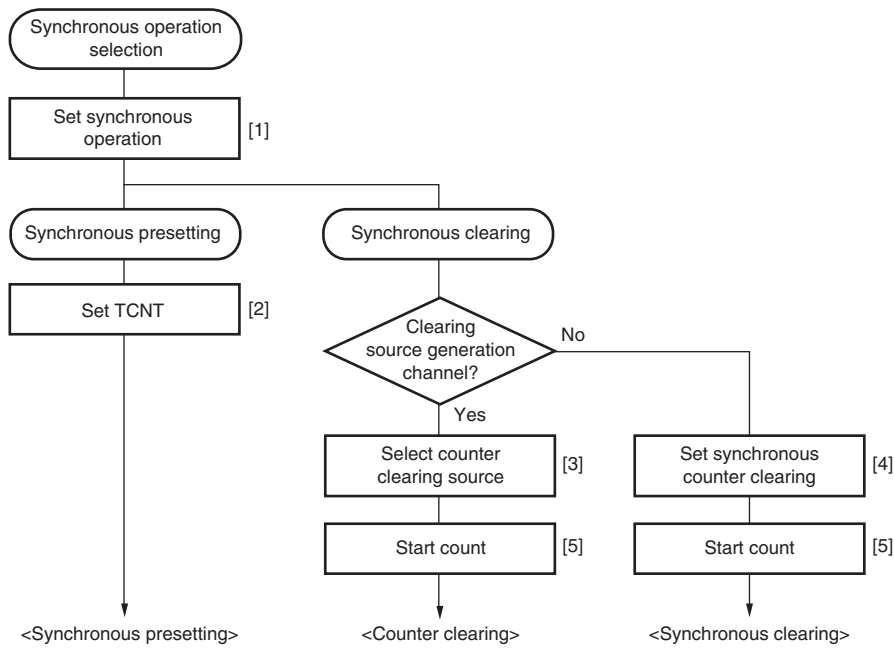
In synchronous operation, the values in a number of TCNT counters can be rewritten simultaneously (synchronous presetting). Also, a number of TCNT counters can be cleared simultaneously by making the appropriate setting in TCR (synchronous clearing).

Synchronous operation enables TGR to be incremented with respect to a single time base.

Channels 0 to 4 can all be designated for synchronous operation. Channel 5 cannot be used for synchronous operation.

(1) Example of Synchronous Operation Setting Procedure

Figure 27.12 shows an example of the synchronous operation setting procedure.



[1] Set to 1 the SYNC bits in TSYR corresponding to the channels to be designated for synchronous operation.
[2] When the TCNT counter of any of the channels designated for synchronous operation is written to, the same value is simultaneously written to the other TCNT counters.
[3] Use bits CCLR2 to CCLR0 in TCR to specify TCNT clearing by input capture/output compare, etc.
[4] Use bits CCLR2 to CCLR0 in TCR to designate synchronous clearing for the counter clearing source.
[5] Set to 1 the CST bits in TSTR for the relevant channels, to start the count operation.

Figure 27.12 Example of Synchronous Operation Setting Procedure

(2) Example of Synchronous Operation

Figure 27.13 shows an example of synchronous operation.

In this example, synchronous operation and PWM mode 1 have been designated for channels 0 to 2, TGRB_0 compare match has been set as the channel 0 counter clearing source, and synchronous clearing has been set for the channel 1 and 2 counter clearing source.

Three-phase PWM waveforms are output from pins TIOC0A, TIOC1A, and TIOC2A. At this time, synchronous presetting, and synchronous clearing by TGRB_0 compare match, are performed for channel 0 to 2 TCNT counters, and the data set in TGRB_0 is used as the PWM cycle.

For details of PWM modes, see section 27.4.5, PWM Modes.

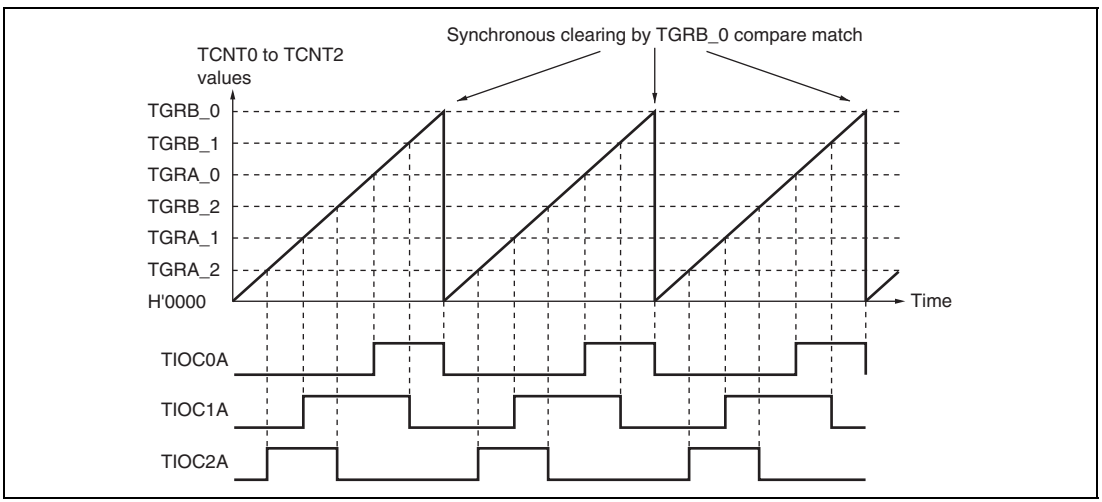


Figure 27.13 Example of Synchronous Operation

27.4.3 Buffer Operation

Buffer operation, provided for channels 0, 3, and 4, enables TGRC and TGRD to be used as buffer registers. In channel 0, TGRF can also be used as a buffer register.

Buffer operation differs depending on whether TGR has been designated as an input capture register or as a compare match register.

Note: TGRE_0 cannot be designated as an input capture register and can only operate as a compare match register.

Table 27.41 shows the register combinations used in buffer operation.

Table 27.41 Register Combinations in Buffer Operation

Channel	Timer General Register	Buffer Register
0	TGRA_0	TGRC_0
	TGRB_0	TGRD_0
	TGRE_0	TGRF_0
3	TGRA_3	TGRC_3
	TGRB_3	TGRD_3
4	TGRA_4	TGRC_4
	TGRB_4	TGRD_4

- When TGR is an output compare register

  When a compare match occurs, the value in the buffer register for the corresponding channel is transferred to the timer general register.

  This operation is illustrated in figure 27.14.

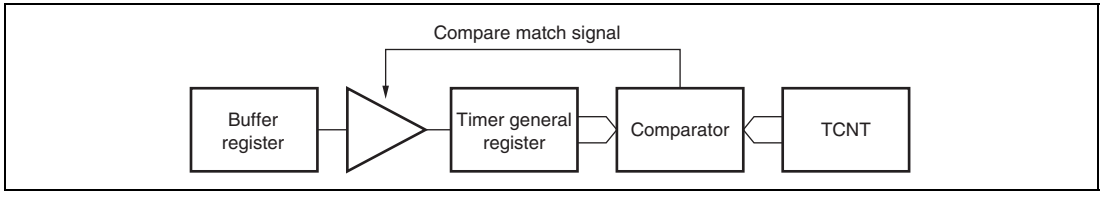


Figure 27.14 Compare Match Buffer Operation

- When TGR is an input capture register

  When input capture occurs, the value in TCNT is transferred to TGR and the value previously held in the timer general register is transferred to the buffer register.

  This operation is illustrated in figure 27.15.

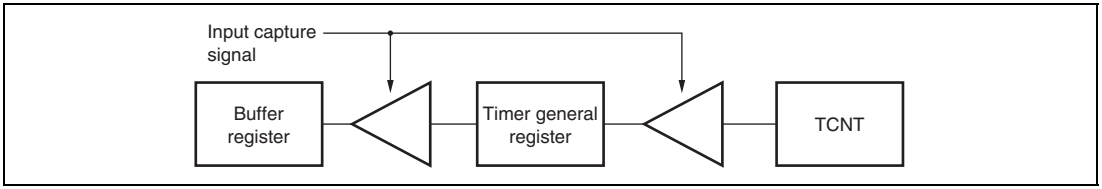


Figure 27.15 Input Capture Buffer Operation

(1) Example of Buffer Operation Setting Procedure

Figure 27.16 shows an example of the buffer operation setting procedure.

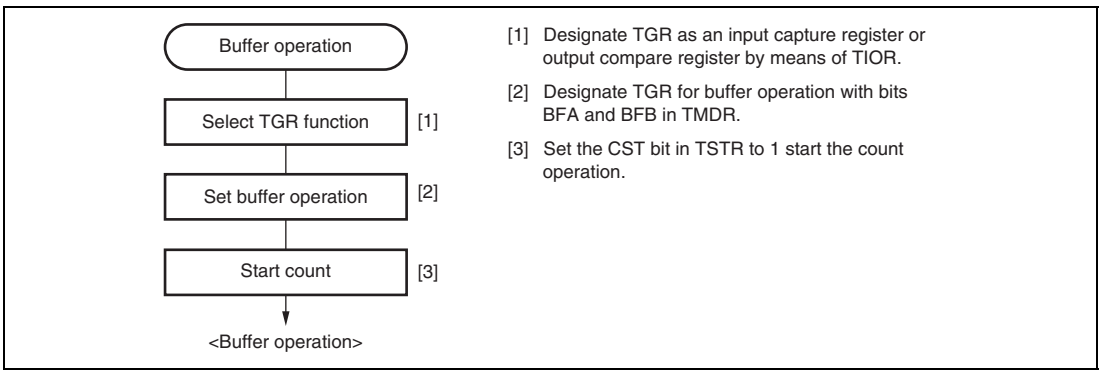


Figure 27.16 Example of Buffer Operation Setting Procedure

(2) Examples of Buffer Operation

(a) When TGR is an output compare register

Figure 27.17 shows an operation example in which PWM mode 1 has been designated for channel 0, and buffer operation has been designated for TGRA and TGRC. The settings used in this example are TCNT clearing by compare match B, 1 output at compare match A, and 0 output at compare match B. In this example, the TTSA bit in TBTM is cleared to 0.

As buffer operation has been set, when compare match A occurs the output changes and the value in buffer register TGRC is simultaneously transferred to timer general register TGRA. This operation is repeated each time that compare match A occurs.

For details of PWM modes, see section 27.4.5, PWM Modes.

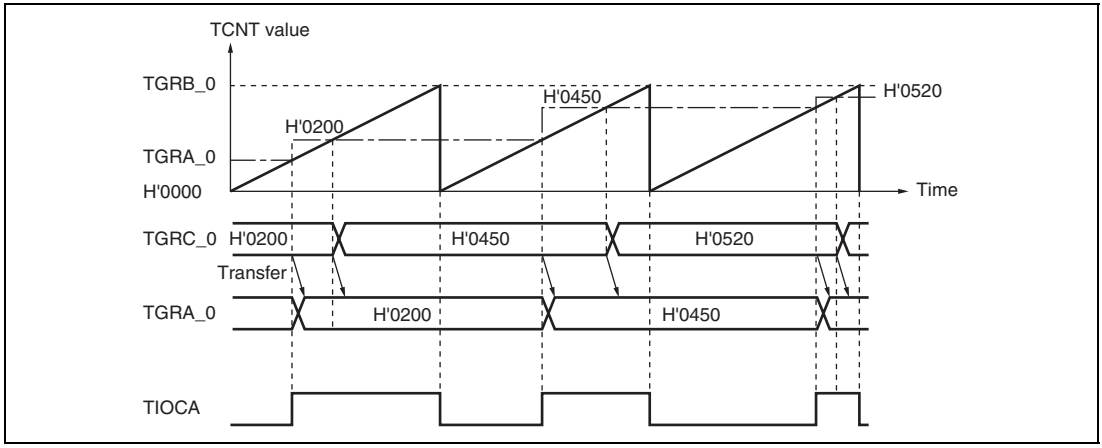


Figure 27.17 Example of Buffer Operation (1)

(b) When TGR is an input capture register

Figure 27.18 shows an operation example in which TGRA has been designated as an input capture register, and buffer operation has been designated for TGRA and TGRC.

Counter clearing by TGRA input capture has been set for TCNT, and both rising and falling edges have been selected as the TIOCA pin input capture input edge.

As buffer operation has been set, when the TCNT value is stored in TGRA upon the occurrence of input capture A, the value previously stored in TGRA is simultaneously transferred to TGRC.

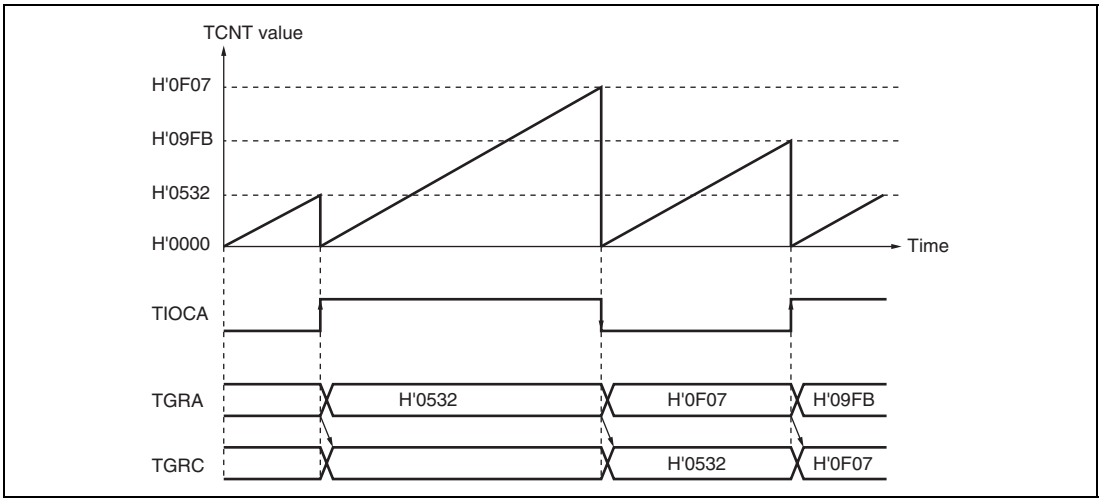


Figure 27.18 Example of Buffer Operation (2)

(3) Selecting Timing for Transfer from Buffer Registers to Timer General Registers in Buffer Operation

The timing for transfer from buffer registers to timer general registers can be selected in PWM mode 1 or 2 for channel 0 or in PWM mode 1 for channels 3 and 4 by setting the buffer operation transfer mode registers (TBTM_0, TBTM_3, and TBTM_4). Either compare match (initial setting) or TCNT clearing can be selected for the transfer timing. TCNT clearing as transfer timing is one of the following cases.

- When TCNT overflows (H'FFFF to H'0000)
- When H'0000 is written to TCNT during counting
- When TCNT is cleared to H'0000 under the condition specified in the CCLR2 to CCLR0 bits in TCR

Note: TBTM must be modified only while TCNT stops.

Figure 27.19 shows an operation example in which PWM mode 1 is designated for channel 0 and buffer operation is designated for TGRA_0 and TGRC_0. The settings used in this example are TCNT_0 clearing by compare match B, 1 output at compare match A, and 0 output at compare match B. The TTSA bit in TBTM_0 is set to 1.

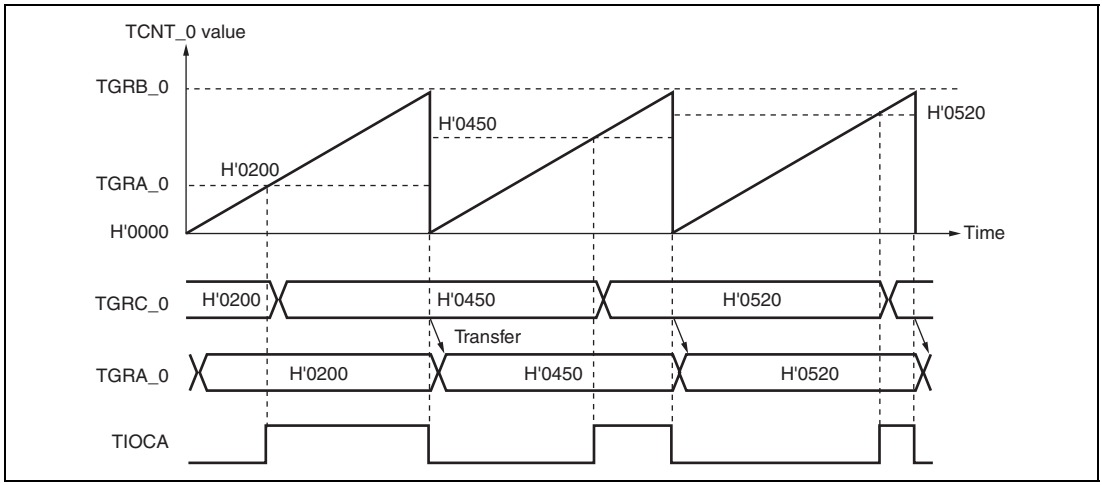


Figure 27.19 Example of Buffer Operation When TCNT_0 Clearing is Selected for TGRC_0 to TGRA_0 Transfer Timing

27.4.4 Cascaded Operation

In cascaded operation, two 16-bit counters for different channels are used together as a 32-bit counter.

This function works by counting the channel 1 counter clock upon overflow/underflow of TCNT_2 as set in bits TPSC0 to TPSC2 in TCR.

Underflow occurs only when the lower 16-bit TCNT is in phase-counting mode.

Table 27.42 shows the register combinations used in cascaded operation.

Note: When phase counting mode is set for channel 1, the counter clock setting is invalid and the counters operates independently in phase counting mode.

Table 27.42 Cascaded Combinations

Combination	Upper 16 Bits	Lower 16 Bits
Channels 1 and 2	TCNT_1	TCNT_2

For simultaneous input capture of TCNT_1 and TCNT_2 during cascaded operation, additional input capture input pins can be specified by the input capture control register (TICCR). For input capture in cascade connection, refer to section 27.7.22, Simultaneous Capture of TCNT_1 and TCNT_2 in Cascade Connection.

Table 27.43 show the TICCR setting and input capture input pins.

Table 27.43 TICCR Setting and Input Capture Input Pins

Target Input Capture	TICCR Setting	Input Capture Input Pins
Input capture from TCNT_1 to TGRA_1	I2AE bit = 0 (initial value)	TIOC1A
	I2AE bit = 1	TIOC1A, TIOC2A
Input capture from TCNT_1 to TGRB_1	I2BE bit = 0 (initial value)	TIOC1B
	I2BE bit = 1	TIOC1B, TIOC2B
Input capture from TCNT_2 to TGRA_2	I1AE bit = 0 (initial value)	TIOC2A
	I1AE bit = 1	TIOC2A, TIOC1A
Input capture from TCNT_2 to TGRB_2	I1BE bit = 0 (initial value)	TIOC2B
	I1BE bit = 1	TIOC2B, TIOC1B

(1) Example of Cascaded Operation Setting Procedure

Figure 27.20 shows an example of the setting procedure for cascaded operation.

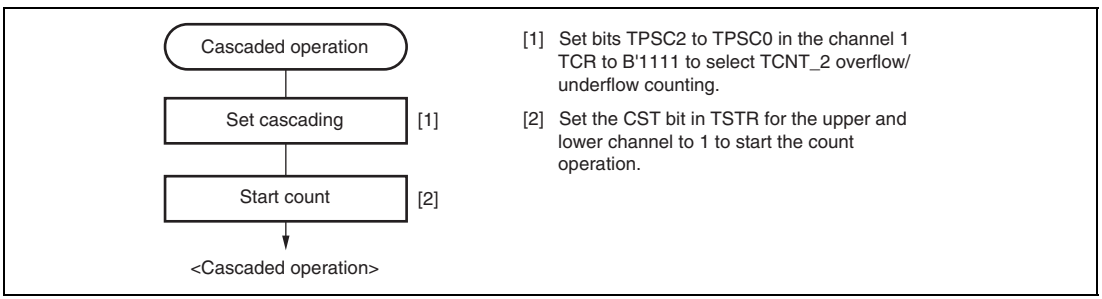


Figure 27.20 Cascaded Operation Setting Procedure

(2) Cascaded Operation Example (a)

Figure 27.21 illustrates the operation when TCNT_2 overflow/underflow counting has been set for TCNT_1 and phase counting mode has been designated for channel 2.

TCNT_1 is incremented by TCNT_2 overflow and decremented by TCNT_2 underflow.

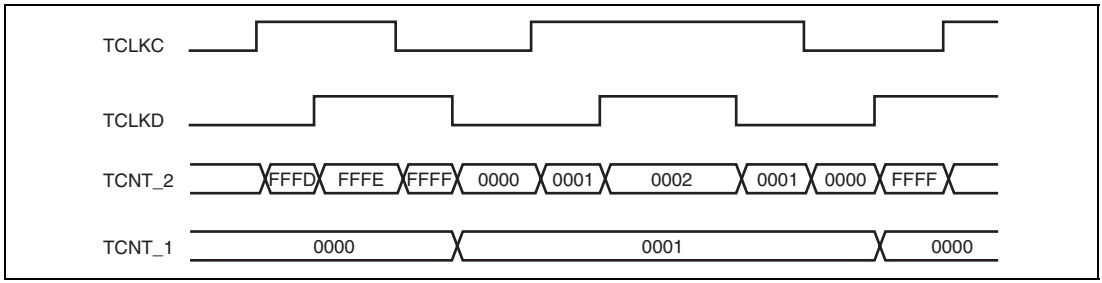


Figure 27.21 Cascaded Operation Example (a)

(3) Cascaded Operation Example (b)

Figure 27.22 illustrates the operation when TCNT_1 and TCNT_2 have been cascaded and the I2AE bit in TICCR has been set to 1 to include the TIOC2A pin in the TGRA_1 input capture conditions. In this example, the IOA0 to IOA3 bits in TIOR_1 have selected the TIOC1A rising edge for the input capture timing while the IOA0 to IOA3 bits in TIOR_2 have selected the TIOC2A rising edge for the input capture timing.

Under these conditions, the rising edge of both TIOC1A and TIOC2A is used for the TGRA_1 input capture condition. For the TGRA_2 input capture condition, the TIOC2A rising edge is used.

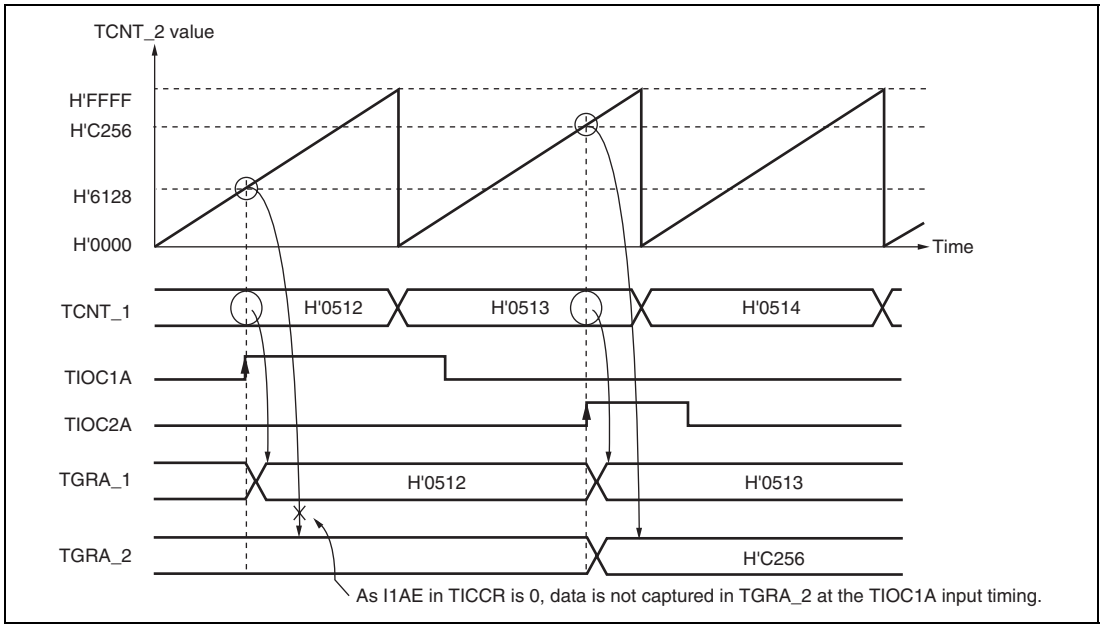


Figure 27.22 Cascaded Operation Example (b)

(4) Cascaded Operation Example (c)

Figure 27.23 illustrates the operation when TCNT_1 and TCNT_2 have been cascaded and the I2AE and I1AE bits in TICCR have been set to 1 to include the TIOC2A and TIOC1A pins in the TGRA_1 and TGRA_2 input capture conditions, respectively. In this example, the IOA0 to IOA3 bits in both TIOR_1 and TIOR_2 have selected both the rising and falling edges for the input capture timing. Under these conditions, the ORed result of TIOC1A and TIOC2A input is used for the TGRA_1 and TGRA_2 input capture conditions.

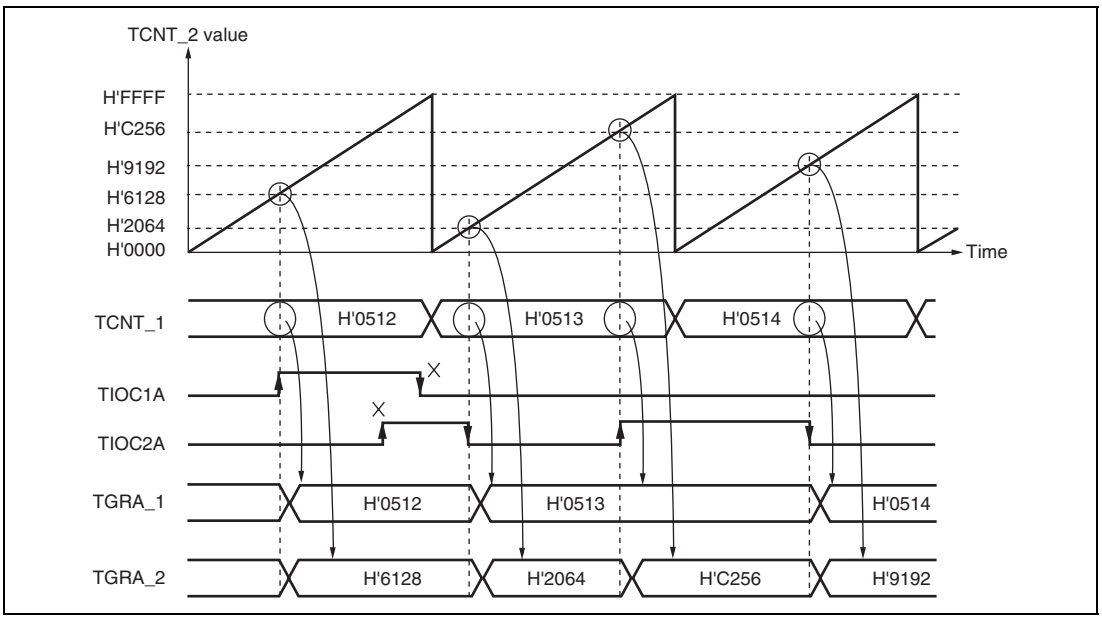


Figure 27.23 Cascaded Operation Example (c)

(5) Cascaded Operation Example (d)

Figure 27.24 illustrates the operation when TCNT_1 and TCNT_2 have been cascaded and the I2AE bit in TICCR has been set to 1 to include the TIOC2A pin in the TGRA_1 input capture conditions. In this example, the IOA0 to IOA3 bits in TIOR_1 have selected TGRA_0 compare match or input capture occurrence for the input capture timing while the IOA0 to IOA3 bits in TIOR_2 have selected the TIOC2A rising edge for the input capture timing.

Under these conditions, as TIOR_1 has selected TGRA_0 compare match or input capture occurrence for the input capture timing, the TIOC2A edge is not used for TGRA_1 input capture condition although the I2AE bit in TICCR has been set to 1.

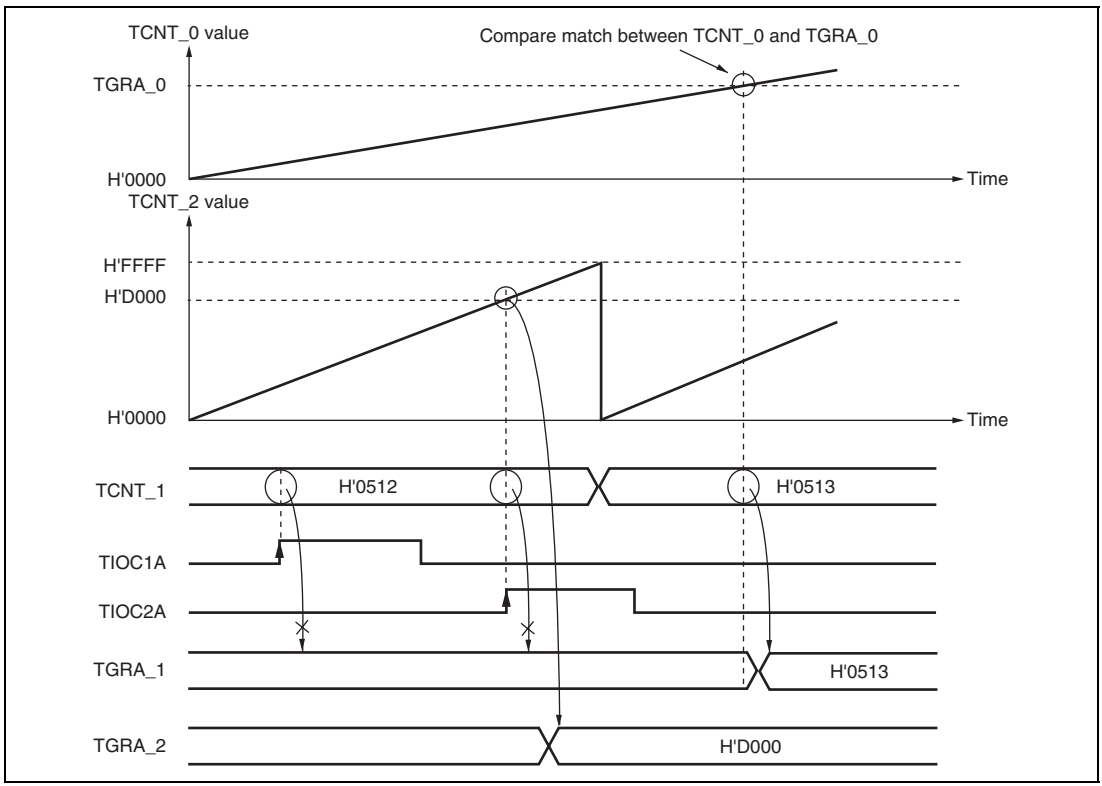


Figure 27.24 Cascaded Operation Example (d)

27.4.5 PWM Modes

In PWM mode, PWM waveforms are output from the output pins. The output level can be selected as 0, 1, or toggle output in response to a compare match of each TGR.

TGR registers settings can be used to output a PWM waveform in the range of 0% to 100% duty.

Designating TGR compare match as the counter clearing source enables the period to be set in that register. All channels can be designated for PWM mode independently. Synchronous operation is also possible.

There are two PWM modes, as described below.

- PWM mode 1

  PWM output is generated from the TIOCA and TIOCC pins by pairing TGRA with TGRB and TGRC with TGRD. The output specified by bits IOA0 to IOA3 and IOC0 to IOC3 in TIOR is output from the TIOCA and TIOCC pins at compare matches A and C, and the output specified by bits IOB0 to IOB3 and IOD0 to IOD3 in TIOR is output at compare matches B and D. The initial output value is the value set in TGRA or TGRC. If the set values of paired TGRs are identical, the output value does not change when a compare match occurs.

  In PWM mode 1, a maximum 8-phase PWM output is possible.

- PWM mode 2

  PWM output is generated using one TGR as the cycle register and the others as duty registers. The output specified in TIOR is performed by means of compare matches. Upon counter clearing by a synchronization register compare match, the output value of each pin is the initial value set in TIOR. If the set values of the cycle and duty registers are identical, the output value does not change when a compare match occurs.

  In PWM mode 2, a maximum 8-phase PWM output is possible in combination use with synchronous operation.

The correspondence between PWM output pins and registers is shown in table 27.44.

Table 27.44 PWM Output Registers and Output Pins

		Output Pins	
Channel	**Registers**	**PWM Mode 1**	**PWM Mode 2**
0	TGRA_0	TIOC0A	TIOC0A
	TGRB_0		TIOC0B
	TGRC_0	TIOC0C	TIOC0C
	TGRD_0		TIOC0D
1	TGRA_1	TIOC1A	TIOC1A
	TGRB_1		TIOC1B
2	TGRA_2	TIOC2A	TIOC2A
	TGRB_2		TIOC2B
3	TGRA_3	TIOC3A	Cannot be set
	TGRB_3		Cannot be set
	TGRC_3	TIOC3C	Cannot be set
	TGRD_3		Cannot be set
4	TGRA_4	TIOC4A	Cannot be set
	TGRB_4		Cannot be set
	TGRC_4	TIOC4C	Cannot be set
	TGRD_4		Cannot be set

Note: In PWM mode 2, PWM output is not possible for the TGR register in which the period is set.

(1) Example of PWM Mode Setting Procedure

Figure 27.25 shows an example of the PWM mode setting procedure.

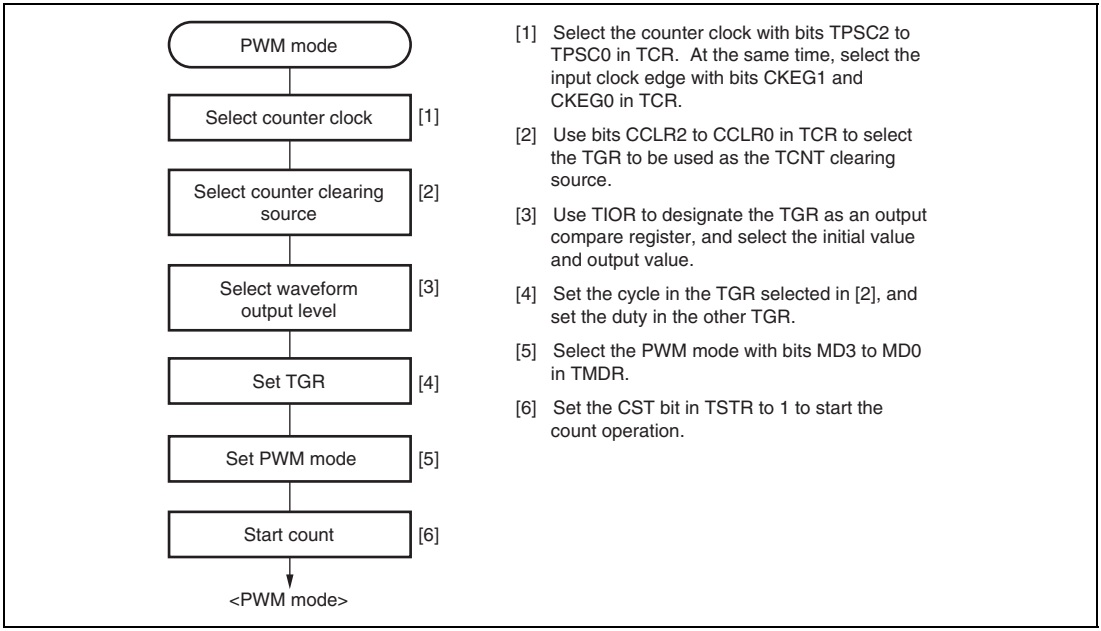


Figure 27.25 Example of PWM Mode Setting Procedure

(2) Examples of PWM Mode Operation

Figure 27.26 shows an example of PWM mode 1 operation.

In this example, TGRA compare match is set as the TCNT clearing source, 0 is set for the TGRA initial output value and output value, and 1 is set as the TGRB output value.

In this case, the value set in TGRA is used as the period, and the values set in the TGRB registers are used as the duty levels.

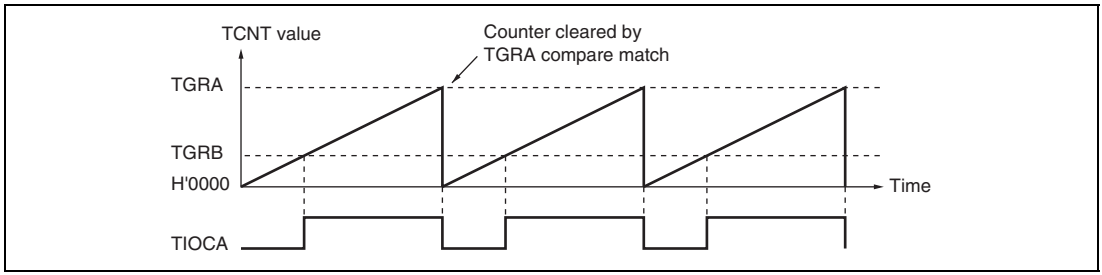


Figure 27.26 Example of PWM Mode Operation (1)

Figure 27.27 shows an example of PWM mode 2 operation.

In this example, synchronous operation is designated for channels 0 and 1, TGRB_1 compare match is set as the TCNT clearing source, and 0 is set for the initial output value and 1 for the output value of the other TGR registers (TGRA_0 to TGRD_0, TGRA_1), outputting a 5-phase PWM waveform.

In this case, the value set in TGRB_1 is used as the cycle, and the values set in the other TGRs are used as the duty levels.

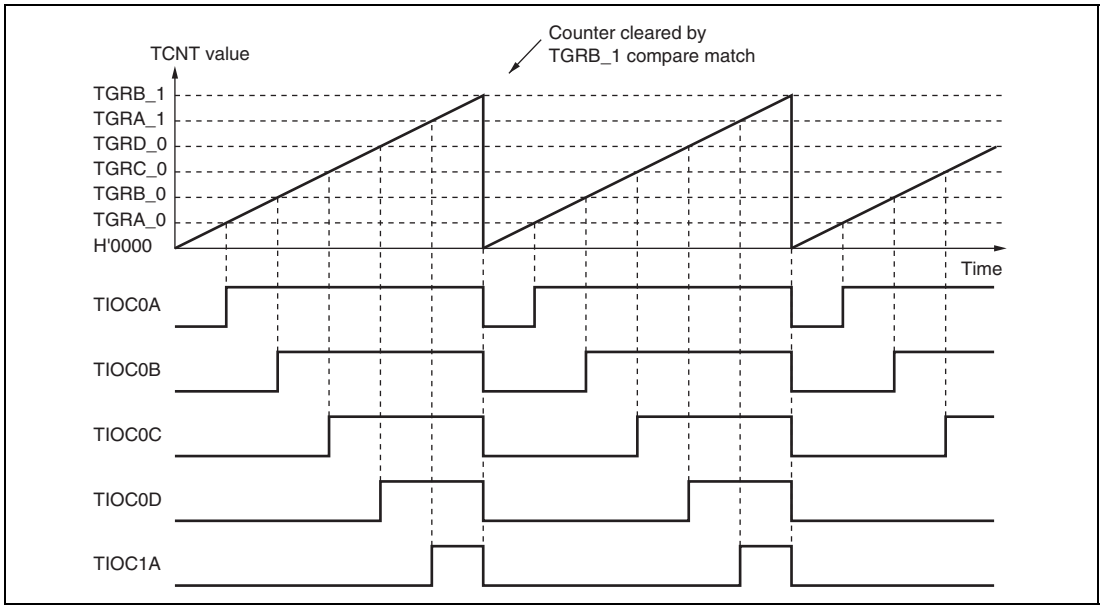


Figure 27.27 Example of PWM Mode Operation (2)

Figure 27.28 shows examples of PWM waveform output with 0% duty and 100% duty in PWM mode.

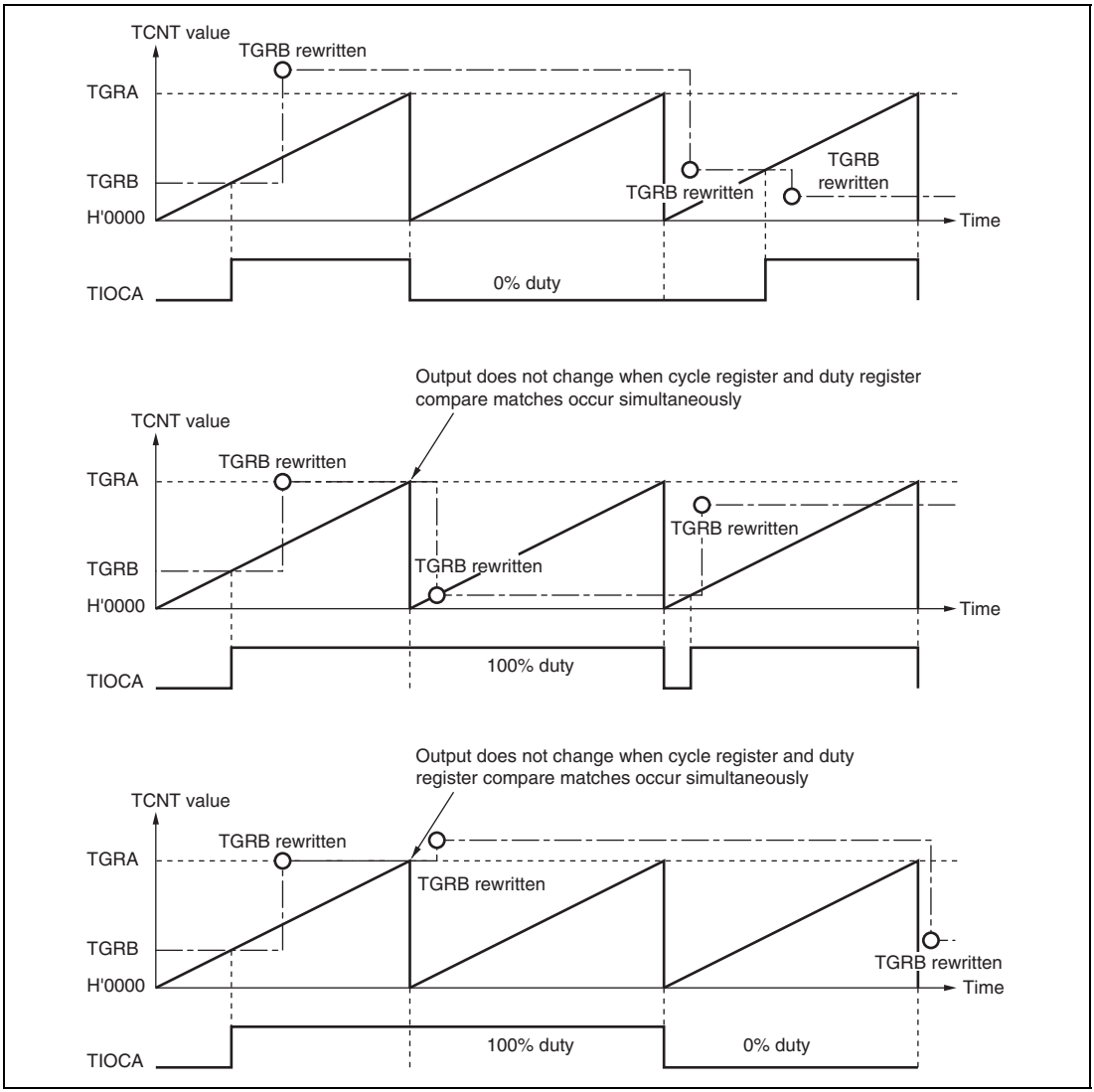


Figure 27.28 Example of PWM Mode Operation (3)

27.4.6 Phase Counting Mode

In phase counting mode, the phase difference between two external clock inputs is detected and TCNT is incremented/decremented accordingly. This mode can be set for channels 1 and 2.

When phase counting mode is set, an external clock is selected as the counter input clock and TCNT operates as an up/down-counter regardless of the setting of bits TPSC0 to TPSC2 and bits CKEG0 and CKEG1 in TCR. However, the functions of bits CCLR0 and CCLR1 in TCR, and of TIOR, TIER, and TGR, are valid, and input capture/compare match and interrupt functions can be used.

This can be used for two-phase encoder pulse input.

If overflow occurs when TCNT is counting up, the TCFV flag in TSR is set; if underflow occurs when TCNT is counting down, the TCFU flag is set.

The TCFD bit in TSR is the count direction flag. Reading the TCFD flag reveals whether TCNT is counting up or down.

Table 27.45 shows the correspondence between external clock pins and channels.

Table 27.45 Phase Counting Mode Clock Input Pins

	External Clock Pins	
Channels	**A-Phase**	**B-Phase**
When channel 1 is set to phase counting mode	TCLKA	TCLKB
When channel 2 is set to phase counting mode	TCLKC	TCLKD

(1) Example of Phase Counting Mode Setting Procedure

Figure 27.29 shows an example of the phase counting mode setting procedure.

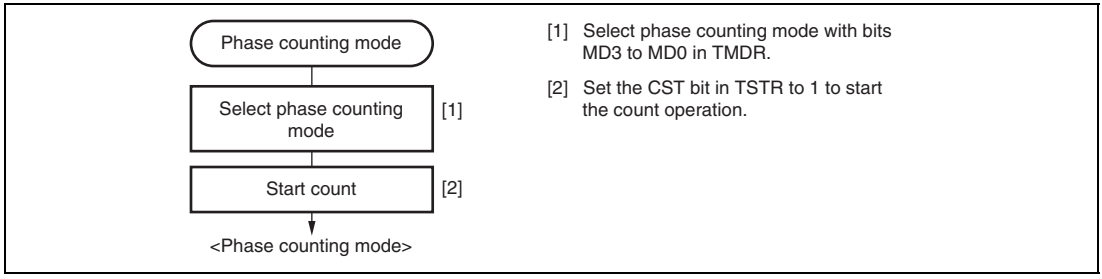


Figure 27.29 Example of Phase Counting Mode Setting Procedure

(2) Examples of Phase Counting Mode Operation

In phase counting mode, TCNT counts up or down according to the phase difference between two external clocks. There are four modes, according to the count conditions.

(a) Phase counting mode 1

Figure 27.30 shows an example of phase counting mode 1 operation, and table 27.46 summarizes the TCNT up/down-count conditions.

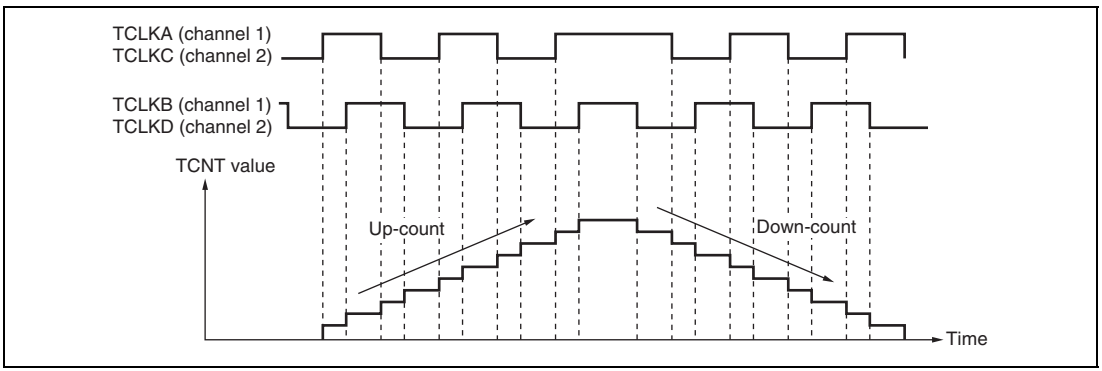


Figure 27.30 Example of Phase Counting Mode 1 Operation

Table 27.46 Up/Down-Count Conditions in Phase Counting Mode 1

TCLKA (Channel 1) TCLKC (Channel 2)	TCLKB (Channel 1) TCLKD (Channel 2)	Operation
High level	↑	Up-count
Low level	↓	
↑	Low level	
↓	High level	
High level	↓	Down-count
Low level	↑	
↑	High level	
↓	Low level	

[Legend]

↑: Rising edge

↓: Falling edge

(b) Phase counting mode 2

Figure 27.31 shows an example of phase counting mode 2 operation, and table 27.47 summarizes the TCNT up/down-count conditions.

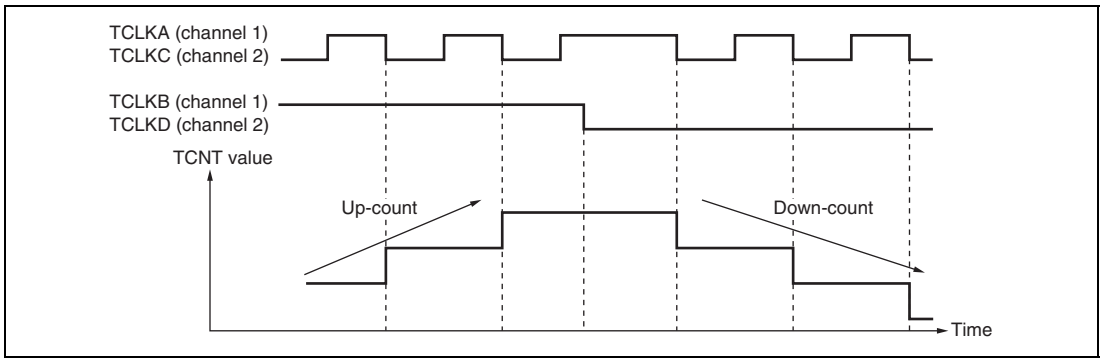


Figure 27.31 Example of Phase Counting Mode 2 Operation

Table 27.47 Up/Down-Count Conditions in Phase Counting Mode 2

TCLKA (Channel 1) TCLKC (Channel 2)	TCLKB (Channel 1) TCLKD (Channel 2)	Operation
High level	↑	Don't care
Low level	↓	Don't care
↑	Low level	Don't care
↓	High level	Up-count
High level	↓	Don't care
Low level	↑	Don't care
↑	High level	Don't care
↓	Low level	Down-count

[Legend]

↑: Rising edge

↓: Falling edge

(c) Phase counting mode 3

Figure 27.32 shows an example of phase counting mode 3 operation, and table 27.48 summarizes the TCNT up/down-count conditions.

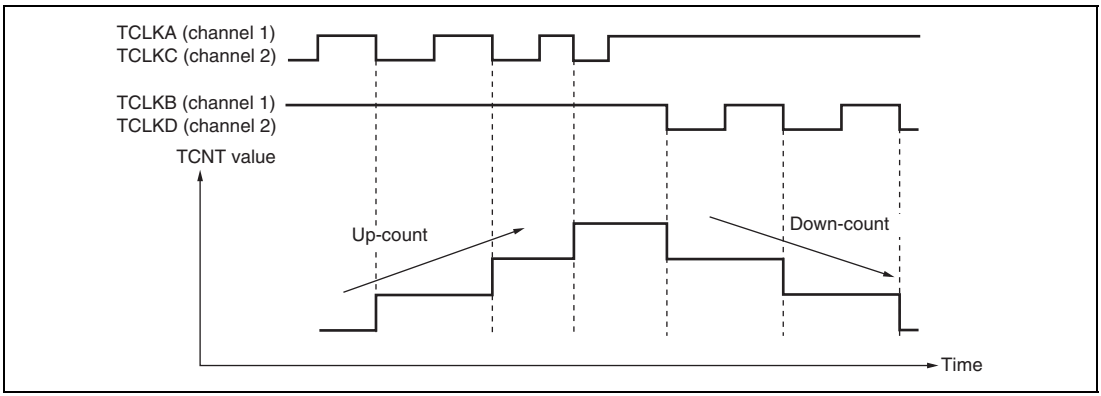


Figure 27.32 Example of Phase Counting Mode 3 Operation

Table 27.48 Up/Down-Count Conditions in Phase Counting Mode 3

TCLKA (Channel 1) TCLKC (Channel 2)	TCLKB (Channel 1) TCLKD (Channel 2)	Operation
High level	↑	Don't care
Low level	↓	Don't care
↑	Low level	Don't care
↓	High level	Up-count
High level	↓	Down-count
Low level	↑	Don't care
↑	High level	Don't care
↓	Low level	Don't care

[Legend]
↑: Rising edge
↓: Falling edge

(d) Phase counting mode 4

Figure 27.33 shows an example of phase counting mode 4 operation, and table 27.49 summarizes the TCNT up/down-count conditions.

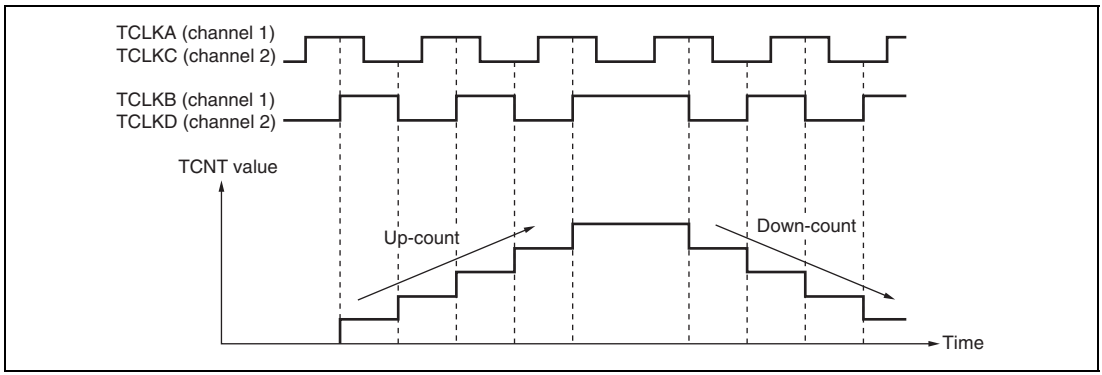


Figure 27.33 Example of Phase Counting Mode 4 Operation

Table 27.49 Up/Down-Count Conditions in Phase Counting Mode 4

TCLKA (Channel 1) TCLKC (Channel 2)	TCLKB (Channel 1) TCLKD (Channel 2)	Operation
High level	⌜	Up-count
Low level	⌞	
⌜	Low level	Don't care
⌞	High level	
High level	⌞	Down-count
Low level	⌜	
⌜	High level	Don't care
⌞	Low level	

[Legend]
⌜: Rising edge
⌞: Falling edge

(3) Phase Counting Mode Application Example

Figure 27.34 shows an example in which channel 1 is in phase counting mode, and channel 1 is coupled with channel 0 to input servo motor 2-phase encoder pulses in order to detect position or speed.

Channel 1 is set to phase counting mode 1, and the encoder pulse A-phase and B-phase are input to TCLKA and TCLKB.

Channel 0 operates with TCNT counter clearing by TGRC_0 compare match; TGRA_0 and TGRC_0 are used for the compare match function and are set with the speed control period and position control period. TGRB_0 is used for input capture, with TGRB_0 and TGRD_0 operating in buffer mode. The channel 1 counter input clock is designated as the TGRB_0 input capture source, and the pulse widths of 2-phase encoder 4-multiplication pulses are detected.

TGRA_1 and TGRB_1 for channel 1 are designated for input capture, and channel 0 TGRA_0 and TGRC_0 compare matches are selected as the input capture source and store the up/down-counter values for the control periods.

This procedure enables the accurate detection of position and speed.

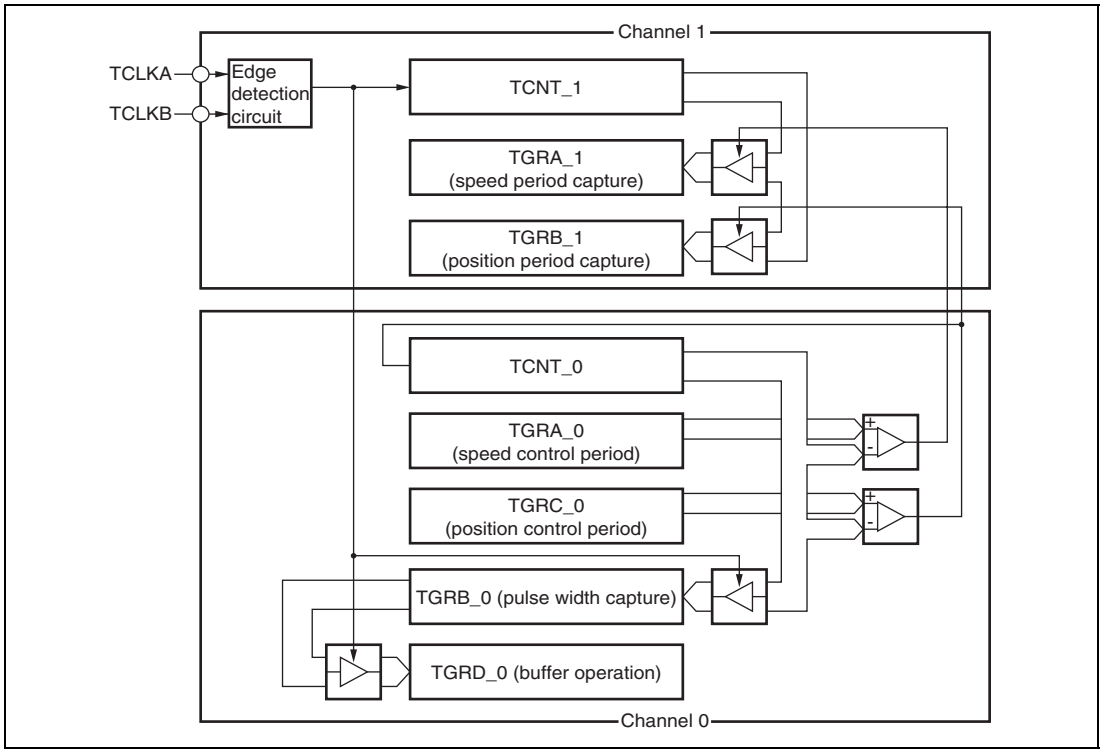


Figure 27.34 Phase Counting Mode Application Example

27.4.7 Reset-Synchronized PWM Mode

In the reset-synchronized PWM mode, three-phase output of positive and negative PWM waveforms that share a common wave transition point can be obtained by combining channels 3 and 4.

When set for reset-synchronized PWM mode, the TIOC3B, TIOC3D, TIOC4A, TIOC4C, TIOC4B, and TIOC4D pins function as PWM output pins and TCNT3 functions as an upcounter.

Table 27.50 shows the PWM output pins used. Table 27.51 shows the settings of the registers.

Table 27.50 Output Pins for Reset-Synchronized PWM Mode

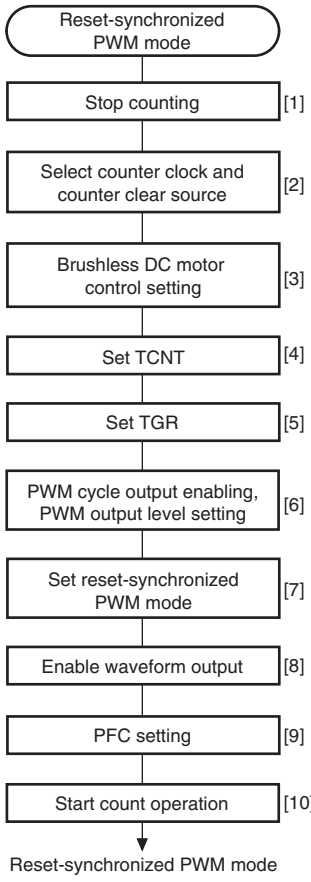
Channel	Output Pin	Description
3	TIOC3B	PWM output pin 1
	TIOC3D	PWM output pin 1' (negative-phase waveform of PWM output 1)
4	TIOC4A	PWM output pin 2
	TIOC4C	PWM output pin 2' (negative-phase waveform of PWM output 2)
	TIOC4B	PWM output pin 3
	TIOC4D	PWM output pin 3' (negative-phase waveform of PWM output 3)

Table 27.51 Register Settings for Reset-Synchronized PWM Mode

Register	Description of Setting
TCNT_3	Initial setting of H'0000
TCNT_4	Initial setting of H'0000
TGRA_3	Set count cycle for TCNT_3
TGRB_3	Sets the turning point for PWM waveform output by the TIOC3B and TIOC3D pins
TGRA_4	Sets the turning point for PWM waveform output by the TIOC4A and TIOC4C pins
TGRB_4	Sets the turning point for PWM waveform output by the TIOC4B and TIOC4D pins

(1) Procedure for Selecting the Reset-Synchronized PWM Mode

Figure 27.35 shows an example of procedure for selecting the reset synchronized PWM mode.



[1] Clear the CST3 and CST4 bits in the TSTR to 0 to halt the counting of TCNT. The reset-synchronized PWM mode must be set up while TCNT_3 and TCNT_4 are halted.

[2] Set bits TPSC2-TPSC0 and CKEG1 and CKEG0 in the TCR_3 to select the counter clock and clock edge for channel 3. Set bits CCLR2-CCLR0 in the TCR_3 to select TGRA compare-match as a counter clear source.

[3] When performing brushless DC motor control, set bit BDC in the timer gate control register (TGCR) and set the feedback signal input source and output chopping or gate signal direct output.

[4] Reset TCNT_3 and TCNT_4 to H'0000.

[5] TGRA_3 is the period register. Set the waveform period value in TGRA_3. Set the transition timing of the PWM output waveforms in TGRB_3, TGRA_4, and TGRB_4. Set times within the compare-match range of TCNT_3.
$X \leq$ TGRA_3 (X: set value).

[6] Select enabling/disabling of toggle output synchronized with the PMW cycle using bit PSYE in the timer output control register (TOCR), and set the PWM output level with bits OLSP and OLSN. When specifying the PWM output level by using TOLBR as a buffer for TOCR2, see figure 27.3.

[7] Set bits MD3-MD0 in TMDR_3 to B'1000 to select the reset-synchronized PWM mode. Do not set to TMDR_4.

[8] Set the enabling/disabling of the PWM waveform output pin in TOER.

[9] Set the port control register and the port I/O register.

[10] Set the CST3 bit in the TSTR to 1 to start the count operation.

Note: The output waveform starts to toggle operation at the point of TCNT_3 = TGRA_3 = X by setting X = TGRA, i.e., cycle = duty.

Figure 27.35 Procedure for Selecting Reset-Synchronized PWM Mode

(2) Reset-Synchronized PWM Mode Operation

Figure 27.36 shows an example of operation in the reset-synchronized PWM mode. TCNT_3 and TCNT_4 operate as upcounters. The counter is cleared when a TCNT_3 and TGRA_3 compare-match occurs, and then begins incrementing from H'0000. The PWM output pin output toggles with each occurrence of a TGRB_3, TGRA_4, TGRB_4 compare-match, and upon counter clears.

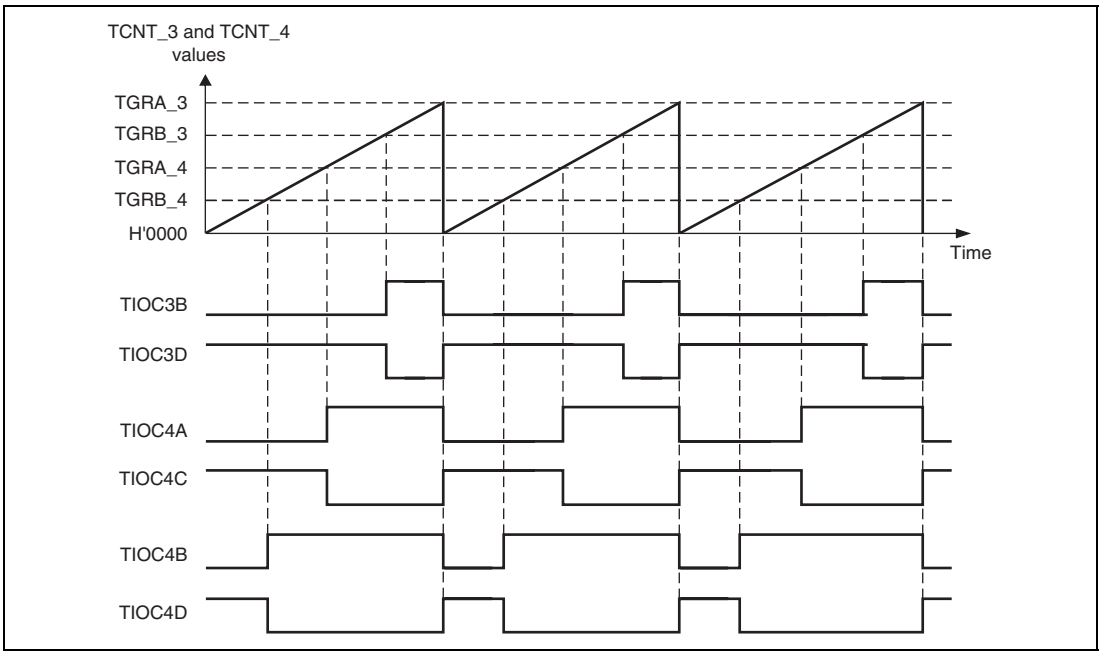


Figure 27.36 Reset-Synchronized PWM Mode Operation Example (When TOCR's OLSN = 1 and OLSP = 1)

27.4.8 Complementary PWM Mode

In the complementary PWM mode, three-phase output of non-overlapping positive and negative PWM waveforms can be obtained by combining channels 3 and 4. PWM waveforms without non-overlapping interval are also available.

In complementary PWM mode, TIOC3B, TIOC3D, TIOC4A, TIOC4B, TIOC4C, and TIOC4D pins function as PWM output pins, the TIOC3A pin can be set for toggle output synchronized with the PWM period. TCNT_3 and TCNT_4 function as up/down counters.

Table 27.52 shows the PWM output pins used. Table 27.53 shows the settings of the registers used.

A function to directly cut off the PWM output by using an external signal is supported as a port function.

Table 27.52 Output Pins for Complementary PWM Mode

Channel	Output Pin	Description
3	TIOC3A	Toggle output synchronized with PWM period (or I/O port)
	TIOC3B	PWM output pin 1
	TIOC3C	I/O port*
	TIOC3D	PWM output pin 1' (non-overlapping negative-phase waveform of PWM output 1; PWM output without non-overlapping interval is also available)
4	TIOC4A	PWM output pin 2
	TIOC4B	PWM output pin 3
	TIOC4C	PWM output pin 2' (non-overlapping negative-phase waveform of PWM output 2; PWM output without non-overlapping interval is also available)
	TIOC4D	PWM output pin 3' (non-overlapping negative-phase waveform of PWM output 3; PWM output without non-overlapping interval is also available)

Note: * Avoid setting the TIOC3C pin as a timer I/O pin in the complementary PWM mode.

Table 27.53 Register Settings for Complementary PWM Mode

Channel	Counter/Register	Description	Read/Write from CPU
3	TCNT_3	Start of up-count from value set in dead time register	Maskable by TRWER setting*
	TGRA_3	Set TCNT_3 upper limit value (1/2 carrier cycle + dead time)	Maskable by TRWER setting*
	TGRB_3	PWM output 1 compare register	Maskable by TRWER setting*
	TGRC_3	TGRA_3 buffer register	Always readable/writable
	TGRD_3	PWM output 1/TGRB_3 buffer register	Always readable/writable
4	TCNT_4	Up-count start, initialized to H'0000	Maskable by TRWER setting*
	TGRA_4	PWM output 2 compare register	Maskable by TRWER setting*
	TGRB_4	PWM output 3 compare register	Maskable by TRWER setting*
	TGRC_4	PWM output 2/TGRA_4 buffer register	Always readable/writable
	TGRD_4	PWM output 3/TGRB_4 buffer register	Always readable/writable
Timer dead time data register (TDDR)		Set TCNT_4 and TCNT_3 offset value (dead time value)	Maskable by TRWER setting*
Timer cycle data register (TCDR)		Set TCNT_4 upper limit value (1/2 carrier cycle)	Maskable by TRWER setting*
Timer cycle buffer register (TCBR)		TCDR buffer register	Always readable/writable
Subcounter (TCNTS)		Subcounter for dead time generation	Read-only
Temporary register 1 (TEMP1)		PWM output 1/TGRB_3 temporary register	Not readable/writable
Temporary register 2 (TEMP2)		PWM output 2/TGRA_4 temporary register	Not readable/writable
Temporary register 3 (TEMP3)		PWM output 3/TGRB_4 temporary register	Not readable/writable

Note: * Access can be enabled or disabled according to the setting of bit 0 (RWE) in TRWER (timer read/write enable register).

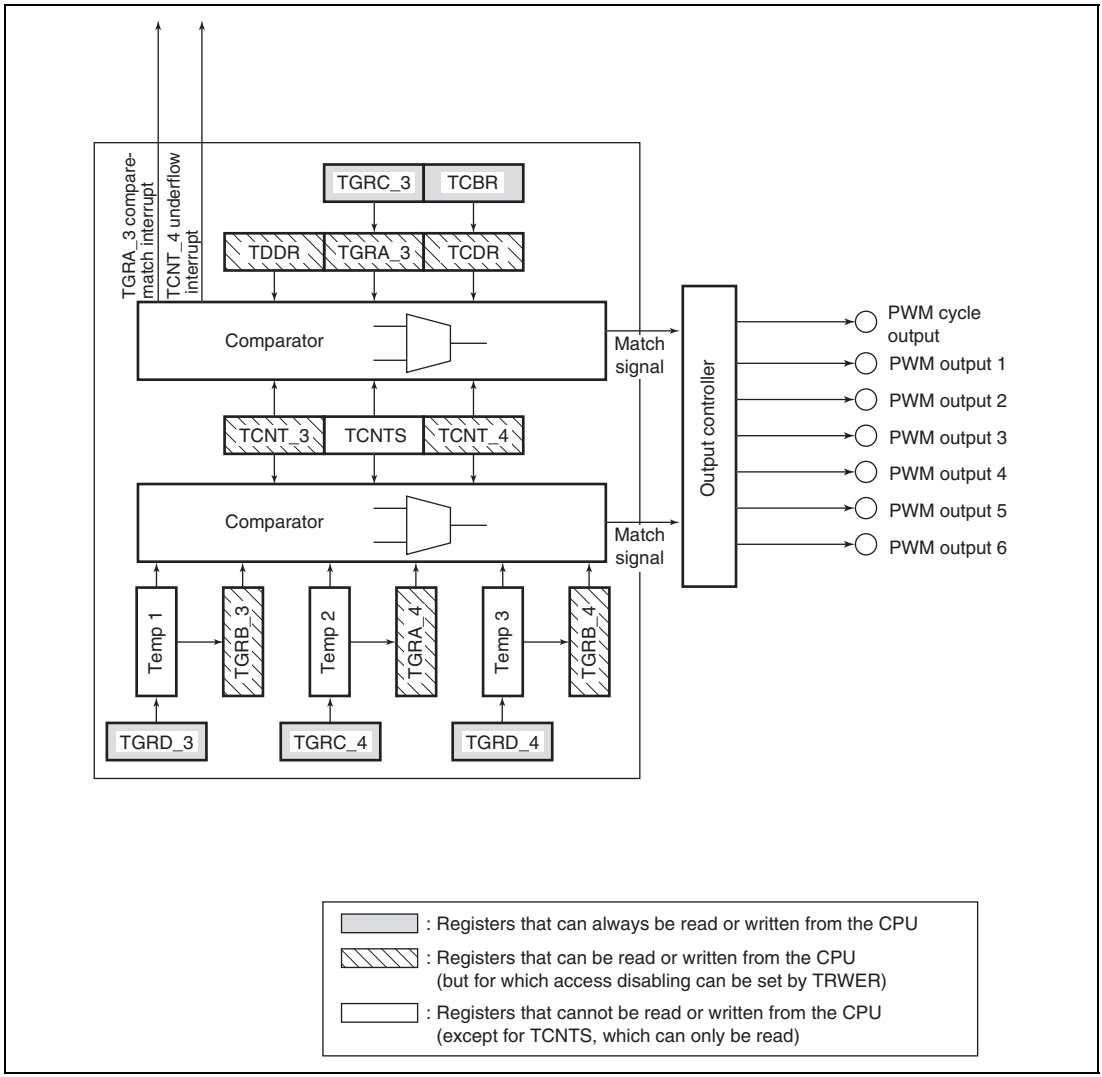
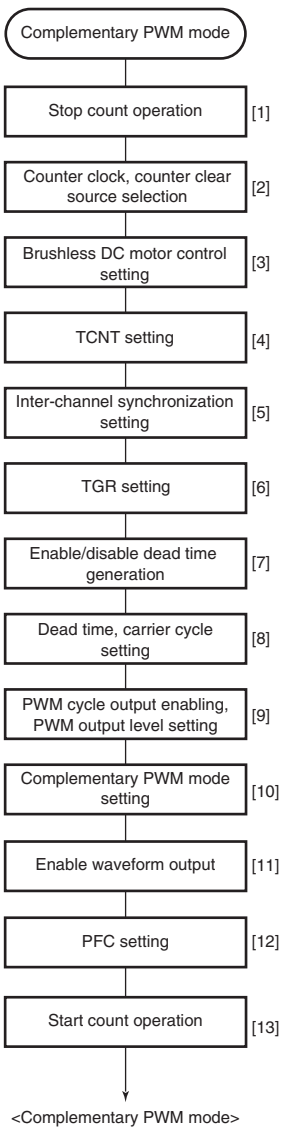


Figure 27.37 Block Diagram of Channels 3 and 4 in Complementary PWM Mode

(1) Example of Complementary PWM Mode Setting Procedure

An example of the complementary PWM mode setting procedure is shown in figure 27.38.



[1] Clear bits CST3 and CST4 in the timer start register (TSTR) to 0, and halt timer counter (TCNT) operation. Perform complementary PWM mode setting when TCNT_3 and TCNT_4 are stopped.

[2] Set the same counter clock and clock edge for channels 3 and 4 with bits TPSC2-TPSC0 and bits CKEG1 and CKEG0 in the timer control register (TCR). Use bits CCLR2-CCLR0 to set synchronous clearing only when restarting by a synchronous clear from another channel during complementary PWM mode operation.

[3] When performing brushless DC motor control, set bit BDC in the timer gate control register (TGCR) and set the feedback signal input source and output chopping or gate signal direct output.

[4] Set the dead time in TCNT_3. Set TCNT_4 to H'0000.

[5] Set only when restarting by a synchronous clear from another channel during complementary PWM mode operation. In this case, synchronize the channel generating the synchronous clear with channels 3 and 4 using the timer synchronous register (TSYR).

[6] Set the output PWM duty in the duty registers (TGRB_3, TGRA_4, TGRB_4) and buffer registers (TGRD_3, TGRC_4, TGRD_4). Set the same initial value in each corresponding TGR.

[7] This setting is necessary only when no dead time should be generated. Make appropriate settings in the timer dead time enable register (TDER) so that no dead time is generated.

[8] Set the dead time in the dead time register (TDDR), 1/2 the carrier cycle in the carrier cycle data register (TCDR) and carrier cycle buffer register (TCBR), and 1/2 the carrier cycle plus the dead time in TGRA_3 and TGRC_3. When no dead time generation is selected, set 1 in TDDR and 1/2 the carrier cycle + 1 in TGRA_3 and TGRC_3.

[9] Select enabling/disabling of toggle output synchronized with the PWM cycle using bit PSYE in the timer output control register 1 (TOCR1), and set the PWM output level with bits OLSP and OLSN. When specifying the PWM output level by using TOLBR as a buffer for TOCR2, see figure 27.3.

[10] Select complementary PWM mode in timer mode register 3 (TMDR_3). Do not set in TMDR_4.

[11] Set enabling/disabling of PWM waveform output pin output in the timer output master enable register (TOER).

[12] Set the port control register and the port I/O register.

[13] Set bits CST3 and CST4 in TSTR to 1 simultaneously to start the count operation.

Figure 27.38 Example of Complementary PWM Mode Setting Procedure

(2) Outline of Complementary PWM Mode Operation

In complementary PWM mode, 6-phase PWM output is possible. Figure 27.39 illustrates counter operation in complementary PWM mode, and figure 27.40 shows an example of complementary PWM mode operation.

(a) Counter Operation

In complementary PWM mode, three counters—TCNT_3, TCNT_4, and TCNTS—perform up/down-count operations.

TCNT_3 is automatically initialized to the value set in TDDR when complementary PWM mode is selected and the CST bit in TSTR is 0.

When the CST bit is set to 1, TCNT_3 counts up to the value set in TGRA_3, then switches to down-counting when it matches TGRA_3. When the TCNT3 value matches TDDR, the counter switches to up-counting, and the operation is repeated in this way.

TCNT_4 is initialized to H'0000.

When the CST bit is set to 1, TCNT4 counts up in synchronization with TCNT_3, and switches to down-counting when it matches TCDR. On reaching H'0000, TCNT4 switches to up-counting, and the operation is repeated in this way.

TCNTS is a read-only counter. It need not be initialized.

When TCNT_3 matches TCDR during TCNT_3 and TCNT_4 up/down-counting, down-counting is started, and when TCNTS matches TCDR, the operation switches to up-counting. When TCNTS matches TGRA_3, it is cleared to H'0000.

When TCNT_4 matches TDDR during TCNT_3 and TCNT_4 down-counting, up-counting is started, and when TCNTS matches TDDR, the operation switches to down-counting. When TCNTS reaches H'0000, it is set with the value in TGRA_3.

TCNTS is compared with the compare register and temporary register in which the PWM duty is set during the count operation only.

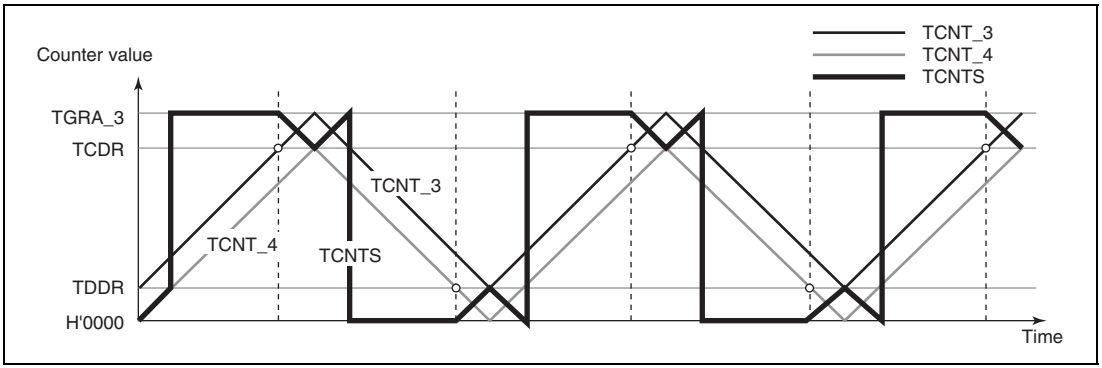


Figure 27.39 Complementary PWM Mode Counter Operation

(b) Register Operation

In complementary PWM mode, nine registers are used, comprising compare registers, buffer registers, and temporary registers. Figure 27.40 shows an example of complementary PWM mode operation.

The registers which are constantly compared with the counters to perform PWM output are TGRB_3, TGRA_4, and TGRB_4. When these registers match the counter, the value set in bits OLSN and OLSP in the timer output control register (TOCR) is output.

The buffer registers for these compare registers are TGRD_3, TGRC_4, and TGRD_4.

Between a buffer register and compare register there is a temporary register. The temporary registers cannot be accessed by the CPU.

Data in a compare register is changed by writing the new data to the corresponding buffer register. The buffer registers can be read or written at any time.

The data written to a buffer register is constantly transferred to the temporary register in the Ta interval. Data is not transferred to the temporary register in the Tb interval. Data written to a buffer register in this interval is transferred to the temporary register at the end of the Tb interval.

The value transferred to a temporary register is transferred to the compare register when TCNTS for which the Tb interval ends matches TGRA_3 when counting up, or H'0000 when counting down. The timing for transfer from the temporary register to the compare register can be selected with bits MD3 to MD0 in the timer mode register (TMDR). Figure 27.40 shows an example in which the mode is selected in which the change is made in the trough.

In the tb interval (tb1 in figure 27.40) in which data transfer to the temporary register is not performed, the temporary register has the same function as the compare register, and is compared with the counter. In this interval, therefore, there are two compare match registers for one-phase output, with the compare register containing the pre-change data, and the temporary register containing the new data. In this interval, the three counters—TCNT_3, TCNT_4, and TCNTS—and two registers—compare register and temporary register—are compared, and PWM output controlled accordingly.

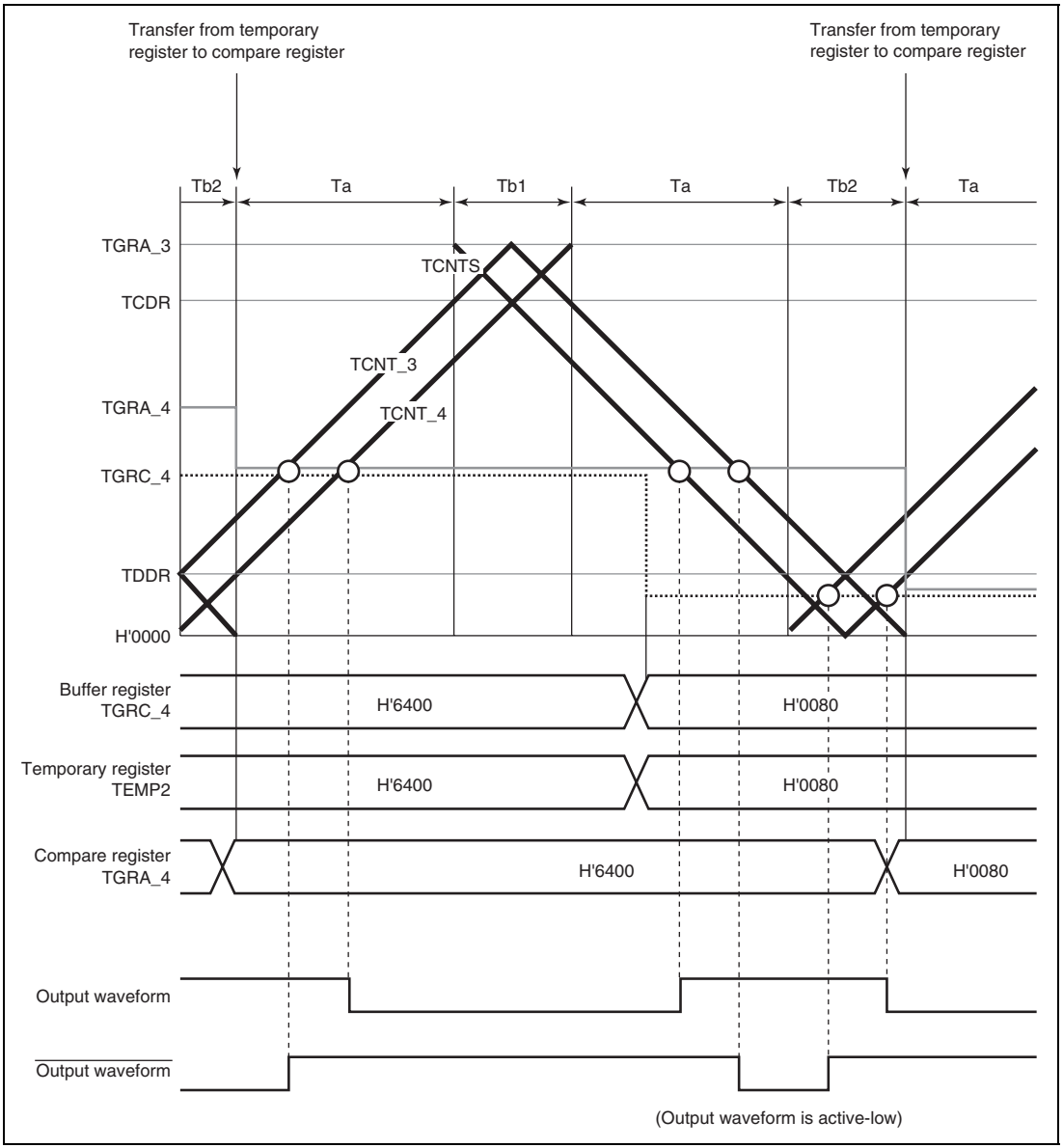


Figure 27.40 Example of Complementary PWM Mode Operation

(c) Initialization

In complementary PWM mode, there are six registers that must be initialized. In addition, there is a register that specifies whether to generate dead time (it should be used only when dead time generation should be disabled).

Before setting complementary PWM mode with bits MD3 to MD0 in the timer mode register (TMDR), the following initial register values must be set.

TGRC_3 operates as the buffer register for TGRA_3, and should be set with 1/2 the PWM carrier cycle + dead time Td. The timer cycle buffer register (TCBR) operates as the buffer register for the timer cycle data register (TCDR), and should be set with 1/2 the PWM carrier cycle. Set dead time Td in the timer dead time data register (TDDR).

When dead time is not needed, the TDER bit in the timer dead time enable register (TDER) should be cleared to 0, TGRC_3 and TGRA_3 should be set to 1/2 the PWM carrier cycle + 1, and TDDR should be set to 1.

Set the respective initial PWM duty values in buffer registers TGRD_3, TGRC_4, and TGRD_4.

The values set in the five buffer registers excluding TDDR are transferred simultaneously to the corresponding compare registers when complementary PWM mode is set.

Set TCNT_4 to H'0000 before setting complementary PWM mode.

Table 27.54 Registers and Counters Requiring Initialization

Register/Counter	Set Value
TGRC_3	1/2 PWM carrier cycle + dead time Td (1/2 PWM carrier cycle + 1 when dead time generation is disabled by TDER)
TDDR	Dead time Td (1 when dead time generation is disabled by TDER)
TCBR	1/2 PWM carrier cycle
TGRD_3, TGRC_4, TGRD_4	Initial PWM duty value for each phase
TCNT_4	H'0000

Note: The TGRC_3 set value must be the sum of 1/2 the PWM carrier cycle set in TCBR and dead time Td set in TDDR. When dead time generation is disabled by TDER, TGRC_3 must be set to 1/2 the PWM carrier cycle + 1.

(d) PWM Output Level Setting

In complementary PWM mode, the PWM pulse output level is set with bits OLSN and OLSP in timer output control register 1 (TOCR1) or bits OLS1P to OLS3P and OLS1N to OLS3N in timer output control register 2 (TOCR2).

The output level can be set for each of the three positive phases and three negative phases of 6-phase output.

Complementary PWM mode should be cleared before setting or changing output levels.

(e) Dead Time Setting

In complementary PWM mode, PWM pulses are output with a non-overlapping relationship between the positive and negative phases. This non-overlap time is called the dead time.

The non-overlap time is set in the timer dead time data register (TDDR). The value set in TDDR is used as the TCNT_3 counter start value, and creates non-overlap between TCNT_3 and TCNT_4. Complementary PWM mode should be cleared before changing the contents of TDDR.

(f) Dead Time Suppressing

Dead time generation is suppressed by clearing the TDER bit in the timer dead time enable register (TDER) to 0. TDER can be cleared to 0 only when 0 is written to it after reading TDER = 1.

TGRA_3 and TGRC_3 should be set to 1/2 PWM carrier cycle + 1 and the timer dead time data register (TDDR) should be set to 1.

By the above settings, PWM waveforms without dead time can be obtained. Figure 27.41 shows an example of operation without dead time.

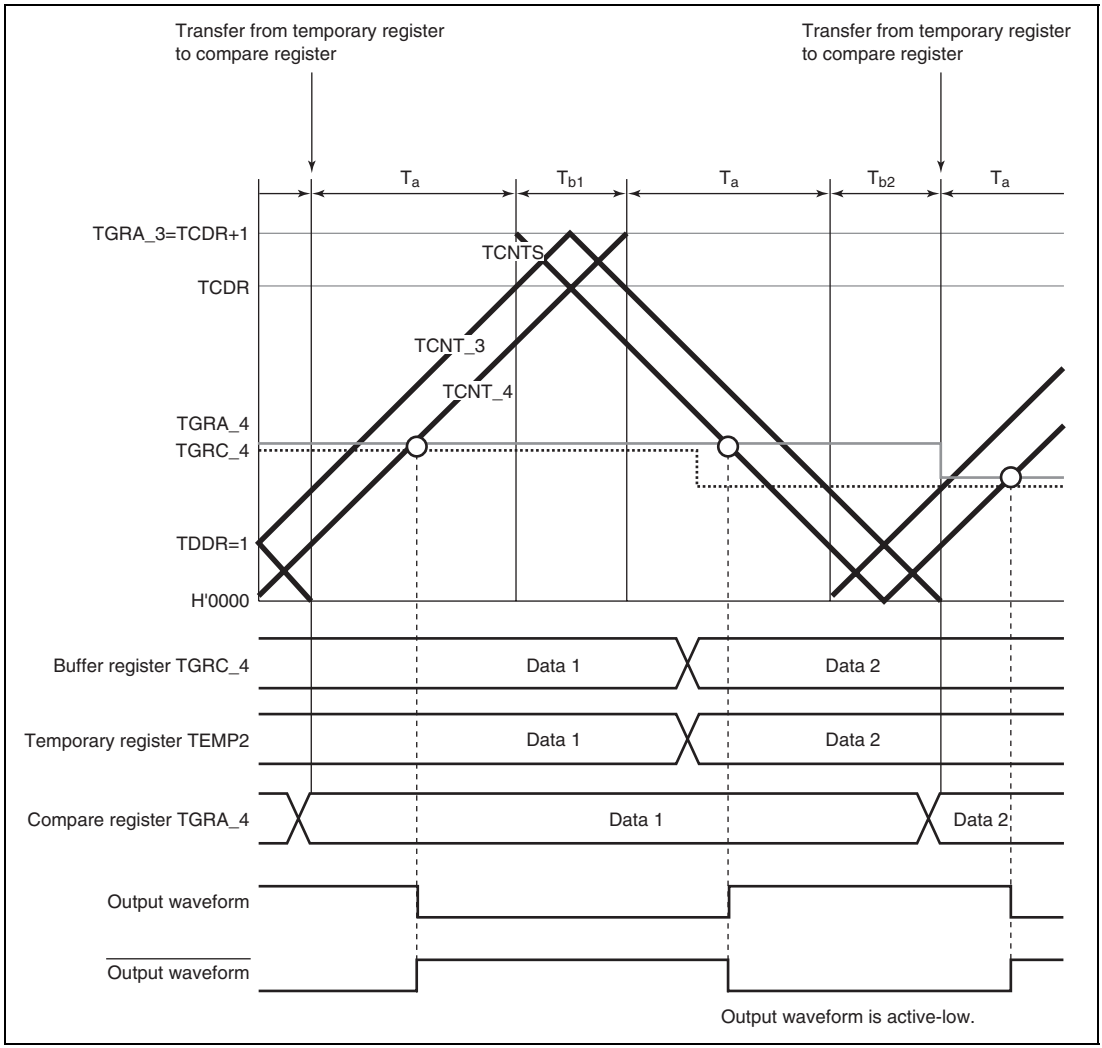


Figure 27.41 Example of Operation without Dead Time

(g) PWM Cycle Setting

In complementary PWM mode, the PWM pulse cycle is set in two registers—TGRA_3, in which the TCNT_3 upper limit value is set, and TCDR, in which the TCNT_4 upper limit value is set. The settings should be made so as to achieve the following relationship between these two registers:

With dead time: TGRA_3 set value = TCDR set value + TDDR set value
Without dead time: TGRA_3 set value = TCDR set value + 1

The TGRA_3 and TCDR settings are made by setting the values in buffer registers TGRC_3 and TCBR. The values set in TGRC_3 and TCBR are transferred simultaneously to TGRA_3 and TCDR in accordance with the transfer timing selected with bits MD3 to MD0 in the timer mode register (TMDR).

The updated PWM cycle is reflected from the next cycle when the data update is performed at the crest, and from the current cycle when performed in the trough. Figure 27.42 illustrates the operation when the PWM cycle is updated at the crest.

See (h) Register Data Updating, for the method of updating the data in each buffer register.

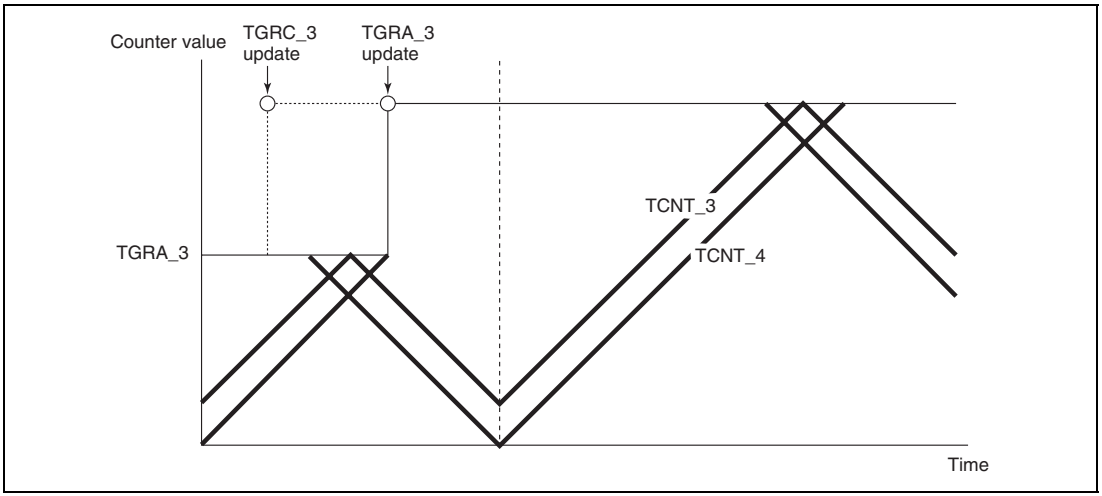


Figure 27.42 Example of PWM Cycle Updating

(h) Register Data Updating

In complementary PWM mode, the buffer register is used to update the data in a compare register. The update data can be written to the buffer register at any time. There are five PWM duty and carrier cycle registers that have buffer registers and can be updated during operation.

There is a temporary register between each of these registers and its buffer register. When subcounter TCNTS is not counting, if buffer register data is updated, the temporary register value is also rewritten. Transfer is not performed from buffer registers to temporary registers when TCNTS is counting; in this case, the value written to a buffer register is transferred after TCNTS halts.

The temporary register value is transferred to the compare register at the data update timing set with bits MD3 to MD0 in the timer mode register (TMDR). Figure 27.43 shows an example of data updating in complementary PWM mode. This example shows the mode in which data updating is performed at both the counter crest and trough.

When rewriting buffer register data, a write to TGRD_4 must be performed at the end of the update. Data transfer from the buffer registers to the temporary registers is performed simultaneously for all five registers after the write to TGRD_4.

A write to TGRD_4 must be performed after writing data to the registers to be updated, even when not updating all five registers, or when updating the TGRD_4 data. In this case, the data written to TGRD_4 should be the same as the data prior to the write operation.

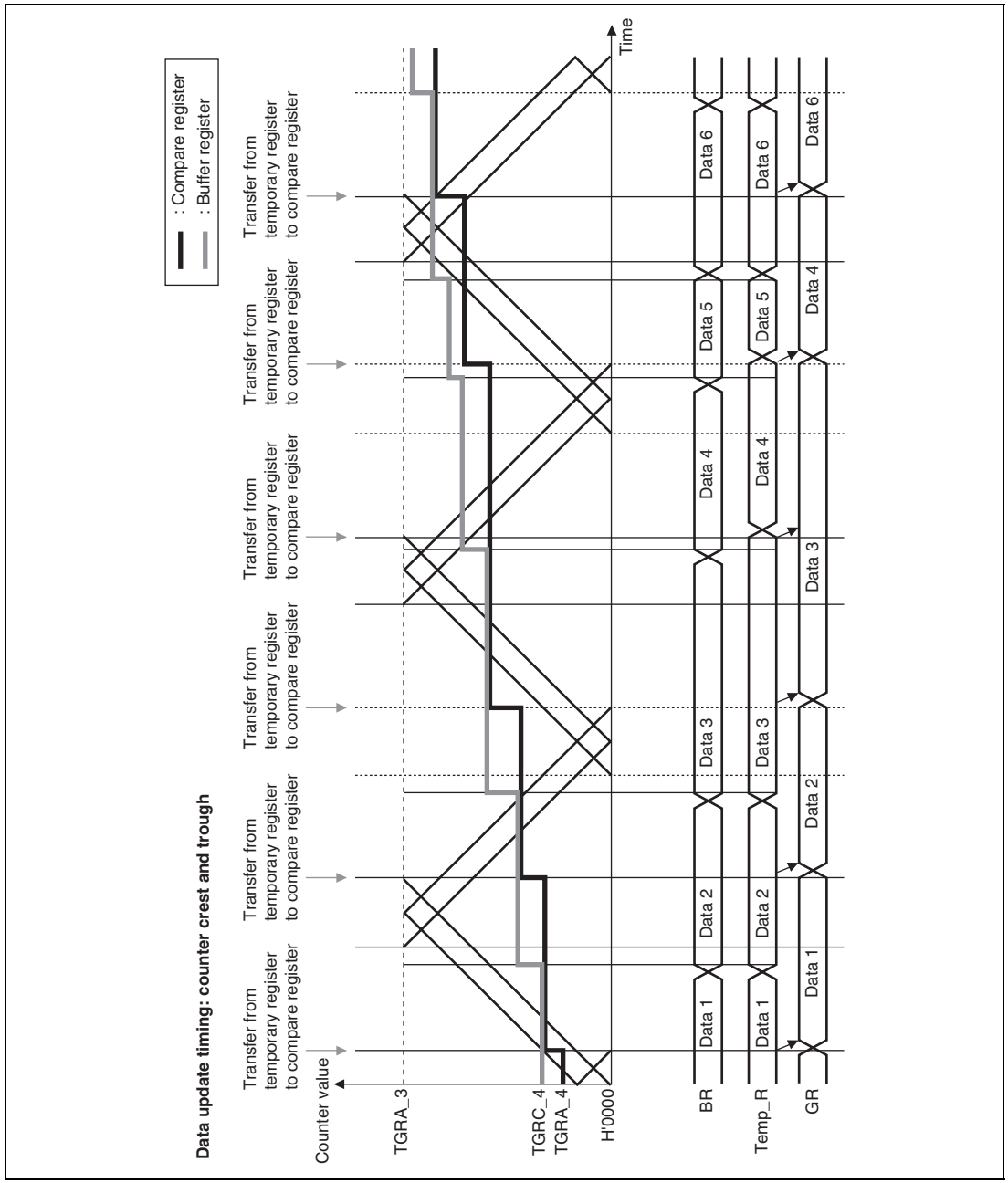


Figure 27.43 Example of Data Update in Complementary PWM Mode

(i) Initial Output in Complementary PWM Mode

In complementary PWM mode, the initial output is determined by the setting of bits OLSN and OLSP in timer output control register 1 (TOCR1) or bits OLS1N to OLS3N and OLS1P to OLS3P in timer output control register 2 (TOCR2).

This initial output is the PWM pulse non-active level, and is output from when complementary PWM mode is set with the timer mode register (TMDR) until TCNT_4 exceeds the value set in the dead time register (TDDR). Figure 27.44 shows an example of the initial output in complementary PWM mode.

An example of the waveform when the initial PWM duty value is smaller than the TDDR value is shown in figure 27.45.

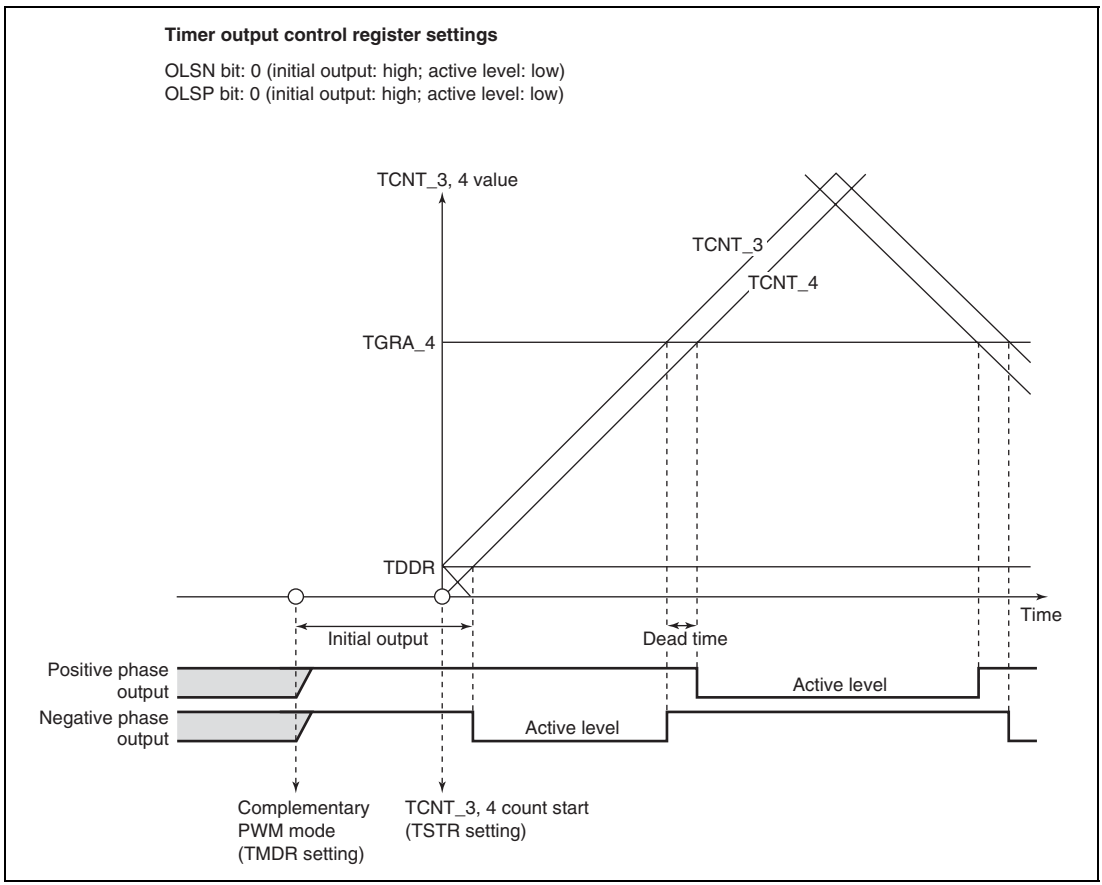


Figure 27.44 Example of Initial Output in Complementary PWM Mode (1)

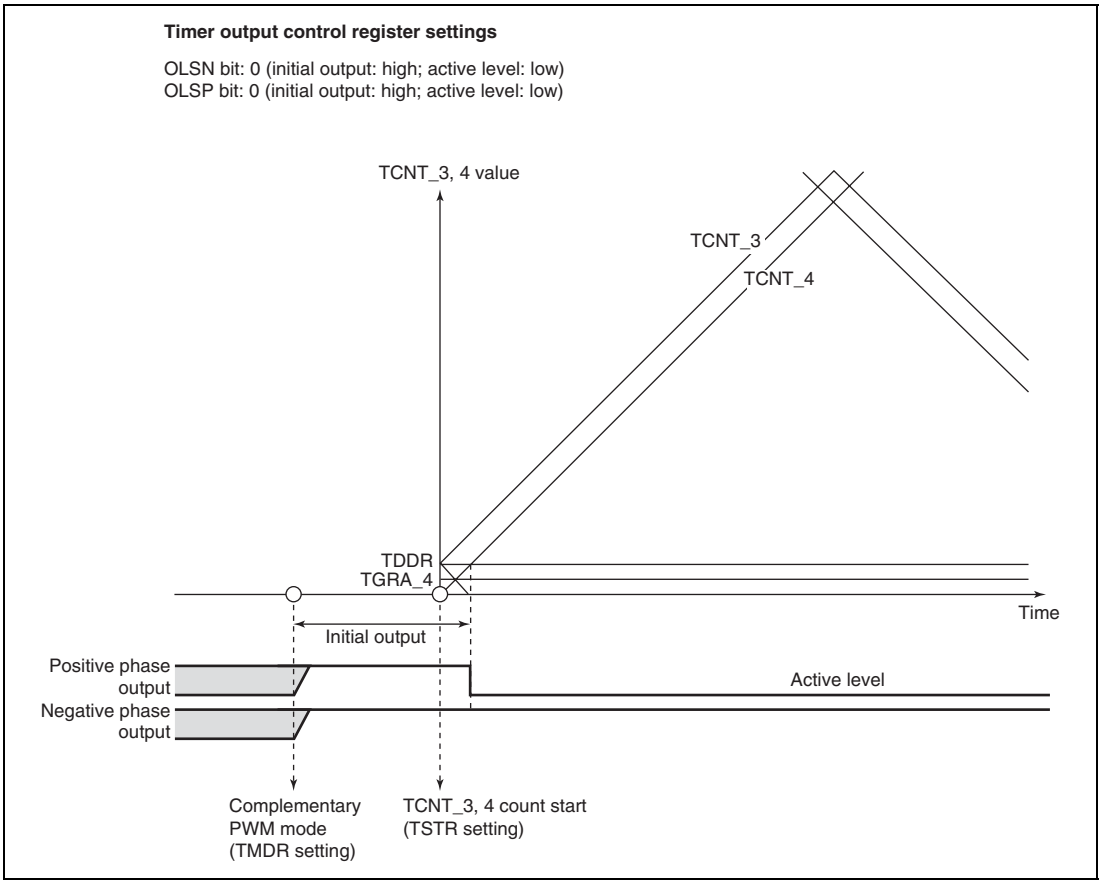


Figure 27.45 Example of Initial Output in Complementary PWM Mode (2)

(j) Complementary PWM Mode PWM Output Generation Method

In complementary PWM mode, 3-phase output is performed of PWM waveforms with a non-overlap time between the positive and negative phases. This non-overlap time is called the dead time.

A PWM waveform is generated by output of the output level selected in the timer output control register in the event of a compare-match between a counter and data register. While TCNTS is counting, data register and temporary register values are simultaneously compared to create consecutive PWM pulses from 0 to 100%. The relative timing of on and off compare-match occurrence may vary, but the compare-match that turns off each phase takes precedence to secure the dead time and ensure that the positive phase and negative phase on times do not overlap. Figures 27.46 to 27.48 show examples of waveform generation in complementary PWM mode.

The positive phase/negative phase off timing is generated by a compare-match with the solid-line counter, and the on timing by a compare-match with the dotted-line counter operating with a delay of the dead time behind the solid-line counter. In the T1 period, compare-match **a** that turns off the negative phase has the highest priority, and compare-matches occurring prior to **a** are ignored. In the T2 period, compare-match **c** that turns off the positive phase has the highest priority, and compare-matches occurring prior to **c** are ignored.

In normal cases, compare-matches occur in the order **a** → **b** → **c** → **d** (or **c** → **d** → **a'** → **b'**), as shown in figure 27.46.

If compare-matches deviate from the **a** → **b** → **c** → **d** order, since the time for which the negative phase is off is less than twice the dead time, the figure shows the positive phase is not being turned on. If compare-matches deviate from the **c** → **d** → **a'** → **b'** order, since the time for which the positive phase is off is less than twice the dead time, the figure shows the negative phase is not being turned on.

If compare-match **c** occurs first following compare-match **a**, as shown in figure 27.47, compare-match **b** is ignored, and the negative phase is turned off by compare-match **d**. This is because turning off of the positive phase has priority due to the occurrence of compare-match **c** (positive phase off timing) before compare-match **b** (positive phase on timing) (consequently, the waveform does not change since the positive phase goes from off to off).

Similarly, in the example in figure 27.48, compare-match **a'** with the new data in the temporary register occurs before compare-match **c**, but other compare-matches occurring up to **c**, which turns off the positive phase, are ignored. As a result, the negative phase is not turned on.

Thus, in complementary PWM mode, compare-matches at turn-off timings take precedence, and turn-on timing compare-matches that occur before a turn-off timing compare-match are ignored.

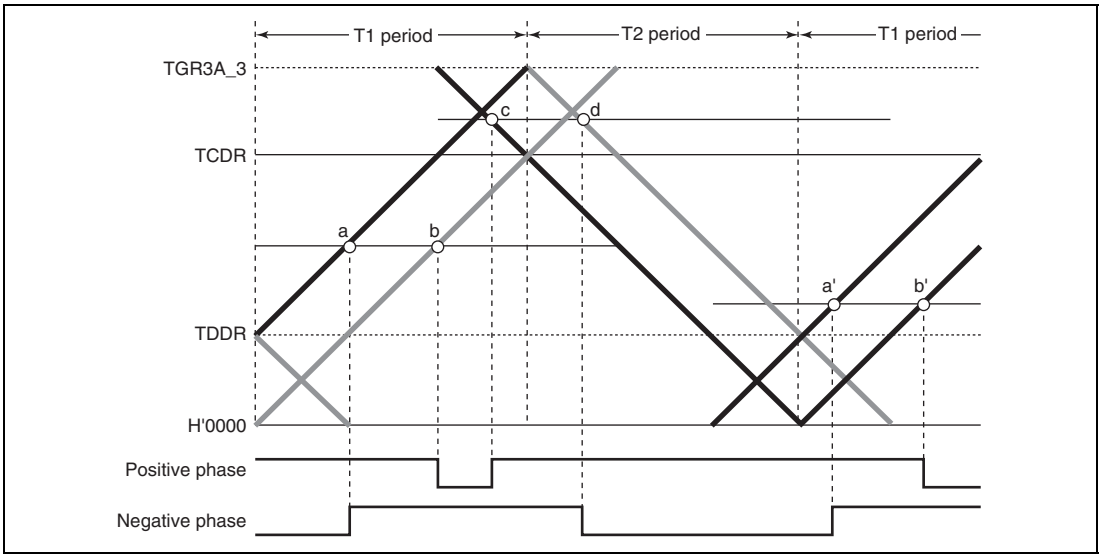


Figure 27.46 Example of Complementary PWM Mode Waveform Output (1)

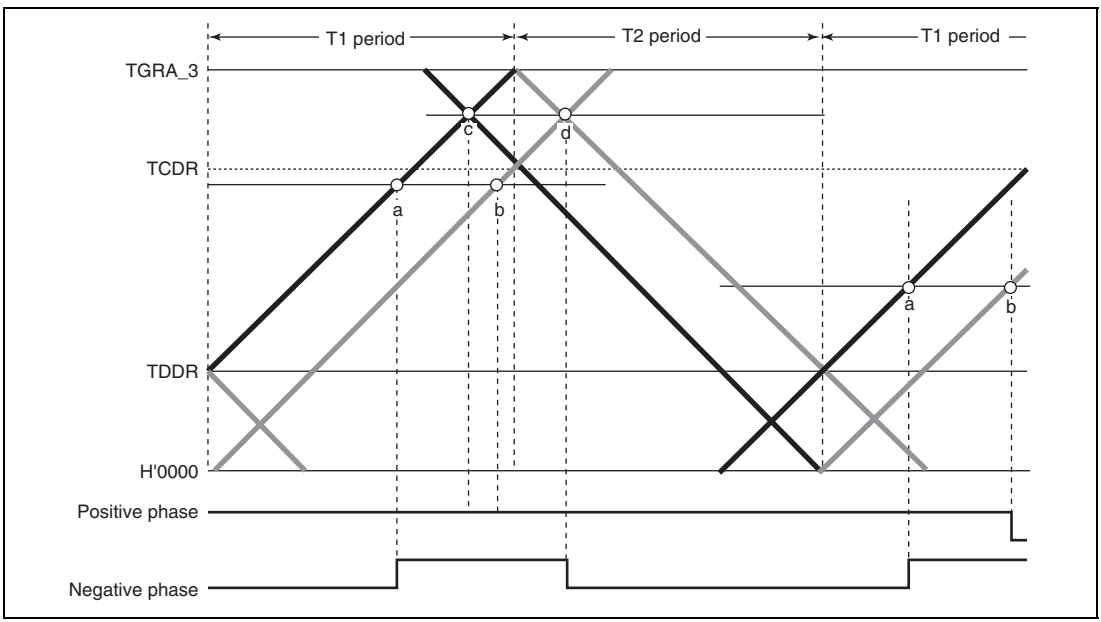


Figure 27.47 Example of Complementary PWM Mode Waveform Output (2)

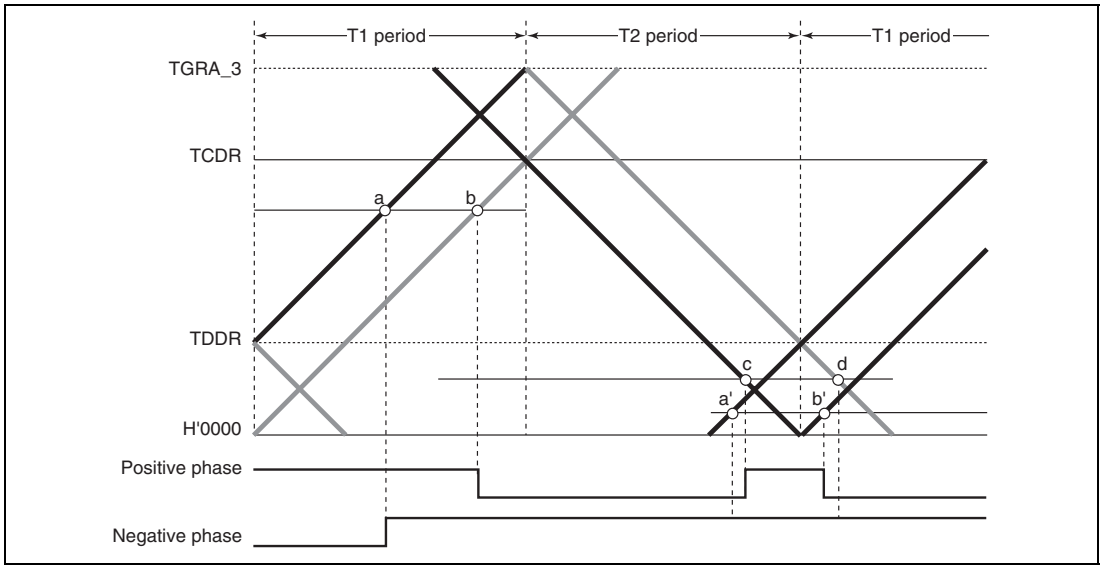


Figure 27.48 Example of Complementary PWM Mode Waveform Output (3)

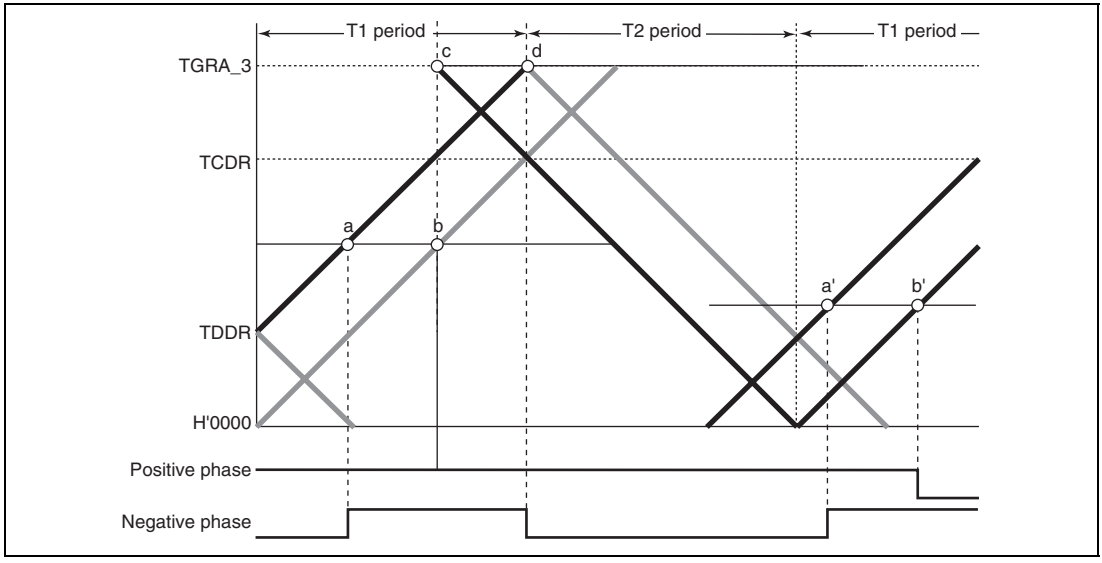


Figure 27.49 Example of Complementary PWM Mode 0% and 100% Waveform Output (1)

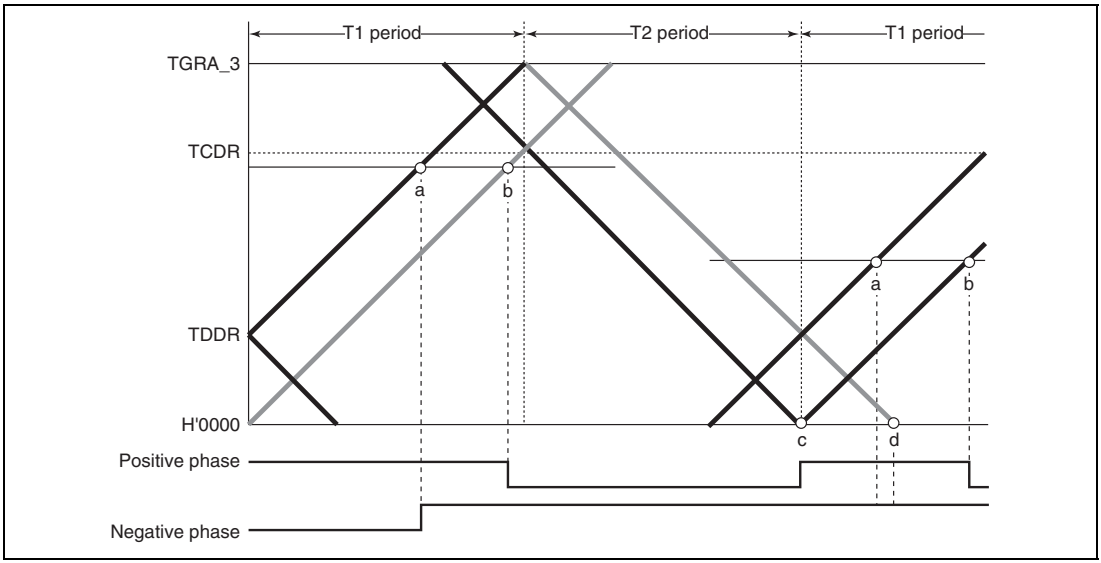


Figure 27.50 Example of Complementary PWM Mode 0% and 100% Waveform Output (2)

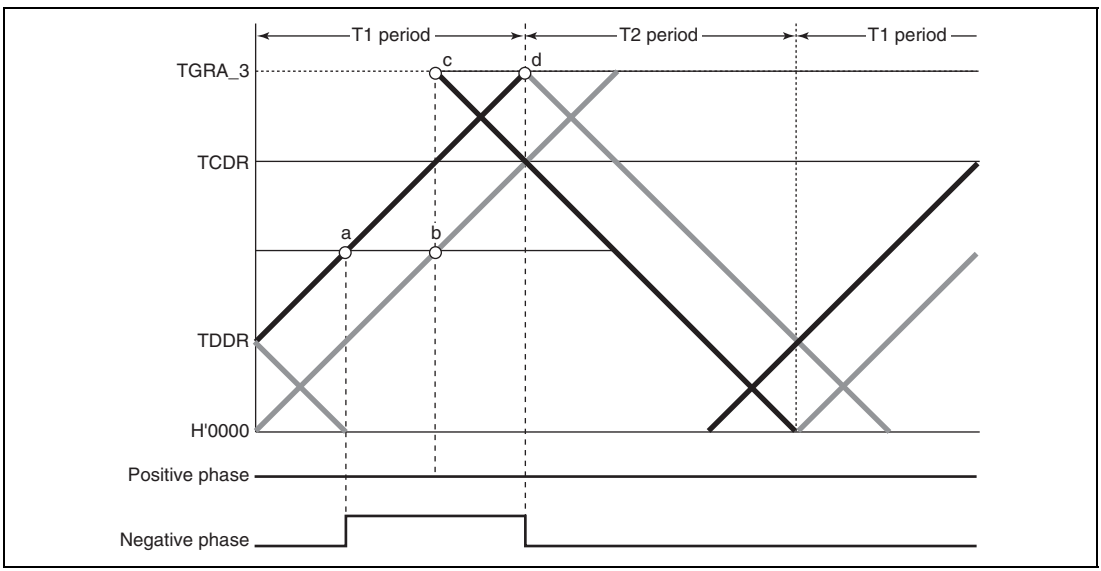


Figure 27.51 Example of Complementary PWM Mode 0% and 100% Waveform Output (3)

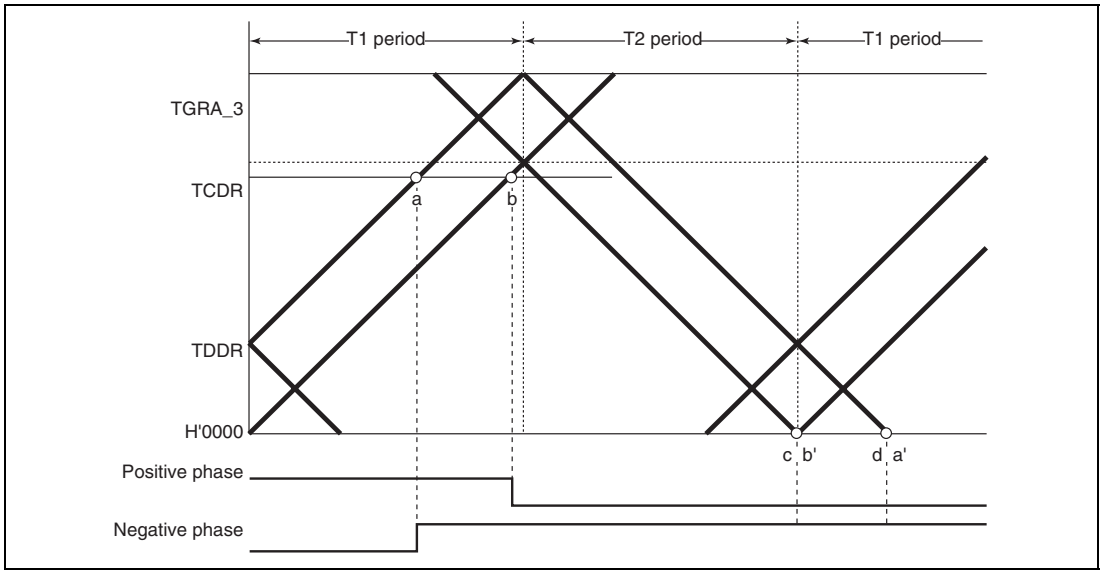


Figure 27.52 Example of Complementary PWM Mode 0% and 100% Waveform Output (4)

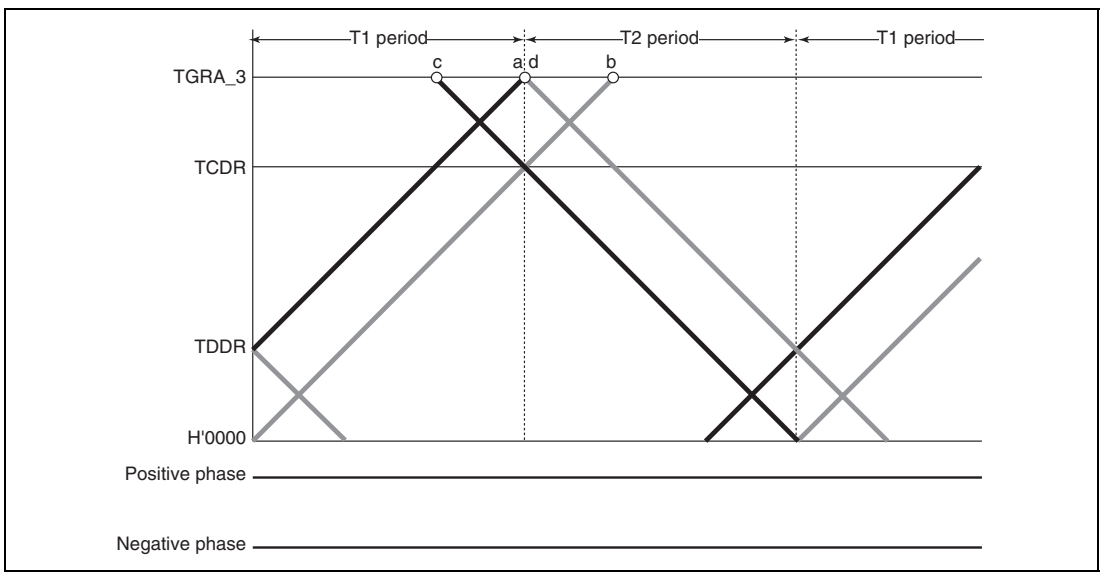


Figure 27.53 Example of Complementary PWM Mode 0% and 100% Waveform Output (5)

(k) Complementary PWM Mode 0% and 100% Duty Output

In complementary PWM mode, 0% and 100% duty cycles can be output as required. Figures 27.49 to 27.53 show output examples.

100% duty output is performed when the data register value is set to H'0000. The waveform in this case has a positive phase with a 100% on-state. 0% duty output is performed when the data register value is set to the same value as TGRA_3. The waveform in this case has a positive phase with a 100% off-state.

On and off compare-matches occur simultaneously, but if a turn-on compare-match and turn-off compare-match for the same phase occur simultaneously, both compare-matches are ignored and the waveform does not change.

(l) Toggle Output Synchronized with PWM Cycle

In complementary PWM mode, toggle output can be performed in synchronization with the PWM carrier cycle by setting the PSYE bit to 1 in the timer output control register (TOCR). An example of a toggle output waveform is shown in figure 27.54.

This output is toggled by a compare-match between TCNT_3 and TGRA_3 and a compare-match between TCNT4 and H'0000.

The output pin for this toggle output is the TIOC3A pin. The initial output is 1.

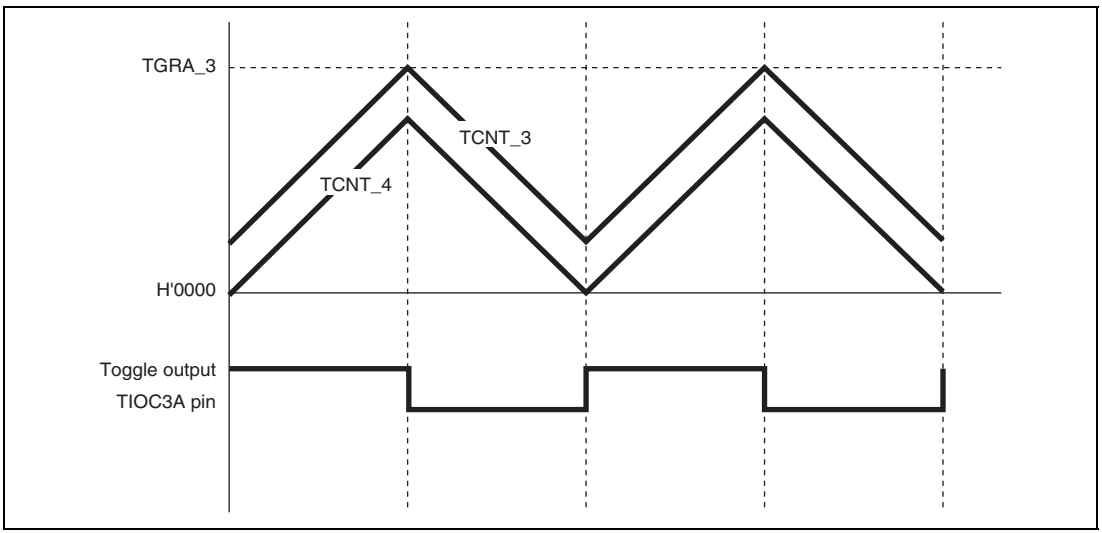


Figure 27.54 Example of Toggle Output Waveform Synchronized with PWM Output

(m) Counter Clearing by Another Channel

In complementary PWM mode, by setting a mode for synchronization with another channel by means of the timer synchronous register (TSYR), and selecting synchronous clearing with bits CCLR2 to CCLR0 in the timer control register (TCR), it is possible to have TCNT_3, TCNT_4, and TCNTS cleared by another channel.

Figure 27.55 illustrates the operation.

Use of this function enables counter clearing and restarting to be performed by means of an external signal.

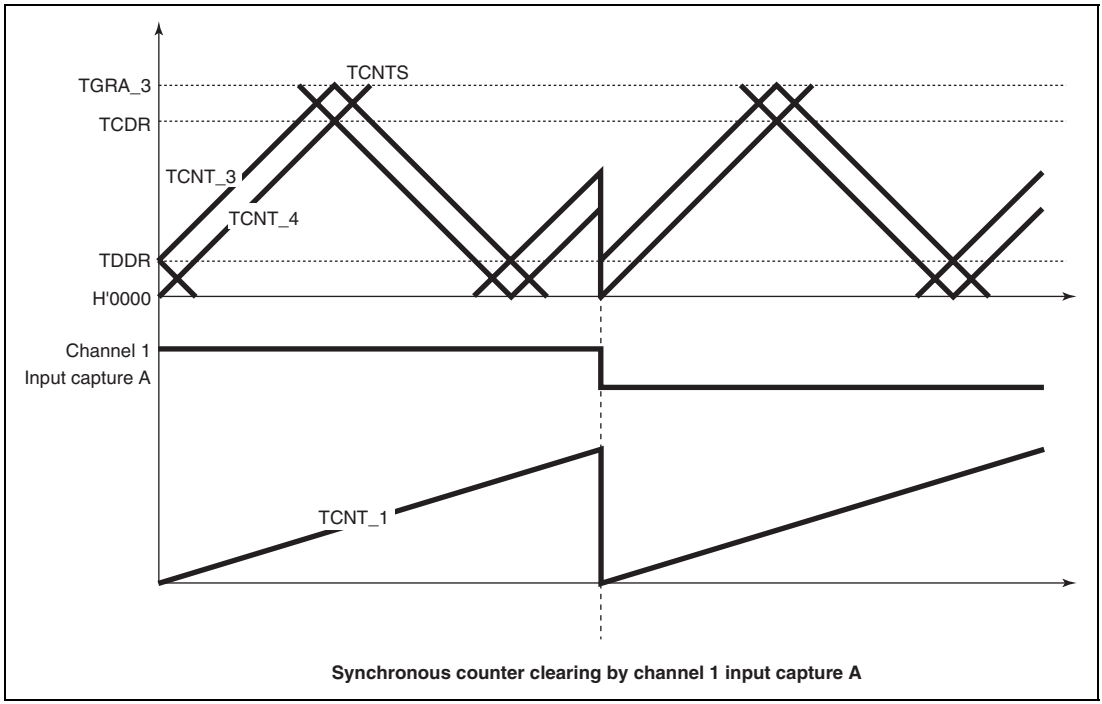


Figure 27.55 Counter Clearing Synchronized with Another Channel

(n) Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode

Setting the WRE bit in TWCR to 1 suppresses initial output when synchronous counter clearing occurs in the Tb interval at the trough in complementary PWM mode and controls abrupt change in duty cycle at synchronous counter clearing.

Initial output suppression is applicable only when synchronous clearing occurs in the Tb interval at the trough as indicated by (10) or (11) in figure 27.56. When synchronous clearing occurs outside that interval, the initial value specified by the OLS bits in TOCR is output. Even in the Tb interval at the trough, if synchronous clearing occurs in the initial value output period (indicated by (1) in figure 27.56) immediately after the counters start operation, initial value output is not suppressed.

To suppress initial output, set all of the comparison registers (TGRB_3, TGRA_4, and TGRB_4) to at least twice the setting of the timer dead time data register (TDDR). If synchronous clearing occurs while a register has a value less than twice that of the TDDR, the dead time in the PWM output may be shortened (or eliminated), or the active level may be incorrectly output on the inverse-phase PWM output pins during the initial-output suppression period. For details, see section 27.7.23, Notes on Control of Output Waveforms in Synchronous Counter Clearing in Complementary PWM Mode.

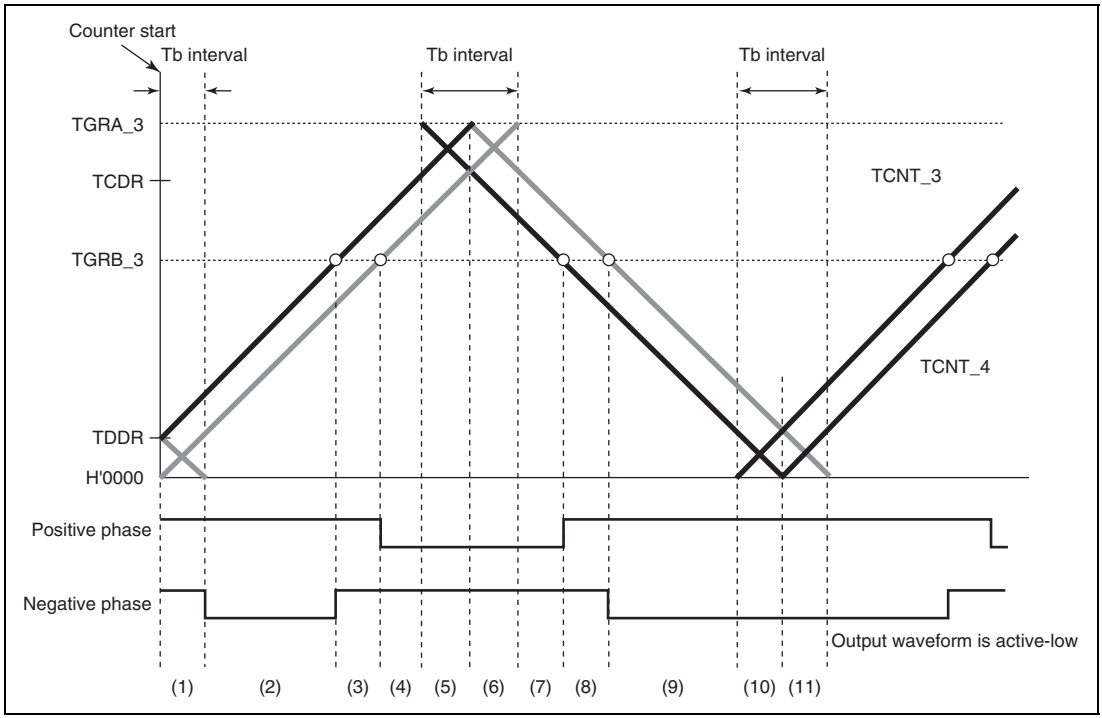


Figure 27.56 Timing for Synchronous Counter Clearing

- Example of Procedure for Setting Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode

  An example of the procedure for setting output waveform control at synchronous counter clearing in complementary PWM mode is shown in figure 27.57.

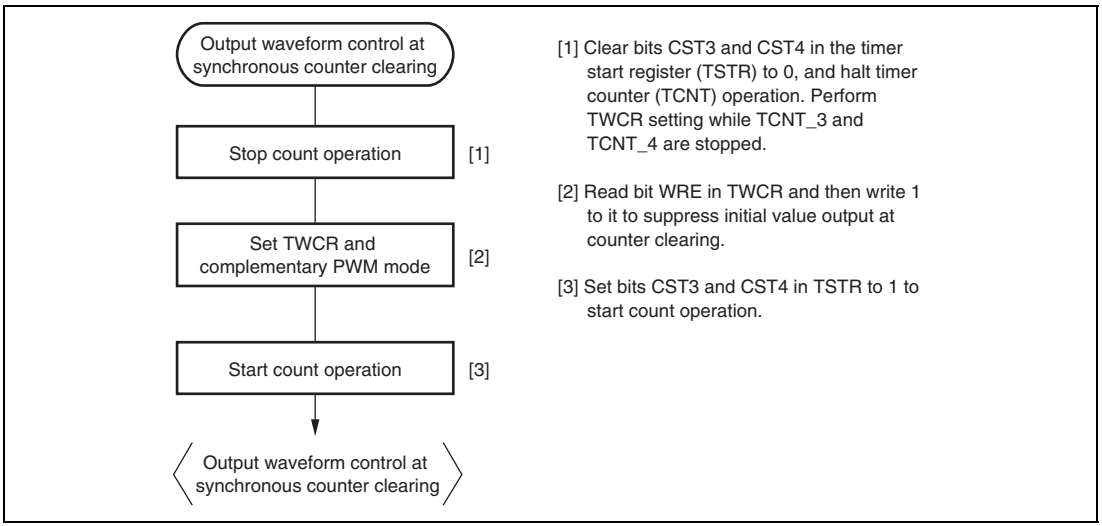


Figure 27.57 Example of Procedure for Setting Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode

- Examples of Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode

  Figures 27.58 to 27.61 show examples of output waveform control in which this module operates in complementary PWM mode and synchronous counter clearing is generated while the WRE bit in TWCR is set to 1. In the examples shown in figures 27.58 to 27.61, synchronous counter clearing occurs at timing (3), (6), (8), and (11) shown in figure 27.56, respectively.

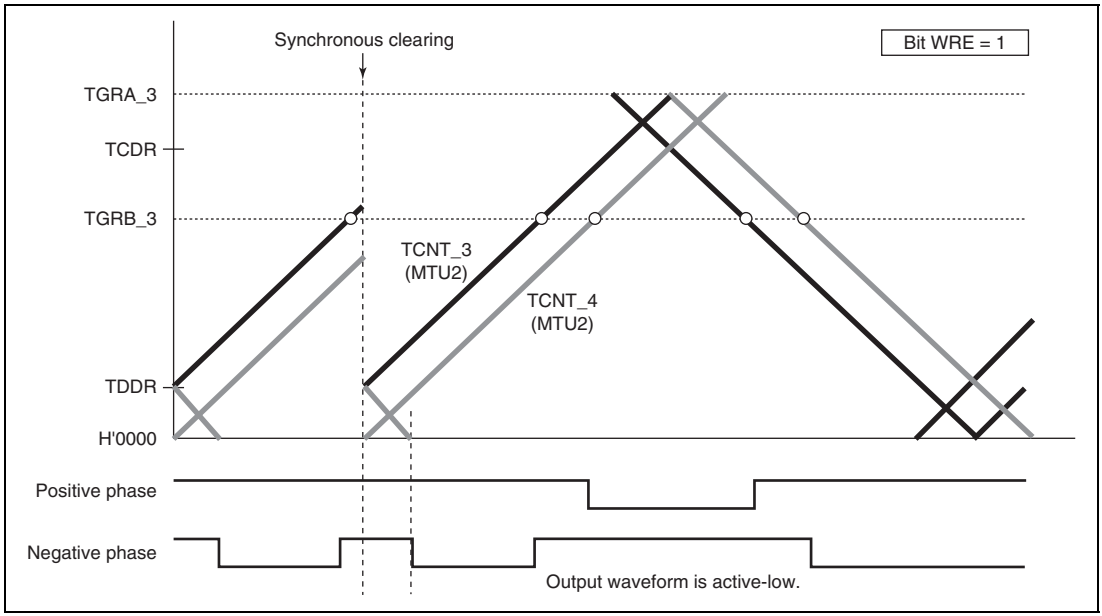


Figure 27.58 Example of Synchronous Clearing in Dead Time during Up-Counting (Timing (3) in Figure 27.56; Bit WRE of TWCR is 1)

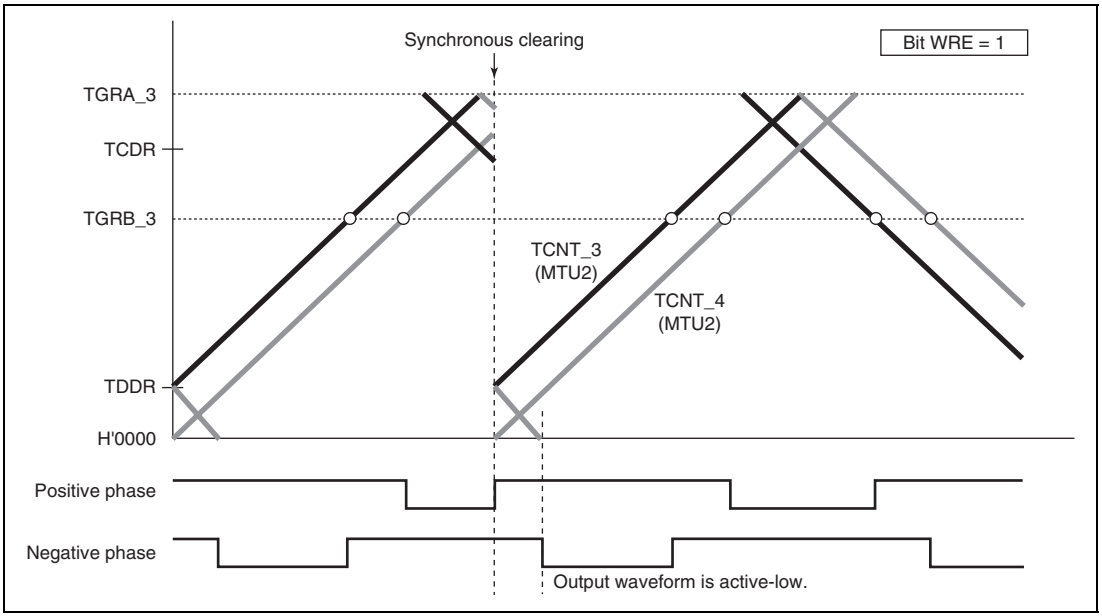


Figure 27.59 Example of Synchronous Clearing in Interval Tb at Crest (Timing (6) in Figure 27.56; Bit WRE of TWCR is 1)

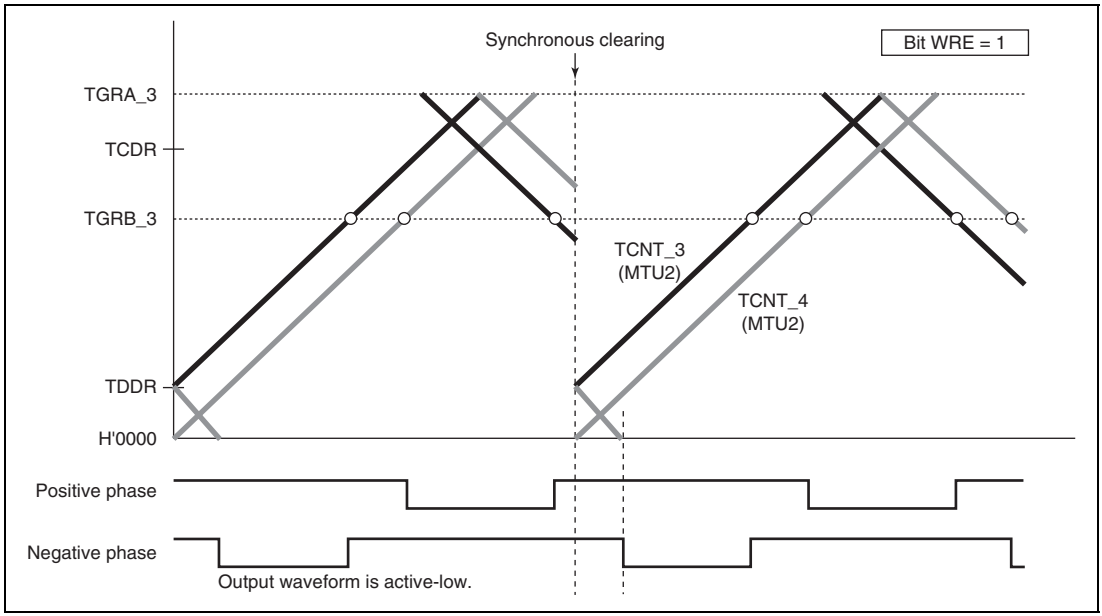


Figure 27.60 Example of Synchronous Clearing in Dead Time during Down-Counting (Timing (8) in Figure 27.56; Bit WRE of TWCR is 1)

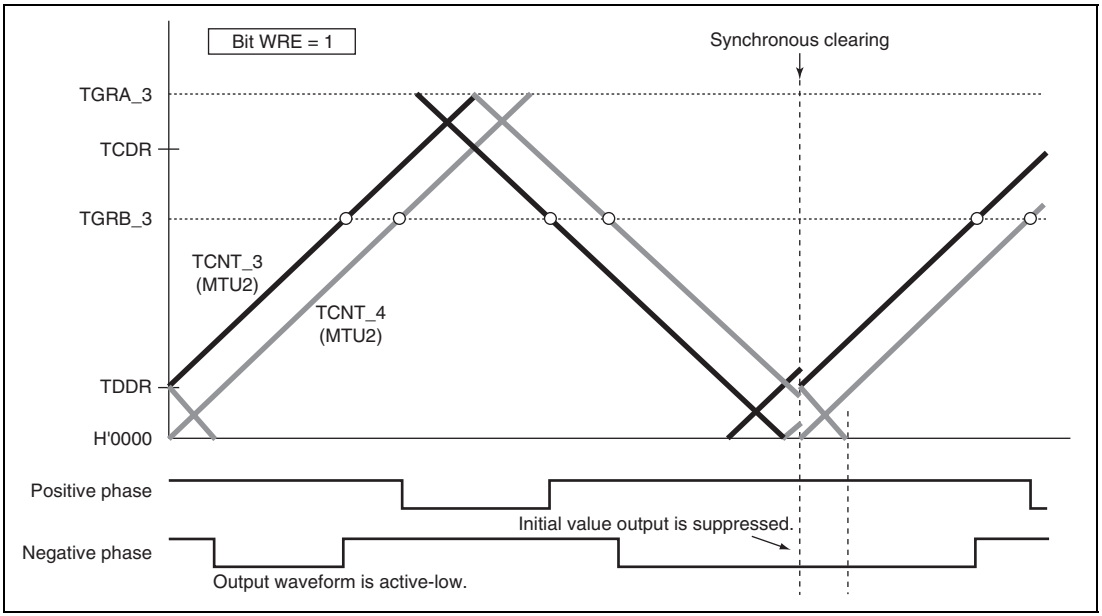


Figure 27.61 Example of Synchronous Clearing in Interval Tb at Trough (Timing (11) in Figure 27.56; Bit WRE of TWCR is 1)

(o) Counter Clearing by TGRA_3 Compare Match

In complementary PWM mode, by setting the CCE bit in the timer waveform control register (TWCR), it is possible to have TCNT_3, TCNT_4, and TCNTS cleared by TGRA_3 compare match.

Figure 27.62 illustrates an operation example.

Notes: 1. Use this function only in complementary PWM mode 1 (transfer at crest)
2. Do not specify synchronous clearing by another channel (do not set the SYNC0 to SYNC4 bits in the timer synchronous register (TSYR) to 1).
3. Do not set the PWM duty value to H'0000.
4. Do not set the PSYE bit in timer output control register 1 (TOCR1) to 1.

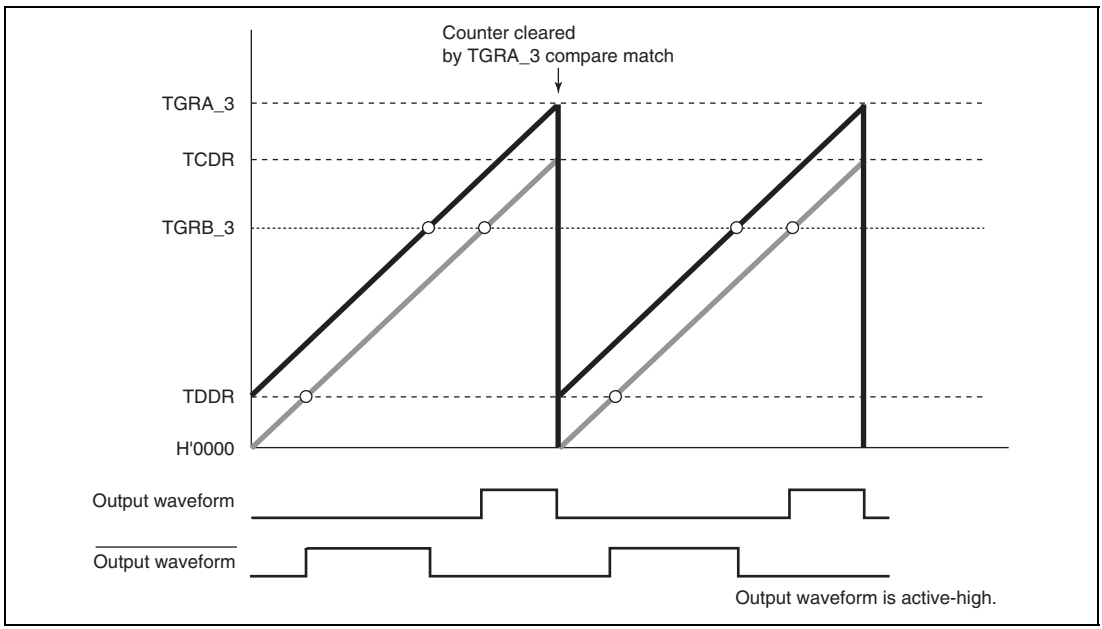


Figure 27.62 Example of Counter Clearing Operation by TGRA_3 Compare Match

(p) Example of AC Synchronous Motor (Brushless DC Motor) Drive Waveform Output

In complementary PWM mode, a brushless DC motor can easily be controlled using the timer gate control register (TGCR). Figures 27.63 to 27.66 show examples of brushless DC motor drive waveforms created using TGCR.

When output phase switching for a 3-phase brushless DC motor is performed by means of external signals detected with a Hall element, etc., clear the FB bit in TGCR to 0. In this case, the external signals indicating the polarity position are input to channel 0 timer input pins TIOC0A, TIOC0B, and TIOC0C (set with the general I/O ports). When an edge is detected at pin TIOC0A, TIOC0B, or TIOC0C, the output on/off state is switched automatically.

When the FB bit is 1, the output on/off state is switched when the UF, VF, or WF bit in TGCR is cleared to 0 or set to 1.

The drive waveforms are output from the complementary PWM mode 6-phase output pins. With this 6-phase output, in the case of on output, it is possible to use complementary PWM mode output and perform chopping output by setting the N bit or P bit to 1. When the N bit or P bit is 0, level output is selected.

The 6-phase output active level (on output level) can be set with the OLSN and OLSP bits in the timer output control register (TOCR) regardless of the setting of the N and P bits.

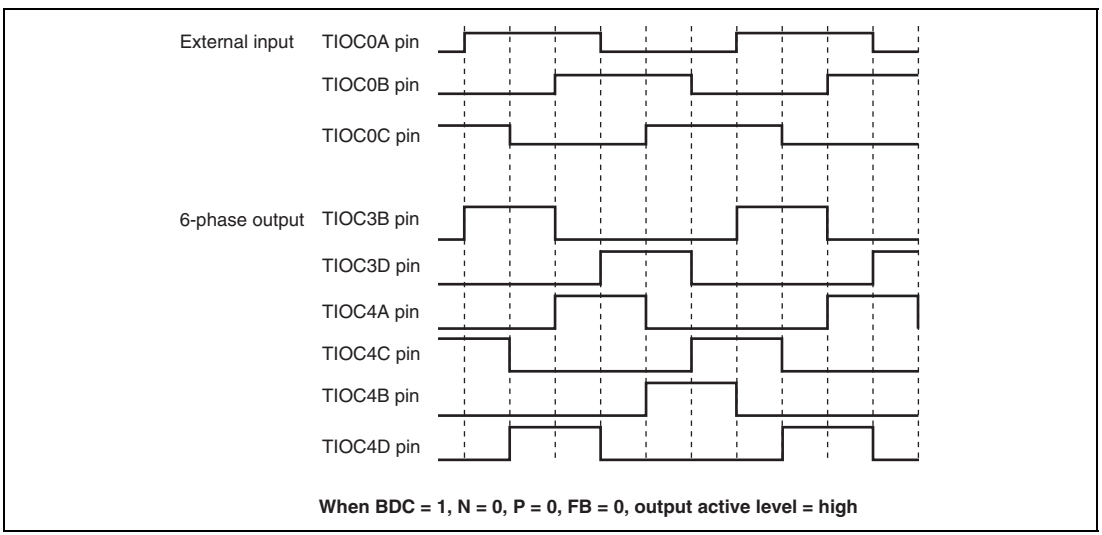


Figure 27.63 Example of Output Phase Switching by External Input (1)

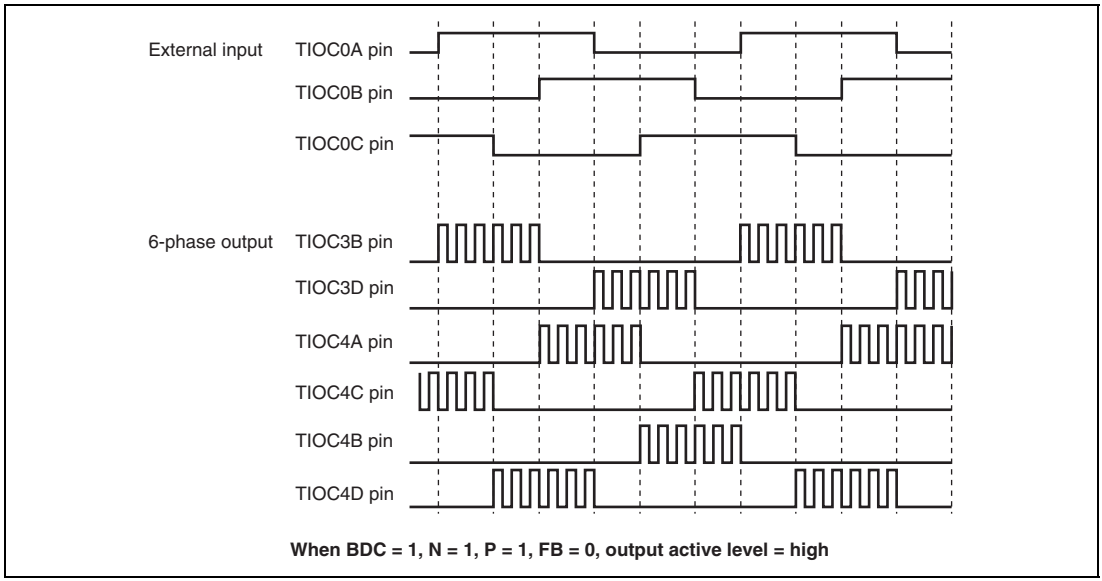


Figure 27.64 Example of Output Phase Switching by External Input (2)

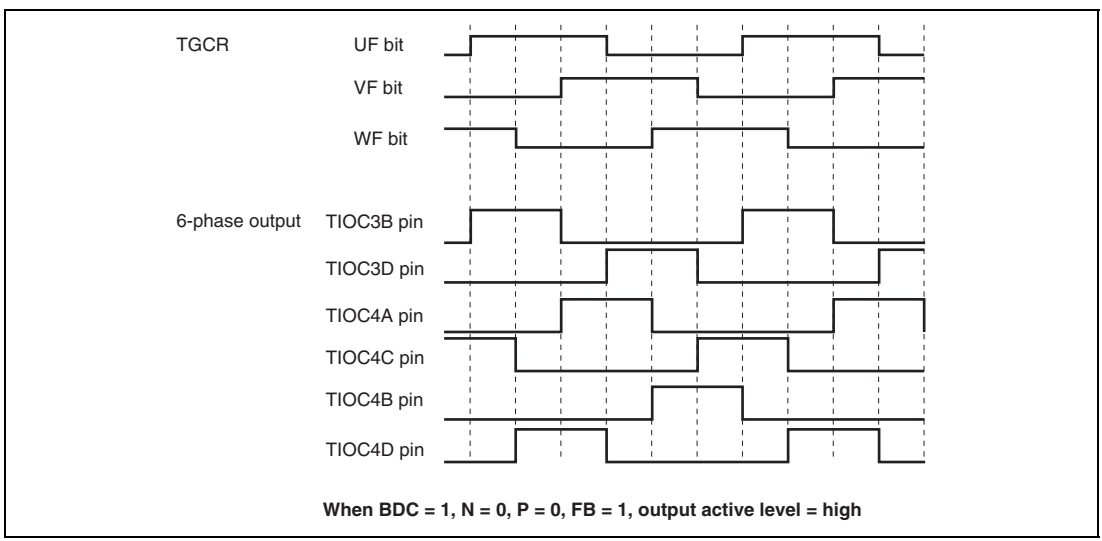


Figure 27.65 Example of Output Phase Switching by Means of UF, VF, WF Bit Settings (1)

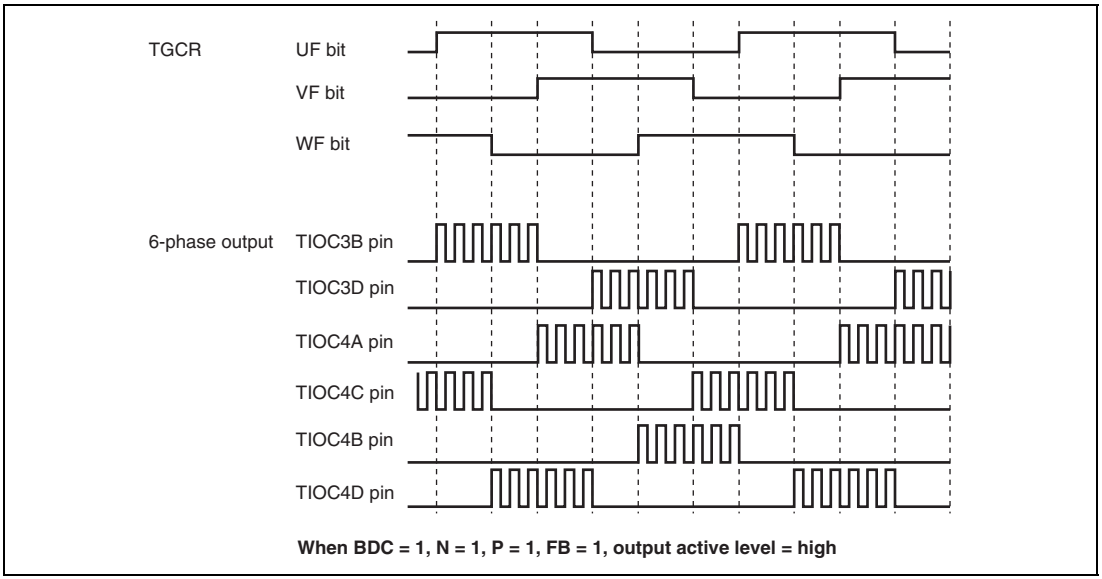


Figure 27.66 Example of Output Phase Switching by Means of UF, VF, WF Bit Settings (2)

(q) A/D Converter Start Request Setting

In complementary PWM mode, an A/D converter start request can be issued using a TGRA_3 compare-match, TCNT_4 underflow (trough), or compare-match on a channel other than channels 3 and 4.

When start requests using a TGRA_3 compare-match are specified, A/D conversion can be started at the crest of the TCNT_3 count.

A/D converter start requests can be set by setting the TTGE bit to 1 in the timer interrupt enable register (TIER). To issue an A/D converter start request at a TCNT_4 underflow (trough), set the TTGE2 bit in TIER_4 to 1.

(3) Interrupt Skipping in Complementary PWM Mode

Interrupts TGIA_3 (at the crest) and TCIV_4 (at the trough) in channels 3 and 4 can be skipped up to seven times by making settings in the timer interrupt skipping set register (TITCR).

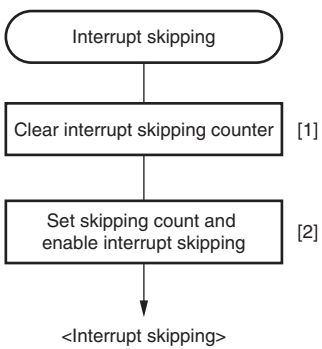
Transfers from a buffer register to a temporary register or a compare register can be skipped in coordination with interrupt skipping by making settings in the timer buffer transfer register (TBTER). For the linkage with buffer registers, refer to description (c), Buffer Transfer Control Linked with Interrupt Skipping, below.

A/D converter start requests generated by the A/D converter start request delaying function can also be skipped in coordination with interrupt skipping by making settings in the timer A/D converter request control register (TADCR). For the linkage with the A/D converter start request delaying function, refer to section 27.4.9, A/D Converter Start Request Delaying Function.

The setting of the timer interrupt skipping setting register (TITCR) must be done while the TGIA_3 and TCIV_4 interrupt requests are disabled by the settings of TIER_3 and TIER_4 along with under the conditions in which TGFA_3 and TCFV_4 flag settings by compare match never occur. Before changing the skipping count, be sure to clear the T3AEN and T4VEN bits to 0 to clear the skipping counter.

(a) Example of Interrupt Skipping Operation Setting Procedure

Figure 27.67 shows an example of the interrupt skipping operation setting procedure. Figure 27.68 shows the periods during which interrupt skipping count can be changed.



[1] Set bits T3AEN and T4VEN in the timer interrupt skipping set register (TITCR) to 0 to clear the skipping counter.

[2] Specify the interrupt skipping count within the range from 0 to 7 times in bits 3ACOR2 to 3ACOR0 and 4VCOR2 to 4VCOR0 in TITCR, and enable interrupt skipping through bits T3AEN and T4VEN.

Note: The setting of TITCR must be done while the TGIA_3 and TCIV_4 interrupt requests are disabled by the settings of TIER_3 and TIER_4 along with under the conditions in which TGFA_3 and TCFV_4 flag settings by compare match never occur. Before changing the skipping count, be sure to clear the T3AEN and T4VEN bits to 0 to clear the skipping counter.

Figure 27.67 Example of Interrupt Skipping Operation Setting Procedure

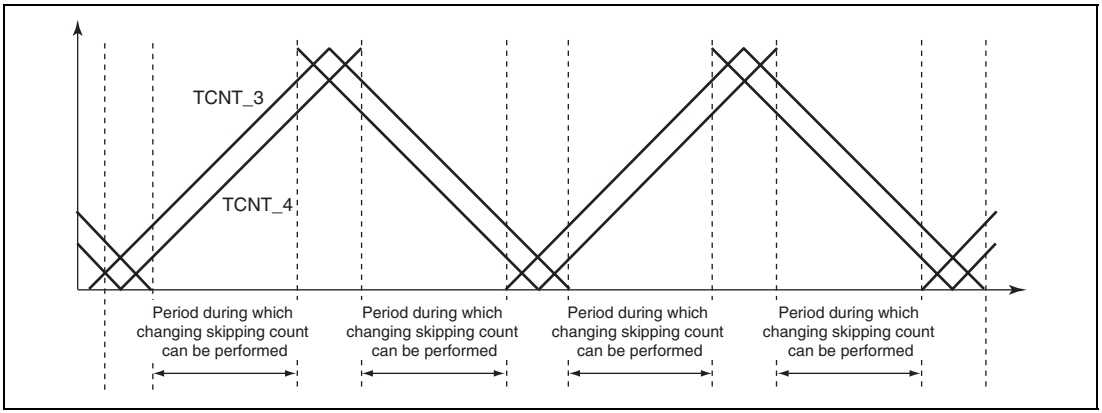


Figure 27.68 Periods during which Interrupt Skipping Count can be Changed

(b) Example of Interrupt Skipping Operation

Figure 27.69 shows an example of TGIA_3 interrupt skipping in which the interrupt skipping count is set to three by the 3ACOR bit and the T3AEN bit is set to 1 in the timer interrupt skipping set register (TITCR).

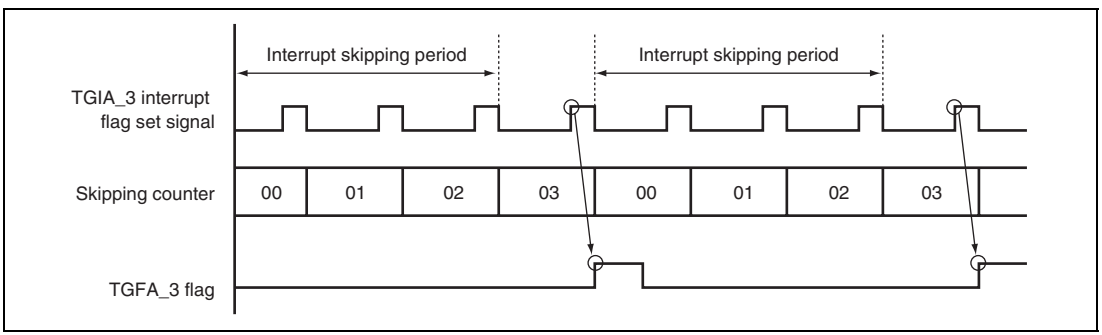


Figure 27.69 Example of Interrupt Skipping Operation

(c) Buffer Transfer Control Linked with Interrupt Skipping

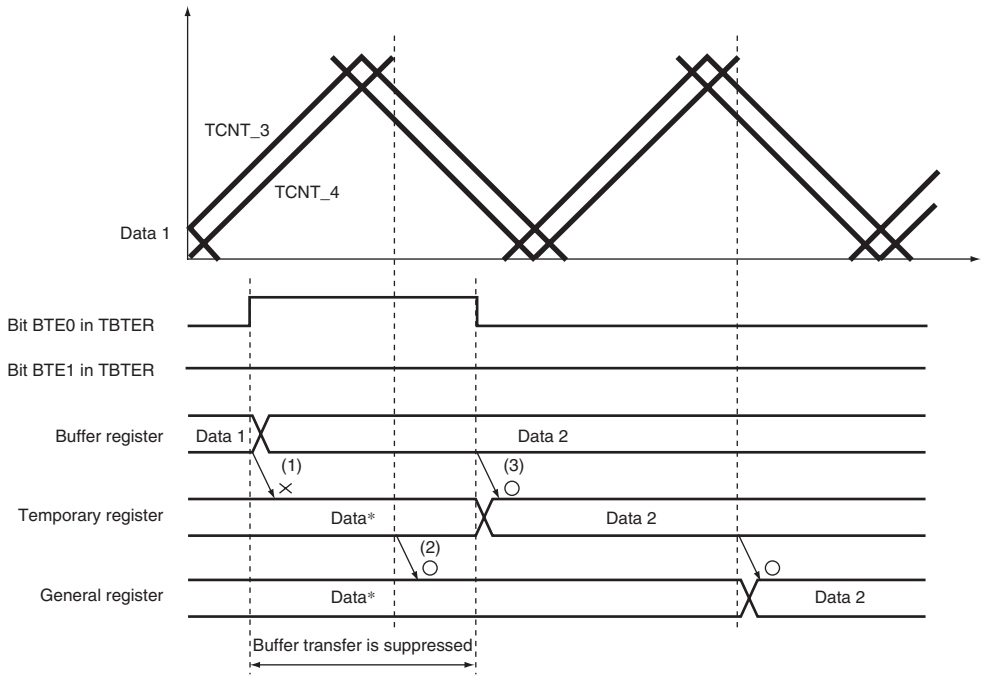
In complementary PWM mode, whether to transfer data from a buffer register to a temporary register and whether to link the transfer with interrupt skipping can be specified with the BTE1 and BTE0 bits in the timer buffer transfer set register (TBTER).

Figure 27.70 shows an example of operation when buffer transfer is suppressed (BTE1 = 0 and BTE0 = 1). While this setting is valid, data is not transferred from the buffer register to the temporary register.

Figure 27.71 shows an example of operation when buffer transfer is linked with interrupt skipping (BTE1 = 1 and BET0 = 0). While this setting is valid, data is not transferred from the buffer register to the temporary register outside the buffer transfer-enabled period. Depending on the rewrite timing from the interrupt generation to the buffer register, there are two types of the transfer timing such as from the buffer register to the temporary register and from the temporary register to the general register.

Note that the buffer transfer-enabled period depends on the T3AEN and T4VEN bit settings in the timer interrupt skipping set register (TITCR). Figure 27.72 shows the relationship between the T3AEN and T4VEN bit settings in TITCR and buffer transfer-enabled period.

Note: This function must always be used in combination with interrupt skipping.
When interrupt skipping is disabled (the T3AEN and T4VEN bits in the timer interrupt skipping set register (TITCR) are cleared to 0 or the skipping count set bits (3ACOR and 4VCOR) in TITCR are cleared to 0), make sure that buffer transfer is not linked with interrupt skipping (clear the BTE1 bit in the timer buffer transfer set register (TBTER) to 0). If buffer transfer is linked with interrupt skipping while interrupt skipping is disabled, buffer transfer is never performed.



[Legend]

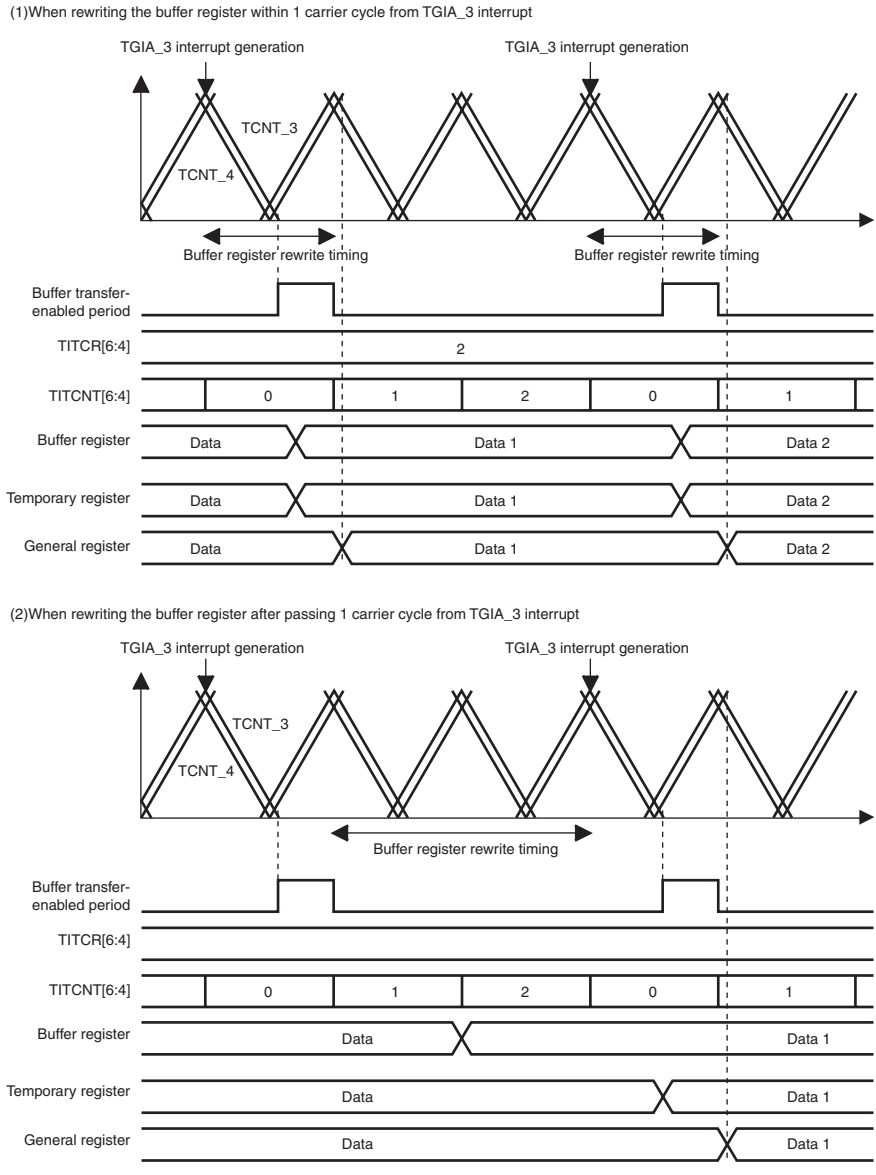
(1) No data is transferred from the buffer register to the temporary register in the buffer transfer-disabled period (bits BTE1 and BTE0 in TBTER are set to 0 and 1, respectively).

(2) Data is transferred from the temporary register to the general register even in the buffer transfer-disabled period.

(3) After buffer transfer is enabled, data is transferred from the buffer register to the temporary register.

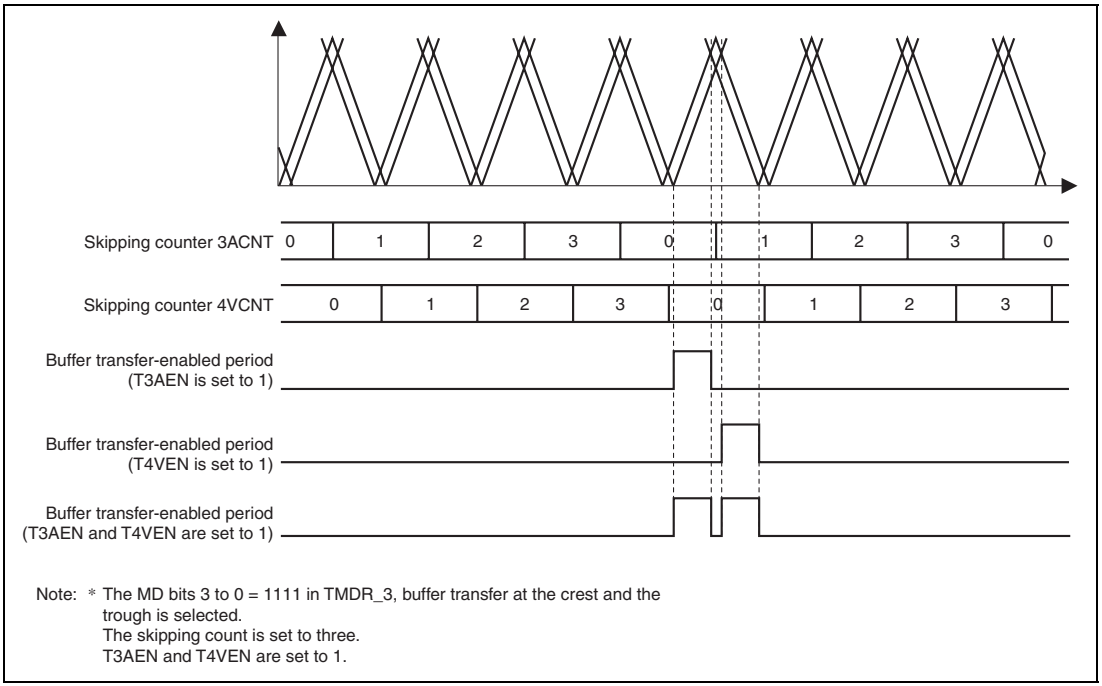
Note: * When buffer transfer at the crest is selected.

Figure 27.70 Example of Operation when Buffer Transfer is Suppressed (BTE1 = 0 and BTE0 = 1)



Note: * The MD bits 3 to 0 = 1101 in TMDR_3, buffer transfer at the crest is selected.
The skipping count is set to two.
T3AEN and T4VEN are set to 1 and 0.

Figure 27.71 Example of Operation when Buffer Transfer is Linked with Interrupt Skipping (BTE1 = 1 and BTE0 = 0)



Note: * The MD bits 3 to 0 = 1111 in TMDR_3, buffer transfer at the crest and the trough is selected.
The skipping count is set to three.
T3AEN and T4VEN are set to 1.

Figure 27.72 Relationship between Bits T3AEN and T4VEN in TITCR and Buffer Transfer-Enabled Period

(4) Complementary PWM Mode Output Protection Function

Complementary PWM mode output has the following protection function.

(a) Register and counter miswrite prevention function

With the exception of the buffer registers, which can be rewritten at any time, access by the CPU can be enabled or disabled for the mode registers, control registers, compare registers, and counters used in complementary PWM mode by means of the RWE bit in the timer read/write enable register (TRWER). The applicable registers are some (21 in total) of the registers in channels 3 and 4 shown in the following:

- TCR_3 and TCR_4, TMDR_3 and TMDR_4, TIORH_3 and TIORH_4, TIORL_3 and TIORL_4, TIER_3 and TIER_4, TCNT_3 and TCNT_4, TGRA_3 and TGRA_4, TGRB_3 and TGRB_4, TOER, TOCR, TGCR, TCDR, and TDDR.

This function enables miswriting due to CPU runaway to be prevented by disabling CPU access to the mode registers, control registers, and counters. When the applicable registers are read in the access-disabled state, undefined values are returned. Writing to these registers is ignored.

27.4.9 A/D Converter Start Request Delaying Function

A/D converter start requests can be issued in channel 4 by making settings in the timer A/D converter start request control register (TADCR), timer A/D converter start request cycle set registers (TADCORA_4 and TADCORB_4), and timer A/D converter start request cycle set buffer registers (TADCOBRA_4 and TADCOBRB_4).

The A/D converter start request delaying function compares TCNT_4 with TADCORA_4 or TADCORB_4, and when their values match, the function issues a respective A/D converter start request (TRG4AN or TRG4BN).

A/D converter start requests (TRG4AN and TRG4BN) can be skipped in coordination with interrupt skipping by setting the ITA3AE, ITA4VE, ITB3AE, and ITB4VE bits in TADCR.

- Example of Procedure for Specifying A/D Converter Start Request Delaying Function

  Figure 27.73 shows an example of procedure for specifying the A/D converter start request delaying function.

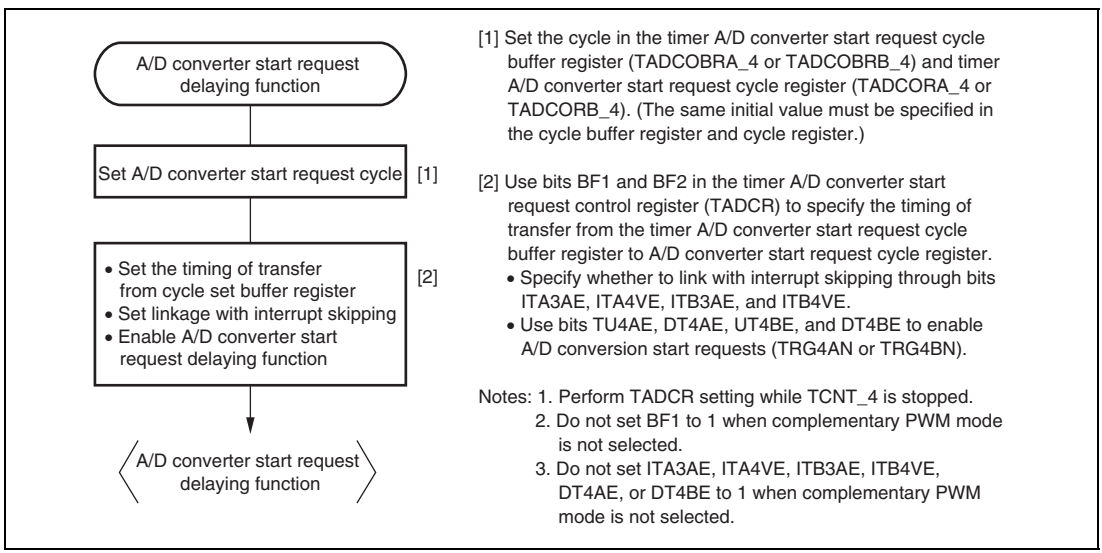


Figure 27.73 Example of Procedure for Specifying A/D Converter Start Request Delaying Function

- Basic Operation Example of A/D Converter Start Request Delaying Function

  Figure 27.74 shows a basic example of A/D converter request signal (TRG4AN) operation when the trough of TCNT_4 is specified for the buffer transfer timing and an A/D converter start request signal is output during TCNT_4 down-counting.

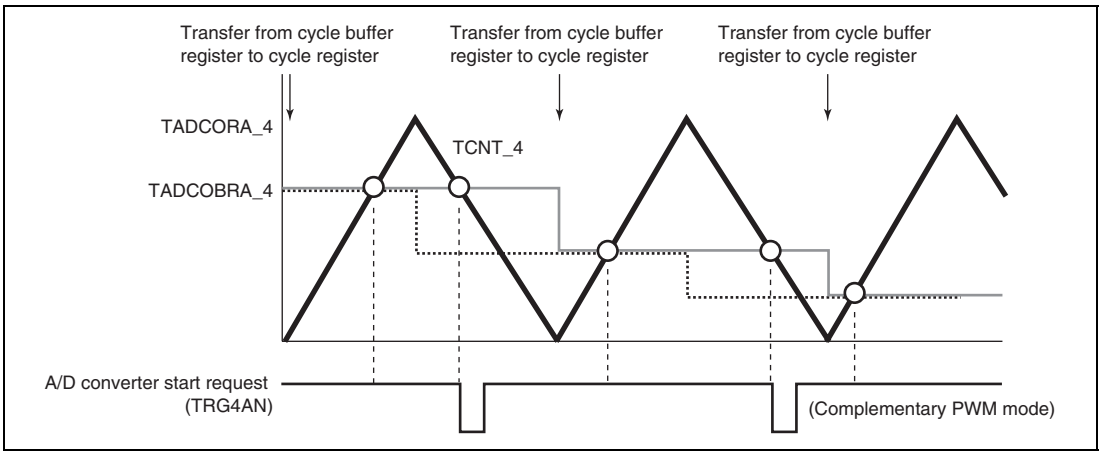


Figure 27.74 Basic Example of A/D Converter Start Request Signal (TRG4AN) Operation

- Buffer Transfer

  The data in the timer A/D converter start request cycle set registers (TADCORA_4 and TADCORB_4) is updated by writing data to the timer A/D converter start request cycle set buffer registers (TADCOBRA_4 and TADCOBRB_4). Data is transferred from the buffer registers to the respective cycle set registers at the timing selected with the BF1 and BF0 bits in the timer A/D converter start request control register (TADCR_4).

- A/D Converter Start Request Delaying Function Linked with Interrupt Skipping

  A/D converter start requests (TRG4AN and TRG4BN) can be issued in coordination with interrupt skipping by making settings in the ITA3AE, ITA4VE, ITB3AE, and ITB4VE bits in the timer A/D converter start request control register (TADCR).

  Figure 27.75 shows an example of A/D converter start request signal (TRG4AN) operation when TRG4AN output is enabled during TCNT_4 up counting and down counting and A/D converter start requests are linked with interrupt skipping.

  Figure 27.76 shows another example of A/D converter start request signal (TRG4AN) operation when TRG4AN output is enabled during TCNT_4 up counting and A/D converter start requests are linked with interrupt skipping.

Note: This function must be used in combination with interrupt skipping.
When interrupt skipping is disabled (the T3AEN and T4VEN bits in the timer interrupt skipping set register (TITCR) are cleared to 0 or the skipping count set bits (3ACOR and 4VCOR) in TITCR are cleared to 0), make sure that A/D converter start requests are not linked with interrupt skipping (clear the ITA3AE, ITA4VE, ITB3AE, and ITB4VE bits in the timer A/D converter start request control register (TADCR) to 0).

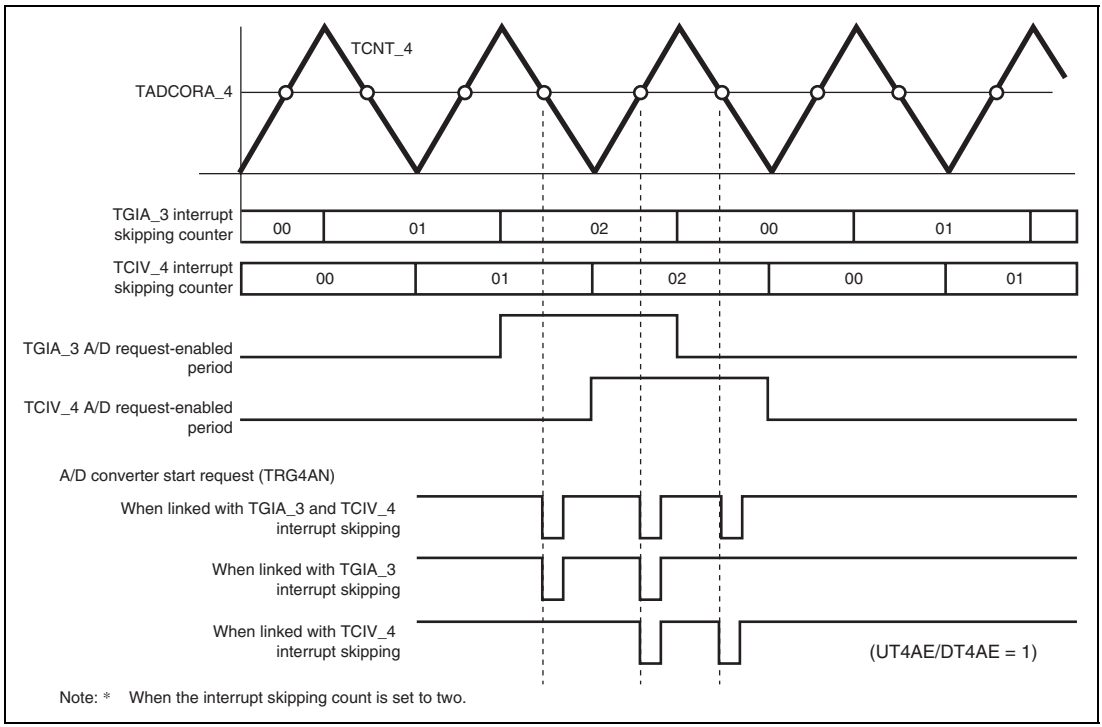


Figure 27.75 Example of A/D Converter Start Request Signal (TRG4AN) Operation Linked with Interrupt Skipping

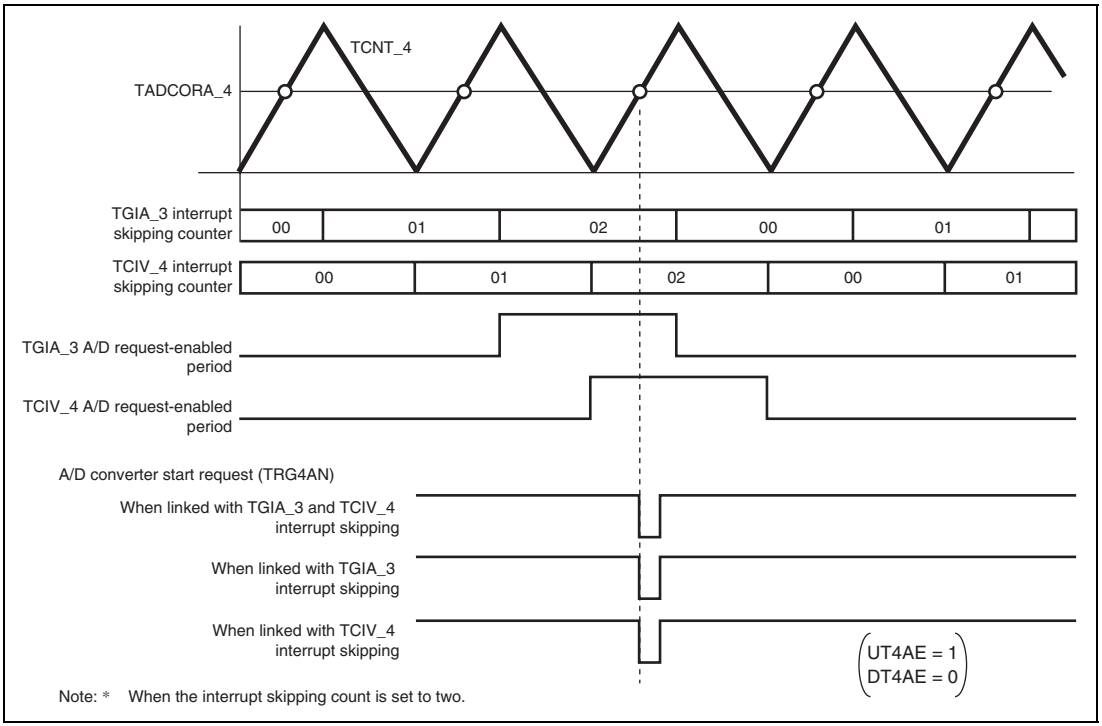


Figure 27.76 Example of A/D Converter Start Request Signal (TRG4AN) Operation Linked with Interrupt Skipping

27.4.10 TCNT Capture at Crest and/or Trough in Complementary PWM Operation

The TCNT value is captured in TGR at either the crest or trough or at both the crest and trough during complementary PWM operation. The timing for capturing in TGR can be selected by TIOR.

Figure 27.77 shows an example in which TCNT is used as a free-running counter without being cleared, and the TCNT value is captured in TGR at the specified timing (either crest or trough, or both crest and trough).

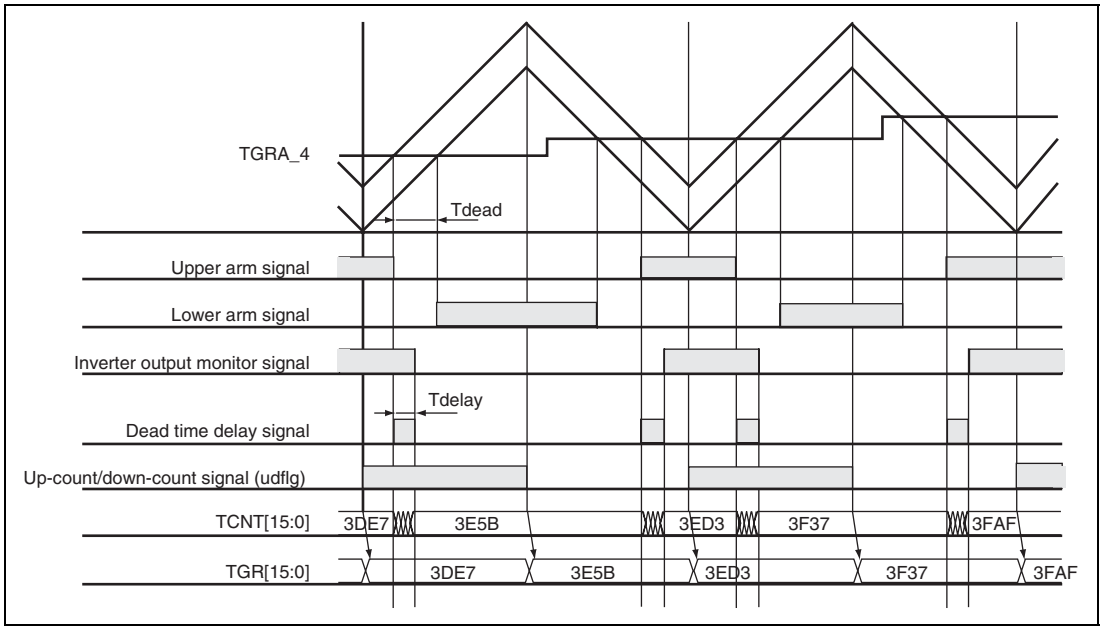


Figure 27.77 TCNT Capturing at Crest and/or Trough in Complementary PWM Operation

27.5 Interrupt Sources

27.5.1 Interrupt Sources and Priorities

This module has three kinds of interrupt sources; TGR input capture/compare match, TCNT overflow, and TCNT underflow. Each interrupt source has its own status flag and enable/disabled bit, allowing the generation of interrupt request signals to be enabled or disabled individually.

When an interrupt request is generated, the corresponding status flag in TSR is set to 1. If the corresponding enable/disable bit in TIER is set to 1 at this time, an interrupt is requested. The interrupt request is cleared by clearing the status flag to 0.

Unlike SH2A products, the interrupt controller handles requests from all channels as having equal priority. For details, see section 7, INTC/INTC2.

Table 27.55 lists the interrupt sources of this module.

Table 27.55 Interrupts of Multi-Function Timer Pulse Unit 2

Channel	Name	Interrupt Source	Interrupt Flag	Activation of Direct Memory Access Controller
0	TGIA_0	TGRA_0 input capture/compare match	TGFA_0	Possible
	TGIB_0	TGRB_0 input capture/compare match	TGFB_0	Not possible
	TGIC_0	TGRC_0 input capture/compare match	TGFC_0	Not possible
	TGID_0	TGRD_0 input capture/compare match	TGFD_0	Not possible
	TCIV_0	TCNT_0 overflow	TCFV_0	Not possible
	TGIE_0	TGRE_0 compare match	TGFE_0	Not possible
	TGIF_0	TGRF_0 compare match	TGFF_0	Not possible
1	TGIA_1	TGRA_1 input capture/compare match	TGFA_1	Possible
	TGIB_1	TGRB_1 input capture/compare match	TGFB_1	Not possible
	TCIV_1	TCNT_1 overflow	TCFV_1	Not possible
	TCIU_1	TCNT_1 underflow	TCFU_1	Not possible

Channel	Name	Interrupt Source	Interrupt Flag	Activation of Direct Memory Access Controller
2	TGIA_2	TGRA_2 input capture/compare match	TGFA_2	Possible
	TGIB_2	TGRB_2 input capture/compare match	TGFB_2	Not possible
	TCIV_2	TCNT_2 overflow	TCFV_2	Not possible
	TCIU_2	TCNT_2 underflow	TCFU_2	Not possible
3	TGIA_3	TGRA_3 input capture/compare match	TGFA_3	Possible
	TGIB_3	TGRB_3 input capture/compare match	TGFB_3	Not possible
	TGIC_3	TGRC_3 input capture/compare match	TGFC_3	Not possible
	TGID_3	TGRD_3 input capture/compare match	TGFD_3	Not possible
	TCIV_3	TCNT_3 overflow	TCFV_3	Not possible
4	TGIA_4	TGRA_4 input capture/compare match	TGFA_4	Possible
	TGIB_4	TGRB_4 input capture/compare match	TGFB_4	Not possible
	TGIC_4	TGRC_4 input capture/compare match	TGFC_4	Not possible
	TGID_4	TGRD_4 input capture/compare match	TGFD_4	Not possible
	TCIV_4	TCNT_4 overflow/underflow	TCFV_4	Not possible

Note: This table shows the initial state immediately after a reset. The relative channel priorities can be changed by the interrupt controller.

(1) Input Capture/Compare Match Interrupt

An interrupt is requested if the TGIE bit in TIER is set to 1 when the TGF flag in TSR is set to 1 by the occurrence of a TGR input capture/compare match on a particular channel. The interrupt request is cleared by clearing the TGF flag to 0. This module has eighteen input capture/compare match interrupts, six for channel 0, four each for channels 3 and 4, and two each for channels 1 and 2. The TGFE_0 and TGFF_0 flags in channel 0 are not set by the occurrence of an input capture.

(2) Overflow Interrupt

An interrupt is requested if the TCIEV bit in TIER is set to 1 when the TCFV flag in TSR is set to 1 by the occurrence of TCNT overflow on a channel. The interrupt request is cleared by clearing the TCFV flag to 0. This module has five overflow interrupts, one for each channel.

(3) Underflow Interrupt

An interrupt is requested if the TCIEU bit in TIER is set to 1 when the TCFU flag in TSR is set to 1 by the occurrence of TCNT underflow on a channel. The interrupt request is cleared by clearing the TCFU flag to 0. This module has two underflow interrupts, one each for channels 1 and 2.

27.5.2 Activation of Direct Memory Access Controller

The direct memory access controller can be activated by the TGRA input capture/compare match interrupt in each channel. For details, see section 6A, Direct Memory Access Controller for Local Bus and Peripheral Modules (LBSC-DMAC/HPB-DMAC).

In this module, a total of five TGRA input capture/compare match interrupts can be used as direct memory access controller activation sources, one each for channels 0 to 4.

27.5.3 A/D Converter Activation

The A/D converter can be activated by one of the following three methods in this module. Table 27.56 shows the relationship between interrupt sources and A/D converter start request signals.

(1) A/D Converter Activation by TGRA Input Capture/Compare Match or at TCNT_4 Trough in Complementary PWM Mode

The A/D converter can be activated by the occurrence of a TGRA input capture/compare match in each channel. In addition, if complementary PWM operation is performed while the TTGE2 bit in TIER_4 is set to 1, the A/D converter can be activated at the trough of TCNT_4 count (TCNT_4 = H'0000).

A/D converter start request signal TRGAN is issued to the A/D converter under either one of the following conditions.

- When the TGFA flag in TSR is set to 1 by the occurrence of a TGRA input capture/compare match on a particular channel while the TTGE bit in TIER is set to 1
- When the TCNT_4 count reaches the trough (TCNT_4 = H'0000) during complementary PWM operation while the TTGE2 bit in TIER_4 is set to 1

When either condition is satisfied, if A/D converter start signal TRGAN from this module is selected as the trigger in the A/D converter, A/D conversion will start.

(2) A/D Converter Activation by Compare Match between TCNT_0 and TGRE_0

The A/D converter can be activated by generating A/D converter start request signal TRG0N when a compare match occurs between TCNT_0 and TGRE_0 in channel 0.

When the TGFE flag in TSR2_0 is set to 1 by the occurrence of a compare match between TCNT_0 and TGRE_0 in channel 0 while the TTGE2 bit in TIER2_0 is set to 1, A/D converter start request TGR0N is issued to the A/D converter. If A/D converter start signal TGR0N from this module is selected as the trigger in the A/D converter, A/D conversion will start.

(3) A/D Converter Activation by A/D Converter Start Request Delaying Function

The A/D converter can be activated by generating A/D converter start request signal TRG4AN or TRG4BN when the TCNT_4 count matches the TADCORA or TADCORB value if the UT4AE, DT4AE, UT4BE, or DT4BE bit in the A/D converter start request control register (TADCR) is set to 1. For details, refer to section 27.4.9, A/D Converter Start Request Delaying Function.

A/D conversion will start if A/D converter start signal TRG4AN from this module is selected as the trigger in the A/D converter when TRG4AN is generated or if TRG4BN from this module is selected as the trigger in the A/D converter when TRG4BN is generated.

Table 27.56 Interrupt Sources and A/D Converter Start Request Signals

Target Registers	Interrupt Source	A/D Converter Start Request Signal
TGRA_0 and TCNT_0	Input capture/compare match	TRGAN
TGRA_1 and TCNT_1		
TGRA_2 and TCNT_2		
TGRA_3 and TCNT_3		
TGRA_4 and TCNT_4		
TCNT_4	TCNT_4 Trough in complementary PWM mode	
TGRE_0 and TCNT_0	Compare match	TRG0N
TADCORA and TCNT_4		TRG4AN
TADCORB and TCNT_4		TRG4BN

27.6 Operation Timing

27.6.1 Input/Output Timing

(1) TCNT Count Timing

Figure 27.78 shows TCNT count timing in internal clock operation, and Figure 27.79 shows TCNT count timing in external clock operation (normal mode), and Figure 27.80 shows TCNT count timing in external clock operation (phase counting mode).

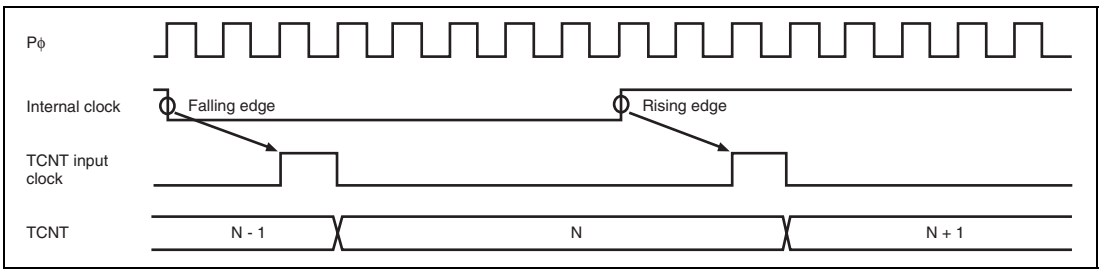


Figure 27.78 Count Timing in Internal Clock Operation

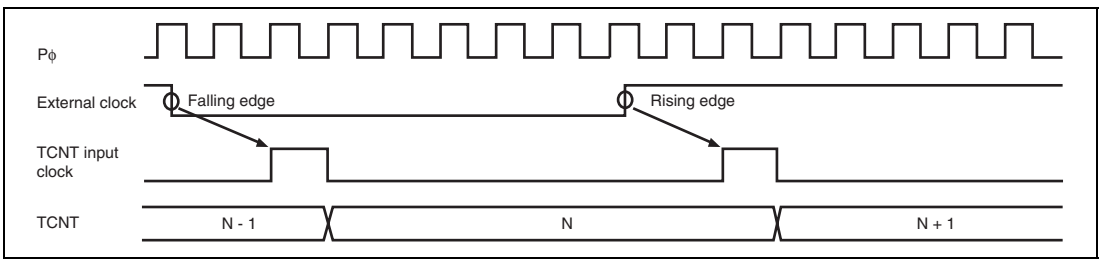


Figure 27.79 Count Timing in External Clock Operation

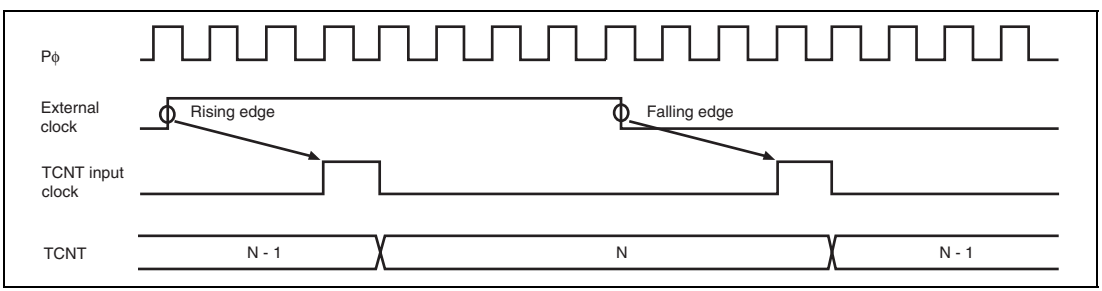


Figure 27.80 Count Timing in External Clock Operation (Phase Counting Mode)

(2) Output Compare Output Timing

A compare match signal is generated in the final state in which TCNT and TGR match (the point at which the count value matched by TCNT is updated). When a compare match signal is generated, the output value set in TIOR is output at the output compare output pin (TIOC pin). After a match between TCNT and TGR, the compare match signal is not generated until the TCNT input clock is generated.

Figure 27.81 shows output compare output timing (normal mode and PWM mode) and Figure 27.82 shows output compare output timing (complementary PWM mode and reset synchronous PWM mode).

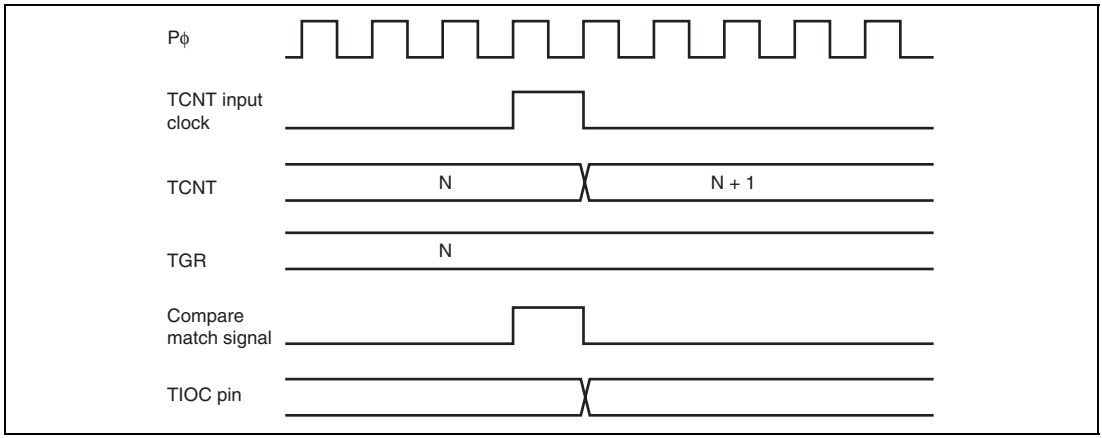


Figure 27.81 Output Compare Output Timing (Normal Mode/PWM Mode)

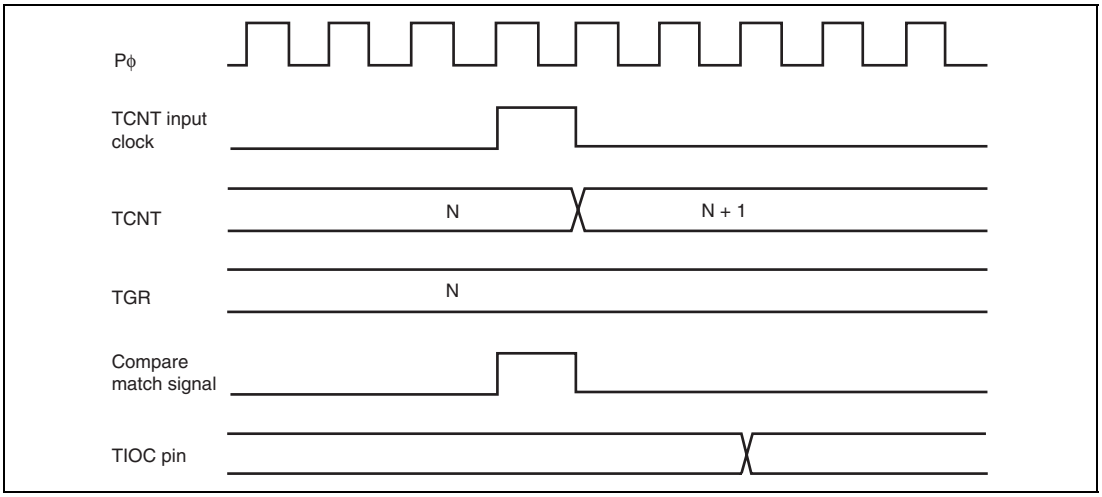


Figure 27.82 Output Compare Output Timing (Complementary PWM Mode/Reset Synchronous PWM Mode)

(3) Input Capture Signal Timing

Figure 27.83 shows input capture signal timing.

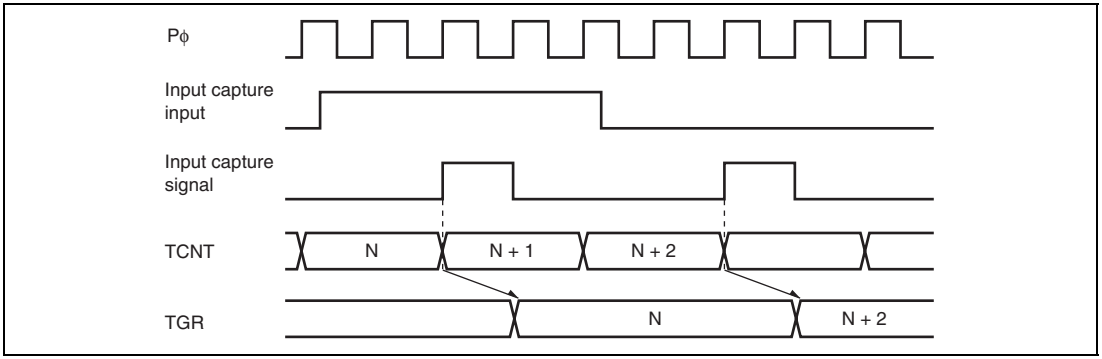


Figure 27.83 Input Capture Input Signal Timing

(4) Timing for Counter Clearing by Compare Match/Input Capture

Figure 27.84 shows the timing when counter clearing on compare match is specified, and Figure 27.85 shows the timing when counter clearing on input capture is specified.

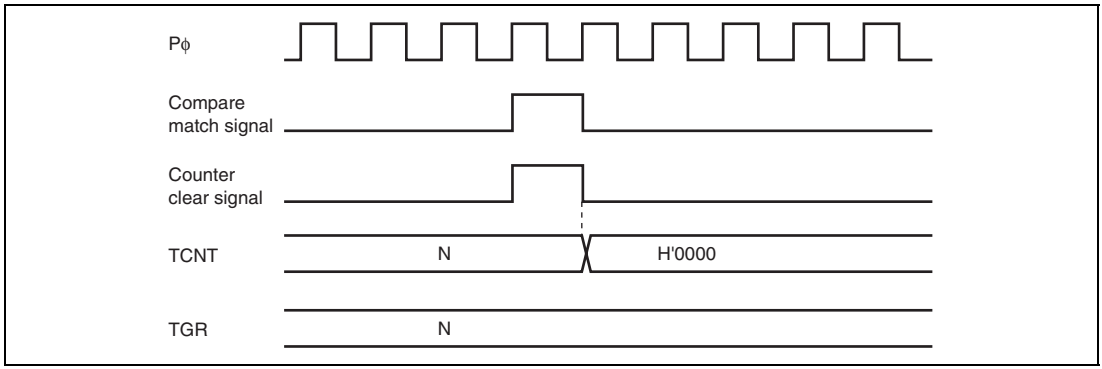


Figure 27.84 Counter Clear Timing (Compare Match)

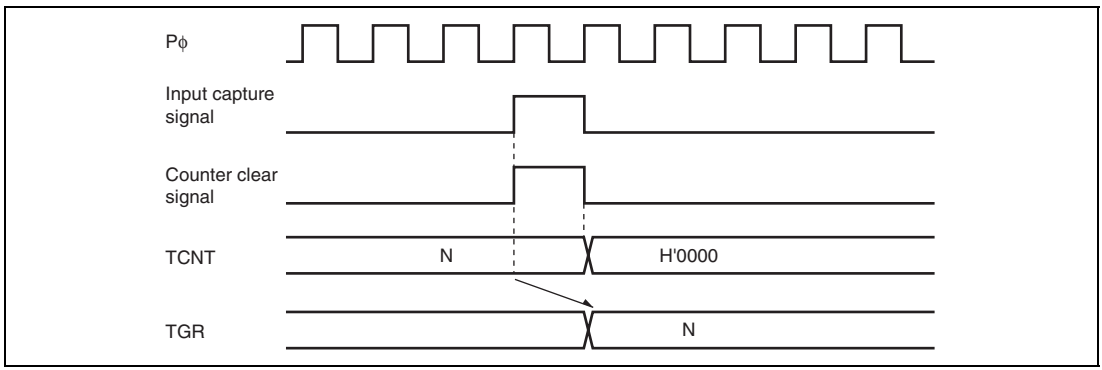


Figure 27.85 Counter Clear Timing (Input Capture)

(5) Buffer Operation Timing

Figures 27.86 to 27.88 show the timing in buffer operation.

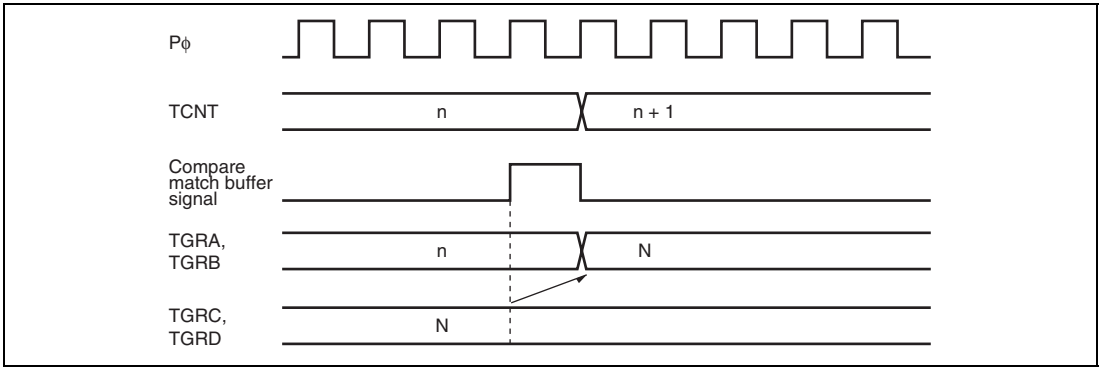


Figure 27.86 Buffer Operation Timing (Compare Match)

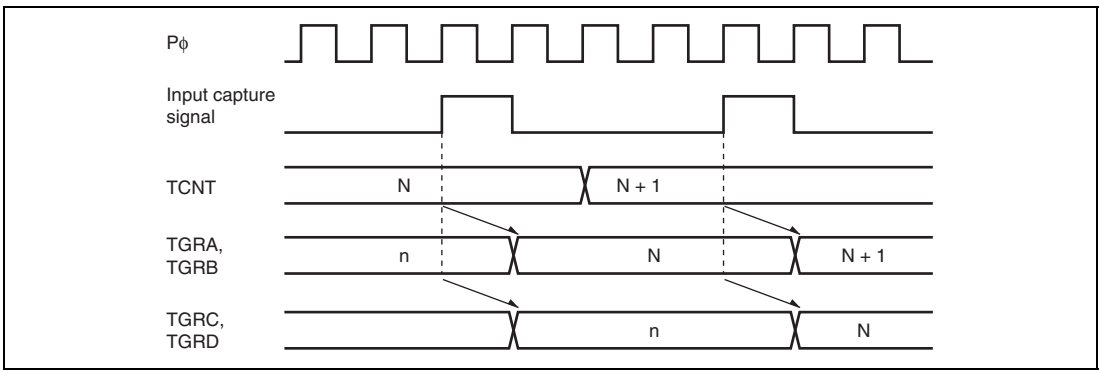


Figure 27.87 Buffer Operation Timing (Input Capture)

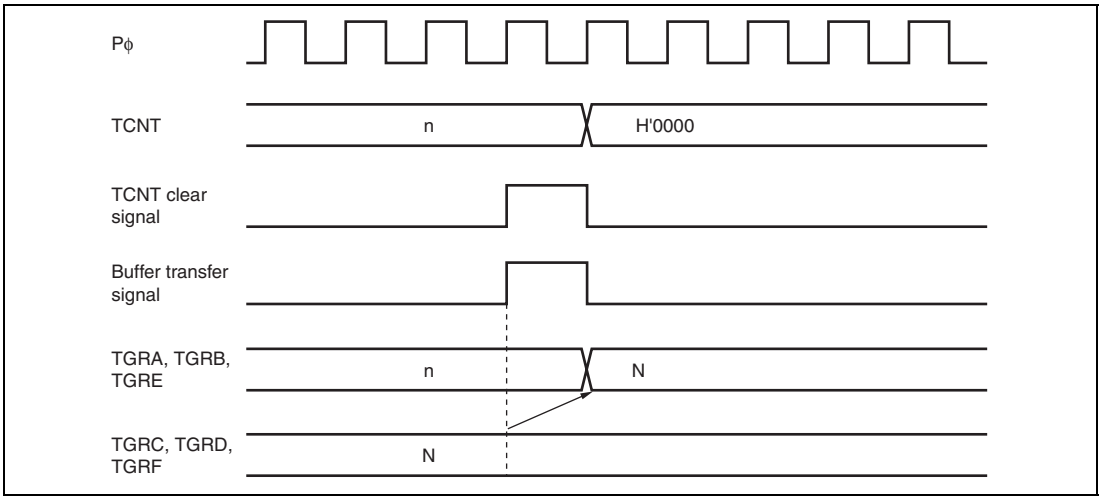


Figure 27.88 Buffer Transfer Timing (when TCNT Cleared)

(6) Buffer Transfer Timing (Complementary PWM Mode)

Figures 27.89 to 27.91 show the buffer transfer timing in complementary PWM mode.

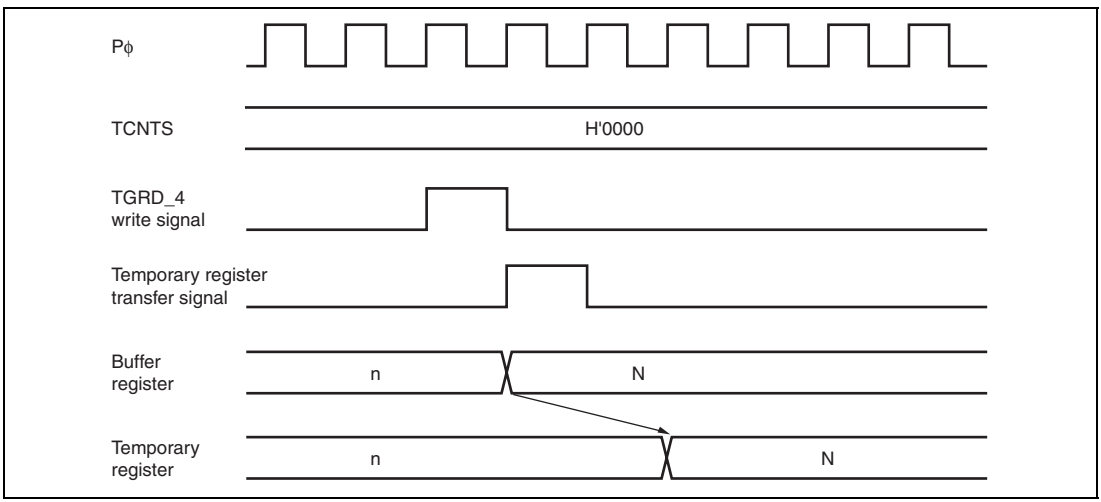


Figure 27.89 Transfer Timing from Buffer Register to Temporary Register (TCNTS Stop)

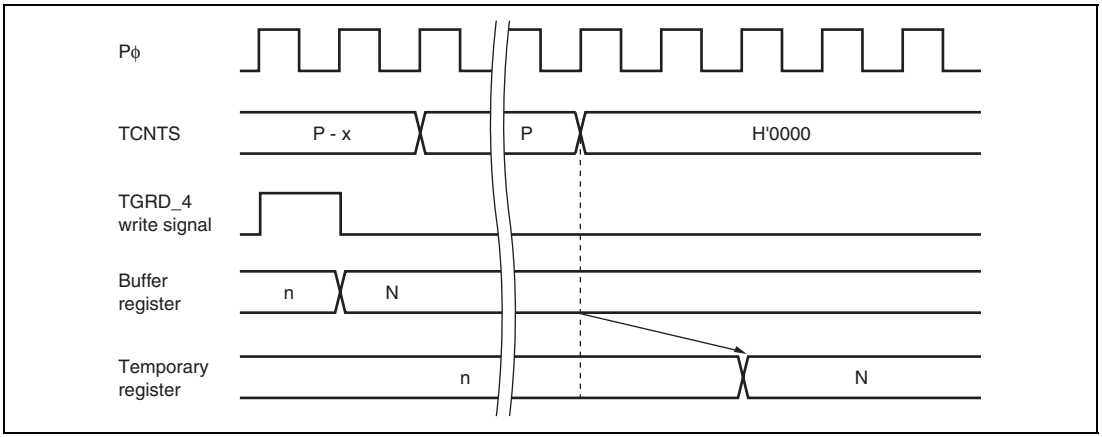


Figure 27.90 Transfer Timing from Buffer Register to Temporary Register (TCNTS Operating)

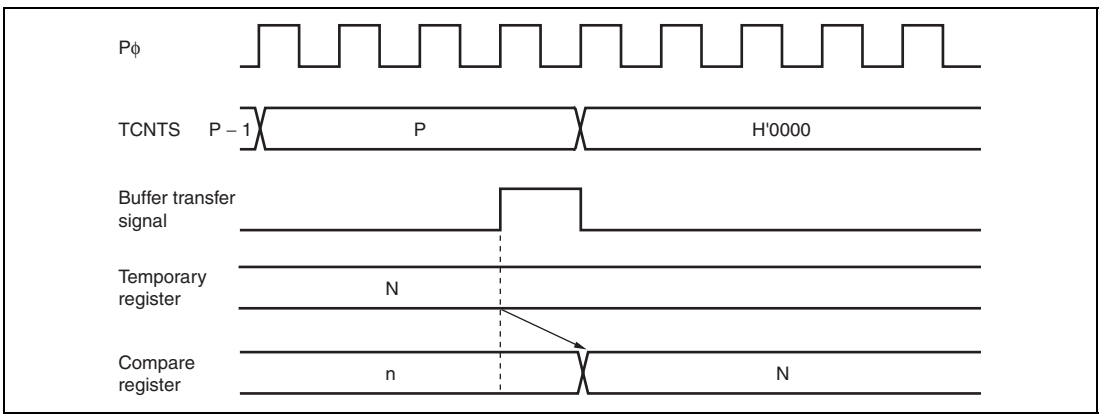


Figure 27.91 Transfer Timing from Temporary Register to Compare Register

27.6.2 Interrupt Signal Timing

(1) TGF Flag Setting Timing in Case of Compare Match

Figure 27.92 shows the timing for setting of the TGF flag in TSR on compare match, and TGI interrupt request signal timing.

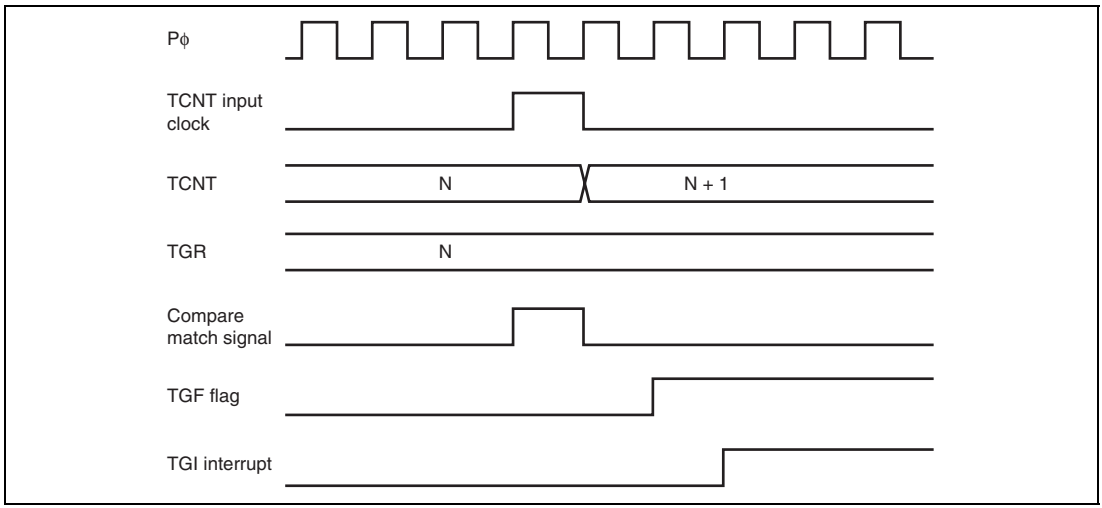


Figure 27.92 TGI Interrupt Timing (Compare Match)

(2) TGF Flag Setting Timing in Case of Input Capture

Figure 27.93 shows the timing for setting of the TGF flag in TSR on input capture, and TGI interrupt request signal timing.

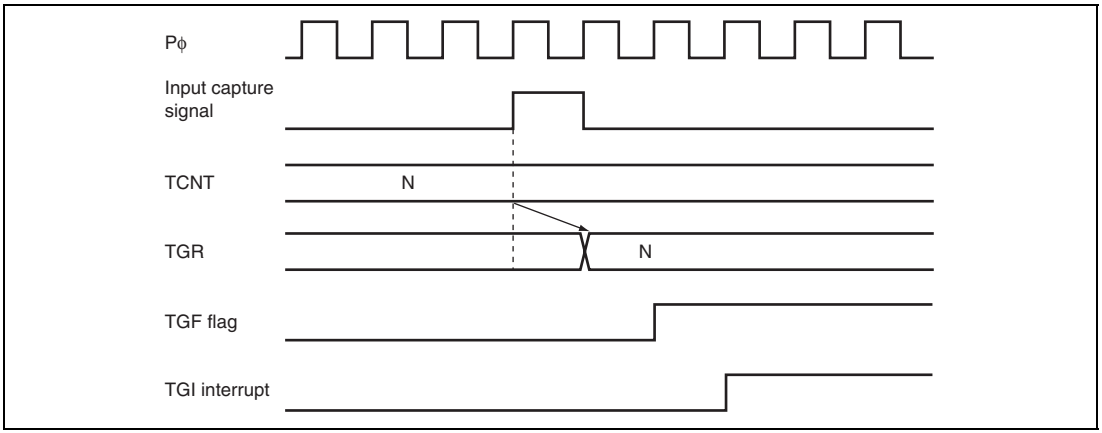


Figure 27.93 TGI Interrupt Timing (Input Capture)

(3) TCFV Flag/TCFU Flag Setting Timing

Figure 27.94 shows the timing for setting of the TCFV flag in TSR on overflow, and TCIV interrupt request signal timing.

Figure 27.95 shows the timing for setting of the TCFU flag in TSR on underflow, and TCIU interrupt request signal timing.

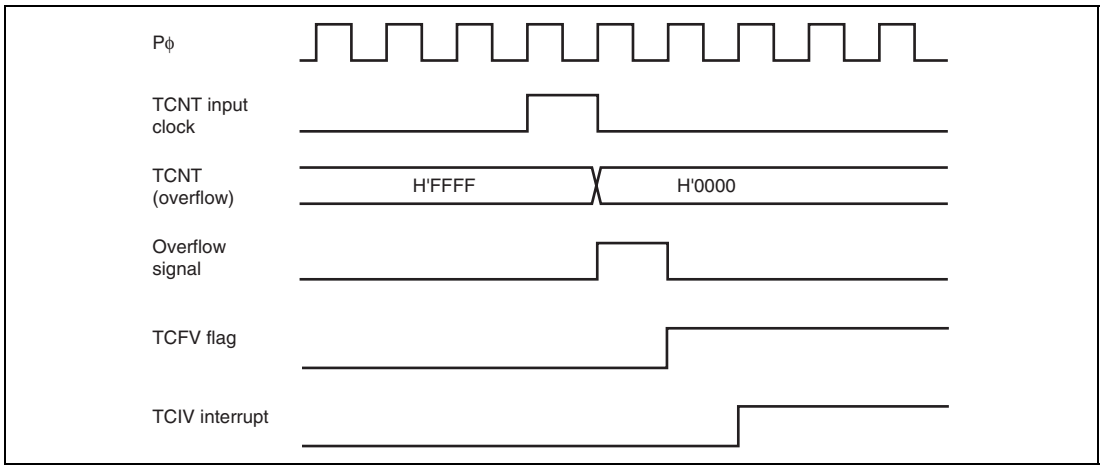


Figure 27.94 TCIV Interrupt Setting Timing

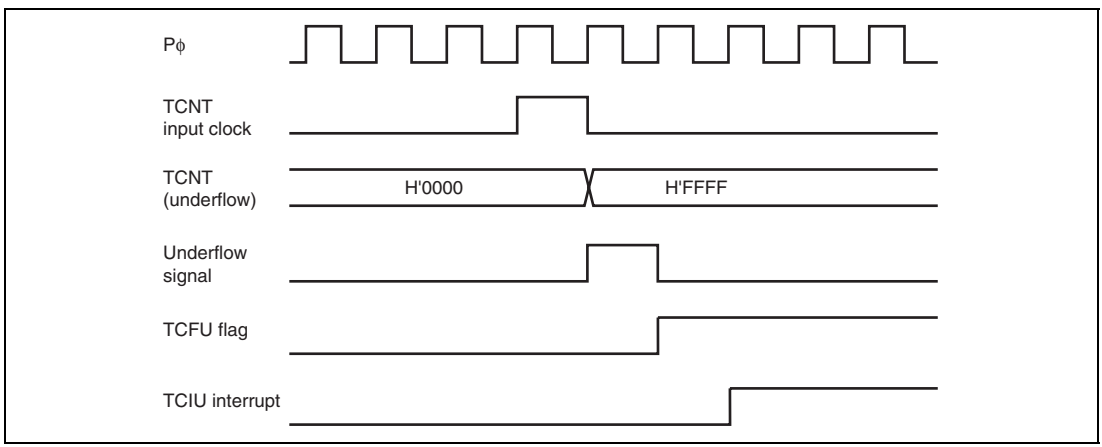


Figure 27.95 TCIU Interrupt Setting Timing

(4) Status Flag Clearing Timing

After a status flag is read as 1 by the CPU, it is cleared by writing 0 to it. When the direct memory access controller is activated, the flag is cleared automatically. Figure 27.96 shows the timing for status flag clearing by the CPU, and Figure 27.97 shows the timing for status flag clearing by the direct memory access controller.

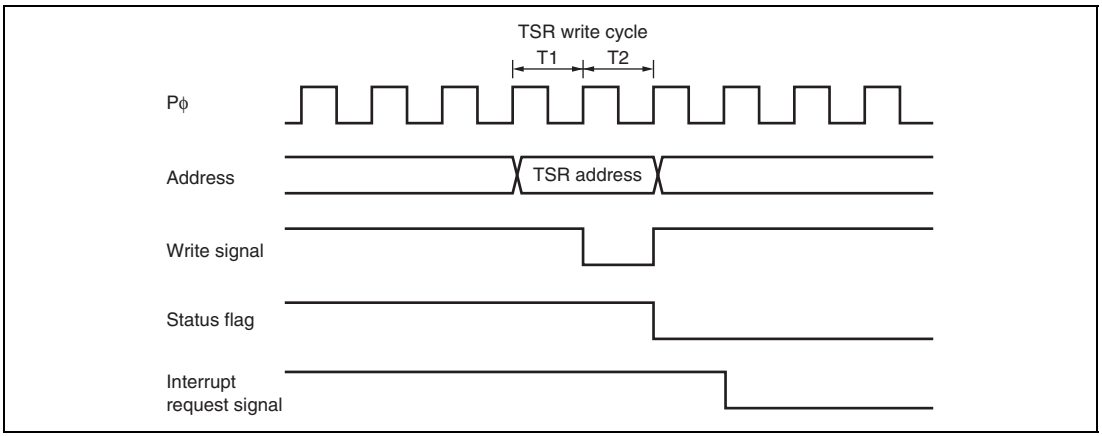


Figure 27.96 Timing for Status Flag Clearing by CPU

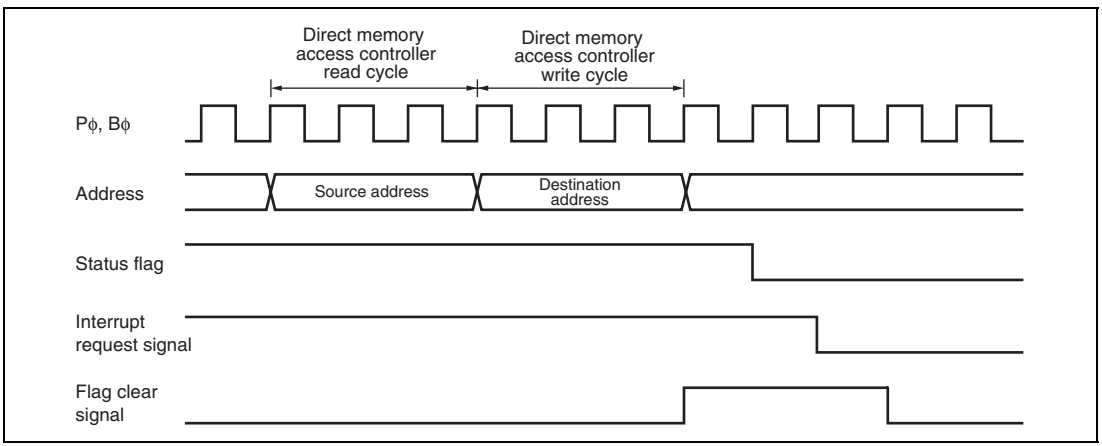


Figure 27.97 Timing for Status Flag Clearing by Direct Memory Access Controller Activation

27.7 Usage Notes

27.7.1 Module Standby Mode Setting

Operation of this module can be disabled or enabled using the standby control register. The initial setting is for the operation to be halted. Register access is enabled by clearing module standby mode. For details, refer to section 9, Operating Modes and Power-Down Modes.

27.7.2 Input Clock Restrictions

The input clock pulse width must be at least 1.5 states in the case of single-edge detection, and at least 2.5 states in the case of both-edge detection. This module will not operate properly at narrower pulse widths.

In phase counting mode, the phase difference and overlap between the two input clocks must be at least 1.5 states, and the pulse width must be at least 2.5 states. Figure 27.98 shows the input clock conditions in phase counting mode.

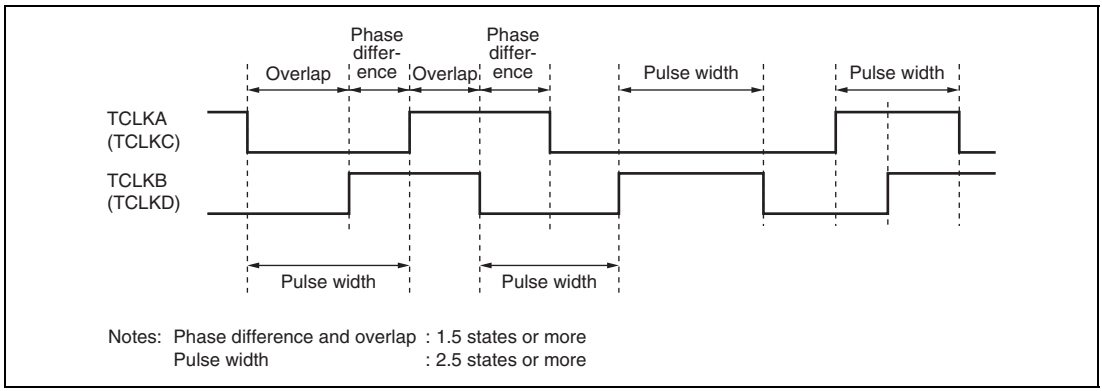


Figure 27.98 Phase Difference, Overlap, and Pulse Width in Phase Counting Mode

27.7.3 Caution on Period Setting

When counter clearing on compare match is set, TCNT is cleared in the final state in which it matches the TGR value (the point at which the count value matched by TCNT is updated). Consequently, the actual counter frequency is given by the following formula:

$$f = \frac{P\phi}{(N + 1)}$$

Where f: Counter frequency
Pϕ: Peripheral clock operating frequency for this module
N: TGR set value

27.7.4 Contention between TCNT Write and Clear Operations

If the counter clear signal is generated in the T2 state of a TCNT write cycle, TCNT clearing takes precedence and the TCNT write is not performed.

Figure 27.99 shows the timing in this case.

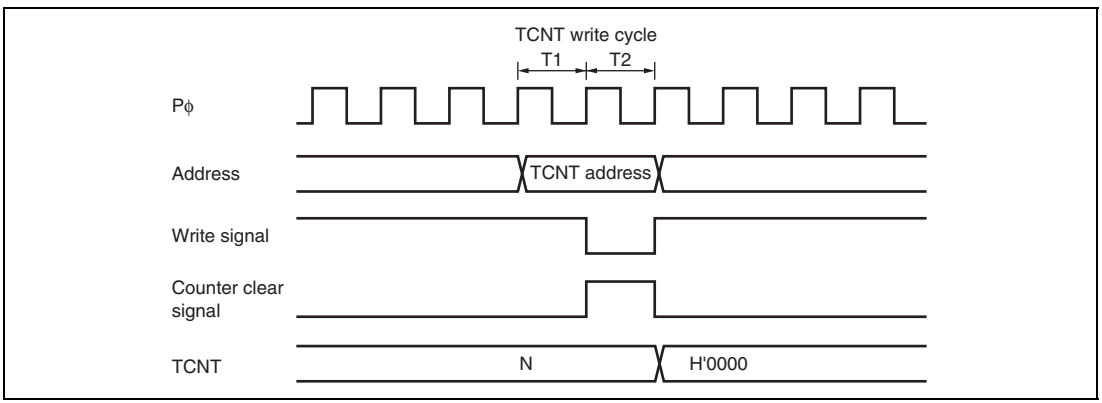


Figure 27.99 Contention between TCNT Write and Clear Operations

27.7.5 Contention between TCNT Write and Increment Operations

If incrementing occurs in the T2 state of a TCNT write cycle, the TCNT write takes precedence and TCNT is not incremented.

Figure 27.100 shows the timing in this case.

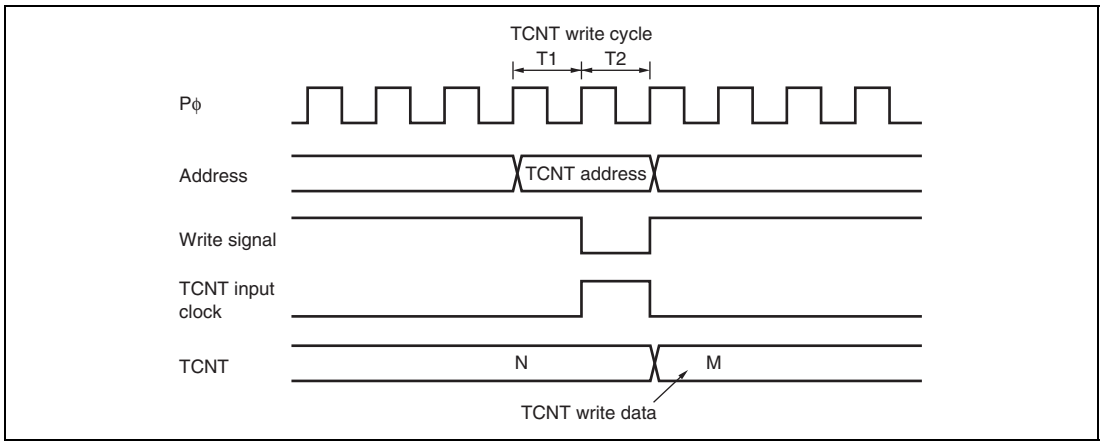


Figure 27.100 Contention between TCNT Write and Increment Operations

27.7.6 Contention between TGR Write and Compare Match

If a compare match occurs in the T2 state of a TGR write cycle, the TGR write is executed and the compare match signal is also generated.

Figure 27.101 shows the timing in this case.

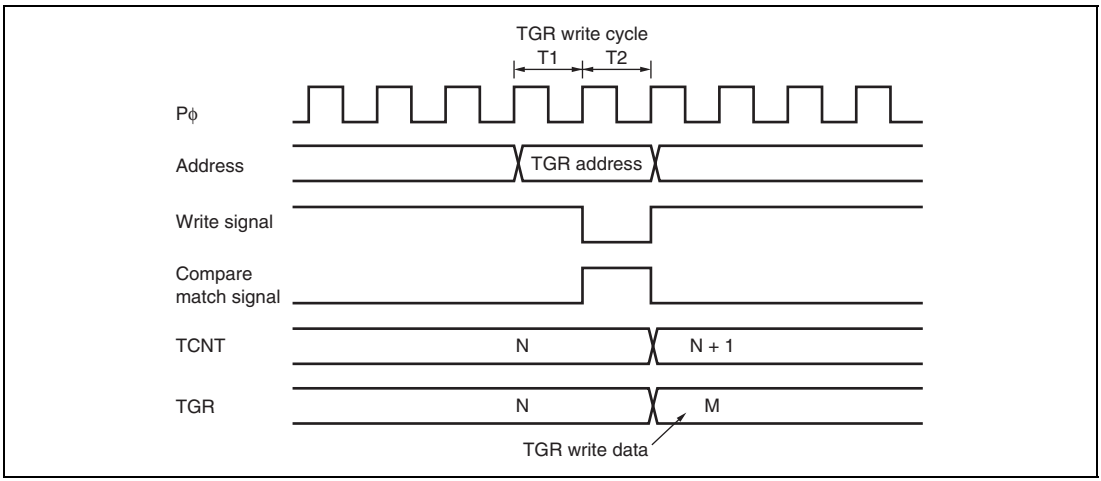


Figure 27.101 Contention between TGR Write and Compare Match

27.7.7 Contention between Buffer Register Write and Compare Match

If a compare match occurs in the T2 state of a TGR write cycle, the data that is transferred to TGR by the buffer operation is the data after write.

Figure 27.102 shows the timing in this case.

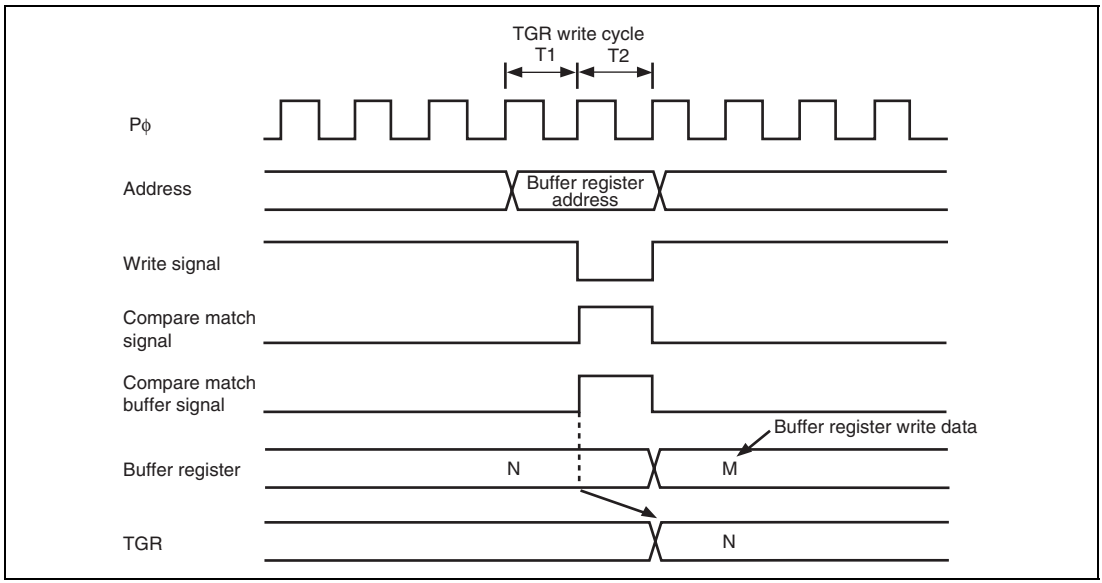


Figure 27.102 Contention between Buffer Register Write and Compare Match

27.7.8 Contention between Buffer Register Write and TCNT Clear

When the buffer transfer timing is set at the TCNT clear by the buffer transfer mode register (TBTM), if TCNT clear occurs in the T2 state of a TGR write cycle, the data that is transferred to TGR by the buffer operation is the data before write.

Figure 27.103 shows the timing in this case.

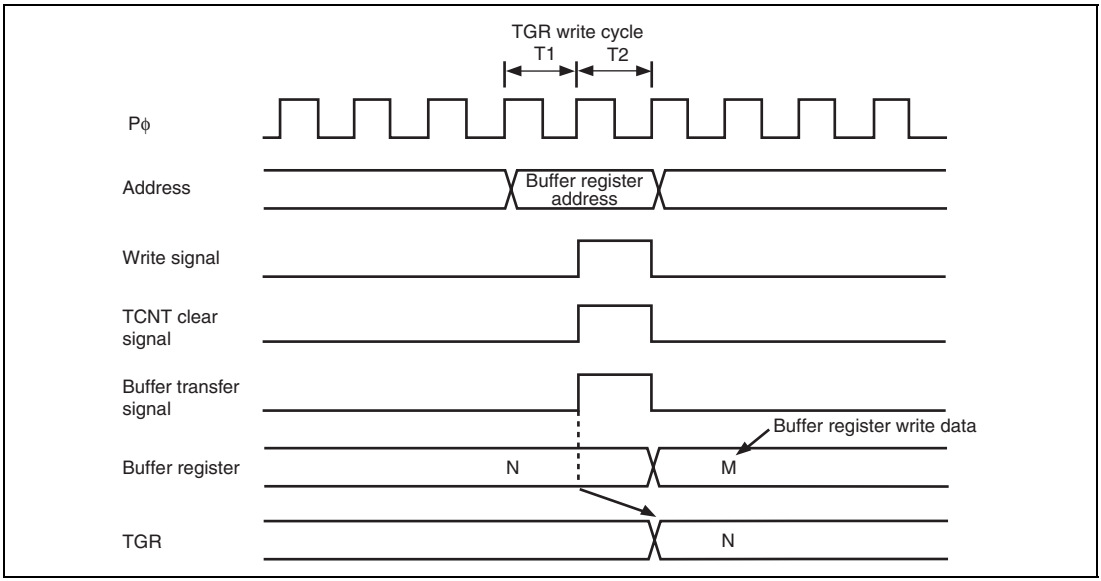


Figure 27.103 Contention between Buffer Register Write and TCNT Clear

27.7.9 Contention between TGR Read and Input Capture

If an input capture signal is generated in the T1 state of a TGR read cycle, the data that is read will be the data in the buffer before input capture transfer.

Figure 27.104 shows the timing in this case.

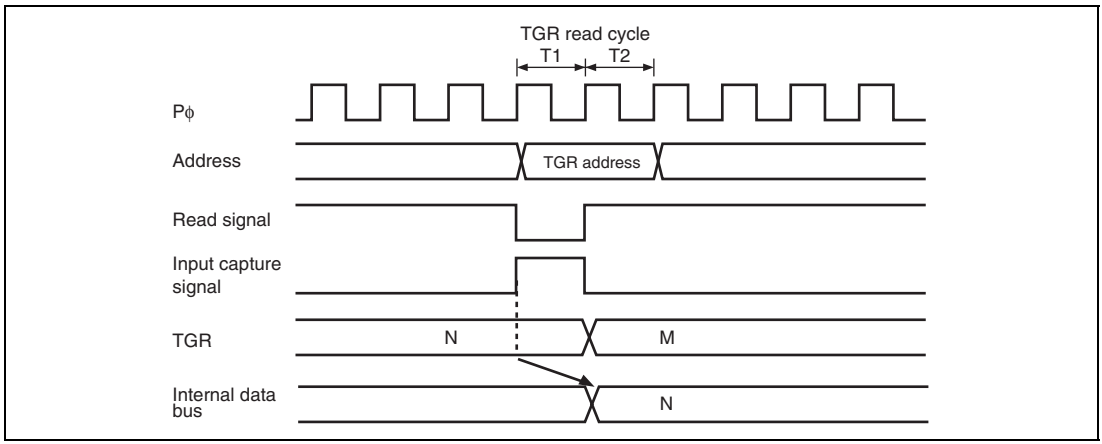


Figure 27.104 Contention between TGR Read and Input Capture

27.7.10 Contention between TGR Write and Input Capture

If an input capture signal is generated in the T2 state of a TGR write cycle, the input capture operation takes precedence and the write to TGR is not performed.

Figure 27.105 shows the timing in this case.

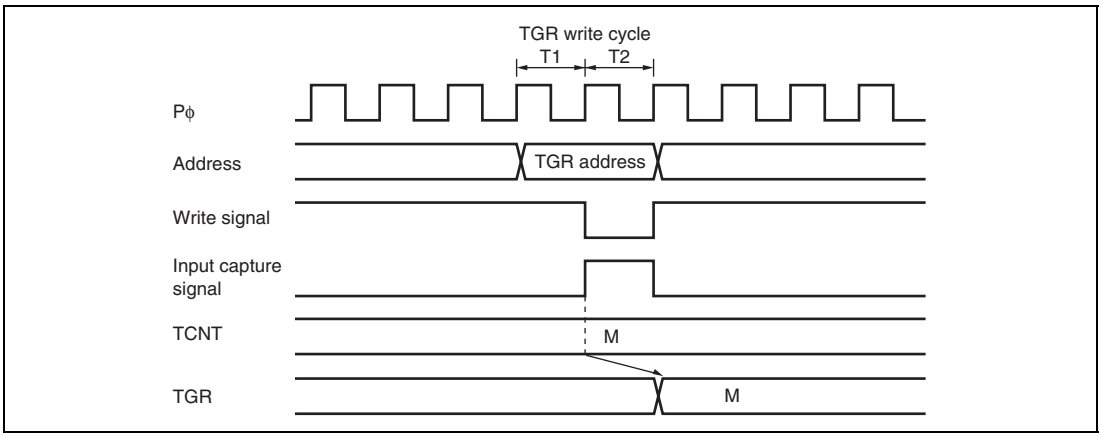


Figure 27.105 Contention between TGR Write and Input Capture

27.7.11 Contention between Buffer Register Write and Input Capture

If an input capture signal is generated in the T2 state of a buffer register write cycle, the buffer operation takes precedence and the write to the buffer register is not performed.

Figure 27.106 shows the timing in this case.

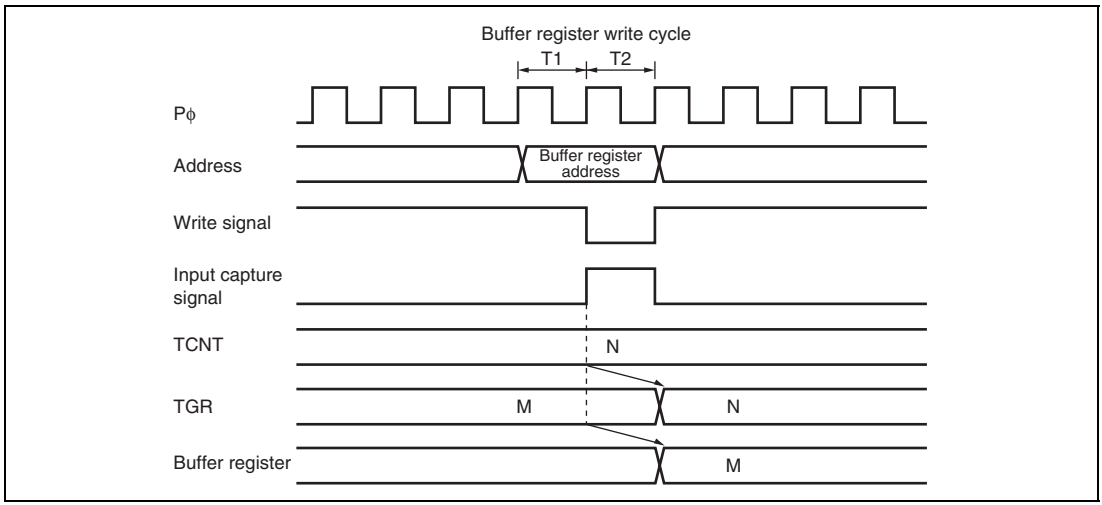


Figure 27.106 Contention between Buffer Register Write and Input Capture

27.7.12 TCNT2 Write and Overflow/Underflow Contention in Cascade Connection

With timer counters TCNT1 and TCNT2 in a cascade connection, when a contention occurs during TCNT_1 count (during a TCNT_2 overflow/underflow) in the T_2 state of the TCNT_2 write cycle, the write to TCNT_2 is conducted, and the TCNT_1 count signal is disabled. At this point, if there is match with TGRA_1 and the TCNT_1 value, a compare signal is issued. Furthermore, when the TCNT_1 count clock is selected as the input capture source of channel 0, TGRA_0 to D_0 carry out the input capture operation. In addition, when the compare match/input capture is selected as the input capture source of TGRB_1, TGRB_1 carries out input capture operation. The timing is shown in figure 27.107.

For cascade connections, be sure to synchronize settings for channels 1 and 2 when setting TCNT clearing.

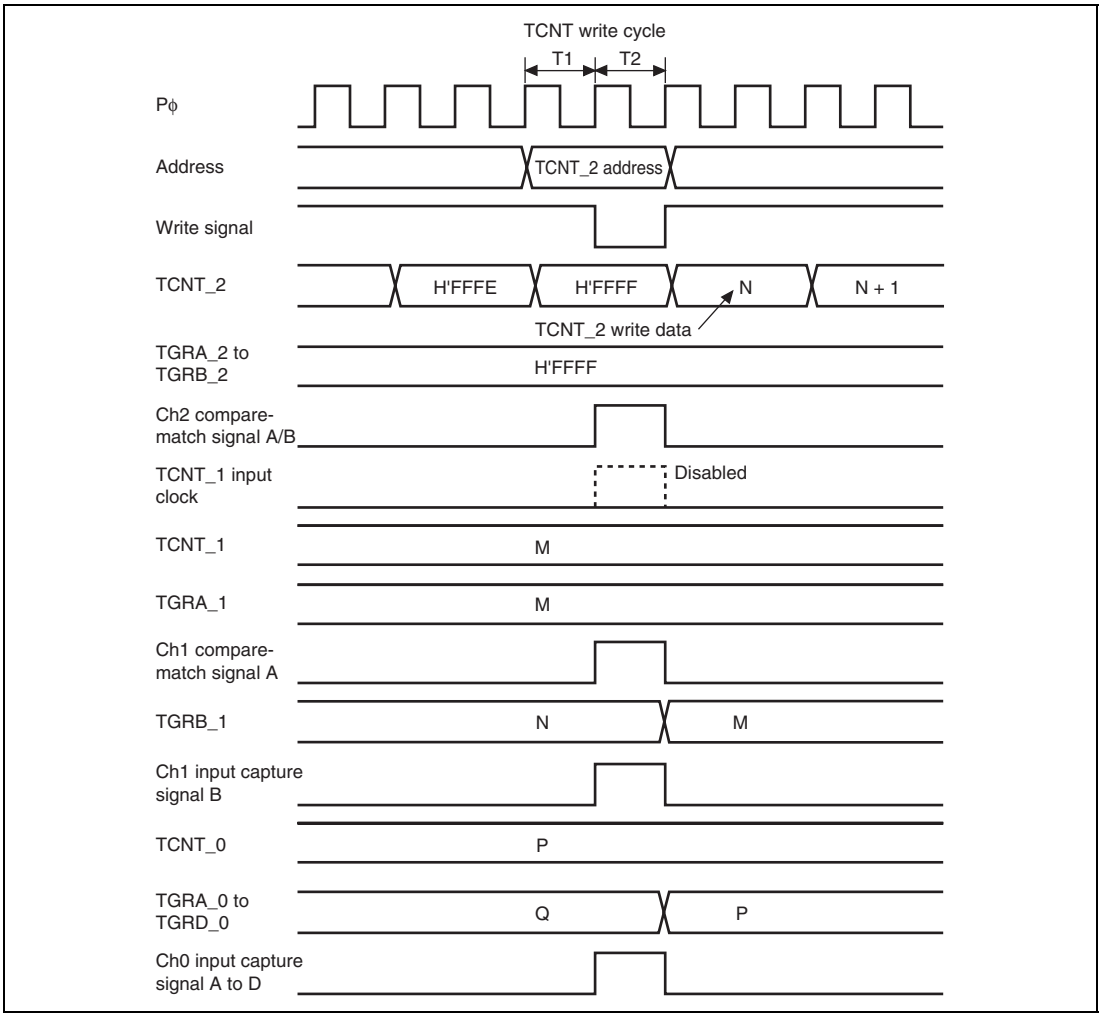


Figure 27.107 TCNT_2 Write and Overflow/Underflow Contention with Cascade Connection

27.7.13 Counter Value during Complementary PWM Mode Stop

When counting operation is suspended with TCNT_3 and TCNT_4 in complementary PWM mode, TCNT_3 has the timer dead time register (TDDR) value, and TCNT_4 is held at H'0000.

When restarting complementary PWM mode, counting begins automatically from the initialized state. This explanatory diagram is shown in figure 27.108.

When counting begins in another operating mode, be sure that TCNT_3 and TCNT_4 are set to the initial values.

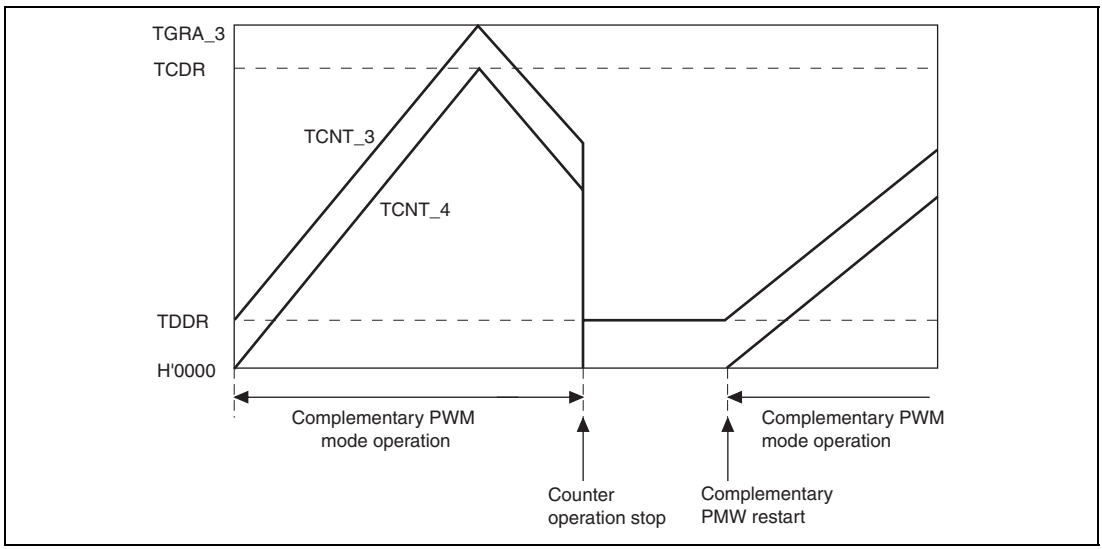


Figure 27.108 Counter Value during Complementary PWM Mode Stop

27.7.14 Buffer Operation Setting in Complementary PWM Mode

In complementary PWM mode, conduct rewrites by buffer operation for the PWM cycle setting register (TGRA_3), timer cycle data register (TCDR), and duty setting registers (TGRB_3, TGRA_4, and TGRB_4).

In complementary PWM mode, channel 3 and channel 4 buffers operate in accordance with bit settings BFA and BFB of TMDR_3. When TMDR_3's BFA bit is set to 1, TGRC_3 functions as a buffer register for TGRA_3. At the same time, TGRC_4 functions as the buffer register for TGRA_4, and TCBR functions as the TCDR's buffer register.

27.7.15 Reset Sync PWM Mode Buffer Operation and Compare Match Flag

When setting buffer operation for reset sync PWM mode, set the BFA and BFB bits of TMDR_4 to 0. The TIOC4C pin will be unable to produce its waveform output if the BFA bit of TMDR_4 is set to 1.

In reset sync PWM mode, the channel 3 and channel 4 buffers operate in accordance with the BFA and BFB bit settings of TMDR_3. For example, if the BFA bit of TMDR_3 is set to 1, TGRC_3 functions as the buffer register for TGRA_3. At the same time, TGRC_4 functions as the buffer register for TGRA_4.

The TGFC bit and TGFD bit of TSR_3 and TSR_4 are not set when TGRC_3 and TGRD_3 are operating as buffer registers.

Figure 27.109 shows an example of operations for TGR_3, TGR_4, TIOC3, and TIOC4, with TMDR_3's BFA and BFB bits set to 1, and TMDR_4's BFA and BFB bits set to 0.

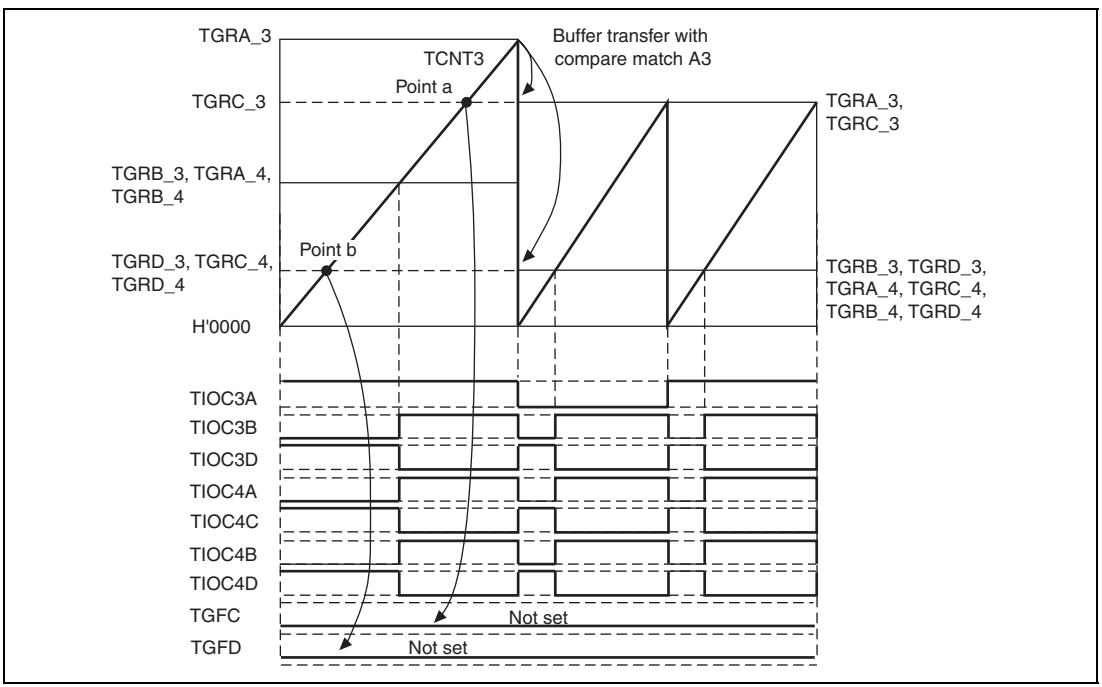


Figure 27.109 Buffer Operation and Compare-Match Flags in Reset Synchronous PWM Mode

27.7.16 Overflow Flags in Reset Synchronous PWM Mode

When set to reset synchronous PWM mode, TCNT_3 and TCNT_4 start counting when the CST3 bit of TSTR is set to 1. At this point, TCNT_4's count clock source and count edge obey the TCR_3 setting.

In reset synchronous PWM mode, with cycle register TGRA_3's set value at H'FFFF, when specifying TGR3A compare-match for the counter clear source, TCNT_3 and TCNT_4 count up to H'FFFF, then a compare-match occurs with TGRA_3, and TCNT_3 and TCNT_4 are both cleared. At this point, TSR's overflow flag TCFV bit is not set.

Figure 27.110 shows a TCFV bit operation example in reset synchronous PWM mode with a set value for cycle register TGRA_3 of H'FFFF, when a TGRA_3 compare-match has been specified without synchronous setting for the counter clear source.

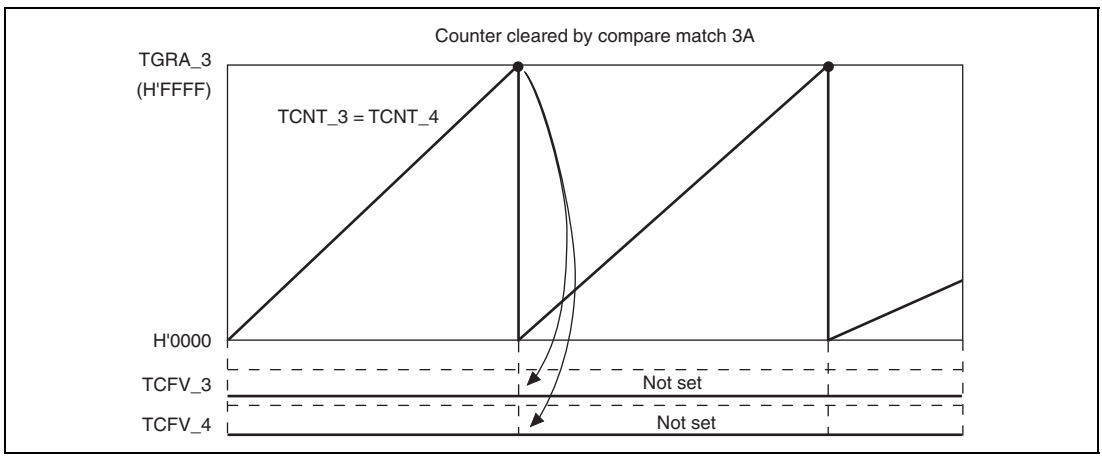


Figure 27.110 Reset Synchronous PWM Mode Overflow Flag

27.7.17 Contention between Overflow/Underflow and Counter Clearing

If overflow/underflow and counter clearing occur simultaneously, the TCFV/TCFU flag in TSR is not set and TCNT clearing takes precedence.

Figure 27.111 shows the operation timing when a TGR compare match is specified as the clearing source, and when H'FFFF is set in TGR.

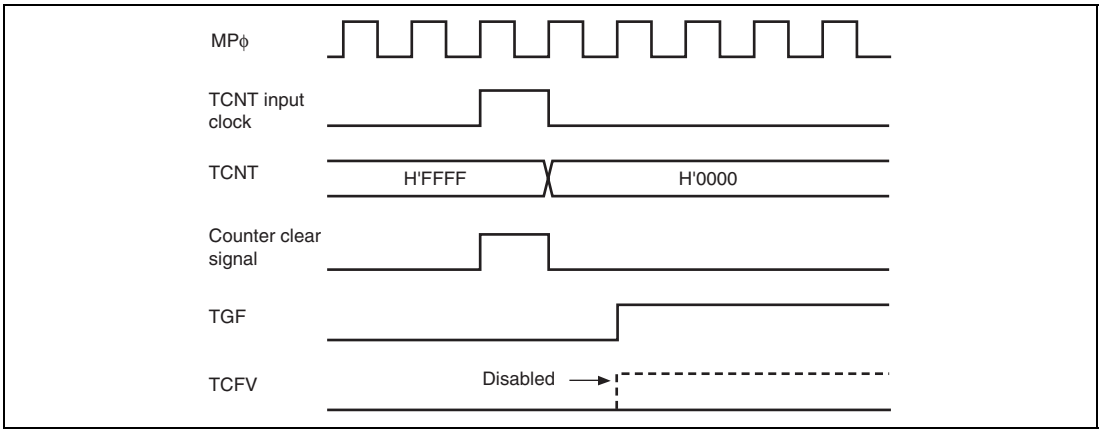


Figure 27.111 Contention between Overflow and Counter Clearing

27.7.18 Contention between TCNT Write and Overflow/Underflow

If there is an up-count or down-count in the T2 state of a TCNT write cycle, and overflow/underflow occurs, the TCNT write takes precedence and the TCFV/TCFU flag in TSR is not set.

Figure 27.112 shows the operation timing when there is contention between TCNT write and overflow.

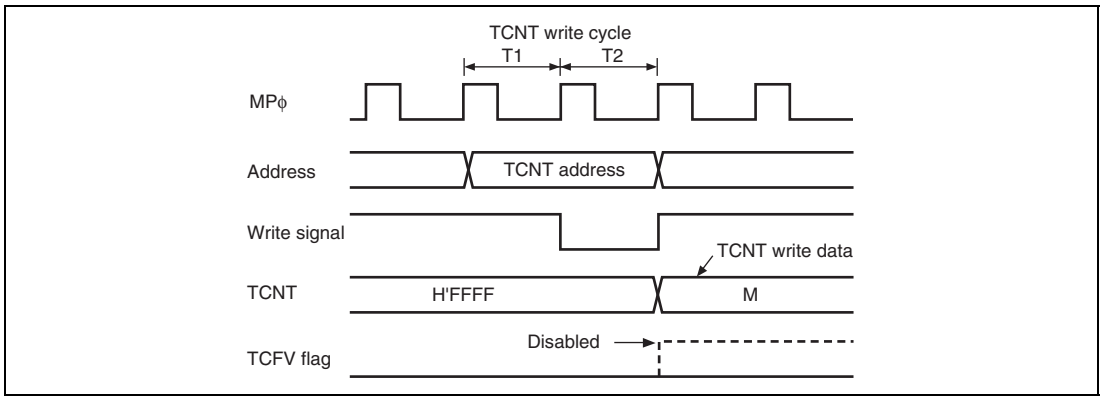


Figure 27.112 Contention between TCNT Write and Overflow

27.7.19 Cautions on Transition from Normal Operation or PWM Mode 1 to Reset-Synchronized PWM Mode

When making a transition from channel 3 or 4 normal operation or PWM mode 1 to reset-synchronized PWM mode, if the counter is halted with the output pins (TIOC3B, TIOC3D, TIOC4A, TIOC4C, TIOC4B, TIOC4D) in the high-level state, followed by the transition to reset-synchronized PWM mode and operation in that mode, the initial pin output will not be correct.

When making a transition from normal operation to reset-synchronized PWM mode, write H'11 to registers TIORH_3, TIORL_3, TIORH_4, and TIORL_4 to initialize the output pins to low level output, then set an initial register value of H'00 before making the mode transition.

When making a transition from PWM mode 1 to reset-synchronized PWM mode, first switch to normal operation, then initialize the output pins to low level output and set an initial register value of H'00 before making the transition to reset-synchronized PWM mode.

27.7.20 Output Level in Complementary PWM Mode and Reset-Synchronized PWM Mode

When channels 3 and 4 are in complementary PWM mode or reset-synchronized PWM mode, the PWM waveform output level is set with the OLSP and OLSN bits in the timer output control register (TOCR). In the case of complementary PWM mode or reset-synchronized PWM mode, TIOR should be set to H'00.

27.7.21 Interrupts in Module Standby Mode

If module standby mode is entered when an interrupt has been requested, it will not be possible to clear the CPU interrupt source or the direct memory access controller activation source. Interrupts should therefore be disabled before entering module standby mode.

27.7.22 Simultaneous Capture of TCNT_1 and TCNT_2 in Cascade Connection

When timer counters 1 and 2 (TCNT_1 and TCNT_2) are operated as a 32-bit counter in cascade connection, the cascade counter value cannot be captured successfully even if input-capture input is simultaneously done to TIOC1A and TIOC2A or to TIOC1B and TIOC2B. This is because the input timing of TIOC1A and TIOC2A or of TIOC1B and TIOC2B may not be the same when external input-capture signals to be input into TCNT_1 and TCNT_2 are taken in synchronization with the internal clock. For example, TCNT_1 (the counter for upper 16 bits) does not capture the count-up value by overflow from TCNT_2 (the counter for lower 16 bits) but captures the count value before the count-up. In this case, the values of TCNT_1 = H'FFF1 and TCNT_2 = H'0000 should be transferred to TGRA_1 and TGRA_2 or to TGRB_1 and TGRB_2, but the values of TCNT_1 = H'FFF0 and TCNT_2 = H'0000 are erroneously transferred.

27.7.23 Notes on Control of Output Waveforms in Synchronous Counter Clearing in Complementary PWM Mode

When either condition (1) or (2) below is satisfied while control of output waveforms at synchronous counter clearing is enabled (the WRE bit in the TWCR register is 1) in complementary PWM mode, the MTU2 will operate as follows.

- The dead time for the PWM output pins will be shorter or no dead time will be generated.
- The negative-phase PWM output pins will output the active level outside the active-level output period.

Condition (1): Synchronous clearing occurs during the PWM output dead time in initial output disabled period (10) (figure 27.113)

Condition (2): Synchronous clearing occurs when TGRB_3 ≤ TDDR, TGRA_4 ≤ TDDR, or TGRB_4 ≤ TDDR is satisfied in initial output disabled period (10) or (11) (figure 27.114).

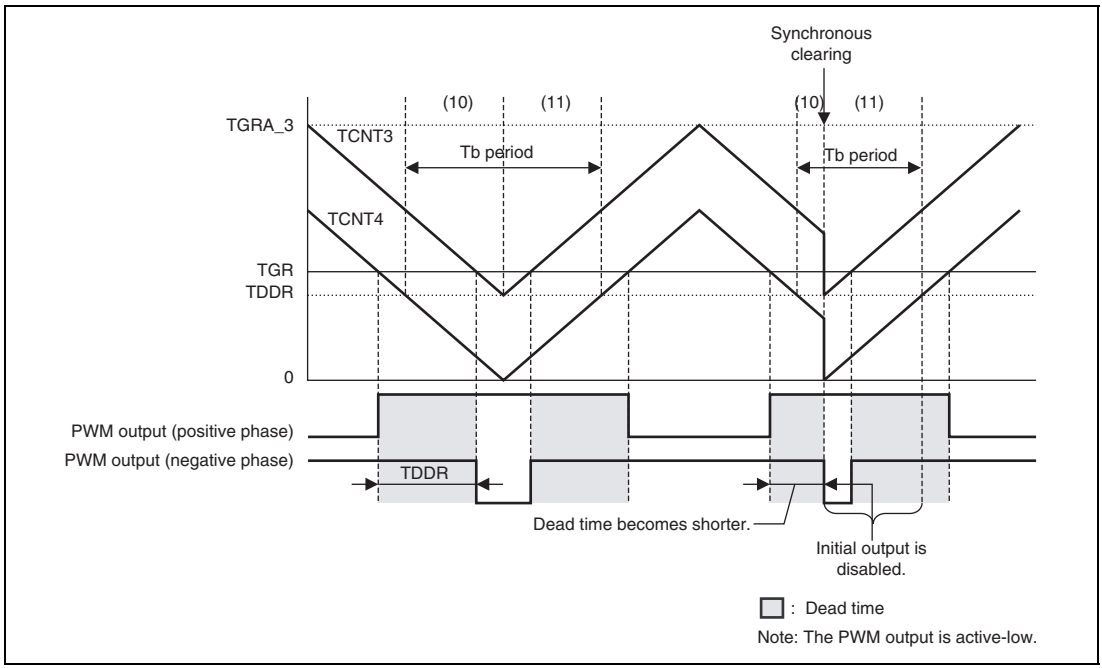


Figure 27.113 Example of Synchronous Clearing under Condition (1)

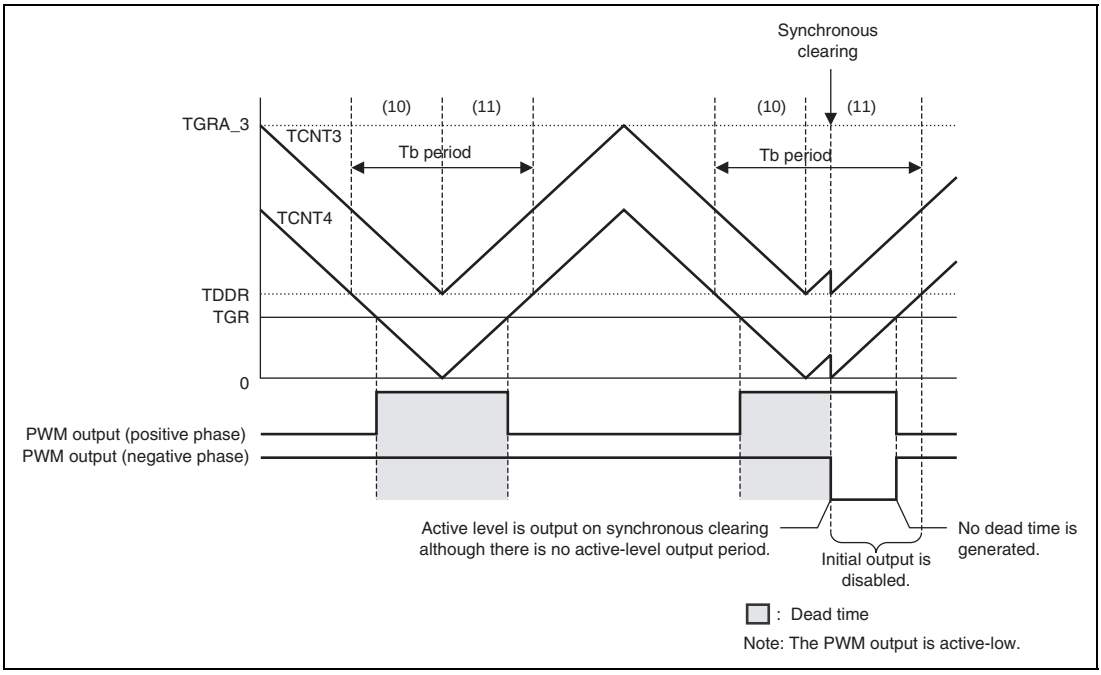


Figure 27.114 Example of Synchronous Clearing under Condition (2)

To avoid this, make sure that synchronous clearing only proceeds when each of the comparison registers (TGRB_3, TGRA_4, and TGRB_4) is set to a value at least twice that in the timer dead time data register (TDDR).

27.8 Output Pin Initialization for Multi-Function Timer Pulse Unit 2

27.8.1 Operating Modes

This module has the following six operating modes. Waveform output is possible in all of these modes.

- Normal mode (channels 0 to 4)
- PWM mode 1 (channels 0 to 4)
- PWM mode 2 (channels 0 to 2)
- Phase counting modes 1 to 4 (channels 1 and 2)
- Complementary PWM mode (channels 3 and 4)
- Reset-synchronized PWM mode (channels 3 and 4)

The output pin initialization method for each of these modes is described in this section.

27.8.2 Reset Start Operation

The output pins of this module (TIOC*) are initialized low by a reset and in standby mode. Since the pin functions are selected using the general I/O ports, when the general I/O port is set, the pin states at that point are output to the ports. When this module output is selected by the general I/O port immediately after a reset, the initial output level, low, is output directly at the port. When the active level is low, the system will operate at this point, and therefore the general I/O port setting should be made after the initialization of the output pins is completed.

Note: Channel number and port notation are substituted for *.

27.8.3 Operation in Case of Re-Setting Due to Error during Operation, etc.

If an error occurs during operation of this module, the module output should be cut by the system. Cutoff is performed by switching the pin output to port output with the general I/O port and outputting the inverse of the active level. The pin initialization procedures for re-setting due to an error during operation, etc., and the procedures for restarting in a different mode after re-setting, are shown below.

This module has six operating modes, as stated above. There are thus 36 mode transition combinations, but some transitions are not available with certain channel and mode combinations. Possible mode transition combinations are shown in table 27.57.

Table 27.57 Mode Transition Combinations

	After					
Before	**Normal**	**PWM1**	**PWM2**	**PCM**	**CPWM**	**RPWM**
Normal	(1)	(2)	(3)	(4)	(5)	(6)
PWM1	(7)	(8)	(9)	(10)	(11)	(12)
PWM2	(13)	(14)	(15)	(16)	None	None
PCM	(17)	(18)	(19)	(20)	None	None
CPWM	(21)	(22)	None	None	(23) (24)	(25)
RPWM	(26)	(27)	None	None	(28)	(29)

[Legend]
Normal: Normal mode
PWM1: PWM mode 1
PWM2: PWM mode 2
PCM: Phase counting modes 1 to 4
CPWM: Complementary PWM mode
RPWM: Reset-synchronized PWM mode

27.8.4 Overview of Initialization Procedures and Mode Transitions in Case of Error during Operation, etc.

- When making a transition to a mode (Normal, PWM1, PWM2, PCM) in which the pin output level is selected by the timer I/O control register (TIOR) setting, initialize the pins by means of a TIOR setting.
- In PWM mode 1, since a waveform is not output to the TIOC*B (TIOC *D) pin, setting TIOR will not initialize the pins. If initialization is required, carry it out in normal mode, then switch to PWM mode 1.
- In PWM mode 2, since a waveform is not output to the cycle register pin, setting TIOR will not initialize the pins. If initialization is required, carry it out in normal mode, then switch to PWM mode 2.
- In normal mode or PWM mode 2, if TGRC and TGRD operate as buffer registers, setting TIOR will not initialize the buffer register pins. If initialization is required, clear buffer mode, carry out initialization, then set buffer mode again.
- In PWM mode 1, if either TGRC or TGRD operates as a buffer register, setting TIOR will not initialize the TGRC pin. To initialize the TGRC pin, clear buffer mode, carry out initialization, then set buffer mode again.
- When making a transition to a mode (CPWM, RPWM) in which the pin output level is selected by the timer output control register (TOCR) setting, switch to normal mode and perform initialization with TIOR, then restore TIOR to its initial value, and temporarily disable channel 3 and 4 output with the timer output master enable register (TOER). Then operate the unit in accordance with the mode setting procedure (TOCR setting, TMDR setting, TOER setting).

Note: Channel number is substituted for * indicated in this article.

Pin initialization procedures are described below for the numbered combinations in table 27.57. The active level is assumed to be low.

(1) Operation when Error Occurs during Normal Mode Operation, and Operation is Restarted in Normal Mode

Figure 27.115 shows an explanatory diagram of the case where an error occurs in normal mode and operation is restarted in normal mode after re-setting.

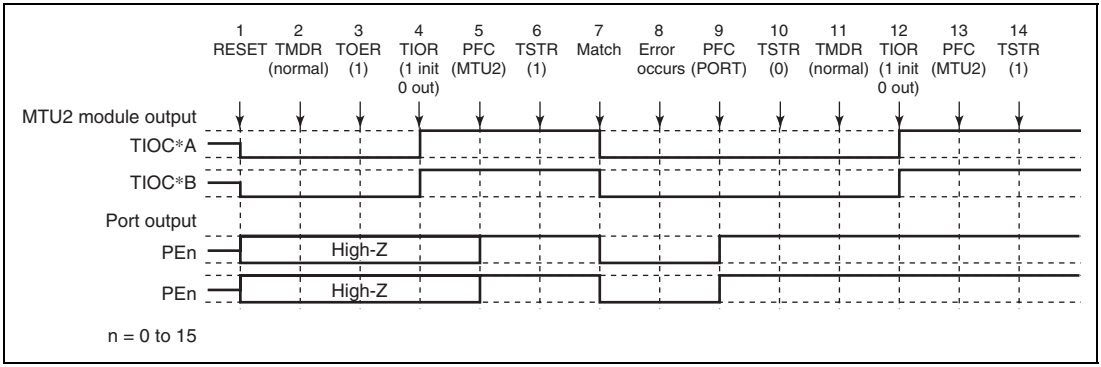


Figure 27.115 Error Occurrence in Normal Mode, Recovery in Normal Mode

1. After a reset, the module output is low and ports are in the high-impedance state.
2. After a reset, the TMDR setting is for normal mode.
3. For channels 3 and 4, enable output with TOER before initializing the pins with TIOR.
4. Initialize the pins with TIOR. (The example shows initial high output, with low output on compare-match occurrence.)
5. Set the multi-function timer pulse unit 2 output with the general I/O port.
6. The count operation is started by TSTR.
7. Output goes low on compare-match occurrence.
8. An error occurs.
9. Set port output with the general I/O port and output the inverse of the active level.
10. The count operation is stopped by TSTR.
11. Not necessary when restarting in normal mode.
12. Initialize the pins with TIOR.
13. Set the multi-function timer pulse unit 2 output with the general I/O port.
14. Operation is restarted by TSTR.

(2) Operation when Error Occurs during Normal Mode Operation, and Operation is Restarted in PWM Mode 1

Figure 27.116 shows an explanatory diagram of the case where an error occurs in normal mode and operation is restarted in PWM mode 1 after re-setting.

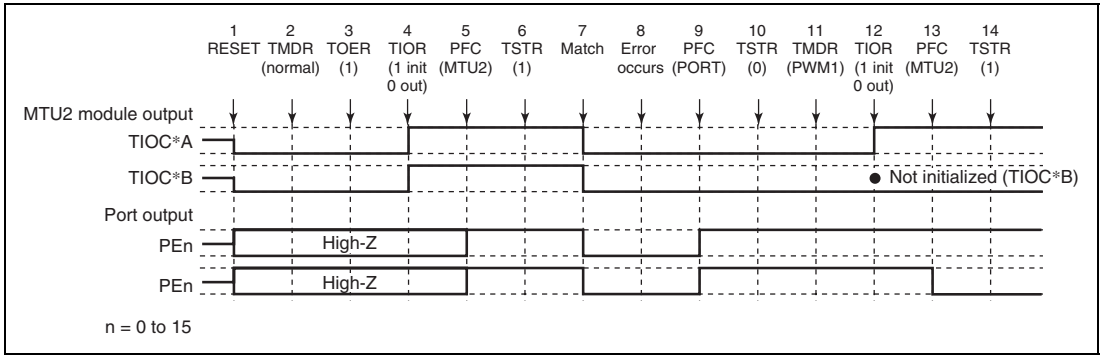


Figure 27.116 Error Occurrence in Normal Mode, Recovery in PWM Mode 1

1 to 10 are the same as in figure 27.115.

11. Set PWM mode 1.
12. Initialize the pins with TIOR. (In PWM mode 1, the TIOC*B side is not initialized. If initialization is required, initialize in normal mode, and then switch to PWM mode 1.)
13. Set the multi-function timer pulse unit 2 output with the general I/O port.
14. Operation is restarted by TSTR.

(3) Operation when Error Occurs during Normal Mode Operation, and Operation is Restarted in PWM Mode 2

Figure 27.117 shows an explanatory diagram of the case where an error occurs in normal mode and operation is restarted in PWM mode 2 after re-setting.

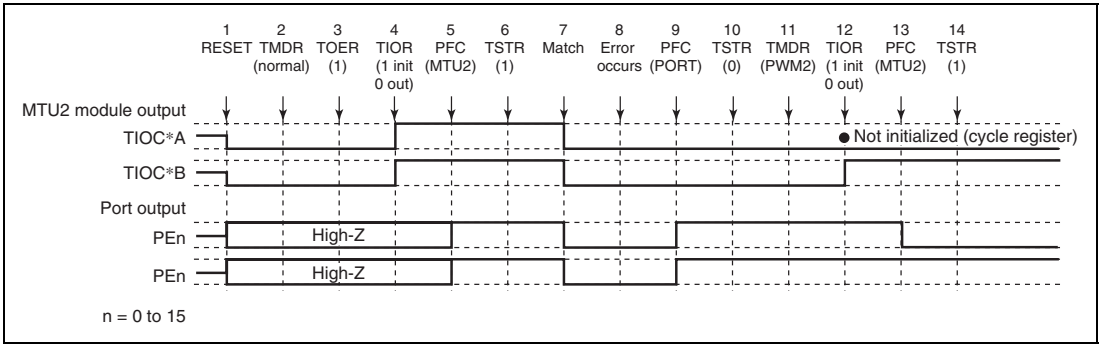


Figure 27.117 Error Occurrence in Normal Mode, Recovery in PWM Mode 2

1 to 10 are the same as in figure 27.115.

11. Set PWM mode 2.
12. Initialize the pins with TIOR. (In PWM mode 2, the cycle register pins are not initialized. If initialization is required, initialize in normal mode, and then switch to PWM mode 2.)
13. Set the multi-function timer pulse unit 2 output with the general I/O port.
14. Operation is restarted by TSTR.

Note: PWM mode 2 can only be set for channels 0 to 2, and therefore TOER setting is not necessary.

(4) Operation when Error Occurs during Normal Mode Operation, and Operation is Restarted in Phase Counting Mode

Figure 27.118 shows an explanatory diagram of the case where an error occurs in normal mode and operation is restarted in phase counting mode after re-setting.

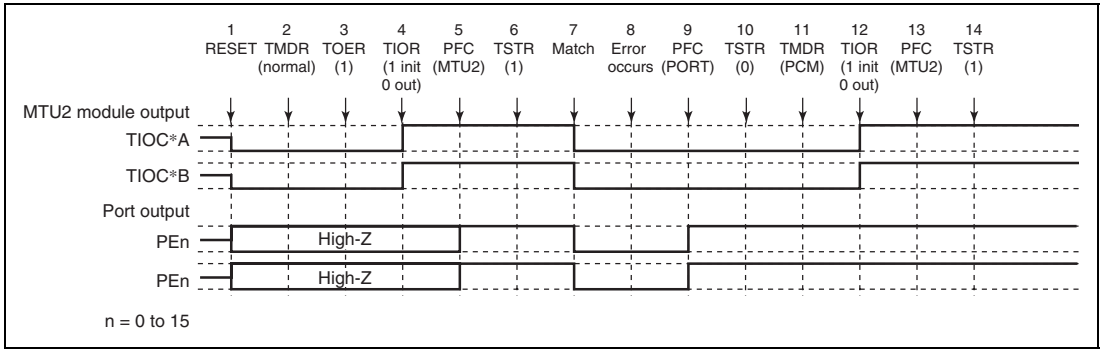


Figure 27.118 Error Occurrence in Normal Mode, Recovery in Phase Counting Mode

1 to 10 are the same as in figure 27.115.

11. Set phase counting mode.
12. Initialize the pins with TIOR.
13. Set the multi-function timer pulse unit 2 output with the general I/O port.
14. Operation is restarted by TSTR.

Note: Phase counting mode can only be set for channels 1 and 2, and therefore TOER setting is not necessary.

(5) Operation when Error Occurs during Normal Mode Operation, and Operation is Restarted in Complementary PWM Mode

Figure 27.119 shows an explanatory diagram of the case where an error occurs in normal mode and operation is restarted in complementary PWM mode after re-setting.

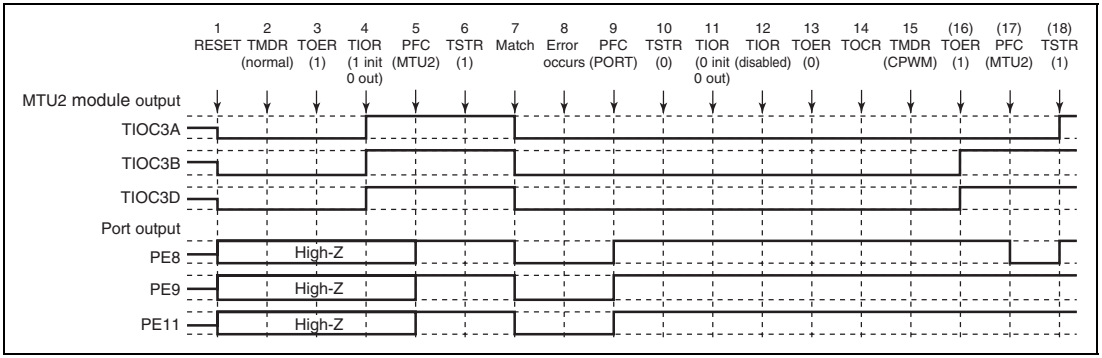


Figure 27.119 Error Occurrence in Normal Mode, Recovery in Complementary PWM Mode

1 to 10 are the same as in figure 27.115.

11. Initialize the normal mode waveform generation section with TIOR.
12. Disable operation of the normal mode waveform generation section with TIOR.
13. Disable channel 3 and 4 output with TOER.
14. Select the complementary PWM output level and cyclic output enabling/disabling with TOCR.
15. Set complementary PWM.
16. Enable channel 3 and 4 output with TOER.
17. Set the multi-function timer pulse unit 2 output with the general I/O port.
18. Operation is restarted by TSTR.

(6) Operation when Error Occurs during Normal Mode Operation, and Operation is Restarted in Reset-Synchronized PWM Mode

Figure 27.120 shows an explanatory diagram of the case where an error occurs in normal mode and operation is restarted in reset-synchronized PWM mode after re-setting.

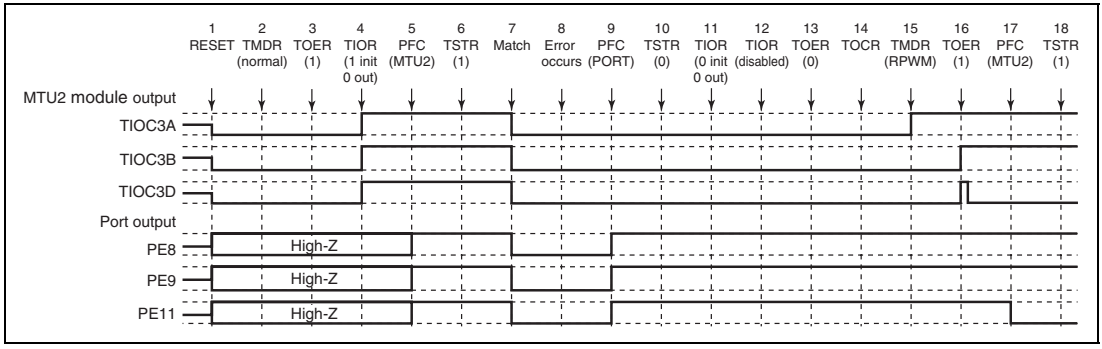


Figure 27.120 Error Occurrence in Normal Mode, Recovery in Reset-Synchronized PWM Mode

1 to 13 are the same as in figure 27.115.

14. Select the reset-synchronized PWM output level and cyclic output enabling/disabling with TOCR.
15. Set reset-synchronized PWM.
16. Enable channel 3 and 4 output with TOER.
17. Set the multi-function timer pulse unit 2 output with the general I/O port.
18. Operation is restarted by TSTR.

(7) Operation when Error Occurs during PWM Mode 1 Operation, and Operation is Restarted in Normal Mode

Figure 27.121 shows an explanatory diagram of the case where an error occurs in PWM mode 1 and operation is restarted in normal mode after re-setting.

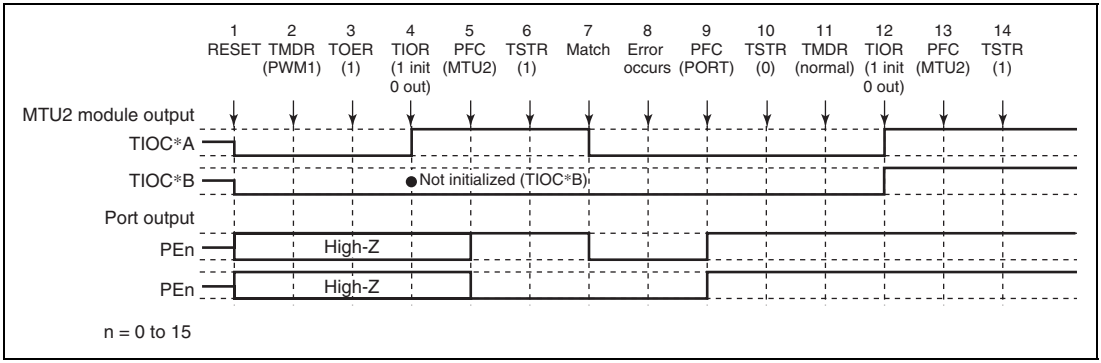


Figure 27.121 Error Occurrence in PWM Mode 1, Recovery in Normal Mode

1. After a reset, the module output is low and ports are in the high-impedance state.
2. Set PWM mode 1.
3. For channels 3 and 4, enable output with TOER before initializing the pins with TIOR.
4. Initialize the pins with TIOR. (The example shows initial high output, with low output on compare-match occurrence. In PWM mode 1, the TIOC*B side is not initialized.)
5. Set the multi-function timer pulse unit 2 output with the general I/O port.
6. The count operation is started by TSTR.
7. Output goes low on compare-match occurrence.
8. An error occurs.
9. Set port output with the general I/O port and output the inverse of the active level.
10. The count operation is stopped by TSTR.
11. Set normal mode.
12. Initialize the pins with TIOR.
13. Set the multi-function timer pulse unit 2 output with the general I/O port.
14. Operation is restarted by TSTR.

(8) Operation when Error Occurs during PWM Mode 1 Operation, and Operation is Restarted in PWM Mode 1

Figure 27.122 shows an explanatory diagram of the case where an error occurs in PWM mode 1 and operation is restarted in PWM mode 1 after re-setting.

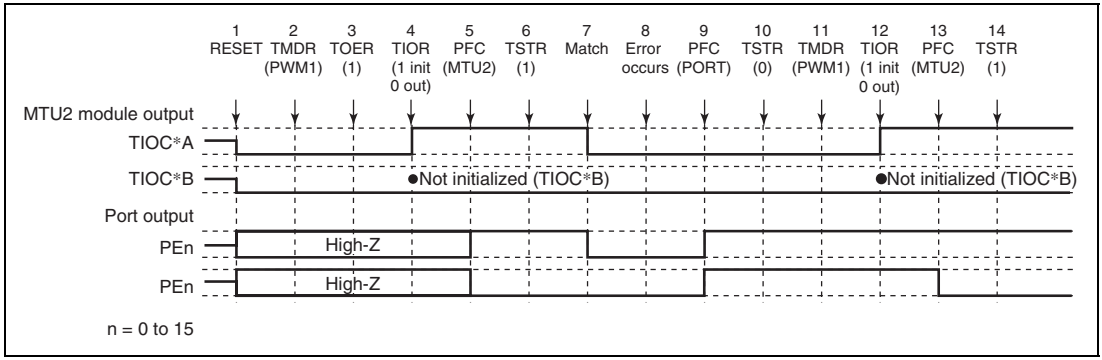


Figure 27.122 Error Occurrence in PWM Mode 1, Recovery in PWM Mode 1

1 to 10 are the same as in figure 27.121.

11. Not necessary when restarting in PWM mode 1.
12. Initialize the pins with TIOR. (In PWM mode 1, the TIOC*B side is not initialized.)
13. Set the multi-function timer pulse unit 2 output with the general I/O port.
14. Operation is restarted by TSTR.

(9) Operation when Error Occurs during PWM Mode 1 Operation, and Operation is Restarted in PWM Mode 2

Figure 27.123 shows an explanatory diagram of the case where an error occurs in PWM mode 1 and operation is restarted in PWM mode 2 after re-setting.

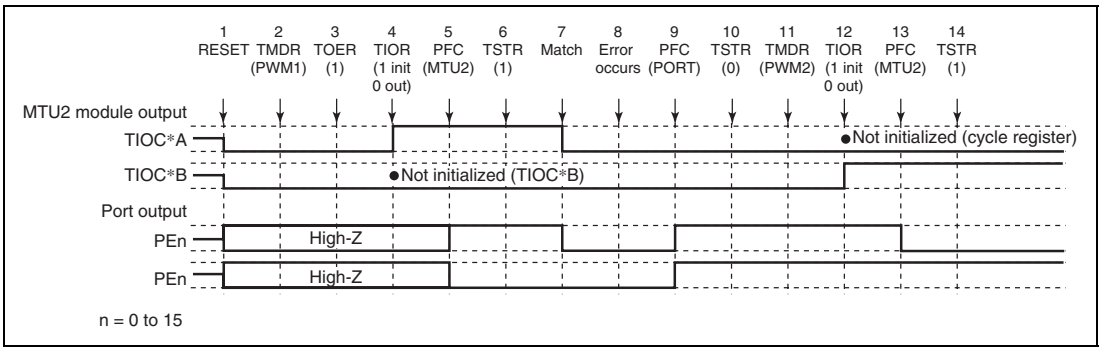


Figure 27.123 Error Occurrence in PWM Mode 1, Recovery in PWM Mode 2

1 to 10 are the same as in figure 27.121.

11. Set PWM mode 2.
12. Initialize the pins with TIOR. (In PWM mode 2, the cycle register pins are not initialized.)
13. Set the multi-function timer pulse unit 2 output with the general I/O port.
14. Operation is restarted by TSTR.

Note: PWM mode 2 can only be set for channels 0 to 2, and therefore TOER setting is not necessary.

(10) Operation when Error Occurs during PWM Mode 1 Operation, and Operation is Restarted in Phase Counting Mode

Figure 27.124 shows an explanatory diagram of the case where an error occurs in PWM mode 1 and operation is restarted in phase counting mode after re-setting.

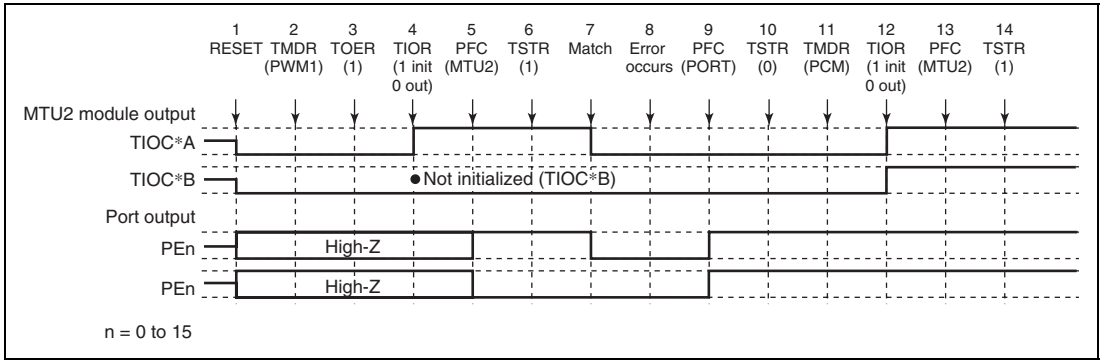


Figure 27.124 Error Occurrence in PWM Mode 1, Recovery in Phase Counting Mode

1 to 10 are the same as in figure 27.121.

11. Set phase counting mode.
12. Initialize the pins with TIOR.
13. Set the multi-function timer pulse unit 2 output with the general I/O port.
14. Operation is restarted by TSTR.

Note: Phase counting mode can only be set for channels 1 and 2, and therefore TOER setting is not necessary.

(11) Operation when Error Occurs during PWM Mode 1 Operation, and Operation is Restarted in Complementary PWM Mode

Figure 27.125 shows an explanatory diagram of the case where an error occurs in PWM mode 1 and operation is restarted in complementary PWM mode after re-setting.

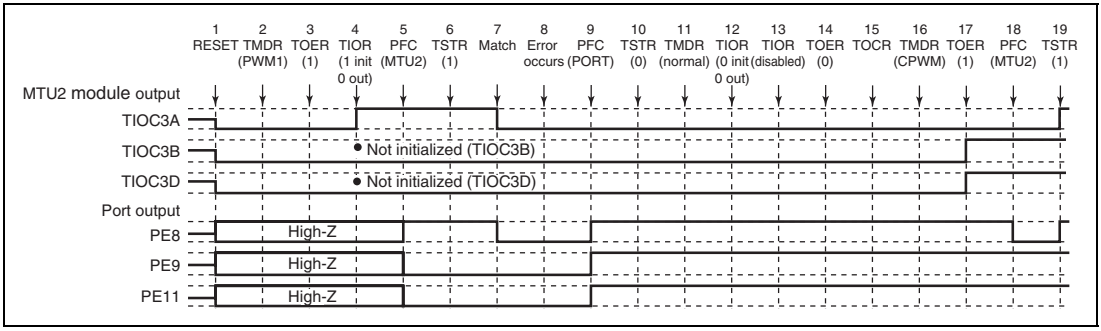


Figure 27.125 Error Occurrence in PWM Mode 1, Recovery in Complementary PWM Mode

1 to 10 are the same as in figure 27.121.

11. Set normal mode for initialization of the normal mode waveform generation section.
12. Initialize the PWM mode 1 waveform generation section with TIOR.
13. Disable operation of the PWM mode 1 waveform generation section with TIOR.
14. Disable channel 3 and 4 output with TOER.
15. Select the complementary PWM output level and cyclic output enabling/disabling with TOCR.
16. Set complementary PWM.
17. Enable channel 3 and 4 output with TOER.
18. Set the multi-function timer pulse unit 2 output with the general I/O port.
19. Operation is restarted by TSTR.

(12) Operation when Error Occurs during PWM Mode 1 Operation, and Operation is Restarted in Reset-Synchronized PWM Mode

Figure 27.126 shows an explanatory diagram of the case where an error occurs in PWM mode 1 and operation is restarted in reset-synchronized PWM mode after re-setting.

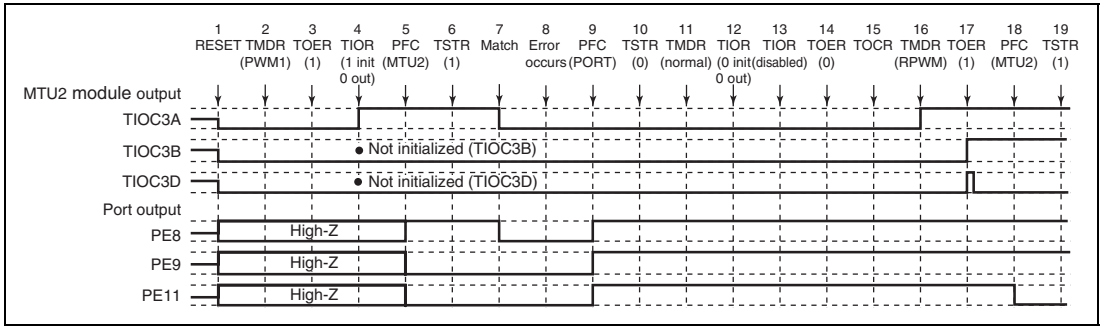


Figure 27.126 Error Occurrence in PWM Mode 1, Recovery in Reset-Synchronized PWM Mode

1 to 14 are the same as in figure 27.125.

15. Select the reset-synchronized PWM output level and cyclic output enabling/disabling with TOCR.
16. Set reset-synchronized PWM.
17. Enable channel 3 and 4 output with TOER.
18. Set the multi-function timer pulse unit 2 output with the general I/O port.
19. Operation is restarted by TSTR.

(13) Operation when Error Occurs during PWM Mode 2 Operation, and Operation is Restarted in Normal Mode

Figure 27.127 shows an explanatory diagram of the case where an error occurs in PWM mode 2 and operation is restarted in normal mode after re-setting.

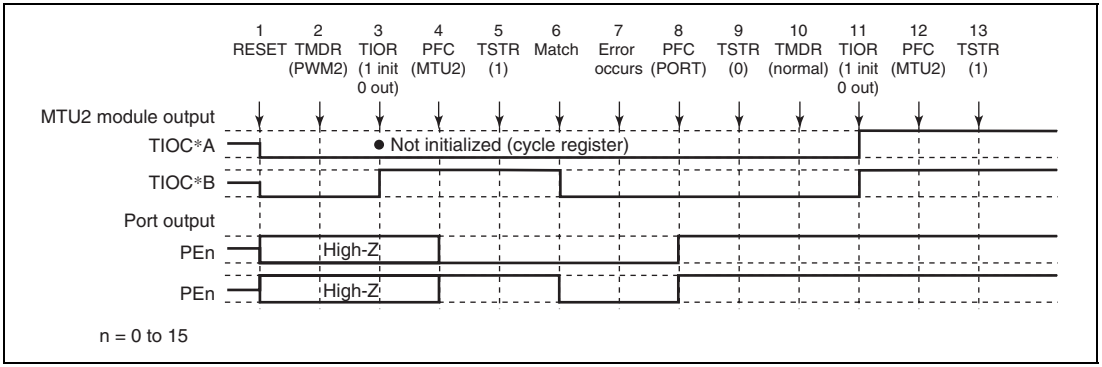


Figure 27.127 Error Occurrence in PWM Mode 2, Recovery in Normal Mode

1. After a reset, the module output is low and ports are in the high-impedance state.
2. Set PWM mode 2.
3. Initialize the pins with TIOR. (The example shows initial high output, with low output on compare-match occurrence. In PWM mode 2, the cycle register pins are not initialized. In the example, TIOC *A is the cycle register.)
4. Set the multi-function timer pulse unit 2 output with the general I/O port.
5. The count operation is started by TSTR.
6. Output goes low on compare-match occurrence.
7. An error occurs.
8. Set port output with the general I/O port and output the inverse of the active level.
9. The count operation is stopped by TSTR.
10. Set normal mode.
11. Initialize the pins with TIOR.
12. Set the multi-function timer pulse unit 2 output with the general I/O port.
13. Operation is restarted by TSTR.

(14) Operation when Error Occurs during PWM Mode 2 Operation, and Operation is Restarted in PWM Mode 1

Figure 27.128 shows an explanatory diagram of the case where an error occurs in PWM mode 2 and operation is restarted in PWM mode 1 after re-setting.

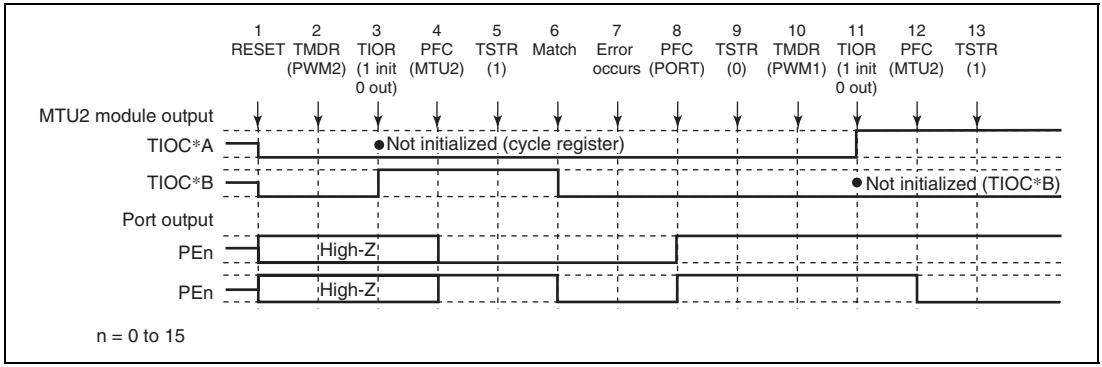


Figure 27.128 Error Occurrence in PWM Mode 2, Recovery in PWM Mode 1

1 to 9 are the same as in figure 27.127.

10. Set PWM mode 1.
11. Initialize the pins with TIOR. (In PWM mode 1, the TIOC*B side is not initialized.)
12. Set the multi-function timer pulse unit 2 output with the general I/O port.
13. Operation is restarted by TSTR.

(15) Operation when Error Occurs during PWM Mode 2 Operation, and Operation is Restarted in PWM Mode 2

Figure 27.129 shows an explanatory diagram of the case where an error occurs in PWM mode 2 and operation is restarted in PWM mode 2 after re-setting.

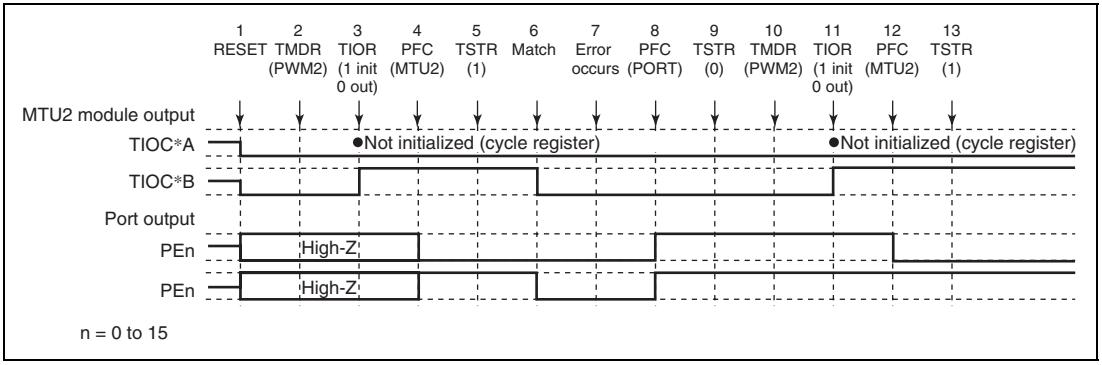


Figure 27.129 Error Occurrence in PWM Mode 2, Recovery in PWM Mode 2

1 to 9 are the same as in figure 27.127.

10. Not necessary when restarting in PWM mode 2.
11. Initialize the pins with TIOR. (In PWM mode 2, the cycle register pins are not initialized.)
12. Set the multi-function timer pulse unit 2 output with the general I/O port.
13. Operation is restarted by TSTR.

(16) Operation when Error Occurs during PWM Mode 2 Operation, and Operation is Restarted in Phase Counting Mode

Figure 27.130 shows an explanatory diagram of the case where an error occurs in PWM mode 2 and operation is restarted in phase counting mode after re-setting.

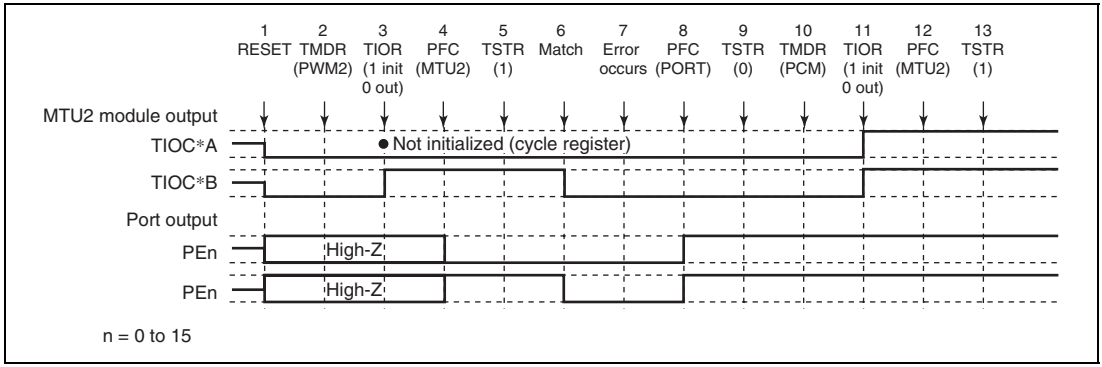


Figure 27.130 Error Occurrence in PWM Mode 2, Recovery in Phase Counting Mode

1 to 9 are the same as in figure 27.127.

10. Set phase counting mode.
11. Initialize the pins with TIOR.
12. Set the multi-function timer pulse unit 2 output with the general I/O port.
13. Operation is restarted by TSTR.

(17) Operation when Error Occurs during Phase Counting Mode Operation, and Operation is Restarted in Normal Mode

Figure 27.131 shows an explanatory diagram of the case where an error occurs in phase counting mode and operation is restarted in normal mode after re-setting.

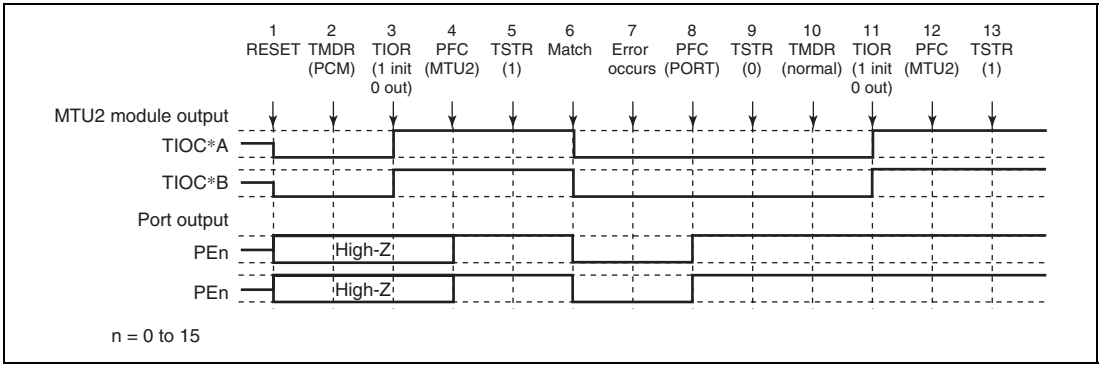


Figure 27.131 Error Occurrence in Phase Counting Mode, Recovery in Normal Mode

1. After a reset, the module output is low and ports are in the high-impedance state.
2. Set phase counting mode.
3. Initialize the pins with TIOR. (The example shows initial high output, with low output on compare-match occurrence.)
4. Set the multi-function timer pulse unit 2 output with the general I/O port.
5. The count operation is started by TSTR.
6. Output goes low on compare-match occurrence.
7. An error occurs.
8. Set port output with the general I/O port and output the inverse of the active level.
9. The count operation is stopped by TSTR.
10. Set in normal mode.
11. Initialize the pins with TIOR.
12. Set the multi-function timer pulse unit 2 output with the general I/O port.
13. Operation is restarted by TSTR.

(18) Operation when Error Occurs during Phase Counting Mode Operation, and Operation is Restarted in PWM Mode 1

Figure 27.132 shows an explanatory diagram of the case where an error occurs in phase counting mode and operation is restarted in PWM mode 1 after re-setting.

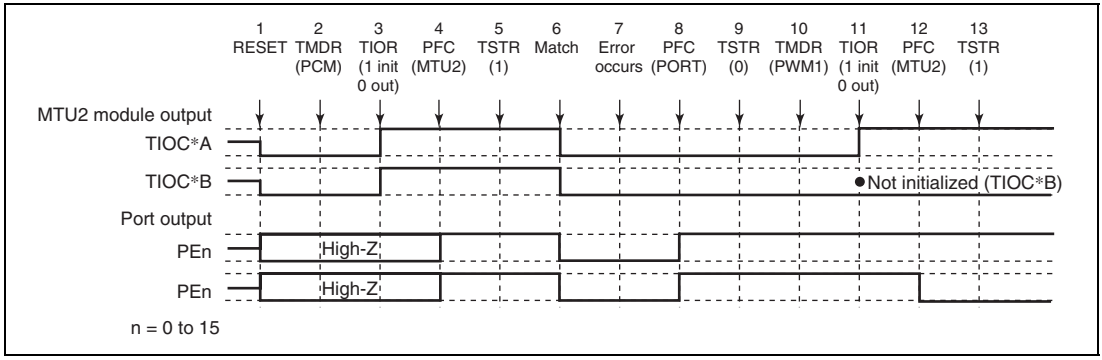


Figure 27.132 Error Occurrence in Phase Counting Mode, Recovery in PWM Mode 1

1 to 9 are the same as in figure 27.131.

10. Set PWM mode 1.
11. Initialize the pins with TIOR. (In PWM mode 1, the TIOC *B side is not initialized.)
12. Set the multi-function timer pulse unit 2 output with the general I/O port.
13. Operation is restarted by TSTR.

(19) Operation when Error Occurs during Phase Counting Mode Operation, and Operation is Restarted in PWM Mode 2

Figure 27.133 shows an explanatory diagram of the case where an error occurs in phase counting mode and operation is restarted in PWM mode 2 after re-setting.

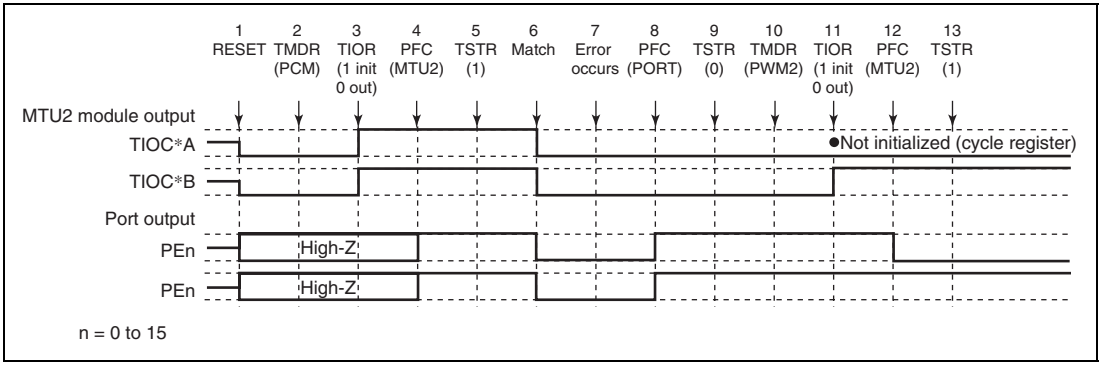


Figure 27.133 Error Occurrence in Phase Counting Mode, Recovery in PWM Mode 2

1 to 9 are the same as in figure 27.131.

10. Set PWM mode 2.
11. Initialize the pins with TIOR. (In PWM mode 2, the cycle register pins are not initialized.)
12. Set the multi-function timer pulse unit 2 output with the general I/O port.
13. Operation is restarted by TSTR.

(20) Operation when Error Occurs during Phase Counting Mode Operation, and Operation is Restarted in Phase Counting Mode

Figure 27.134 shows an explanatory diagram of the case where an error occurs in phase counting mode and operation is restarted in phase counting mode after re-setting.

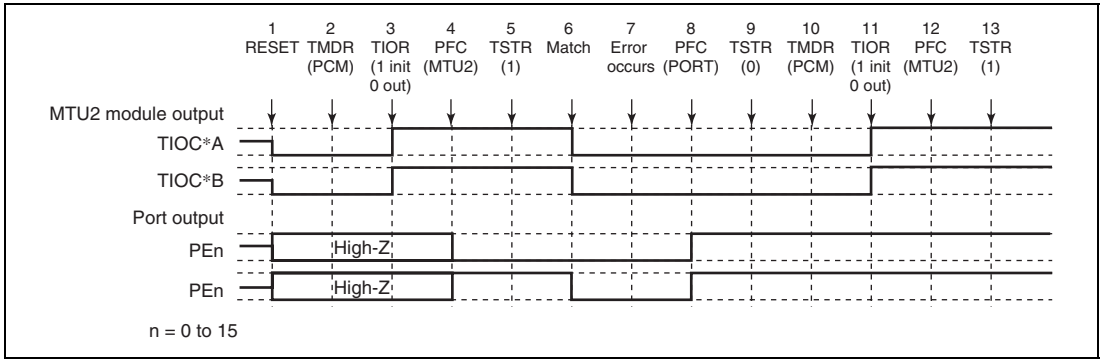


Figure 27.134 Error Occurrence in Phase Counting Mode, Recovery in Phase Counting Mode

1 to 9 are the same as in figure 27.131.

10. Not necessary when restarting in phase counting mode.
11. Initialize the pins with TIOR.
12. Set the multi-function timer pulse unit 2 output with the general I/O port.
13. Operation is restarted by TSTR.

(21) Operation when Error Occurs during Complementary PWM Mode Operation, and Operation is Restarted in Normal Mode

Figure 27.135 shows an explanatory diagram of the case where an error occurs in complementary PWM mode and operation is restarted in normal mode after re-setting.

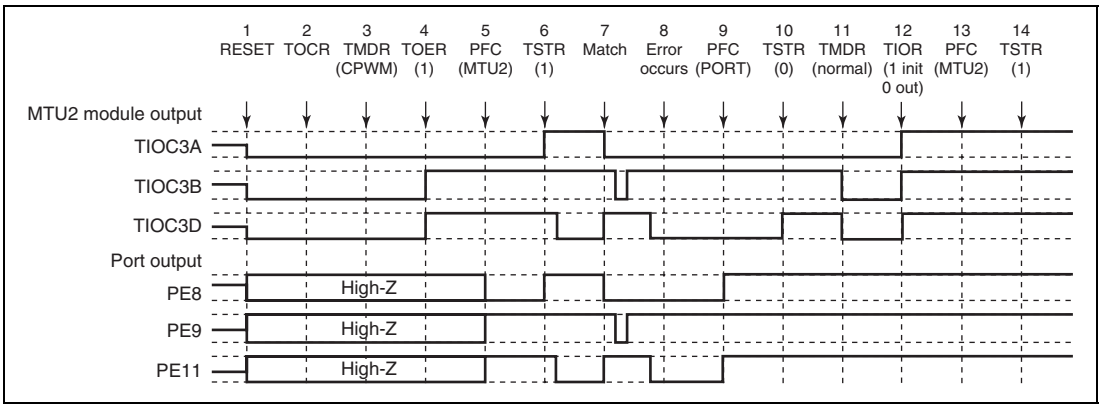


Figure 27.135 Error Occurrence in Complementary PWM Mode, Recovery in Normal Mode

1. After a reset, the module output is low and ports are in the high-impedance state.
2. Select the complementary PWM output level and cyclic output enabling/disabling with TOCR.
3. Set complementary PWM.
4. Enable channel 3 and 4 output with TOER.
5. Set the multi-function timer pulse unit 2 output with the general I/O port.
6. The count operation is started by TSTR.
7. The complementary PWM waveform is output on compare-match occurrence.
8. An error occurs.
9. Set port output with the general I/O port and output the inverse of the active level.
10. The count operation is stopped by TSTR. (This module outputs the same value as the complementary PWM output initial value.)
11. Set normal mode. (This module outputs a low-level signal.)
12. Initialize the pins with TIOR.
13. Set the multi-function timer pulse unit 2 output with the general I/O port.
14. Operation is restarted by TSTR.

(22) Operation when Error Occurs during Complementary PWM Mode Operation, and Operation is Restarted in PWM Mode 1

Figure 27.136 shows an explanatory diagram of the case where an error occurs in complementary PWM mode and operation is restarted in PWM mode 1 after re-setting.

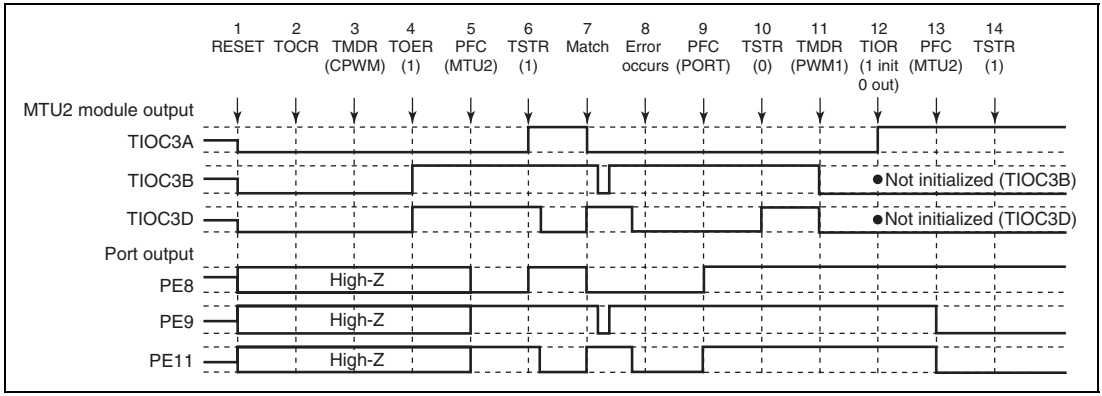


Figure 27.136 Error Occurrence in Complementary PWM Mode, Recovery in PWM Mode 1

1 to 10 are the same as in figure 27.135.

11. Set PWM mode 1. (This module outputs a low-level signal.)
12. Initialize the pins with TIOR. (In PWM mode 1, the TIOC *B side is not initialized.)
13. Set the multi-function timer pulse unit 2 output with the general I/O port.
14. Operation is restarted by TSTR.

(23) Operation when Error Occurs during Complementary PWM Mode Operation, and Operation is Restarted in Complementary PWM Mode

Figure 27.137 shows an explanatory diagram of the case where an error occurs in complementary PWM mode and operation is restarted in complementary PWM mode after re-setting (when operation is restarted using the cycle and duty settings at the time the counter was stopped).

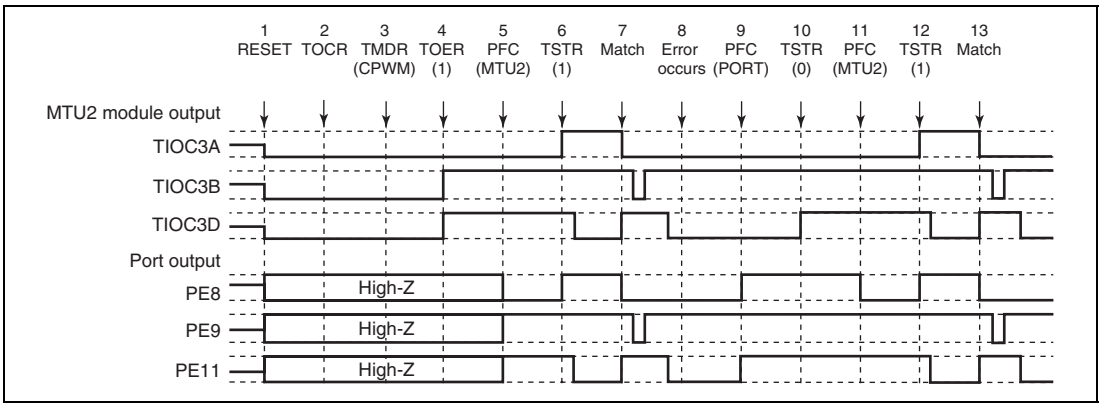


Figure 27.137 Error Occurrence in Complementary PWM Mode, Recovery in Complementary PWM Mode

1 to 10 are the same as in figure 27.135.

11. Set the multi-function timer pulse unit 2 output with the general I/O port.
12. Operation is restarted by TSTR.
13. The complementary PWM waveform is output on compare-match occurrence.

(24) Operation when Error Occurs during Complementary PWM Mode Operation, and Operation is Restarted in Complementary PWM Mode

Figure 27.138 shows an explanatory diagram of the case where an error occurs in complementary PWM mode and operation is restarted in complementary PWM mode after re-setting (when operation is restarted using completely new cycle and duty settings).

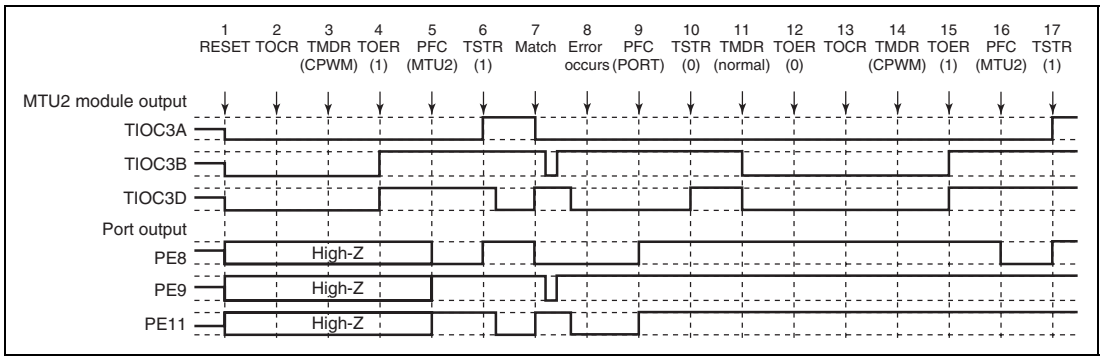


Figure 27.138 Error Occurrence in Complementary PWM Mode, Recovery in Complementary PWM Mode

1 to 10 are the same as in figure 27.135.

11. Set normal mode and make new settings. (This module outputs a low-level signal.)
12. Disable channel 3 and 4 output with TOER.
13. Select the complementary PWM mode output level and cyclic output enabling/disabling with TOCR.
14. Set complementary PWM.
15. Enable channel 3 and 4 output with TOER.
16. Set the multi-function timer pulse unit 2 output with the general I/O port.
17. Operation is restarted by TSTR.

(25) Operation when Error Occurs during Complementary PWM Mode Operation, and Operation is Restarted in Reset-Synchronized PWM Mode

Figure 27.139 shows an explanatory diagram of the case where an error occurs in complementary PWM mode and operation is restarted in reset-synchronized PWM mode.

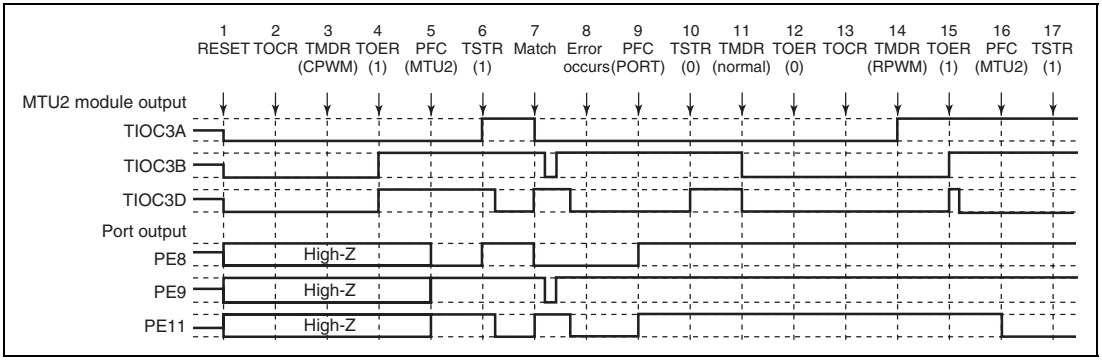


Figure 27.139 Error Occurrence in Complementary PWM Mode, Recovery in Reset-Synchronized PWM Mode

1 to 10 are the same as in figure 27.135.

11. Set normal mode. (This module outputs a low-level signal.)
12. Disable channel 3 and 4 output with TOER.
13. Select the reset-synchronized PWM mode output level and cyclic output enabling/disabling with TOCR.
14. Set reset-synchronized PWM.
15. Enable channel 3 and 4 output with TOER.
16. Set the multi-function timer pulse unit 2 output with the general I/O port.
17. Operation is restarted by TSTR.

(26) Operation when Error Occurs during Reset-Synchronized PWM Mode Operation, and Operation is Restarted in Normal Mode

Figure 27.140 shows an explanatory diagram of the case where an error occurs in reset-synchronized PWM mode and operation is restarted in normal mode after re-setting.

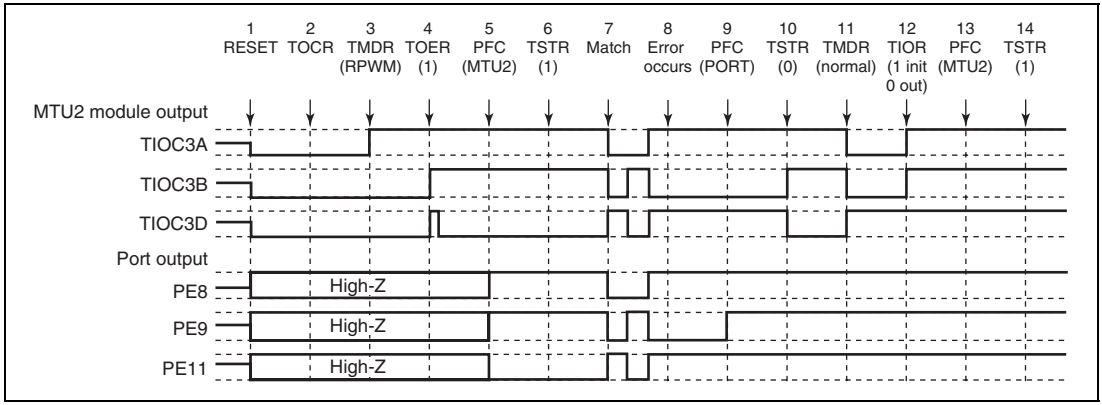


Figure 27.140 Error Occurrence in Reset-Synchronized PWM Mode, Recovery in Normal Mode

1. After a reset, the module output is low and ports are in the high-impedance state.
2. Select the reset-synchronized PWM output level and cyclic output enabling/disabling with TOCR.
3. Set reset-synchronized PWM.
4. Enable channel 3 and 4 output with TOER.
5. Set the multi-function timer pulse unit 2 output with the general I/O port.
6. The count operation is started by TSTR.
7. The reset-synchronized PWM waveform is output on compare-match occurrence.
8. An error occurs.
9. Set port output with the general I/O port and output the inverse of the active level.
10. The count operation is stopped by TSTR. (This module outputs the same value as the reset-synchronized PWM output initial value.)
11. Set normal mode. (The positive phase output from this module is low, and negative phase output is high.)
12. Initialize the pins with TIOR.
13. Set the multi-function timer pulse unit 2 output with the general I/O port.
14. Operation is restarted by TSTR.

(27) Operation when Error Occurs during Reset-Synchronized PWM Mode Operation, and Operation is Restarted in PWM Mode 1

Figure 27.141 shows an explanatory diagram of the case where an error occurs in reset-synchronized PWM mode and operation is restarted in PWM mode 1 after re-setting.

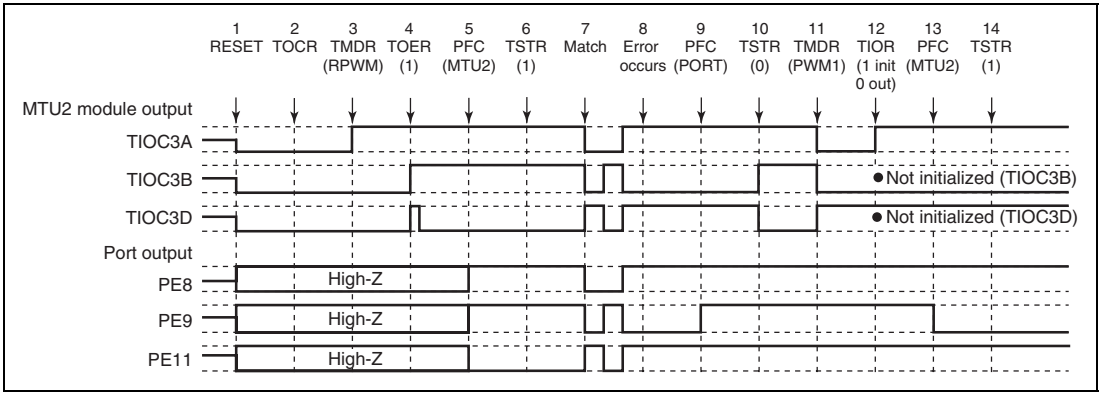


Figure 27.141 Error Occurrence in Reset-Synchronized PWM Mode, Recovery in PWM Mode 1

1 to 10 are the same as in figure 27.140.

11. Set PWM mode 1. (The positive phase output from this module is low, and negative phase output is high.)
12. Initialize the pins with TIOR. (In PWM mode 1, the TIOC *B side is not initialized.)
13. Set the multi-function timer pulse unit 2 output with the general I/O port.
14. Operation is restarted by TSTR.

(28) Operation when Error Occurs during Reset-Synchronized PWM Mode Operation, and Operation is Restarted in Complementary PWM Mode

Figure 27.142 shows an explanatory diagram of the case where an error occurs in reset-synchronized PWM mode and operation is restarted in complementary PWM mode after re-setting.

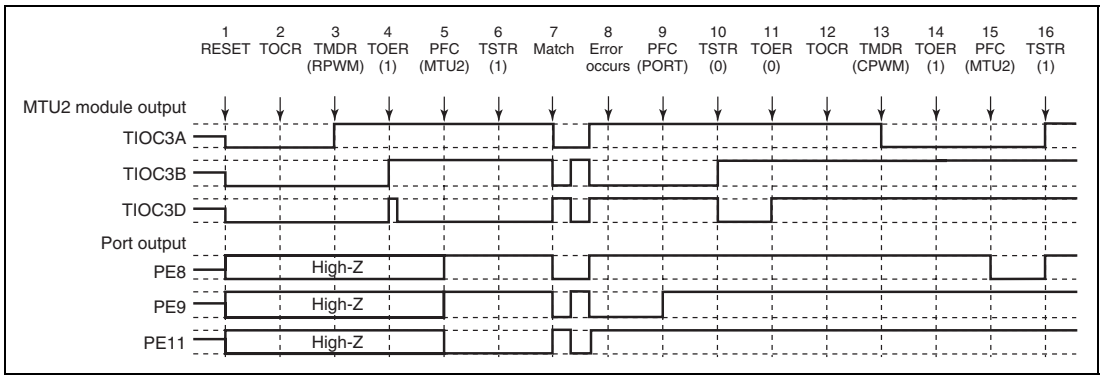


Figure 27.142 Error Occurrence in Reset-Synchronized PWM Mode, Recovery in Complementary PWM Mode

1 to 10 are the same as in figure 27.140.

11. Disable channel 3 and 4 output with TOER.
12. Select the complementary PWM output level and cyclic output enabling/disabling with TOCR.
13. Set complementary PWM. (The cyclic output pin of this module outputs a low-level signal.)
14. Enable channel 3 and 4 output with TOER.
15. Set the multi-function timer pulse unit 2 output with the general I/O port.
16. Operation is restarted by TSTR.

(29) Operation when Error Occurs during Reset-Synchronized PWM Mode Operation, and Operation is Restarted in Reset-Synchronized PWM Mode

Figure 27.143 shows an explanatory diagram of the case where an error occurs in reset-synchronized PWM mode and operation is restarted in reset-synchronized PWM mode after re-setting.

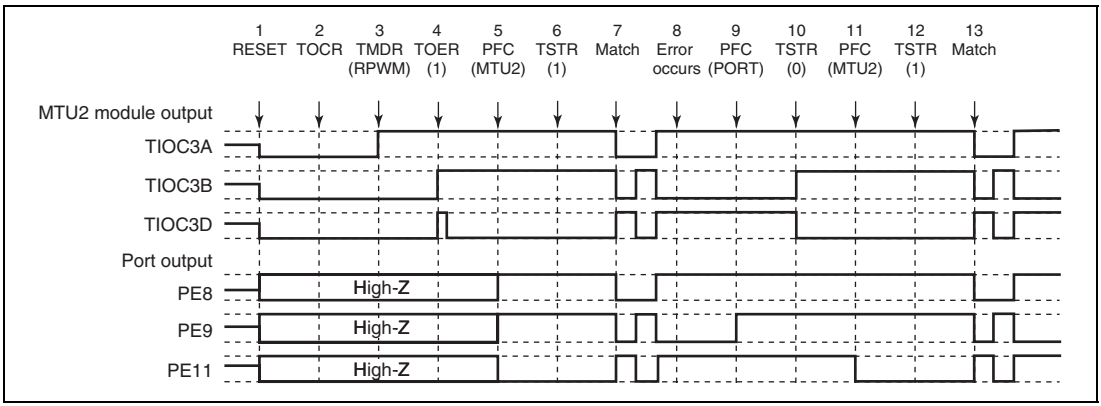


Figure 27.143 Error Occurrence in Reset-Synchronized PWM Mode, Recovery in Reset-Synchronized PWM Mode

1 to 10 are the same as in figure 27.140.

11. Set the multi-function timer pulse unit 2 output with the general I/O port.
12. Operation is restarted by TSTR.
13. The reset-synchronized PWM waveform is output on compare-match occurrence.

Section 28 A/D Converter

This LSI includes a 10-bit successive-approximation A/D converter allowing selection of up to eight analog input channels.

28.1 Features

- Resolution: 10 bits
- Input channels: Eight channels
- Minimum conversion time: 6.0 µs per channel
- Absolute accuracy: ±5 LSB
- Operating modes: Three
  - Single mode: A/D conversion on one channel
  - Multi mode: A/D conversion on one to four channels or on one to eight channels
  - Scan mode: Continuous A/D conversion on one to four channels or on one to eight channels
- Data registers: Eight

  Conversion results are held in a 16-bit data register for each channel
- Sample-and-hold function
- A/D conversion start methods: Three
  - Software
  - Conversion start trigger from the multi-function timer pulse unit 2
  - External trigger signal
- Interrupt source

  An A/D conversion end interrupt (ADI) request can be generated on completion of A/D conversion.
- Module standby mode can be set

Figure 28.1 shows a block diagram of the A/D converter.

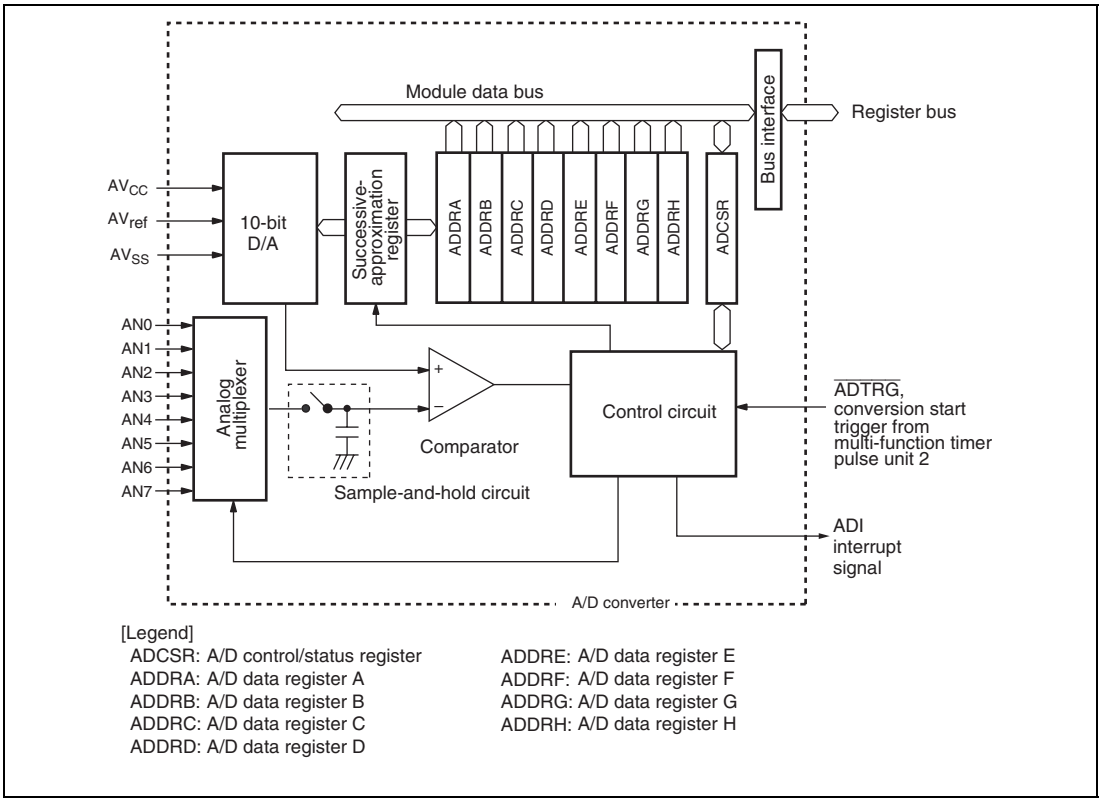


Figure 28.1 Block Diagram of A/D Converter

28.2 Input/Output Pins

Table 28.1 shows the A/D converter pins.

Table 28.1 Pin Configuration

Pin Name	Symbol	I/O	Function
Analog power supply pin	AVcc	Input	Analog power supply pin
Analog ground pin	AVss	Input	Analog ground pin and A/D conversion reference ground
Analog reference voltage pin	AVref	Input	A/D converter reference voltage pin
Analog input pin 0	AN0	Input	Analog input
Analog input pin 1	AN1	Input	
Analog input pin 2	AN2	Input	
Analog input pin 3	AN3	Input	
Analog input pin 4	AN4	Input	
Analog input pin 5	AN5	Input	
Analog input pin 6	AN6	Input	
Analog input pin 7	AN7	Input	
A/D external trigger input pin	$\overline{\text{ADTRG}}$	Input	External trigger input to start A/D conversion

28.3 Register Descriptions

Table 28.2 (1) shows the register configuration.

Table 28.2 (1) Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
A/D data register A	ADDRA	R	H'0000	H'FFFCB800	16
A/D data register B	ADDRB	R	H'0000	H'FFFCB802	16
A/D data register C	ADDRC	R	H'0000	H'FFFCB804	16
A/D data register D	ADDRD	R	H'0000	H'FFFCB806	16
A/D data register E	ADDRE	R	H'0000	H'FFFCB808	16
A/D data register F	ADDRF	R	H'0000	H'FFFCB80A	16
A/D data register G	ADDRG	R	H'0000	H'FFFCB80C	16
A/D data register H	ADDRH	R	H'0000	H'FFFCB80E	16
A/D control/status register	ADCSR	R/W	H'0000	H'FFFCB820	16

Table 28.2 (2) Register State in Each Operating Mode

Abbreviation	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
ADDRA	Initialized	Initialized	Retained	Retained	Initialized	Initialized
ADDRB	Initialized	Initialized	Retained	Retained	Initialized	Initialized
ADDRC	Initialized	Initialized	Retained	Retained	Initialized	Initialized
ADDRD	Initialized	Initialized	Retained	Retained	Initialized	Initialized
ADDRE	Initialized	Initialized	Retained	Retained	Initialized	Initialized
ADDRF	Initialized	Initialized	Retained	Retained	Initialized	Initialized
ADDRG	Initialized	Initialized	Retained	Retained	Initialized	Initialized
ADDRH	Initialized	Initialized	Retained	Retained	Initialized	Initialized
ADCSR	Initialized	Initialized	Retained	Retained	Initialized	Initialized

28.3.1 A/D Data Registers A to H (ADDRA to ADDRH)

The eight A/D data registers, ADDRA to ADDRH, are 16-bit read-only registers that store the results of A/D conversion.

An A/D conversion produces 10-bit data, which is transferred for storage into the ADDR corresponding to the selected channel. The 10 bits of the result are stored in the upper bits (bits 15 to 6) of ADDR. Bits 5 to 0 of ADDR are reserved bits that are always read as 0.

Access to ADDR in 8-bit units is prohibited. ADDR must always be accessed in 16-bit units.

Table 28.3 indicates the pairings of analog input channels and ADDR.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
											-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 6		All 0	R	Bit data (10 bits)
5 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Table 28.3 Analog Input Channels and ADDR

Analog Input Channel	A/D Data Register where Conversion Result is Stored
AN0	ADDRA
AN1	ADDRB
AN2	ADDRC
AN3	ADDRD
AN4	ADDRE
AN5	ADDRF
AN6	ADDRG
AN7	ADDRH

28.3.2 A/D Control/Status Register (ADCSR)

ADCSR is a 16-bit readable/writable register that selects the mode, controls the A/D converter, and enables or disables starting of A/D conversion by external trigger input.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADF	ADIE	ADST	TRGS[3:0]				CKS[2:0]			MDS[2:0]			CH[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/(W)*1	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: *1 Only 0 can be written to clear the flag after 1 is read.

Bit	Bit Name	Initial Value	R/W	Description
15	ADF	0	R/(W)*1	A/D End Flag Status flag indicating the end of A/D conversion. [Clearing conditions] • Cleared by reading ADF while ADF = 1, then writing 0 to ADF • Cleared when the direct memory access controller is activated by ADI interrupt and ADDR is read [Setting conditions] • A/D conversion ends in single mode • A/D conversion ends for the selected channels in multi mode • A/D conversion ends for the selected channels in scan mode
14	ADIE	0	R/W	A/D Interrupt Enable Enables or disables the interrupt (ADI) requested at the end of A/D conversion. Set the ADIE bit while A/D conversion is not being made. 0: A/D end interrupt request (ADI) is disabled 1: A/D end interrupt request (ADI) is enabled

Bit	Bit Name	Initial Value	R/W	Description
13	ADST	0	R/W	A/D Start Starts or stops A/D conversion. This bit remains set to 1 during A/D conversion. 0: A/D conversion is stopped 1: Single mode: A/D conversion starts. This bit is automatically cleared to 0 when A/D conversion ends on the selected channel. Multi mode: A/D conversion starts. This bit is automatically cleared to 0 when A/D conversion is completed cycling through the selected channels. Scan mode: A/D conversion starts. A/D conversion is continuously performed until this bit is cleared to 0 by software, by a power-on reset and so on.
12 to 9	TRGS[3:0]	0000	R/W	Timer Trigger Select These bits enable or disable starting of A/D conversion by a trigger signal. 0000: Start of A/D conversion by external trigger input is disabled 0001: A/D conversion is started by conversion trigger TRGAN from the multi-function timer pulse unit 2 0010: A/D conversion is started by conversion trigger TRG0N from the multi-function timer pulse unit 2 0011: A/D conversion is started by conversion trigger TRG4AN from the multi-function timer pulse unit 2 0100: A/D conversion is started by conversion trigger TRG4BN from the multi-function timer pulse unit 2 1001: A/D conversion is started by $\overline{\text{ADTRG}}$ Other than above: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
8 to 6	CKS[2:0]	000	R/W	Clock Select These bits select the A/D conversion time.*[2] Set the A/D conversion time while A/D conversion is halted (ADST = 0). 000: Conversion time = 412 t_{cyc} (maximum) 001: Conversion time = 480 t_{cyc} (maximum) 010: Conversion time = 548 t_{cyc} (maximum) Other than above: Setting prohibited
5 to 3	MDS[2:0]	000	R/W	Multi-scan Mode These bits select the operating mode for A/D conversion. 0xx: Single mode 100: Multi mode: A/D conversion on 1 to 4 channels 101: Multi mode: A/D conversion on 1 to 8 channels 110: Scan mode: A/D conversion on 1 to 4 channels 111: Scan mode: A/D conversion on 1 to 8 channels

Bit	Bit Name	Initial Value	R/W	Description
2 to 0	CH[2:0]	000	R/W	Channel Select These bits and the MDS bits in ADCSR select the analog input channels.

MDS = 0xx	MDS = 100 or MDS = 110	MDS = 101 or MDS = 111
000: AN0	000: AN0	000: AN0
001: AN1	001: AN0, AN1	001: AN0, AN1
010: AN2	010: AN0 to AN2	010: AN0 to AN2
011: AN3	011: AN0 to AN3	011: AN0 to AN3
100: AN4	100: AN4	100: AN0 to AN4
101: AN5	101: AN4, AN5	101: AN0 to AN5
110: AN6	110: AN4 to AN6	110: AN0 to AN6
111: AN7	111: AN4 to AN7	111: AN0 to AN7

[Legend]

x: Don't care

Notes: 1. Only 0 can be written to clear the flag after 1 is read.
Please note that ADF flag becomes "0" in the following cases, too.
(1) Reading the state of ADF = 1 with CPU.
(2) Clearing ADF flag by having read ADDR with DMAC
(3) Set of ADF flag according to A/D conversion end
(4) Writing 0 in the ADF flag with CPU

2. Set the A/D conversion time to minimum or more values to meet the absolute accuracy of the A/D conversion characteristics.

28.4 Operation

The A/D converter uses the successive-approximation method, and the resolution is 10 bits. It has three operating modes: single mode, multi mode, and scan mode. Switching the operating mode or analog input channels must be done while the ADST bit in ADCSR is 0 to prevent incorrect operation. The ADST bit can be set at the same time as the operating mode or analog input channels are changed.

28.4.1 Single Mode

Single mode should be selected when only A/D conversion on one channel is required.

In single mode, A/D conversion is performed once for the specified one analog input channel, as follows:

1. A/D conversion for the selected channel starts when the ADST bit in ADCSR is set to 1 by software, the multi-function timer pulse unit 2, or external trigger input.
2. When A/D conversion is completed, the A/D conversion result is transferred to the A/D data register corresponding to the channel.
3. After A/D conversion has completed, the ADF bit in ADCSR is set to 1. If the ADIE bit is set to 1 at this time, an ADI interrupt request is generated.
4. The ADST bit that remains 1 during A/D conversion is automatically cleared to 0 when A/D conversion is completed, and the A/D converter becomes idle.

When the operating mode or analog input channel selection must be changed during A/D conversion, to prevent incorrect operation, first clear the ADST bit to 0 to halt A/D conversion. After making the necessary changes, set the ADST bit to 1 to start A/D conversion again. The ADST bit can be set at the same time as the mode or channel selection is switched.

Typical operations when a single channel (AN1) is selected in single mode are described next. Figure 28.2 shows a timing diagram for this example (the bits which are set in this example belong to ADCSR).

1. Single mode is selected, input channel AN1 is selected (CH[2:0] = 001), the A/D interrupt is enabled (ADIE = 1), and A/D conversion is started (ADST = 1).
2. When A/D conversion is completed, the A/D conversion result is transferred into ADDRB. At the same time the ADF flag is set to 1, the ADST bit is cleared to 0, and the A/D converter becomes idle.
3. Since ADF = 1 and ADIE = 1, an ADI interrupt is requested.
4. The A/D interrupt handling routine starts.
5. The routine reads ADF = 1, and then writes 0 to the ADF flag.
6. The routine reads and processes the A/D conversion result (ADDRB).
7. Execution of the A/D interrupts handling routine ends. Then, when the ADST bit is set to 1, A/D conversion starts and steps 2 to 7 are executed.

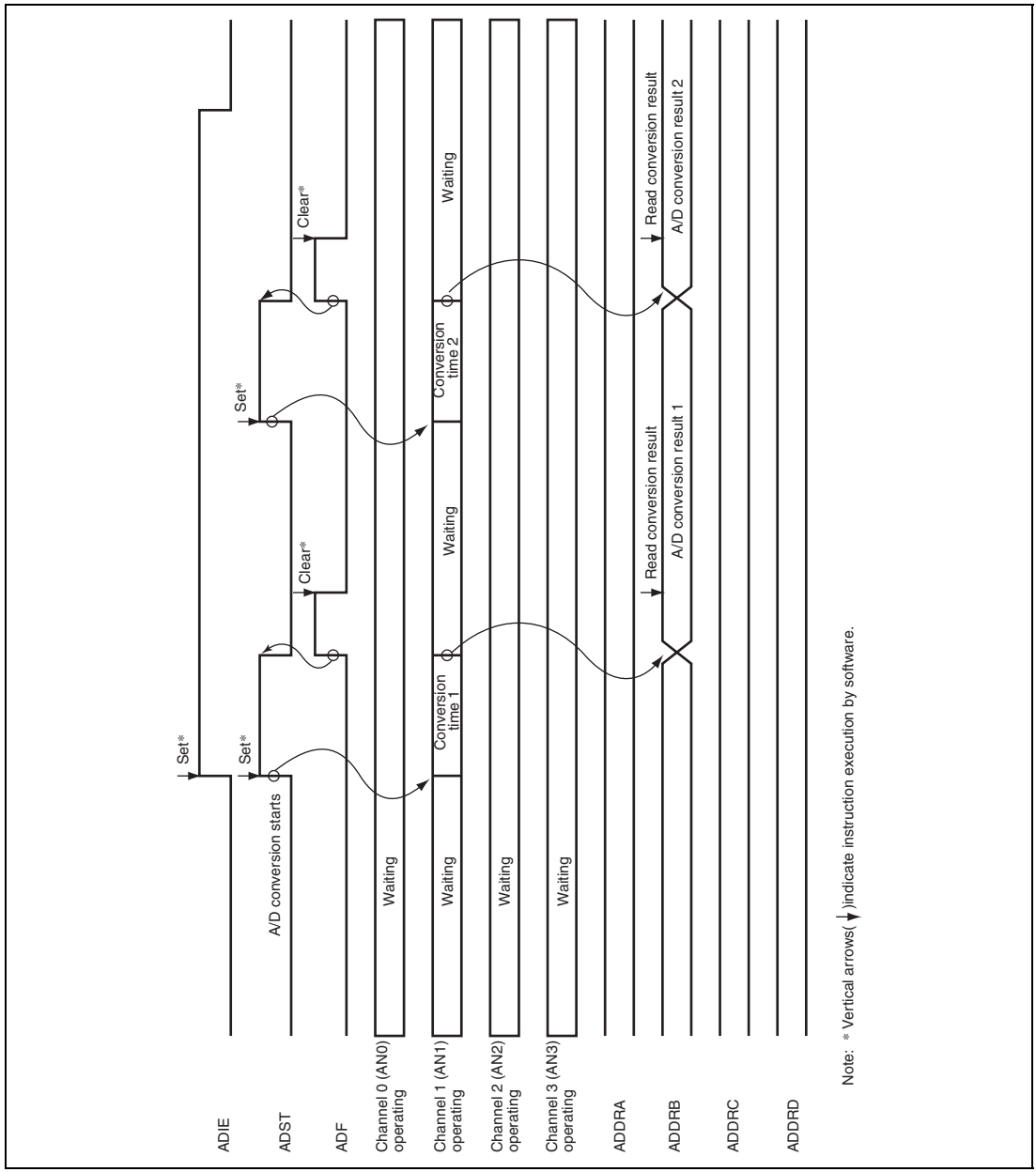


Figure 28.2 Example of A/D Converter Operation (Single Mode, One Channel (AN1) Selected)

28.4.2 Multi Mode

Multi mode should be selected when performing A/D conversion once on one or more channels.

In multi mode, A/D conversion is performed once for a maximum of eight specified analog input channels, as follows:

1. A/D conversion starts from the analog input channel with the lowest number (e.g. AN0, AN1, …, AN3) when the ADST bit in ADCSR is set to 1 by software, the multi-function timer pulse unit 2, or external trigger input.
2. When A/D conversion is completed on each channel, the A/D conversion result is sequentially transferred to the A/D data register corresponding to that channel.
3. After A/D conversion on all selected channels has completed, the ADF bit in ADCSR is set to 1. If the ADIE bit is set to 1 at this time, an ADI interrupt request is generated.
4. The ADST bit that remains 1 during A/D conversion is automatically cleared to 0 when A/D conversion is completed, and the A/D converter becomes idle. If the ADST bit is cleared to 0 during A/D conversion, A/D conversion is halted and the A/D converter becomes idle. The ADF bit is cleared by reading ADF while ADF = 1, then writing 0 to the ADF bit.

A/D conversion is to be performed once on all the specified channels. The conversion results are transferred for storage into the A/D data registers corresponding to the channels.

When the operating mode or analog input channel selection must be changed during A/D conversion, to prevent incorrect operation, first clear the ADST bit to 0 to halt A/D conversion. After making the necessary changes, set the ADST bit to 1. A/D conversion will start again from the first channel in the group. The ADST bit can be set at the same time as the mode or channel selection is changed.

Typical operations when three channels (AN0 to AN2) are selected in multi mode are described next. Figure 28.3 shows a timing diagram for this example.

1. Multi mode is selected (MDS2 = 1, MDS1 = 0), analog input channels AN0 to AN2 are selected (CH[2:0] = 010), and A/D conversion is started (ADST = 1).
2. A/D conversion of the first channel (AN0) starts. When A/D conversion is completed, the A/D conversion result is transferred into ADDRA.
3. Next, the second channel (AN1) is selected automatically and A/D conversion starts.
4. Conversion proceeds in the same way through the third channel (AN2).
5. When conversion of all selected channels (AN0 to AN2) is completed, the ADF flag is set to 1 and the ADST bit cleared to 0.
6. If the ADIE bit is set to 1 at this time, an ADI interrupt is requested.

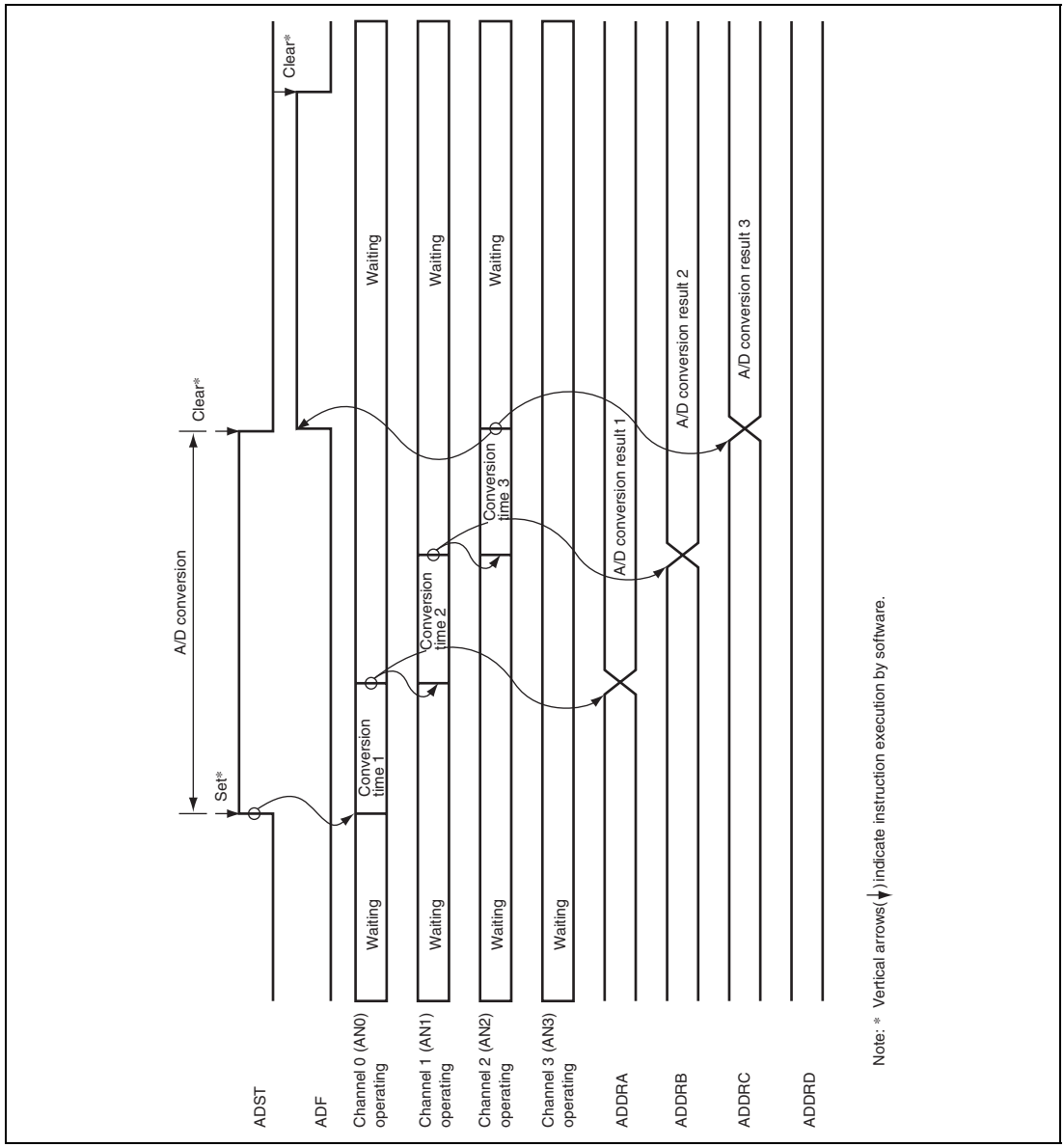


Figure 28.3 Example of A/D Converter Operation (Multi Mode, Three Channels (AN0 to AN2) Selected)

28.4.3 Scan Mode

Scan mode is useful for monitoring analog inputs in a group of one or more channels at all times. In scan mode, A/D conversion is performed sequentially for a maximum of eight specified analog input channels, as follows:

1. A/D conversion starts from the analog input channel with the lowest number (e.g. AN0, AN1, …, AN3) when the ADST bit in ADCSR is set to 1 by software, the multi-function timer pulse unit 2, or external trigger input.
2. When A/D conversion is completed on each channel, the A/D conversion result is sequentially transferred to the A/D data register corresponding to that channel.
3. After A/D conversion on all selected channels has completed, the ADF bit in ADCSR is set to 1. If the ADIE bit is set to 1 at this time, an ADI interrupt request is generated. The A/D converter starts A/D conversion again from the channel with the lowest number.
4. The ADST bit is not cleared automatically, so steps 2. and 3. are repeated as long as the ADST bit remains set to 1. When the ADST bit is cleared to 0, A/D conversion halts and the A/D converter becomes idle.

   The ADF bit is cleared by reading ADF while ADF = 1, then writing 0 to the ADF bit.

When the operating mode or analog input channel selection must be changed during A/D conversion, to prevent incorrect operation, first clear the ADST bit to 0 to halt A/D conversion. After making the necessary changes, set the ADST bit to 1. A/D conversion will start again from the first channel in the group. The ADST bit can be set at the same time as the mode or channel selection is changed.

Typical operations when three channels (AN0 to AN2) are selected in scan mode are described as follows. Figure 28.4 shows a timing diagram for this example.

1. Scan mode is selected (MDS2 = 1, MDS1 = 1), analog input channels AN0 to AN2 are selected (CH[2:0] = 010), and A/D conversion is started (ADST = 1).
2. A/D conversion of the first channel (AN0) starts. When A/D conversion is completed, the A/D conversion result is transferred into ADDRA.
3. Next, the second channel (AN1) is selected automatically and A/D conversion starts.
4. Conversion proceeds in the same way through the third channel (AN2).
5. When conversion of all the selected channels (AN0 to AN2) is completed, the ADF flag is set to 1 and conversion of the first channel (AN0) starts again. If the ADIE bit is set to 1 at this time, an ADI interrupt is requested.

6. The ADST bit is not cleared automatically, so steps 2. to 4. are repeated as long as the ADST bit remains set to 1. When steps 2. to 4. are repeated, the ADF flag is kept to 1. When the ADST bit is cleared to 0, A/D conversion stops. The ADF bit is cleared by reading ADF while ADF = 1, then writing 0 to the ADF bit.

If both the ADF flag and ADIE bit are set to 1 while steps 2. to 4. are repeated, an ADI interrupt is requested at all times. To generate an interrupt on completing conversion of the third channel, clear the ADF bit to 0 after an interrupt is requested.

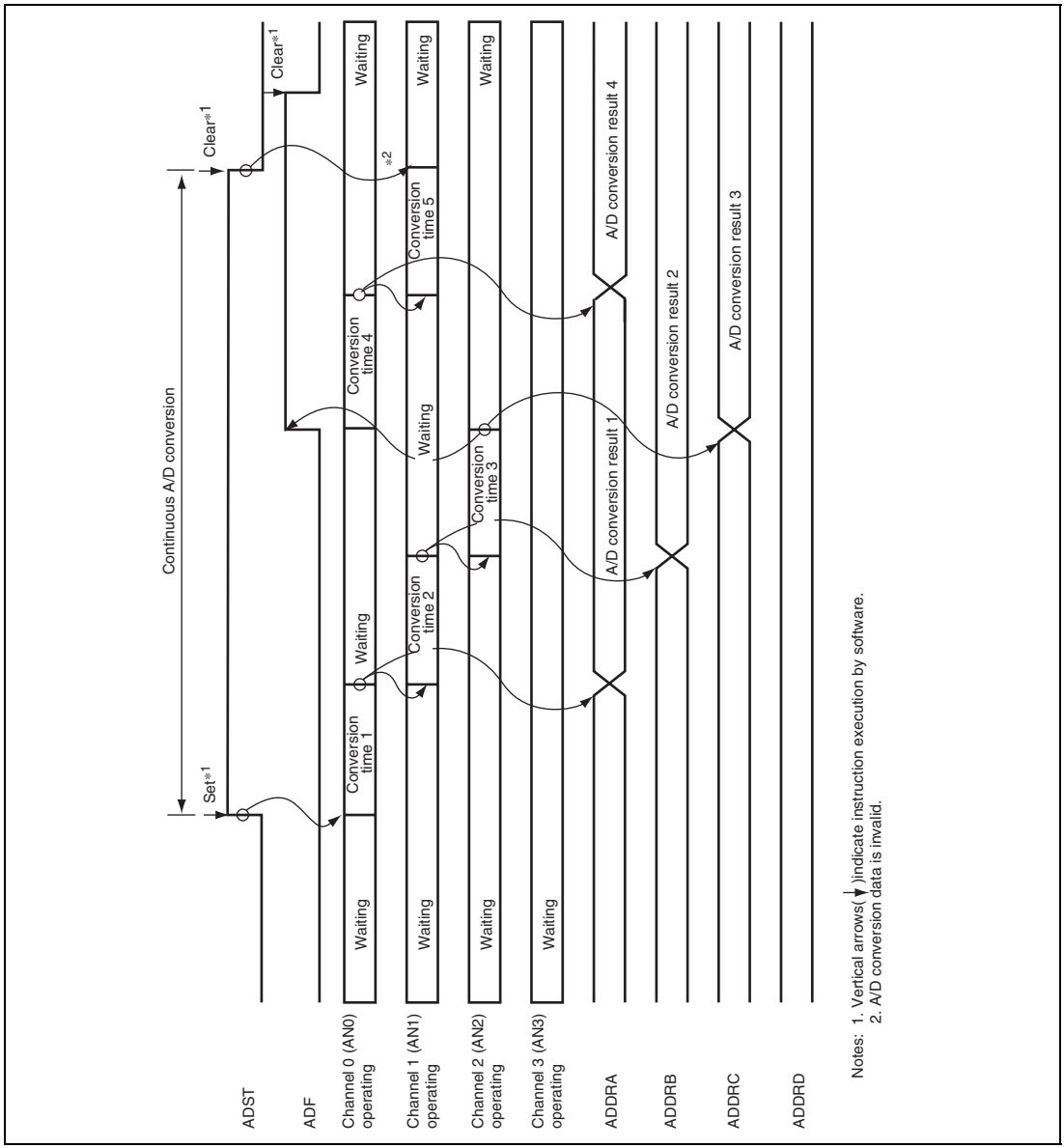


Figure 28.4 Example of A/D Converter Operation (Scan Mode, Three Channels (AN0 to AN2) Selected)

28.4.4 A/D Converter Activation by External Trigger or Multi-Function Timer Pulse Unit 2

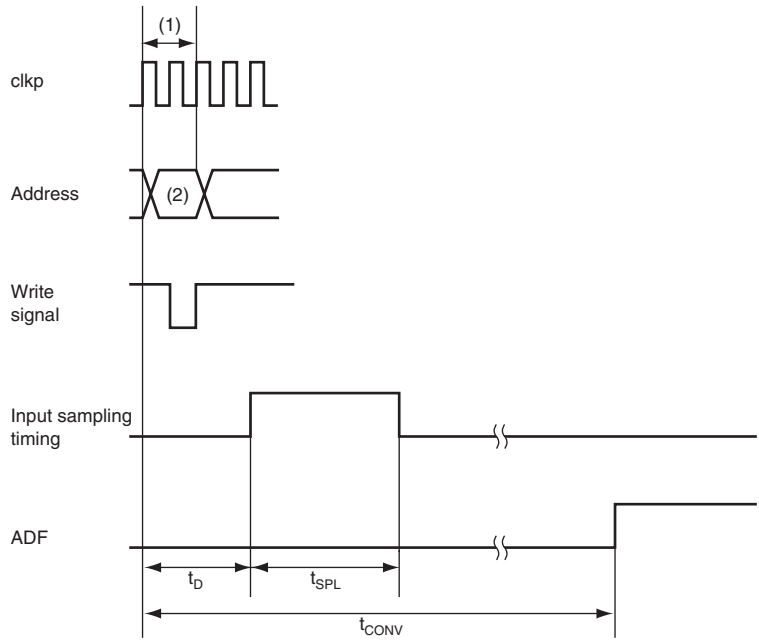
The A/D converter can be independently activated by an external trigger or an A/D conversion request from the multi-function timer pulse unit 2. To activate the A/D converter by an external trigger or the multi-function timer pulse unit 2, set the A/D trigger enable bits (TRGS[3:0]). When an external trigger or an A/D conversion request from the multi-function timer pulse unit 2 is generated with this bit setting, the ADST bit is set to 1 to start A/D conversion. The channel combination is determined by bits CH2 to CH0 in ADCSR. The timing from setting of the ADST bit until the start of A/D conversion is the same as when 1 is written to the ADST bit by software.

28.4.5 Input Sampling and A/D Conversion Time

The A/D converter has a built-in sample-and-hold circuit. The A/D converter samples the analog input at the A/D conversion start delay time (t_D) after the ADST bit in ADCSR is set to 1, then starts conversion. Figure 28.5 shows the A/D conversion timing. Table 28.4 indicates the A/D conversion time.

As indicated in figure 28.5, the A/D conversion time (t_{CONV}) includes t_D and the input sampling time(t_{SPL}). The length of t_D varies depending on the timing of the write access to ADCSR. The total conversion time therefore varies within the ranges indicated in table 28.4.

In multi mode and scan mode, the values given in table 28.4 apply to the first conversion. In the second and subsequent conversions, time is the values given in table 28.5.



[Legend]
(1): ADCSR write cycle
(2): ADCSR address
t_D: A/D conversion start delay time
t_{SPL}: Input sampling time
t_{CONV}: A/D conversion time

Figure 28.5 A/D Conversion Timing

Table 28.4 A/D Conversion Time (Single Mode)

		CKS2 = 0								
		CKS1 = 0						CKS1 = 1		
		CKS0 = 0			CKS0 = 1			CKS0 = 0		
Item	**Symbol**	**Min.**	**Typ.**	**Max.**	**Min.**	**Typ.**	**Max.**	**Min.**	**Typ.**	**Max.**
A/D conversion start delay time	t_D	15	—	26	17	—	30	19	—	34
Input sampling time	t_{SPL}	—	97	—	—	113	—	—	129	—
A/D conversion time	t_{CONV}	401	—	412	467	—	480	533	—	548

Note: Values in the table are represented in terms of t_{cyc}.

Table 28.5 A/D Conversion Time (Multi Mode and Scan Mode)

CKS2	CKS1	CKS0	Conversion Time (t_{cyc})
0	0	0	384 (constant)
		1	448 (constant)
	1	0	512 (constant)

Note: Values in the table are represented in terms of t_{cyc}.

28.4.6 External Trigger Input Timing

A/D conversion can also be externally triggered. When the TRGS[3:0] bits in ADCSR are set to B'1001, an external trigger is input to the $\overline{\text{ADTRG}}$ pin. The ADST bit in ADCSR is set to 1 at the falling edge of the $\overline{\text{ADTRG}}$ pin, thus starting A/D conversion. Other operations, regardless of the operating mode, are the same as when the ADST bit has been set to 1 by software. Figure 28.6 shows the timing.

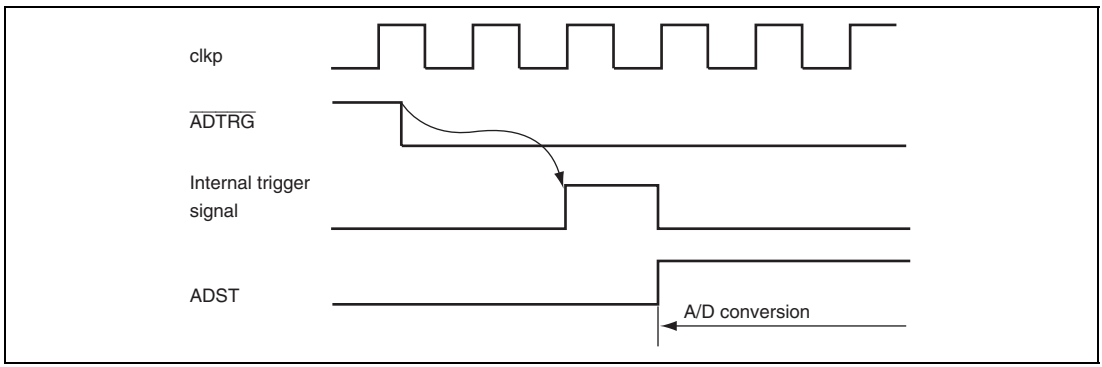


Figure 28.6 External Trigger Input Timing

28.5 Interrupt Sources and DMA Transfer Request

The A/D converter generates an A/D conversion end interrupt (ADI) at the end of A/D conversion. An ADI interrupt request is generated if the ADIE bit is set to 1 when the ADF bit in ADCSR is set to 1 on completion of A/D conversion. Note that the direct memory access controller can be activated by an ADI interrupt depending on the setting of the direct memory access controller. In this case, an interrupt is not issued to the CPU. If the setting to activate the direct memory access controller has not been made, an interrupt request is sent to the CPU. Having the converted data read by the direct memory access controller in response to an ADI interrupt enables continuous conversion to be achieved without imposing a load on software.

In single mode, set the direct memory access controller so that DMA transfer initiated by an ADI interrupt is performed only once. In the case of A/D conversion on multiple channels in scan mode or multi mode, setting the DMA transfer count to one causes DMA transfer to finish after transferring only one channel of data. To make the direct memory access controller transfer all conversion data, set the ADDR where A/D conversion data is stored as the transfer source address, and the number of converted channels as the transfer count.

When the direct memory access controller is activated by ADI, the ADF bit in ADCSR is automatically cleared to 0 when data is transferred by the direct memory access controller.

Table 28.6 Relationship between Interrupt Sources and DMA Transfer Request

Name	Interrupt Source	Interrupt Flag	Direct Memory Access Controller Activation
ADI	A/D conversion end	ADF in ADCSR	Possible

28.6 Definitions of A/D Conversion Accuracy

The A/D converter compares an analog value input from an analog input channel with its analog reference value and converts it to 10-bit digital data. The absolute accuracy of this A/D conversion is the deviation between the input analog value and the output digital value. It includes the following errors:

- Offset error
- Full-scale error
- Quantization error
- Nonlinearity error

These four error quantities are explained below with reference to figure 28.7. In the figure, the 10-bit A/D converter is illustrated as the 3-bit A/D converter for explanation. Offset error is the deviation between actual and ideal A/D conversion characteristics when the digital output value changes from the minimum (zero voltage) B'0000000000 (000 in the figure) to B'000000001 (001 in the figure)(figure 28.7, item (1)). Full-scale error is the deviation between actual and ideal A/D conversion characteristics when the digital output value changes from B'1111111110 (110 in the figure) to the maximum B'1111111111 (111 in the figure)(figure 28.7, item (2)). Quantization error is the intrinsic error of the A/D converter and is expressed as 1/2 LSB (figure 28.7, item (3)). Nonlinearity error is the deviation between actual and ideal A/D conversion characteristics between zero voltage and full-scale voltage (figure 28.7, item (4)). Note that it does not include offset, full-scale, or quantization error.

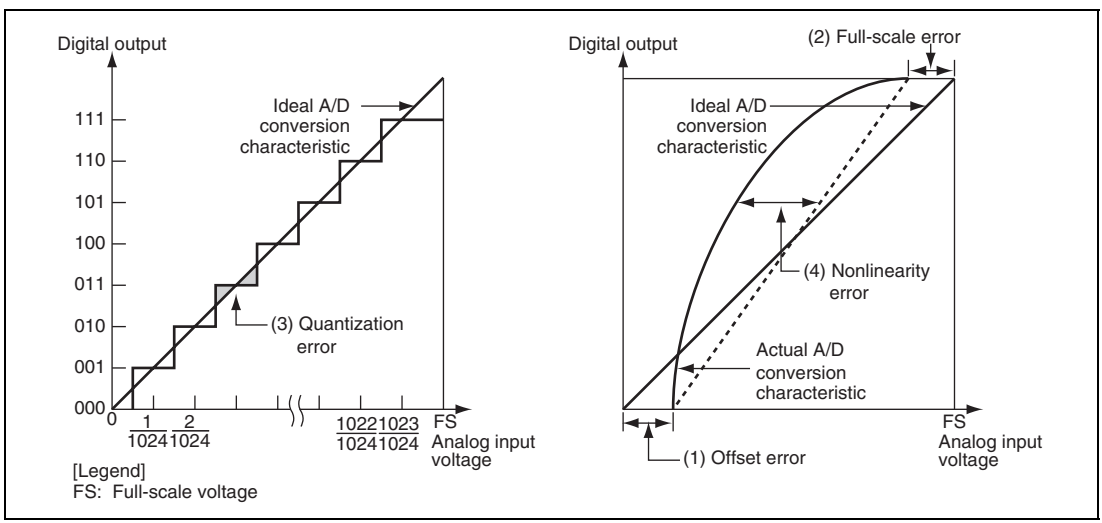


Figure 28.7 Definitions of A/D Conversion Accuracy

28.7 Usage Notes

When using the A/D converter, note the following points.

28.7.1 Module Standby Mode Setting

Operation of the A/D converter can be disabled or enabled using the standby control register. The initial setting is for operation of the A/D converter to be halted. Register access is enabled by clearing module standby mode. For details, see section 9.2, Overview of Power-Down Modes in section 9, Operating Modes and Power-Down Modes.

28.7.2 Setting Analog Input Voltage

Permanent damage to the LSI may result if the following voltage ranges are exceeded.

1. Analog input range

   During A/D conversion, voltages on the analog input pins ANn should not go beyond the following range: AVss ≤ ANn ≤ AVcc (n = 0 to 7).
2. AVcc and AVss input voltages

   Input voltages AVcc and AVss should be PVcc − 0.3 V ≤ AVcc ≤ PVcc and AVss = Vss. Do not leave the AVcc and AVss pins open when the A/D converter is not in use and in software standby mode. When not in use, connect AVcc to the power supply (PVcc) and AVss to the ground (Vss).
3. Setting range of AVref input voltage

   Set the reference voltage range of the AVref pin as 3.0 V ≤ AVref ≤ AVcc.

28.7.3 Notes on Board Design

In board design, digital circuitry and analog circuitry should be as mutually isolated as possible, and layout in which digital circuit signal lines and analog circuit signal lines cross or are in close proximity should be avoided as far as possible. Failure to do so may result in incorrect operation of the analog circuitry due to inductance, adversely affecting A/D conversion values.

Digital circuitry must be isolated from the analog input signals (AN0 to AN7), analog reference voltage (AVref), and analog power supply (AVcc) by the analog ground (AVss). Also, the analog ground (AVss) should be connected at one point to a stable digital ground (Vss) on the board.

28.7.4 Processing of Analog Input Pins

To prevent damage from voltage surges at the analog input pins (AN0 to AN7), connect an input protection circuit like the one shown in figure 28.8. The circuit shown also includes a CR filter to suppress noise. This circuit is shown as an example; the circuit constants should be selected according to actual application conditions.

Figure 28.9 shows an equivalent circuit diagram of the analog input ports and table 28.7 lists the analog input pin specifications.

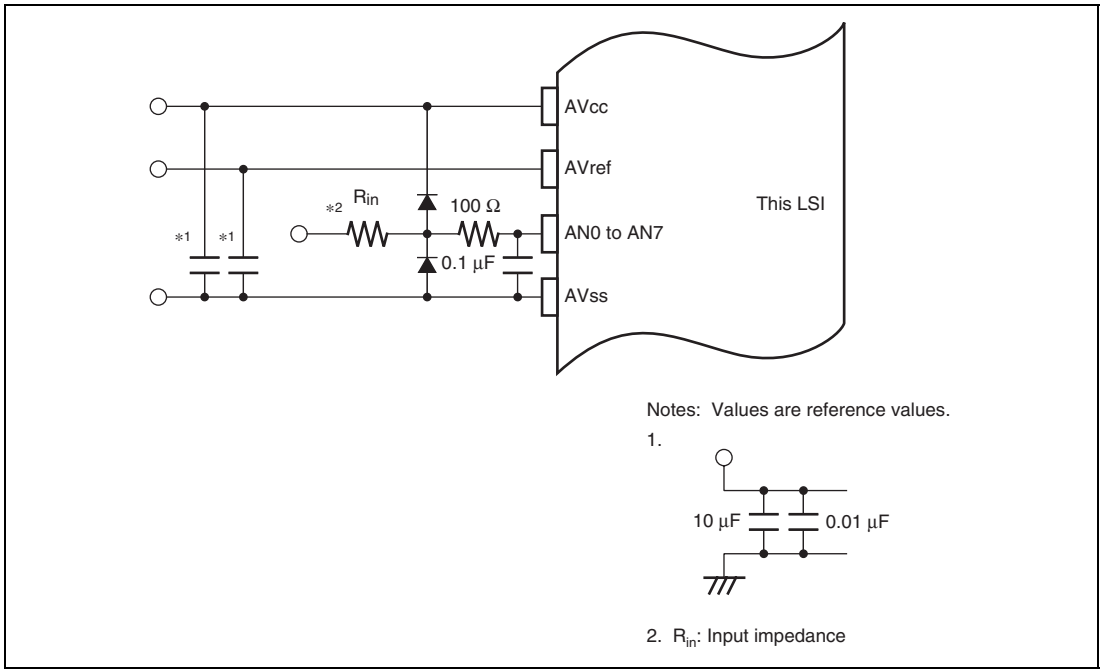
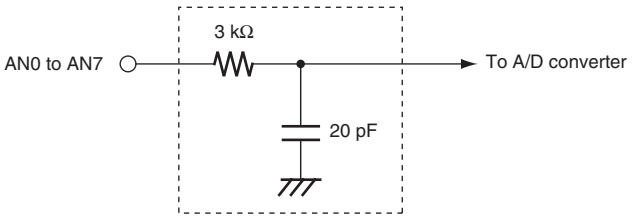


Figure 28.8 Example of Analog Input Protection Circuit



Note: Values are reference values.

Figure 28.9 Analog Input Pin Equivalent Circuit

Table 28.7 Analog Input Pin Ratings

Item	Min.	Max.	Unit
Analog input capacitance	—	20	pF
Allowable signal-source impedance	—	5	kΩ

28.7.5 Permissible Signal Source Impedance

This LSI's analog input is designed such that conversion precision is guaranteed for an input signal for which the signal source impedance is 5 kΩ or less. This specification is provided to enable the A/D converter's sample-and-hold circuit input capacitance to be charged within the sampling time; if the sensor output impedance exceeds 5 kΩ, charging may be insufficient and it may not be possible to guarantee A/D conversion precision. However, for A/D conversion in single mode with a large capacitance provided externally for A/D conversion in single mode, the input load will essentially comprise only the internal input resistance of 3 kΩ, and the signal source impedance is ignored. However, as a low-pass filter effect is obtained in this case, it may not be possible to follow an analog signal with a large differential coefficient (e.g., 5 mV/µs or greater) (see figure 28.10). When converting a high-speed analog signal, a low-impedance buffer should be inserted.

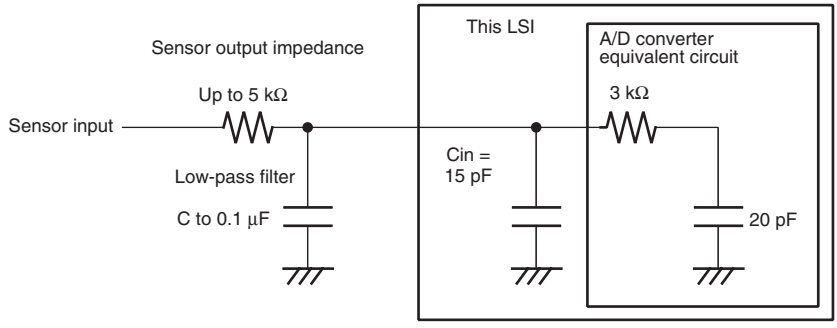


Figure 28.10 Example of Analog Input Circuit

28.7.6 Influences on Absolute Precision

Adding capacitance results in coupling with GND, and therefore noise in GND may adversely affect absolute precision. Be sure to connect AVss, etc. to an electrically stable GND.

Care is also required to insure that filter circuits do not communicate with digital signals on the mounting board (i.e., acting as antennas).

28.7.7 A/D Conversion in Deep Standby Mode

Before entering deep standby mode, disable A/D conversion by clearing the ADST bit to 0. If the LSI enters deep standby mode with A/D conversion enabled, the states on the A/D converter pins are not guaranteed.

28.7.8 Note on Usage in Scan Mode and Multi Mode

Starting conversion immediately after having stopped scan mode or multi mode operation may lead to erroneous results of conversion.

To perform continuous conversion in such cases, set ADST to 0, wait for at least the A/D conversion time for a single channel to elapse, and then start conversion (ADST = 1). (The A/D conversion time for a single channel will vary according to the settings of the ADC registers.)

28.7.9 Notes on Using Single Mode and Multi Mode

If further A/D conversion is attempted in the same mode or a different mode immediately after completion of conversion in single or multi mode, operation may be incorrect.

When successive A/D conversion is to proceed, dummy-read the ADCSR register twice and then set the ADST bit to 1.

Section 29 SD Host Interface (SDHI)

The contents of this section are available upon non-disclosure agreement.
For details, contact your local sales representatives.

Section 30 Multi Media Card Interface (MMC)

30.1 Features

(1) MMC Interface

- Supports 1/4/8-bit MMC bus width
- Supports boot operation
- MMC clock frequency ≤ 1/2 × clks1 frequency.
- MMC clock frequency settings are adjustable in boot mode.
- Error checking functions (CRC7, CRC16)
- Two types of interrupt requests: normal operation interrupts and error/timeout interrupts
- DMA transfer requests: buffer write and buffer read
- Supports MMC mode (does not support SPI mode)

(2) CE-ATA Interface

- Conforms to the CE-ATA Digital Protocol
- Supports 1/4/8-bit bus width
- MMC clock frequency ≤ 1/2 × clks1 frequency.
- Error checking functions (CRC7, CRC16)
- Two types of interrupt requests: normal operation interrupts and error/timeout interrupts
- DMA transfer requests: buffer write and buffer read
- Supports Command Completion Signal (CCS) and Command Completion Signal Disable (CCSD)

Figure 30.1 shows a block diagram of this module.

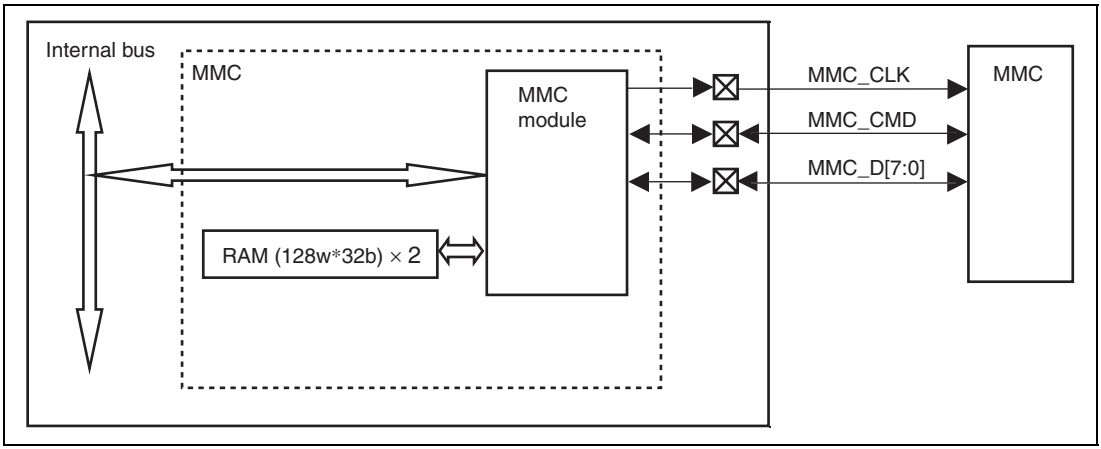


Figure 30.1 Block Diagram of MMC

30.2 Input/Output Pins

Table 30.1 shows the pin configuration of the MMC.

Table 30.1 Pin Configuration

Name	Pin Name	I/O	Function
MMCCLK	D8 (MMC_CLK)	Output	MMC clock
MMCCMD	D9 (MMC_CMD)	Input/output	MMC command/response
MMCDAT[7:0]	D0 (MMC_D0), D1 (MMC_D1), D2 (MMC_D2), D3 (MMC_D3), D4 (MMC_D4), D5 (MMC_D5), D6 (MMC_D6), D7 (MMC_D7)	Input/output	MMC data [7 to 0]

30.3 Register Configuration

Table 30.2 (1) shows the register configuration of the MMC.

The base addresses are as follows.

P4 address: H'FFE4_F000

Area 7 address: H'1FE4_F000

Table 30.2 (1) Register Configuration

Register Name	Abbreviation	R/W	Offset Address from Base Address	Access Size
Command setting register	CE_CMD_SET	R/W	H'00	16, 32
Argument register	CE_ARG	R/W	H'08	16, 32
Argument register for automatically-issued CMD12	CE_ARG_CMD12	R/W	H'0C	16, 32
Command control register	CE_CMD_CTRL	R/W	H'10	16, 32
Transfer block setting register	CE_BLOCK_SET	R/W	H'14	16, 32
Clock control register	CE_CLK_CTRL	R/W	H'18	16, 32
Buffer access configuration register	CE_BUF_ACC	R/W	H'1C	16, 32
Response register 3	CE_RESP3	R/W	H'20	16, 32
Response register 2	CE_RESP2	R/W	H'24	16, 32
Response register 1	CE_RESP1	R/W	H'28	16, 32
Response register 0	CE_RESP0	R/W	H'2C	16, 32
Response register for automatically-issued CMD12	CE_RESP_CMD12	R/W	H'30	16, 32
Data register	CE_DATA	R/W	H'34	16*, 32
Boot operation setting register	CE_BOOT	R/W	H'3C	16, 32
Interrupt flag register	CE_INT	R/W	H'40	16, 32
Interrupt enable register	CE_INT_EN	R/W	H'44	16, 32
Status register 1	CE_HOST_STS1	R/W	H'48	16, 32
Status register 2	CE_HOST_STS2	R/W	H'4C	16, 32
Version register	CE_VERSION	R/W	H'7C	16, 32

Notes: Addresses other than the above must not be write-accessed to. If written to, operation is not guaranteed. If read, an undefined value is read.

* For 16-bit access, H'34 is the only address for access.

Table 30.2 (2) Register State in Each Operating Mode

Abbreviation	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
CE_CMD_SET	H'0000 0000	H'0000 0000	Retained	Retained	Retained	Initialized
CE_ARG	H'0000 0000	H'0000 0000	Retained	Retained	Retained	Initialized
CE_ARG_CMD12	H'0000 0000	H'0000 0000	Retained	Retained	Retained	Initialized
CE_CMD_CTRL	H'0000 0000	H'0000 0000	Retained	Retained	Retained	Initialized
CE_BLOCK_SET	H'0000 0200	H'0000 0200	Retained	Retained	Retained	Initialized
CE_CLK_CTRL	H'0000 0000	H'0000 0000	Retained	Retained	Retained	Initialized
CE_BUF_ACC	H'0000 0000	H'0000 0000	Retained	Retained	Retained	Initialized
CE_RESP3	H'0000 0000	H'0000 0000	Retained	Retained	Retained	Initialized
CE_RESP2	H'0000 0000	H'0000 0000	Retained	Retained	Retained	Initialized
CE_RESP1	H'0000 0000	H'0000 0000	Retained	Retained	Retained	Initialized
CE_RESP0	H'0000 0000	H'0000 0000	Retained	Retained	Retained	Initialized
CE_RESP_CMD12	H'0000 0000	H'0000 0000	Retained	Retained	Retained	Initialized
CE_DATA	H'0000 0000	H'0000 0000	Retained	Retained	Retained	Initialized
CE_BOOT	H'0000 0000	H'0000 0000	Retained	Retained	Retained	Initialized
CE_INT	H'0000 0000	H'0000 0000	Retained	Retained	Retained	Initialized
CE_INT_EN	H'0000 0000	H'0000 0000	Retained	Retained	Retained	Initialized
CE_HOST_STS1	Undefined	Undefined	Retained	Retained	Retained	Initialized
CE_HOST_STS2	H'0000 0000	H'0000 0000	Retained	Retained	Retained	Initialized
CE_VERSION	H'0000 0002	H'0000 0002	Retained	Retained	Retained	Initialized

Note: Initialized: The value is the one written in "Power-on Reset".

30.3.1 Command Setting Register (CE_CMD_SET)

CE_CMD_SET sets a command sequence. The command sequence starts when a command index has been set along with other required settings. If this register is accessed in 16-bit units, the command sequence starts when the settings have been made in bits 31 to 16. Note that writing to CE_CMD_SET is disabled while a command sequence is proceeding (i.e., the value of CMDSEQ in CE_HOST_STS1 is 1). For the setting values of CE_CMD_SET, see section 30.7.17, Setting Values of CE_CMD_SET.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	BOOT	CMD[5:0]						RTYP[1:0]		RBSY	CCSEN	WDAT	DWEN	CMLTE	CMD12EN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RIDXC[1:0]		RCRC7C[1:0]		—	CRC16C	BOOTACK	CRCSTE	TBIT	OPDM	CCSH	—	—	—	DATW[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30	BOOT	0	R/W	Boot Operation 0: Command sequence other than for boot operations 1: Command sequence for boot operations
29 to 24	CMD[5:0]	H'00	R/W	Command Index Note: Setting a command index in these bits initiates the command sequence.
23, 22	RTYP[1:0]	00	R/W	Response Type 00: No response 01: 6-byte response (R1, R1b, R3, R4, R5) 10: 17-byte response (R2) 11: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
21	RBSY	0	R/W	Response Busy Select Selects whether "busy" is involved in response reception. 0: No response busy 1: Response busy involved (R1b)
20	CCSEN	0	R/W	CCS Acceptance 0: Acceptance of CCS disabled 1: Acceptance of CCS enabled
19	WDAT	0	R/W	Presence/Absence of Data 0: No data 1: With data
18	DWEN	0	R/W	Read/Write (valid when “with data” is selected) 0: Read from the card 1: Write to the card
17	CMLTE	0	R/W	Single/Multi Block Transfer Select (valid when “with data” is selected) 0: Single-block transfer 1: Multi-block transfer
16	CMD12EN	0	R/W	Automatic CMD12 Issuance (valid when multi-block transfer is selected) 0: Disables automatic CMD12 issuance 1: Enables automatic CMD12 issuance For details of automatic CMD12 issuance, see section 30.6.2, Automatic CMD12 Issuance. Note: Set the transfer block size to 512 bytes
15, 14	RIDXC[1:0]	00	R/W	Response Index Check 00: Checks the response index 01: Checks the check bits 10: No checking 11: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
13, 12	RCRC7C [1:0]	00	R/W	Response CRC7 Check 00: Checks CRC7 (set the response type to 01) 01: Checks the check bits (set the response type to 01) 10: Checks internal CRC7 (R2 only) (set the response type to 10) 11: No checking
11	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
10	CRC16C	0	R/W	CRC16 Check in Reception 0: Checks CRC16 1: Does not check CRC16 (use when CMD14)
9	BOOTACK	0	R/W	Receive Boot Acknowledge (valid in boot mode) 0: Boot acknowledge is not received. 1: Boot acknowledge is received.
8	CRCSTE	0	R/W	CRC Status Reception (valid when “with data” and “write” are selected) 0: Receives CRC status 1: Does not receive CRC status (use when CMD19)
7	TBIT	0	R/W	Transmission Bit Setting 0: Sets the transmission bit to high 1: Sets the transmission bit to low
6	OPDM	0	R/W	Open-Drain Output Mode 0: Normal output 1: Open-drain output Note: This setting is only applied to the MMCCMD line.
5	CCSH	0	R/W	Output High after CCS Reception 0: Outputs a high level at the third cycle after CCS is received 1: Does not output a high level after CCS is received

Bit	Bit Name	Initial Value	R/W	Description
4 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	DATW[1:0]	00	R/W	Data Bus Width Setting (valid when “with data” is selected) 00: 1 bit 01: 4 bits 10: 8 bits 11: Setting prohibited

30.3.2 Argument Register (CE_ARG)

CE_ARG sets the argument for the command to be transmitted. This register must be set before CMD[5:0] in CE_CMD_SET is set.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ARG[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ARG[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ARG[31:0]	H'0000 0000	R/W	Set bits 31 to 0 of the argument. Note: Set the argument of automatically-issued CMD12 by CE_ARG_CMD12.

30.3.3 Argument Register for Automatically-Issued CMD12 (CE_ARG_CMD12)

CE_ARG_CMD12 is used to set the argument for the automatically-issued CMD12 in multi-block transfer. This register must be set before CMD[5:0] in CE_CMD_SET is set. For automatic issuance of CMD12, see section 30.6.2, Automatic CMD12 Issuance.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	C12ARG[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	C12ARG[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	C12ARG[31:0]	H'0000 0000	R/W	Set bits 31 to 0 of the argument.

30.3.4 Command Control Register (CE_CMD_CTRL)

CE_CMD_CTRL is used to terminate a command sequence forcibly. It also has the function to issue CCSD, which is effective when a CE-ATA device is connected. Do not issue CCSD when an MMC is connected.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CCSD	BREAK
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	CCSD	0	R/W	CCSD Issuance Writing 1 to this bit while it is 0 issues a CCSD. After making sure that the CCSDE bit in CE_INT has become 1, write 0 to this bit to reset. Note: Before issuing a CCSD, make sure that the value of the CMDSEQ bit in CE_HOST_STS1 is 0.
0	BREAK	0	R/W	Forcible Termination of Command Sequence Writing 1 to this bit while it is 0 and then writing 0 to it discontinues the current command sequence. Note: After this bit is set as described above, check if the value of the CMDSEQ bit in CE_HOST_STS1 has become 0, which indicates that the next processing can be performed. Also refer to section 30.8.2, Forcible Termination.

30.3.5 Transfer Block Setting Register (CE_BLOCK_SET)

CE_BLOCK_SET specifies the size of the block and the number of blocks for the data to be transferred. This register must be set before CMD[5:0] in CE_CMD_SET is set.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BLKCNT[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BLKSIZ[15:0]															
Initial value:	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	BLKCNT [15:0]	H'0000	R/W	Number of Blocks for Transfer Note: This setting is valid for multi-block transfer.
15 to 0	BLKSIZ [15:0]	H'0200	R/W	Transfer Block Size Note: Transfer block size should be set as follows. • Single-block transfer: 1 to 512 bytes • Multi-block transfer: 512 bytes

30.3.6 Clock Control Register (CE_CLK_CTRL)

CE_CLK_CTRL controls the MMC clock and sets timeout values. Do not change the setting of this register while a command sequence is in progress.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	CLKEN	—	—	—	—	CLKDIV[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	SRSPTO[1:0]		SRBSYTO[3:0]				SRWDTO[3:0]				SCCSTO[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	CLKEN	0	R/W	MMC Clock Output Control 0: Does not output the MMC clock (fixed to low level) 1: Outputs the MMC clock
23 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
19 to 16	CLKDIV[3:0]	0000	R/W	MMC Clock Frequency Setting 0000: clks1/2^1 0001: clks1/2^2 : 0111: clks1/2^8 1000: clks1/2^9 1001 to 1111: Setting prohibited For details of MMC clock frequency settings in boot operations, see section 30.6.6, MMC Clock Frequency in Boot Mode and section 30.3.11, Boot Operation Setting Register (CE_BOOT).

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
14	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
13, 12	SRSPTO [1:0]	00	R/W	Response Timeout Setting Specifies the timeout period for the RSPTO bit of CE_INT. 00: 64 MMC clock cycles 01: 128 MMC clock cycles 10: 256 MMC clock cycles 11: Setting prohibited
11 to 8	SRBSYTO [3:0]	0000	R/W	Response Busy Timeout Setting Specifies the timeout period for the RBSYTO bit of CE_INT. 0000: 2^{14} MMC clock cycles 0001: 2^{15} MMC clock cycles : 1110: 2^{28} MMC clock cycles 1111: 2^{29} MMC clock cycles
7 to 4	SRWDTO [3:0]	0000	R/W	Write Data/Read Data Timeout Setting Specifies the timeout period for the WDATTO and RDATTO bits of CE_INT. 0000: 2^{14} MMC clock cycles 0001: 2^{15} MMC clock cycles : 1110: 2^{28} MMC clock cycles 1111: 2^{29} MMC clock cycles

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	SCCSTO [3:0]	0000	R/W	CCS Timeout Setting Specifies the timeout period for the CCSTO bit of CE_INT. 0000: 2^{14} MMC clock cycles 0001: 2^{15} MMC clock cycles : 1110: 2^{28} MMC clock cycles 1111: 2^{29} MMC clock cycles

30.3.7 Buffer Access Configuration Register (CE_BUF_ACC)

CE_BUF_ACC configures the method of accessing data registers and mode of DMA transfer. This register must be set before CMD[5:0] in CE_CMD_SET is set. For explanation of the buffers, see section 30.6.3, Buffer Structure.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	DMAWEN	DMAREN	—	—	—	—	—	—	BUSW	ATYP
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

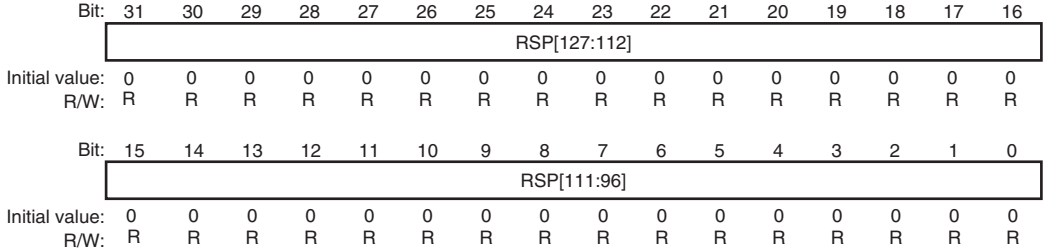
Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
25	DMAWEN	0	R/W	Buffer Write DMA Transfer Request Enable 0: Disables DMA transfer request for buffer writing 1: Enables DMA transfer request for buffer writing
24	DMAREN	0	R/W	Buffer Read DMA Transfer Request Enable 0: Disables DMA transfer request for buffer reading 1: Enables DMA transfer request for buffer reading
23, 22	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
21 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17	BUSW	0	R/W	Data register access size selection 0: When access to CE_DATA in 32-bit. 1: When access to CE_DATA in 16-bit.

Bit	Bit Name	Initial Value	R/W	Description
16	ATYP	0	R/W	Buffer access selection 0: When not swapped byte-wise. 1: When swapped byte-wise. Note: For buffer access, see section 30.6.4, Buffer Access Select Function in Access to CE_DATA.
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

30.3.8 Response Registers 3 to 0 (CE_RESP3 to CE_RESP0)

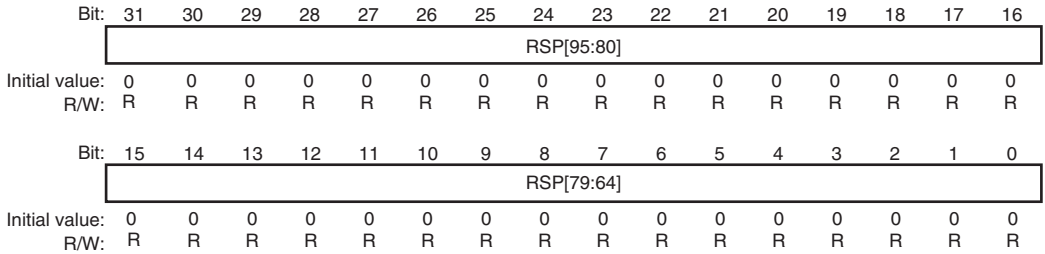
CE_RESP3 to CE_RESP0 are the registers for storing the response that has been received. For the formats of response values, see section 30.6.1, Command/Response Formats.

- CE_RESP3



Bit	Bit Name	Initial Value	R/W	Description (CE_RESP3)
31 to 0	RSP[127:96]	H'00000000	R	R2 response [127:96]

- CE_RESP2



Bit	Bit Name	Initial Value	R/W	Description (CE_RESP2)
31 to 0	RSP[95:64]	H'00000000	R	R2 response [95:64]

- CE_RESP1

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RSP[63:48]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RSP[47:32]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description (CE_RESP1)
31 to 0	RSP[63:32]	H'00000000	R	R2 response [63:32]

- CE_RESP0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RSP[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RSP[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description (CE_RESP0)
31 to 0	RSP[31:0]	H'00000000	R	Response [31:0] or R2 response [31:0]

30.3.9 Response Register for Automatically-Issued CMD12 (CE_RESP_CMD12)

CE_RESP_CMD12 is the register for storing the response to the automatically-issued CMD12.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RSP12[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RSP12[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	RSP12[31:0]	H'00000000	R	CMD12 response [31:0]

30.3.10 Data Register (CE_DATA)

CE_DATA is used to access the buffers of this module. In 16-bit access, only DATA[31:16] is accessible. For the write/read data formats, see section 30.6.5, Data Formats.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DATA[31:0]	H'00000000	R/W	Buffer write/read data [31:0]

30.3.11 Boot Operation Setting Register (CE_BOOT)

CE_BOOT controls the MMC clock and sets timeout values in boot mode. Do not change the setting of this register while a command sequence is in progress.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BTCLKDIV				SBTACKTO				S1STBTDATTO				SBTDATTO			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	BTCLKDIV[3:0]	H'0	R/W	MMC Clock Frequency Setting in Boot Mode 0000: Module clock/2^1 0001: Module clock/2^2 0010: Module clock/2^3 0011: Module clock/2^4 0100 to 1111: Settings prohibited Set these bits to a value lower than that in the CLKDIV bits of CE_CLK_CTRL. For MMC clock frequency in boot mode, see section 30.6.6, MMC Clock Frequency in Boot Mode.
27 to 24	SBTACKTO[3:0]	H'0	R/W	Boot Acknowledge Timeout Setting 0000: 2^14 × MMC clock cycles 0001: 2^15 × MMC clock cycles | 1110: 2^28 × MMC clock cycles 1111: 2^29 × MMC clock cycles

Bit	Bit Name	Initial Value	R/W	Description
23 to 20	S1STBTDATTO [3:0]	H'0	R/W	1st Boot Data Timeout Setting 0000: 2^14 × MMC clock cycles 0001: 2^15 × MMC clock cycles | 1110: 2^28 × MMC clock cycles 1111: 2^29 × MMC clock cycles
19 to 16	SBTDATTO[3:0]	H'0	R/W	Interval Between Boot Data Timeout Setting 0000: 2^14 × MMC clock cycles 0001: 2^15 × MMC clock cycles | 1110: 2^28 × MMC clock cycles 1111: 2^29 × MMC clock cycles
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

30.3.12 Interrupt Flag Register (CE_INT)

CE_INT indicates various statuses during execution of a command sequence. Each bit is set when its setting condition has been met. To clear flag(s), write 0 only to the bit(s) to be cleared and write 1 to the other bits.

For the operation in the case of an error or timeout, see section 30.6.7, Operation in the Case of Error/Timeout.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	CCSDE	—	—	CMD12 DRE	CMD12 RBE	CMD12 CRE	DTRAN E	BUFR E	BUFW EN	BUFR EN	CCS RCV	—	RBSY E	CRSP E
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/WC0	R	R	R/WC0	R/WC0	R/WC0	R/WC0	R/WC0	R/WC0	R/WC0	R/WC0	R	R/WC0	R/WC0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMD VIO	BUF VIO	—	—	WDAT ERR	RDAT ERR	RIDX ERR	RSP ERR	—	—	CCS TO	CRCS TO	WDAT TO	RDAT TO	RBSY TO	RSP TO
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/WC0	R/WC0	R	R	R/WC0	R/WC0	R/WC0	R/WC0	R	R	R/WC0	R/WC0	R/WC0	R/WC0	R/WC0	R/WC0

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved These bits are always read as 0. The writing value should always be 1.
29	CCSDE	0	R/WC0*	CCSD Issuance Complete [Setting conditions] CCSD has been issued [Clearing condition] Writing a 0 to this bit
28, 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 1.

Bit	Bit Name	Initial Value	R/W	Description
26	CMD12DRE	0	R/WC0*	Automatic CMD12 Issuance & Buffer Read Complete [Setting conditions] Response busy for automatically-issued CMD12 and buffer reading have been completed. [Clearing condition] Writing a 0 to this bit Note: When CMD12DRE has been set, CMD12RBE, CMD12CRE, and BUFRE have also been set. So, these bits should be cleared as well.
25	CMD12RBE	0	R/WC0*	Automatic CMD12 Issuance Response Busy Complete [Setting conditions] Reception of the response and response busy for an automatically-issued CMD12 have been completed. [Clearing condition] Writing a 0 to this bit Note: When CMD12RBE has been set, CMD12CRE has also been set. So, this bit should be cleared as well. When CMD12RBE is set during a multi-block write, DTRANE is also set. So clear the bit as well.
24	CMD12CRE	0	R/WC0*	Automatic CMD12 Response Complete [Setting conditions] The response to an automatically-issued CMD12 has been received. [Clearing condition] Writing a 0 to this bit
23	DTRANE	0	R/WC0*	Data Transmission Complete [Setting conditions] Transmission of all blocks of data has been completed. • When configured to receive CRC status: Completion of busy (data busy) after reception of CRC status • When configured not to receive CRC status: Completion of data transmission [Clearing condition] Writing a 0 to this bit

Bit	Bit Name	Initial Value	R/W	Description
22	BUFRE	0	R/WC0*	Buffer Read Complete [Setting conditions] Other than in boot operations All blocks of data have been received and the data have been read from the buffer In boot operations All blocks of data have been received and the data have been read from the buffer, MMCCMD has been modified from 0 to 1, and 48 MMC clock cycles have elapsed. [Clearing condition] Writing a 0 to this bit
21	BUFWEN	0	R/WC0*	Buffer Write Ready [Setting conditions] The buffer has become empty and ready for writing. [Clearing condition] Writing a 0 to this bit Note: This bit is not set when DMA transfer request for buffer writing is enabled.
20	BUFREN	0	R/WC0*	Buffer Read Ready [Setting conditions] Transfer block size of data have been stored in the buffer and it has become ready for reading [Clearing condition] Writing a 0 to this bit Note: This bit is not set when DMA transfer request for buffer reading is enabled.
19	CCSRCV	0	R/WC0*	CCS Reception Complete [Setting conditions] CCS has been received. [Clearing condition] Writing a 0 to this bit
18	—	0	R	Reserved This bit is always read as 0. The write value should always be 1.

Bit	Bit Name	Initial Value	R/W	Description
17	RBSYE	0	R/WC0*	Response Busy Complete [Setting conditions] Reception of a response and response busy have been completed. [Clearing condition] Writing a 0 to this bit Note: When RBSYE has been set, CRSPE has also been set. So, this bit should be cleared as well. Completion of reception of the response and response busy for automatically-issued CMD12 is reflected in CMD12RBE.
16	CRSPE	0	R/WC0*	Command/Response Complete [Setting conditions] • Other than in boot operations When configured not to receive response: A command has been transmitted When configured to receive 6- or 17-byte response: A response has been received • In boot operations When reception of boot acknowledge has been selected: The boot acknowledge pattern has been received. [Clearing condition] Writing a 0 to this bit Note: Completion of reception of the response to automatically-issued CMD12 is reflected in CMD12CRE.

Bit	Bit Name	Initial Value	R/W	Description
15	CMDVIO	0	R/WC0*	Command Issuance Error [Setting conditions] Illegal setting has been made in CE_CMD_SET or CE_BLOCK_SET. • During execution of a command sequence: Writing to CMD[5:0] in CE_CMD_SET (The command sequence is not stopped automatically.) • At the start of command sequence: Writing to CMD[5:0] in CE_CMD_SET when the registers have been set for one of the following combinations of selection — No response + response busy — No response + with data + not during boot operations — No response + acceptance of CCS enabled — No data + automatic CMD12 issuance — With data + single-block transfer + automatic CMD12 issuance — With data + automatic CMD12 issuance + acceptance of CCS enabled — With data + response busy + automatic CMD12 issuance — With data + transfer block size = 0 — With data + transfer block size ≥ 513 — With data + multi-block transfer + number of blocks for transfer = 0 — Boot operations + no data — Boot operations +write — Boot operations + response busy — Boot operations +automatic CMD12 issuance — Boot acknowledge reception + not during boot operations [Clearing condition] Writing a 0 to this bit

Bit	Bit Name	Initial Value	R/W	Description
14	BUFVIO	0	R/WC0*	Buffer Access Error [Setting conditions] Illegal buffer access has been attempted. • CE_ DATA has been accessed exceeding the block size set in BLKSIZ[15:0] in CE_BLOCK_SET • While data is being read from the card: CE_DATA has been accessed with BUFREN not set (when DMA is used, with no DMA transfer request asserted for buffer reading) • While data is being written to the card: CE_DATA has been accessed with BUFWEN not set (when DMA is used, with no DMA transfer request asserted for buffer writing) [Clearing condition] Writing a 0 to this bit Note: When BUFVIO has been set, the command sequence is not stopped automatically.
13, 12	—	All 0	R	Reserved These bits are always read as 0. The writing value should always be 1.
11	WDATERR	0	R/WC0*	Write Data Error [Setting conditions] • Error is found in the data that has been written. • Error is in the status of the CRC status • Error is in the end bits of the CRC status [Clearing condition] Writing a 0 to this bit Note: When WDATERR has been set, the command sequence is stopped automatically.
10	RDATERR	0	R/WC0*	Read Data Error [Setting conditions] Error is found in the read data. • Error is in CRC16 of the read data • Error is in the end bits of the read data [Clearing condition] Writing a 0 to this bit Note: When RDATERR has been set, the command sequence is stopped automatically.

Bit	Bit Name	Initial Value	R/W	Description
9	RIDXERR	0	R/WC0*	Response Index Error [Setting conditions] Error has been found in the index value of the response. • When an error has been found in [45:40] of a 6-byte response (including automatically-issued CMD12) or [133:128] of a 17-byte response (The items to be checked are set by RIDXC in CE_CMD_SET.) [Clearing condition] Writing a 0 to this bit Note: When RIDXERR has been set, the command sequence is stopped automatically.
8	RSPERR	0	R/WC0*	Response Error [Setting conditions] Error has been found in the response values of the response. • Transmission bit in the response is high • Error is in the end bits of the response • When an error has been found in [7:1] of a 6-byte response (including automatically-issued CMD12) or a 17-byte response (The items to be checked are set by RCRC7C in CE_CMD_SET.) • Error in the boot acknowledge pattern • Error in the end bits of the boot acknowledge [Clearing condition] Writing a 0 to this bit Note: When RSPERR has been set, the command sequence is stopped automatically.
7, 6	—	All 0	R	Reserved These bits are always read as 0. The writing value should always be 1.

Bit	Bit Name	Initial Value	R/W	Description
5	CCSTO	0	R/WC0*	CCS Timeout [Setting conditions] CCS could not be received [Clearing condition] Writing a 0 to this bit Note: The period for timeout detection is set in SCCSTO[3:0] of CE_CLK_CTRL. The command sequence is not stopped even if CCSTO is set.
4	CRCSTO	0	R/WC0*	CRC Status Timeout [Setting conditions] CRC status could not be received [Clearing condition] Writing a 0 to this bit Note: The command sequence is not stopped even if CRCSTO is set.
3	WDATTO	0	R/WC0*	Write Data Timeout [Setting conditions] Data busy that follows CRC status does not end [Clearing condition] Writing a 0 to this bit Note: The period for timeout detection is set in SRWDTO[3:0] of CE_CLK_CTRL. The command sequence is not stopped even if WDATTO is set.

Bit	Bit Name	Initial Value	R/W	Description
2	RDATTO	0	R/WC0*	Read Data Timeout [Setting conditions] • Other than in boot operations — Read data could not be received within the period set by SRWDTO in CE_CLK_CTRL after the read command was transmitted — Read data could not be received within the period set by SRWDTO in CE_CLK_CTRL after the read data was received. • In boot operations — The first read data could not be received within the period set by S1STBTDATTO in CE_BOOT. — Read data could not be received within the period set by SBTDATTO in CE_BOOT after the read data was received. [Clearing condition] Writing a 0 to this bit Note: The command sequence is not stopped even if RDATTO is set.

Bit	Bit Name	Initial Value	R/W	Description
1	RBSYTO	0	R/WC0*	Response Busy Timeout [Setting conditions] The busy status remains unchanged after the period set by SRBSYTO in CE_CLK_CTRL after the command (including automatically-issued CMD12) was transmitted. [Clearing condition] Writing a 0 to this bit Note: The command sequence is not stopped even if RBSYTO is set.
0	RSPTO	0	R/WC0*	Response Timeout [Setting conditions] • Other than in boot operations Response could not be received within the period set by SRSPTO in CE_CLK_CTRL after the command (including automatically-issued CMD12) was transmitted. • In boot operations When reception of boot acknowledge has been selected: The boot acknowledge could not be received within the period set by SBTACKTO in CE_BOOT. [Clearing condition] Writing a 0 to this bit Note: The command sequence is not stopped even if RSPTO is set.

Note: * Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored.

30.3.13 Interrupt Enable Register (CE_INT_EN)

CE_INT_EN controls output of the CE_INT-related interrupt signals. If a flag in CE_INT is set to 1 while its corresponding bit in CE_INT_EN is set to 1, an interrupt request is output. For details on interrupt requests, see section 30.4, Interrupt Requests.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	MCCSDE	—	—	MCMD12DRE	MCMD12RBE	MCMD12CRE	MDTRANE	MBUFRE	MBUFWEN	MBUFREN	MCCSRCV	—	MRBSYE	MCRSPE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MCMDVIO	MBUFVIO	—	—	MWDATERR	MRDATERR	MRIDXERR	MRSPERR	—	—	MCCSTO	MCRCSTO	MWDATTO	MRDATTO	MRBSYTO	MRSPTO
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

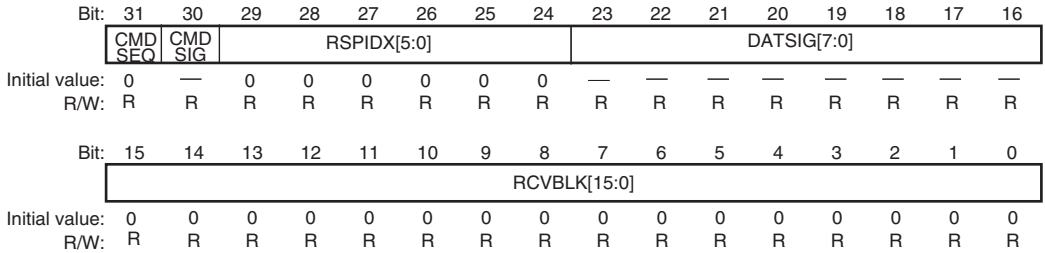
Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
29	MCCSDE	0	R/W	CCSDE Interrupt Mask 0: Disables interrupt output by the CCSDE flag 1: Enables interrupt output by the CCSDE flag
28, 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26	MCMD12DRE	0	R/W	CMD12DRE Interrupt Mask 0: Disables interrupt output by the CMD12DRE flag 1: Enables interrupt output by the CMD12DRE flag
25	MCMD12RBE	0	R/W	CMD12RBE Interrupt Mask 0: Disables interrupt output by the CMD12RBE flag 1: Enables interrupt output by the CMD12RBE flag
24	MCMD12CRE	0	R/W	CMD12CRE Interrupt Mask 0: Disables interrupt output by the CMD12CRE flag 1: Enables interrupt output by the CMD12CRE flag
23	MDTRANE	0	R/W	DTRANE Interrupt Mask 0: Disables interrupt output by the DTRANE flag 1: Enables interrupt output by the DTRANE flag

Bit	Bit Name	Initial Value	R/W	Description
22	MBUFRE	0	R/W	BUFRE Interrupt Mask 0: Disables interrupt output by the BUFRE flag 1: Enables interrupt output by the BUFRE flag
21	MBUFWEN	0	R/W	BUFWEN Interrupt Mask 0: Disables interrupt output by the BUFWEN flag 1: Enables interrupt output by the BUFWEN flag
20	MBUFREN	0	R/W	BUFREN Interrupt Mask 0: Disables interrupt output by the BUFREN flag 1: Enables interrupt output by the BUFREN flag
19	MCCSRCV	0	R/W	CCSRCV Interrupt Mask 0: Disables interrupt output by the CCSRCV flag 1: Enables interrupt output by the CCSRCV flag
18	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
17	MRBSYE	0	R/W	RBSYE Interrupt Mask 0: Disables interrupt output by the RBSYE flag 1: Enables interrupt output by the RBSYE flag
16	MCRSPE	0	R/W	CRSPE Interrupt Mask 0: Disables interrupt output by the CRSPE flag 1: Enables interrupt output by the CRSPE flag
15	MCMDVIO	0	R/W	CMDVIO Interrupt Mask 0: Disables interrupt output by the CMDVIO flag 1: Enables interrupt output by the CMDVIO flag
14	MBUFVIO	0	R/W	BUFVIO Interrupt Mask 0: Disables interrupt output by the BUFVIO flag 1: Enables interrupt output by the BUFVIO flag
13, 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11	MWDATERR	0	R/W	WDATERR Interrupt Mask 0: Disables interrupt output by the WDATERR flag 1: Enables interrupt output by the WDATERR flag

Bit	Bit Name	Initial Value	R/W	Description
10	MRDATERR	0	R/W	RDATERR Interrupt Mask 0: Disables interrupt output by the RDATERR flag 1: Enables interrupt output by the RDATERR flag
9	MRIDXERR	0	R/W	RIDXERR Interrupt Mask 0: Disables interrupt output by the RIDXERR flag 1: Enables interrupt output by the RIDXERR flag
8	MRSPERR	0	R/W	RSPERR Interrupt Mask 0: Disables interrupt output by the RSPERR flag 1: Enables interrupt output by the RSPERR flag
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	MCCSTO	0	R/W	CCSTO Interrupt Mask 0: Disables interrupt output by the CCSTO flag 1: Enables interrupt output by the CCSTO flag
4	MCRCSTO	0	R/W	CRCSTO Interrupt Mask 0: Disables interrupt output by the CRCSTO flag 1: Enables interrupt output by the CRCSTO flag
3	MWDATTO	0	R/W	WDATTO Interrupt Mask 0: Disables interrupt output by the WDATTO flag 1: Enables interrupt output by the WDATTO flag
2	MRDATTO	0	R/W	RDATTO Interrupt Mask 0: Disables interrupt output by the RDATTO flag 1: Enables interrupt output by the RDATTO flag
1	MRBSYTO	0	R/W	RBSYTO Interrupt Mask 0: Disables interrupt output by the RBSYTO flag 1: Enables interrupt output by the RBSYTO flag
0	MRSPTO	0	R/W	RSPTO Interrupt Mask 0: Disables interrupt output by the RSPTO flag 1: Enables interrupt output by the RSPTO flag

30.3.14 Status Register 1 (CE_HOST_STS1)

CE_HOST _STS1 indicates the number of blocks that have been transferred, the states of the MMCCMD line and MMCDAT lines, the index of the response that has been received, and whether a command sequence is in progress.



Bit	Bit Name	Initial Value	R/W	Description
31	CMDSEQ	0	R	Command Sequence in Progress 0: Command sequence is in the initial state 1: Command sequence is being executed
30	CMDSIG	Undefined	R	CMD Line Status Indicates the state on the command line.
29 to 24	RSPIDX [5:0]	H'00	R	Response Index Indicate [45:40] of a 6-byte response or [133:128] of a 17-byte response.
23 to 16	DATSIG [7:0]	Undefined	R	DAT[7:0] Status Indicate the states on the MMCDAT[7:0] lines. Note: When a communication error or a timeout error occurs, MMCDAT[0] may remain 0.
15 to 0	RCVBLK [15:0]	H'0000	R	Number of Transferred Blocks Indicate the number blocks that have been transferred. • When the DWEN bit in CE_CMD_SET is 0 Number of blocks read from the card • When the DWEN bit in CE_CMD_SET is 1 Number of blocks written to the card

30.3.15 Status Register 2 (CE_HOST_STS2)

CE_HOST _STS2 indicates timeout and error statuses.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRC STE	CRC 16E	AC12 CRCE	RSP CRC7E	CRC STEBE	RDAT EBE	AC12R EBE	RSP EBE	AC12 IDXE	RSP IDXE	BTACK PATE	BTACK EBE	—	CRCST[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	STCC STO	STRD ATTO	DATBS YTO	CRCST TO	AC12 BSYTO	RSPBS YTO	AC12 RSPTO	STRS PTO	BTAC KTO	1STBT DATTO	BTDA TTO	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	CRCSTE	0	R	CRC Status Error This bit is set to 1 when an error is found in the CRC status value.
30	CRC16E	0	R	Read Data CRC16 Error This bit is set to 1 when an error is found in CRC16 in the read data.
29	AC12CRCE	0	R	Automatic CMD12 Response CRC7 Error This bit is set to 1 when an error is found in [7:1] of the response to the automatically-issued CMD12. Note: The items to be checked are set by RCRC7C in CE_CMD_SET.
28	RSPCRC7E	0	R	Command Response CRC7 Error (other than automatically-issued CMD12) This bit is set to 1 when an error is found in [7:1] of a 6-byte response or a 17-byte response. Note: The items to be checked are set by RCRC7C in CE_CMD_SET.
27	CRCSTEBE	0	R	CRC Status End Bit Error This bit is set to 1 when an error is found in the end bits in CRC status.
26	RDATEBE	0	R	Read Data End Bit Error This bit is set to 1 when an error is found in the end bits in the read data.

Bit	Bit Name	Initial Value	R/W	Description
25	AC12REBE	0	R	Automatic CMD12 Response End Bit Error This bit is set to 1 when an error is found in the end bits of the response to the automatically-issued CMD12.
24	RSPEBE	0	R	Command Response End Bit Error (other than automatically-issued CMD12) This bit is set to 1 when an error is found in the end bits of the response.
23	AC12IDXE	0	R	Automatic CMD12 Response Index Error This bit is set to 1 when an error is found in [45:40] of the response to the automatically-issued CMD12. Note: The items to be checked are set by RIDXC in CE_CMD_SET.
22	RSPIDXE	0	R	Command Response Index Error (other than automatically-issued CMD12) This bit is set to 1 when an error is found in [45:40] of a 6-byte response or [133:128] of a 17-byte response. Note: The items to be checked are set by RIDXC in CE_CMD_SET.
21	BTACKPATE	0	R	Boot Acknowledge Pattern Error This bit is set to 1 when an error is found in the boot acknowledge pattern.
20	BTACKEBE	0	R	Boot Acknowledge End Bit Error This bit is set to 1 when an error is found in the end bits of the boot acknowledge.
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
18 to 16	CRCST [2:0]	000	R	CRC Status Indicate the value of the CRC status that has been received.
15	STCCSTO	0	R	CCS Timeout This bit is set to 1 if CCS is not received within the period set by the SCCSTO bits in CE_CLK_CTRL.

Bit	Bit Name	Initial Value	R/W	Description
14	STRDATTO	0	R	Read Data Timeout (Valid other than in boot operations) • This bit is set to 1 if read data is not received within the period set by the SRWDTO bits in CE_CLK_CTRL after a read command was transmitted. • This bit is set to 1 if read data is not received within the period set by the SRWDTO bits in CE_CLK_CTRL after a read data was received.
13	DATBSYTO	0	R	Data Busy Timeout This bit is set to 1 if busy status remains unchanged after the period set by the SRWDTO bits in CE_CLK_CTRL after the CRC status was received.
12	CRCSTTO	0	R	CRC Status Timeout This bit is set to 1 if CRC status could not be received.
11	AC12BSYTO	0	R	Automatic CMD12 Response Busy Timeout This bit is set to 1 if busy state remains unchanged after the period set by the SRBSYTO bits in CE_CLK_CTRL after the automatically-issued CMD12 was transmitted.
10	RSPBSYTO	0	R	Response Busy Timeout This bit is set to 1 if busy state remains unchanged after the period set by the SRBSYTO bits in CE_CLK_CTRL after a command (other than automatically-issued CMD12) was transmitted.
9	AC12RSPTO	0	R	Automatic CMD12 Response Timeout This bit is set to 1 if the response is not received within the period set by the SRSPTO bits in CE_CLK_CTRL after the automatically-issued CMD12 was transmitted.
8	STRSPTO	0	R	Response Timeout This bit is set to 1 if the response is not received within the period set by the SRSPTO bits in CE_CLK_CTRL after a command (other than automatically-issued CMD12) was transmitted.
7	BTACKTO	0	R	Boot Acknowledge Timeout In boot operations, this bit is set to 1 if boot acknowledge is not received within the period set by the SBTACKTO bits in CE_BOOT.

Bit	Bit Name	Initial Value	R/W	Description
6	1STBTDATTO	0	R	1st Boot Data Timeout In boot operations, this bit is set to 1 if the 1st read data is not received within the period set by the S1STBTDATTO bits in CE_BOOT.
5	BTDATTO	0	R	Interval between Boot Data Timeout In boot operations, this bit is set to 1 if read data is not received within the period set by the SBTDATTO bits in CE_BOOT after a read data was received.
4 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

30.3.16 Version Register (CE_VERSION)

CE_VERSION indicates the version number and controls software reset of this module.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SW RST	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VERSION[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	SWRST	0	R/W	Software Reset 0: Software reset cleared (normal operation) 1: Executes software reset
30 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	VERSION [15:0]	H'0002	R	Version Information Indicates the version number of this module.

30.4 Interrupt Requests

This module generates two types of interrupt requests: normal operation and error/timeout. The interrupt flags are accommodated in the CE_INT. When a bit in the flag register is set to 1 and also the corresponding bit in the interrupt enable register is set to 1 (enabled), an interrupt request is generated. Table 30.3 shows the specification of the interrupt requests.

Table 30.3 Specification of the interrupt requests

Flag Register	Bit	Enable Register	Bit	Interrupt Request
CE_INT	CCSDE	CE_INT_EN	MCCSDE	Normal operation interrupt
	CMD12DRE		MCMD12DRE	
	CMD12RBE		MCMD12RBE	
	CMD12CRE		MCMD12CRE	
	DTRANE		MDTRANE	
	BUFRE		MBUFRE	
	BUFWEN		MBUFWEN	
	BUFREN		MBUFREN	
	CCSRCV		MCCSRCV	
	RBSYE		MRBSYE	
	CRSPE		MCRSPE	
	CMDVIO		MCMDVIO	Error/timeout interrupt
	BUFVIO		MBUFVIO	
	WDATERR		MWDATERR	
	RDATERR		MRDATERR	
	RIDXERR		MRIDXERR	
	RSPERR		MRSPERR	
	CCSTO		MCCSTO	
	CRCSTO		MCRCSTO	
	WDATTO		MWDATTO	
	RDATTO		MRDATTO	
	RBSYTO		MRBSYTO	
	RSPTO		MRSPTO	

30.5 DMA Specifications

This module has two channels of DMA transfer requests: for buffer reading and for buffer writing.

The method of DMA transfer is configured by CE_BUF_ACC.

30.5.1 DMA for Buffer Writing

The DMA transfer request is asserted for buffer writing when the buffer has become empty while the DMAWEN bit in CE_BUF_ACC is set to 1.

The DMA transfer request stays asserted for the amount of data specified by BLKSIZ (the block size set in CE_BLOCK_SET) × BLKCNT (the number of blocks for transfer set in CE_BLOCK_SET), and negated after the last block has been transferred. Note that the BUFWEN bit in CE_INT will not be asserted during DMA transfer.

If an error has occurred during DMA transfer or DMA transfer is forcibly terminated, the command sequence is stopped automatically, which causes the DMA transfer request to be negated.

30.5.2 DMA for Buffer Reading

The DMA transfer request is asserted for buffer reading when the buffer stores data of the block size specified in CE_BLOCK_SET while the DMAREN bit in CE_BUF_ACC is set to 1.

The DMA transfer request stays asserted for the amount of data specified by BLKSIZ (the block size set in CE_BLOCK_SET) × BLKCNT (the number of blocks for transfer set in CE_BLOCK_SET), and negated after the last block has been transferred. Note that the BUFREN bit in CE_INT will not be asserted during DMA transfer.

If an error has occurred during DMA transfer or DMA transfer is forcibly terminated, the command sequence is stopped automatically, which causes the DMA transfer request to be negated.

30.6 Operation

This section describes the formats of the command and response, timing of automatic CMD12 issuance, buffer structures, buffer access select function, and the operation when an error has occurred.

30.6.1 Command/Response Formats

Figure 30.2 shows the format of the command to be transferred. The command index that is set in CMD[5:0] of CE_CMD_SET and the argument set in ARG[31:0] of CE_ARG are reflected in the command.

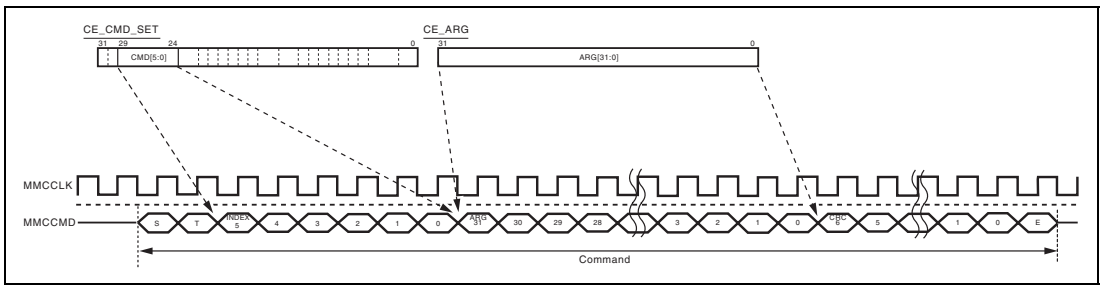


Figure 30.2 Command Format

Figures 30.3 and 30.4 show the formats when a 6-byte response and 17-byte response (R2) are received, respectively. The response index is stored in RSPIDX[5:0] of CE_HOST_STS1, and the status value of the response is stored to CE_RESP0 or CE_RESP3 to CE_RESP0.

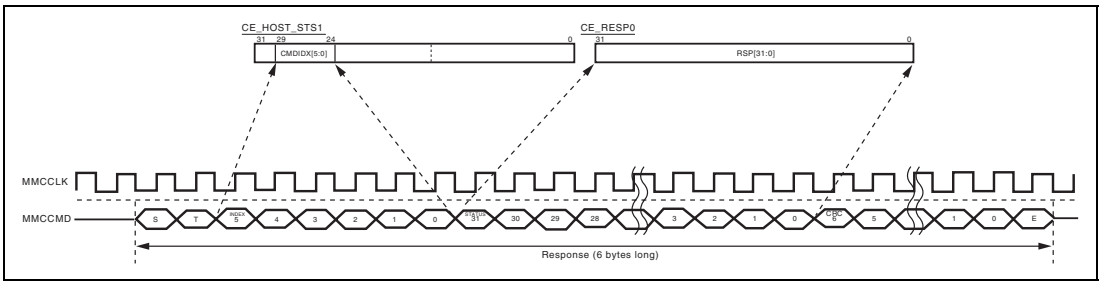


Figure 30.3 Format of 6-Byte Response

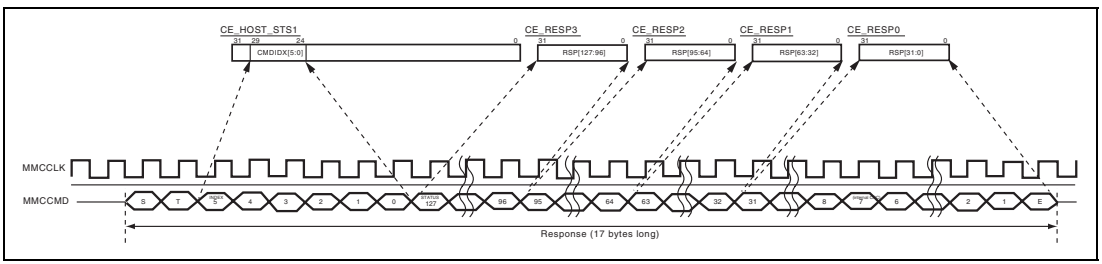


Figure 30.4 Format of 17-Byte Response (R2)

30.6.2 Automatic CMD12 Issuance

This module has the function that automatically issues CMD12 when multi-block transfer is performed with the CMD12EN in CE_CMD_SET set to 1. Timing of automatic CMD12 issuance is described for the case of multi-block reading and multi-block writing.

Figure 30.5 shows the timing of automatic CMD12 issuance in multi-block read. CMD12 is issued such that the end bit of the command is sent two bits before the end bit of the data during reception of the last block.

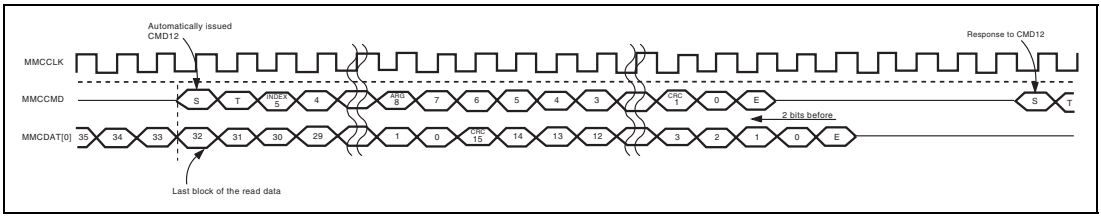


Figure 30.5 Timing of Automatically-Issued CMD12 in Multi-Block Read (1-Bit Mode)

Figure 30.6 shows the timing of automatic CMD12 issuance in multi-block write. CMD12 is issued after the data busy after transmission of the last block has ended.

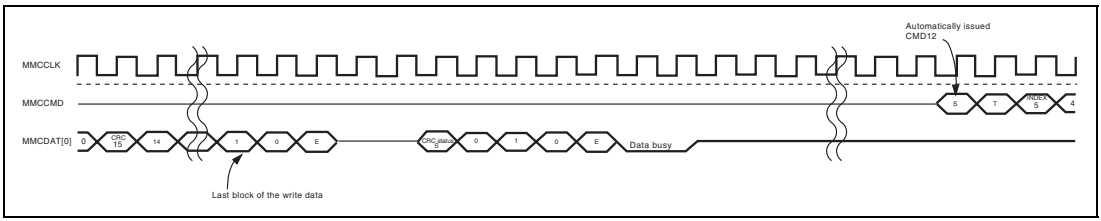


Figure 30.6 Timing of Automatically-Issued CMD12 in Multi-Block Write (1-Bit Mode)

30.6.3 Buffer Structure

This module has two 512-byte RAM units which are used for double buffering. If the transfer block size is set to 4 × n + 1 or 4 × n + 3, access should be made for 4 × n + 2 bytes or 4 × (n + 1) bytes in 16-bit access, and for 4 × (n + 1) bytes in 32-bit access. (n = 0, 1, 2, …, 127)

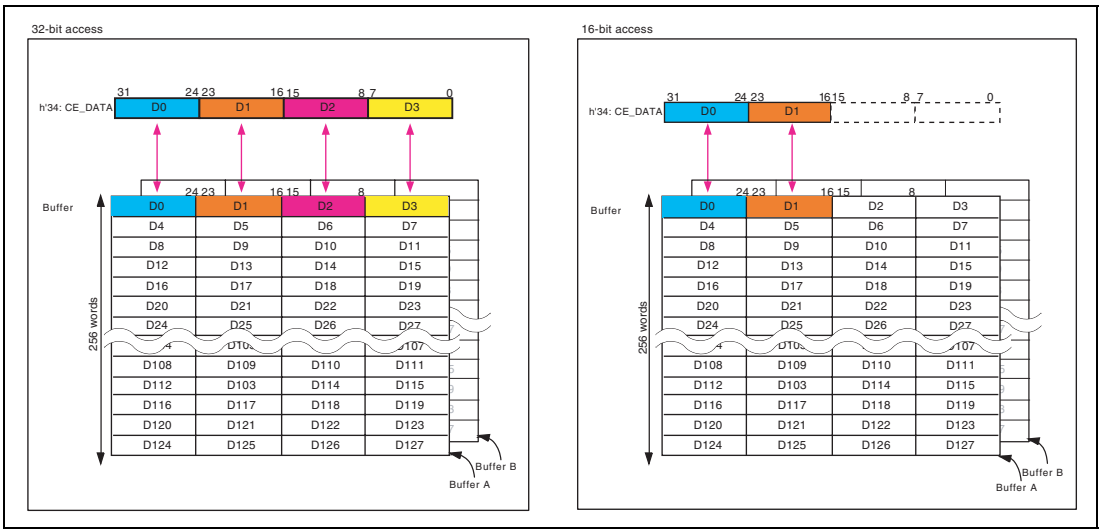


Figure 30.7 Double Buffer Structure

30.6.4 Buffer Access Select Function in Access to CE_DATA

This module has the buffer access select function that allows byte-wise swapping of data when the buffer is accessed by writing to or reading from CE_DATA. This function is enabled by the setting of CE_BUFF_ACC. Figure 30.8 shows the specification of 32-bit and 16-bit accesses.

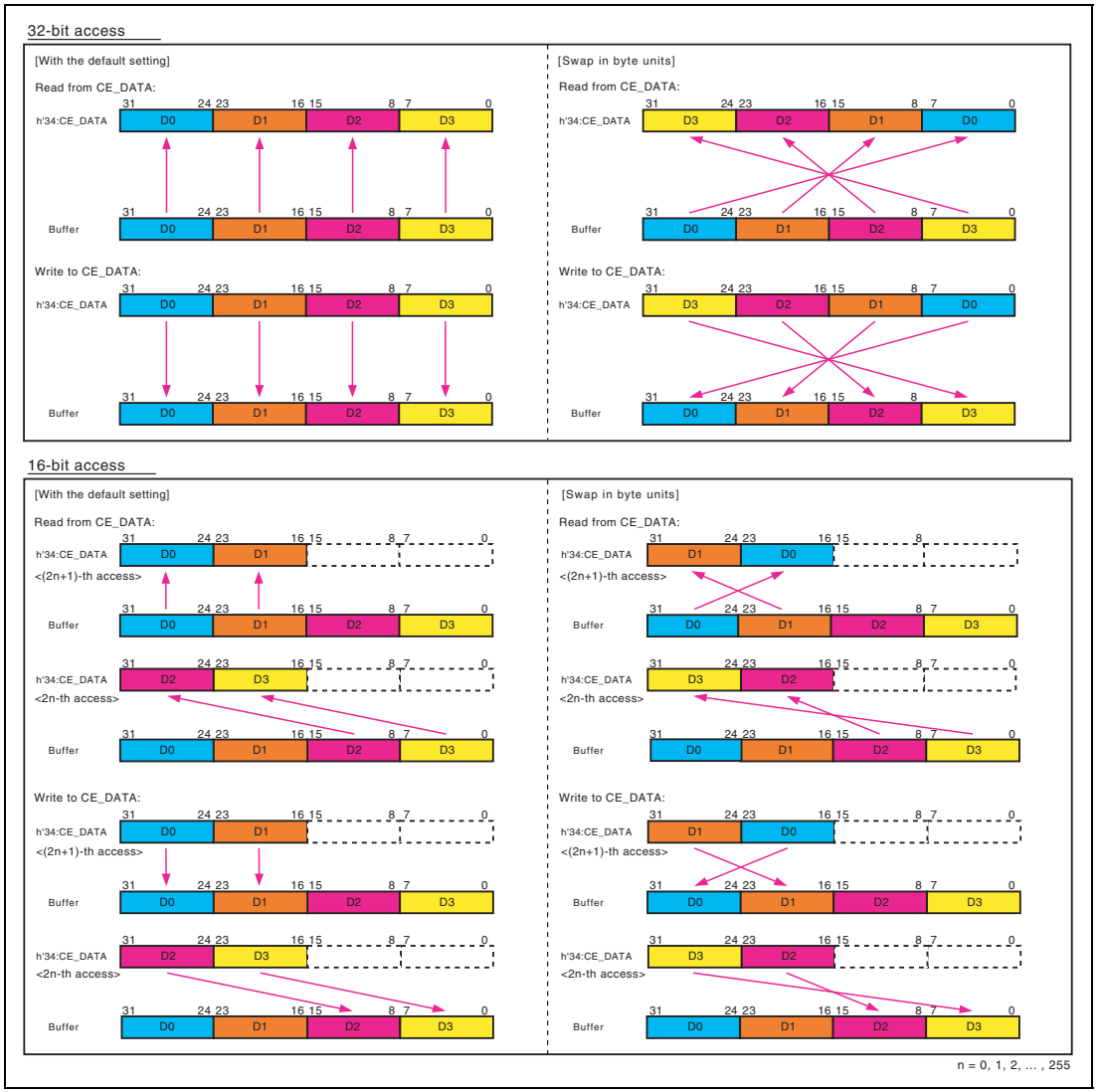


Figure 30.8 Specification of Byte-Swapping in 32/16-Bit Accesses

30.6.5 Data Formats

Figures 30.9 to 30.11 show the formats of data. In transmission, the value written to the buffer is reflected on the data lines, and in reception, the value of the received data is stored in the buffer.

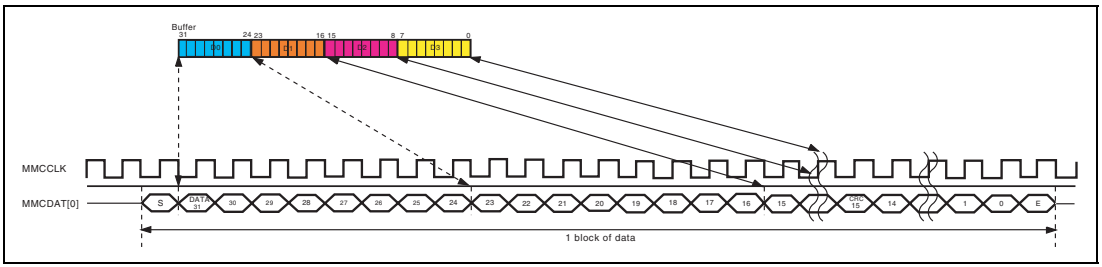


Figure 30.9 Data Format (1-Bit Mode)

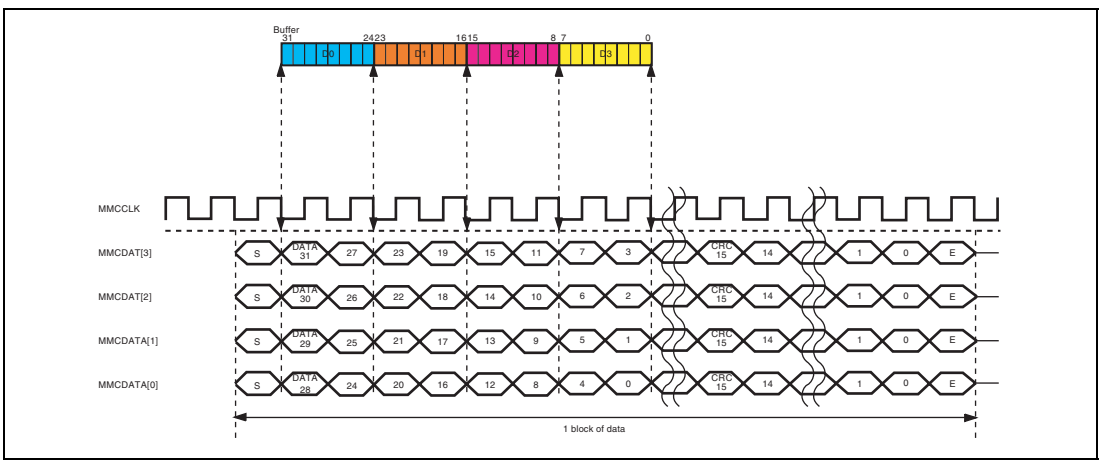


Figure 30.10 Data Format (4-Bit Mode)

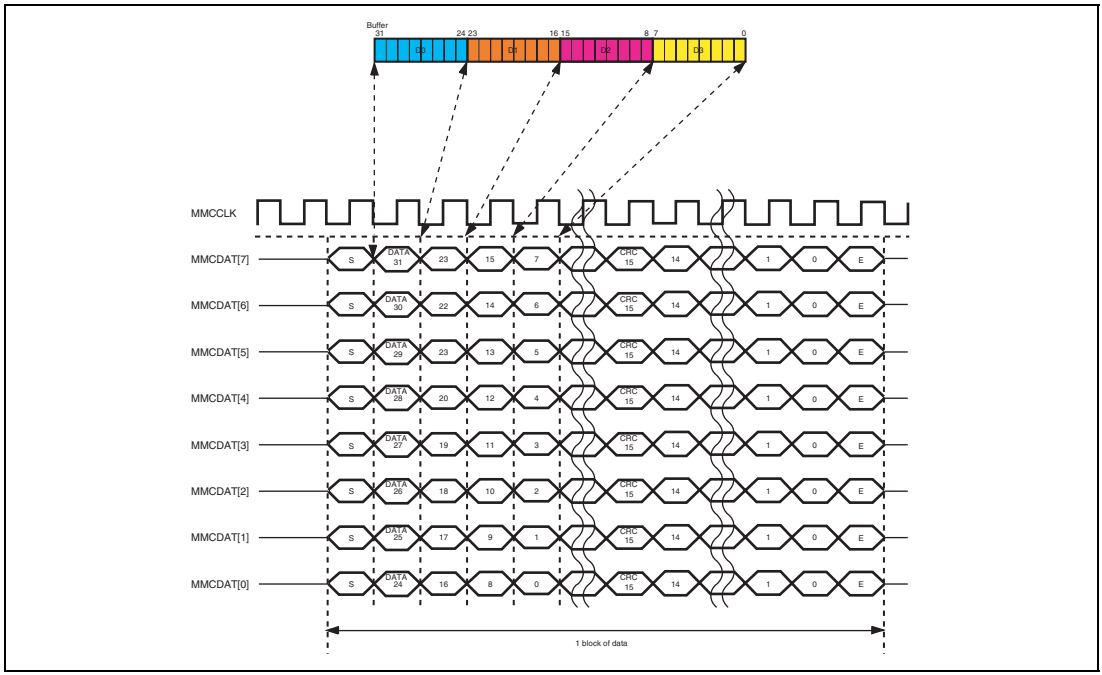


Figure 30.11 Data Format (8-Bit Mode)

30.6.6 MMC Clock Frequency in Boot Mode

Figure 30.12 shows the timing for changing the MMC clock frequency in boot mode. In boot mode, the MMC clock frequency can be switched to the value corresponding to the setting of the BTCLKDIV bits of CE_BOOT 74 MMC clock cycles after MMCCMD is modified from 1 to 0. Alternatively, it can be switched to the value corresponding to the setting of the CLKDIV bits of CE_CLK_CTRL 48 MMC clock cycles after MMCCMD is modified from 0 to 1.

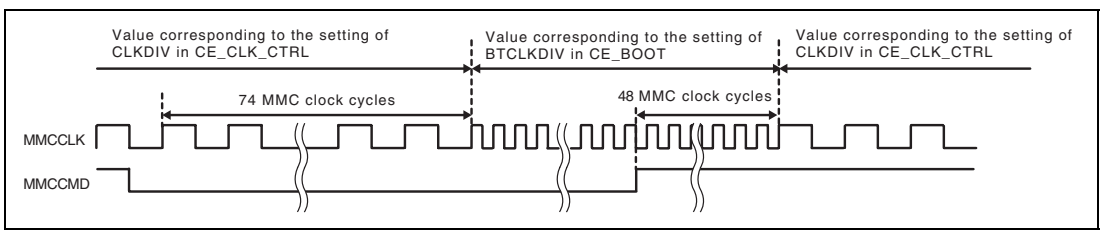


Figure 30.12 Timing for Changing MMC Clock Frequency in Boot Mode

30.6.7 Operation in the Case of Error/Timeout

The data for transmission or received data that had been stored in the buffers at the time of error occurrence are not guaranteed. After the error is recognized, check status register 1 and if the command sequence is still in progress, terminate it forcibly. Then, initialize the module and execute the command sequence again.

This module is not stopped when a timeout has occurred. If the command sequence does not end normally when the timeout has occurred and the flag in the status register 1 is still indicating that the command sequence is in progress, terminate the sequence forcibly and initialize the module.

For forcible termination, refer to section 30.8, Usage Notes.

30.7 Examples of Setting

This section shows the procedures for executing typical command sequences.

30.7.1 Legends

Figure 30.13 shows the legends for the symbols used in the figures in the following subsections.

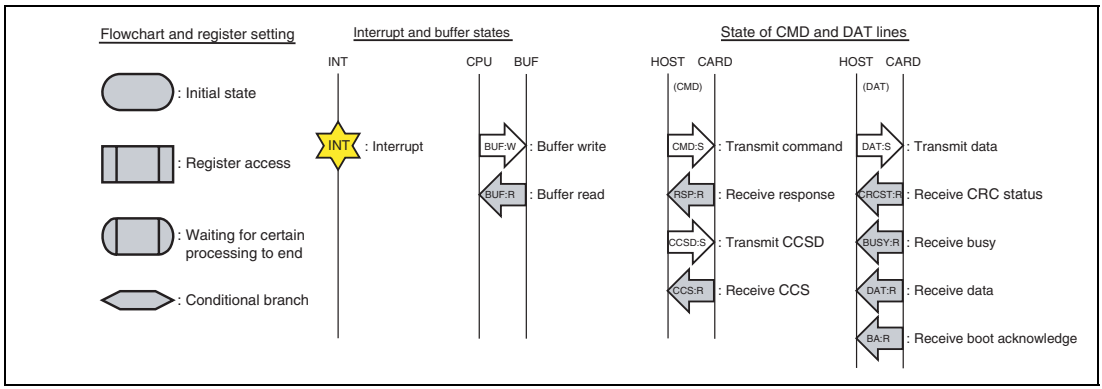


Figure 30.13 Legends for the Symbols Used in the Figures

30.7.2 Command Transmission

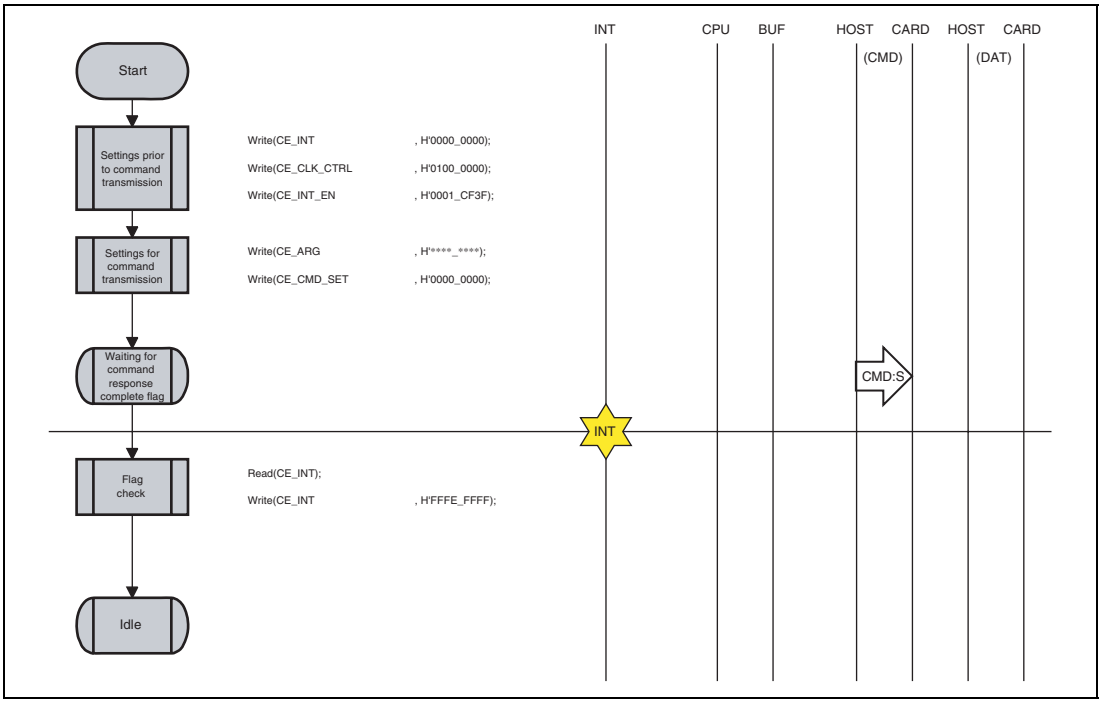


Figure 30.14 Command Transmission (CMD0)

30.7.3 Command Transmission → Response Reception

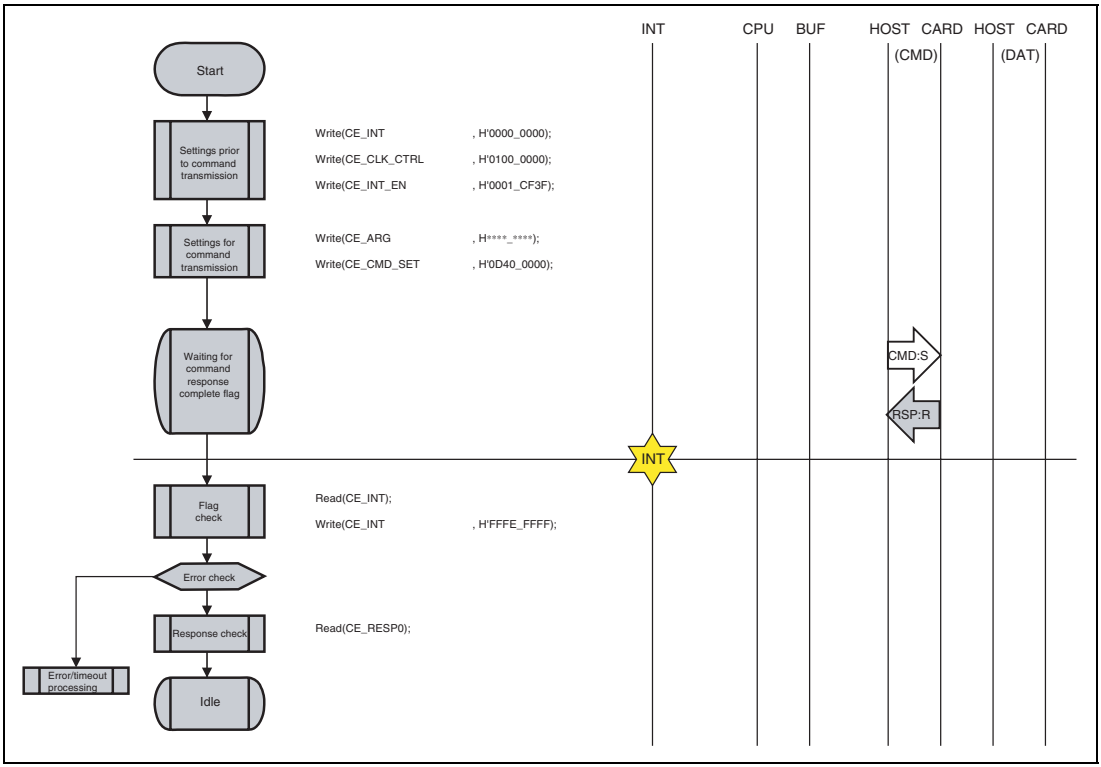


Figure 30.15 Command Transmission → Response Reception (CMD3)

30.7.4 Command Transmission → Response Reception (with Response Busy)

(1) When the busy time period is less than the period set by SRBSYTO in CE_CLK_CTRL

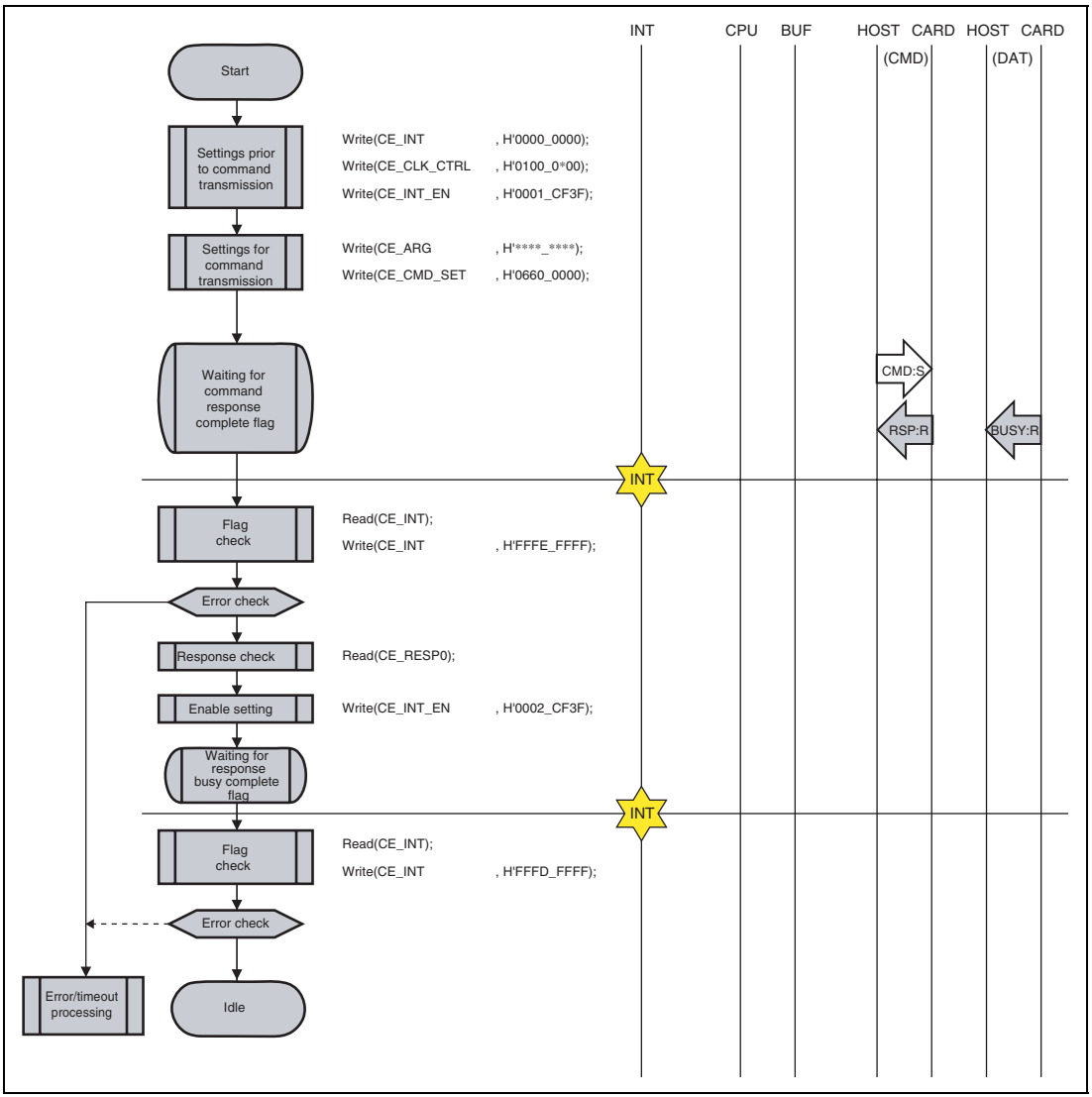


Figure 30.16 Command Transmission → Response Reception (with Response Busy) (CMD6)

(2) When the busy time period may be equal to or beyond the period set by SRBSYTO in CE_CLK_CTRL

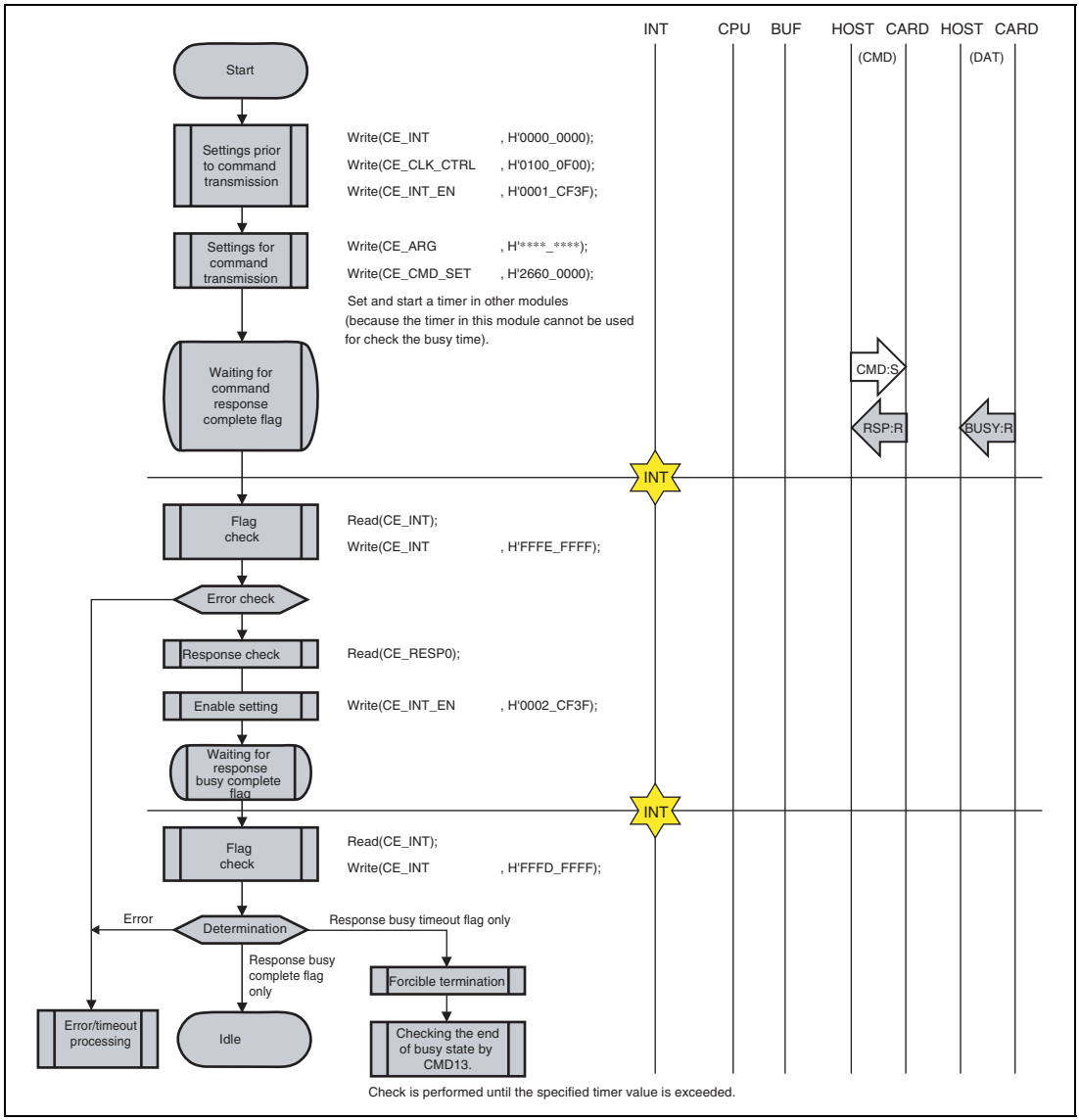


Figure 30.17 Command Transmission → Response Reception (with Response Busy) (CMD38)

30.7.5 Single-Block Read

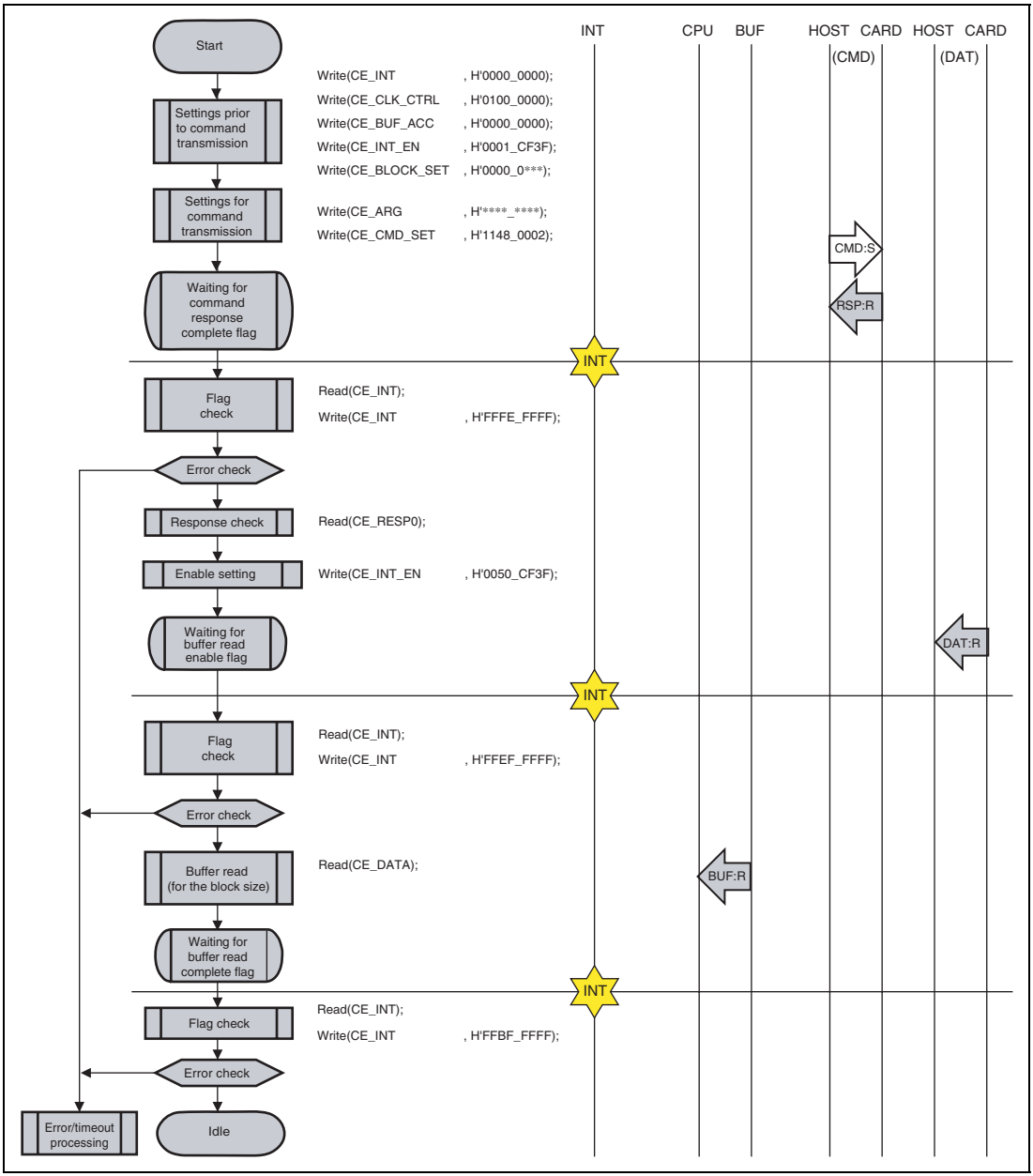


Figure 30.18 Single-Block Read (CMD17)

30.7.6 Multi-Block Read

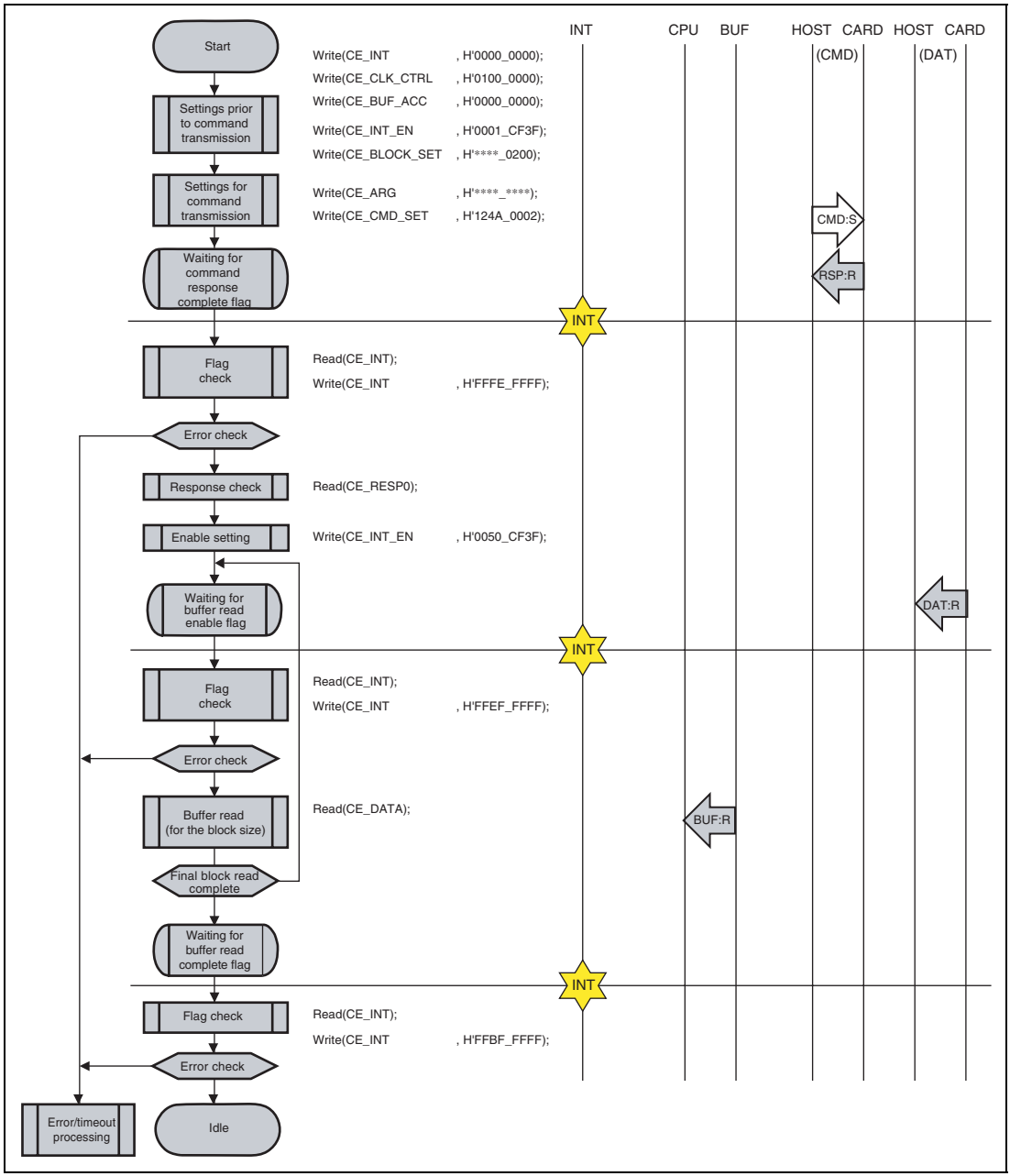


Figure 30.19 Multi-Block Read (CMD18 Pre-Defined)

30.7.7 Multi-Block Read (with Automatic CMD12 Issuance)

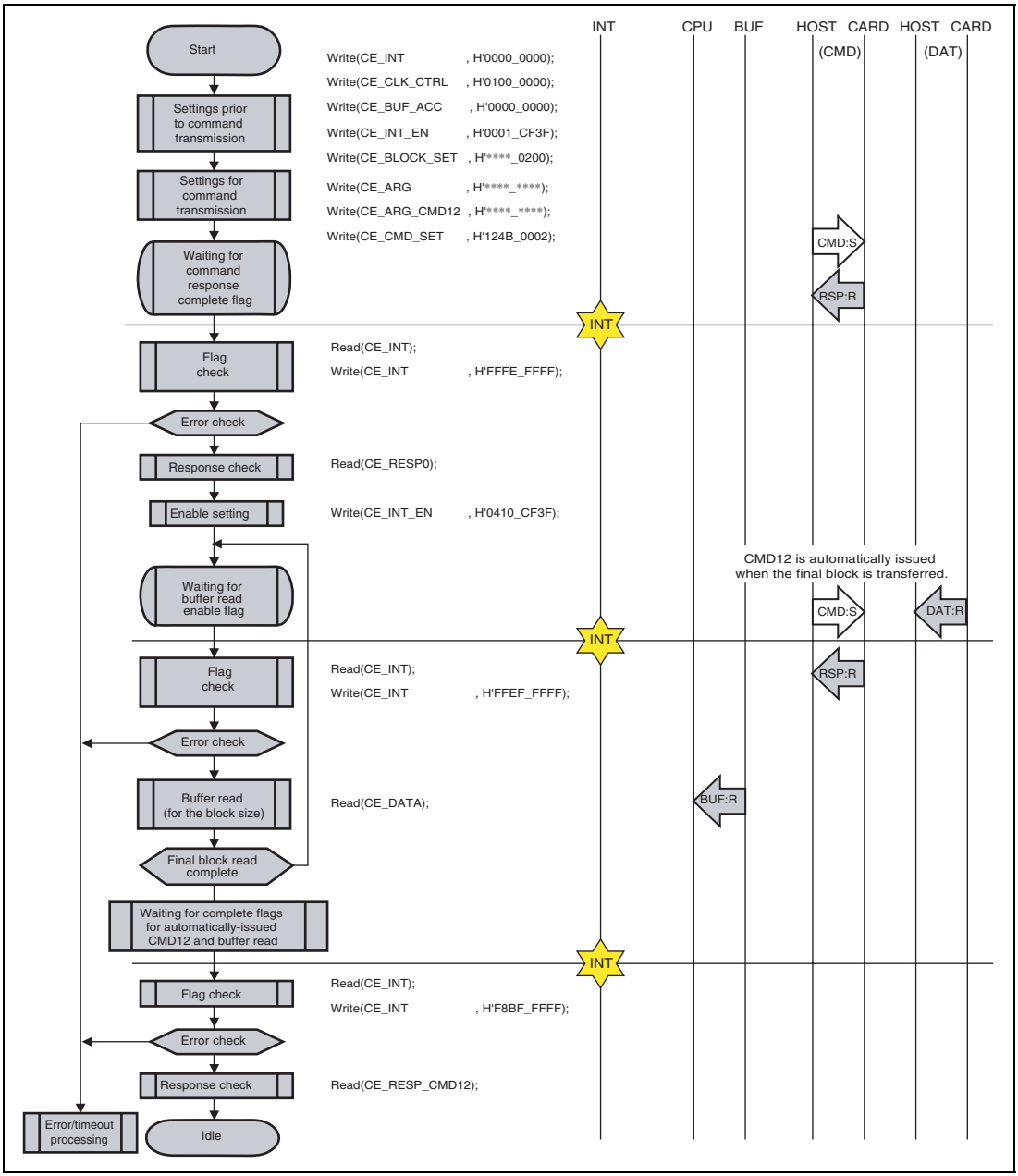


Figure 30.20 Multi-Block Read (with Automatic CMD12 Issuance) (CMD18 Open-Ended)

30.7.8 Single-Block Write

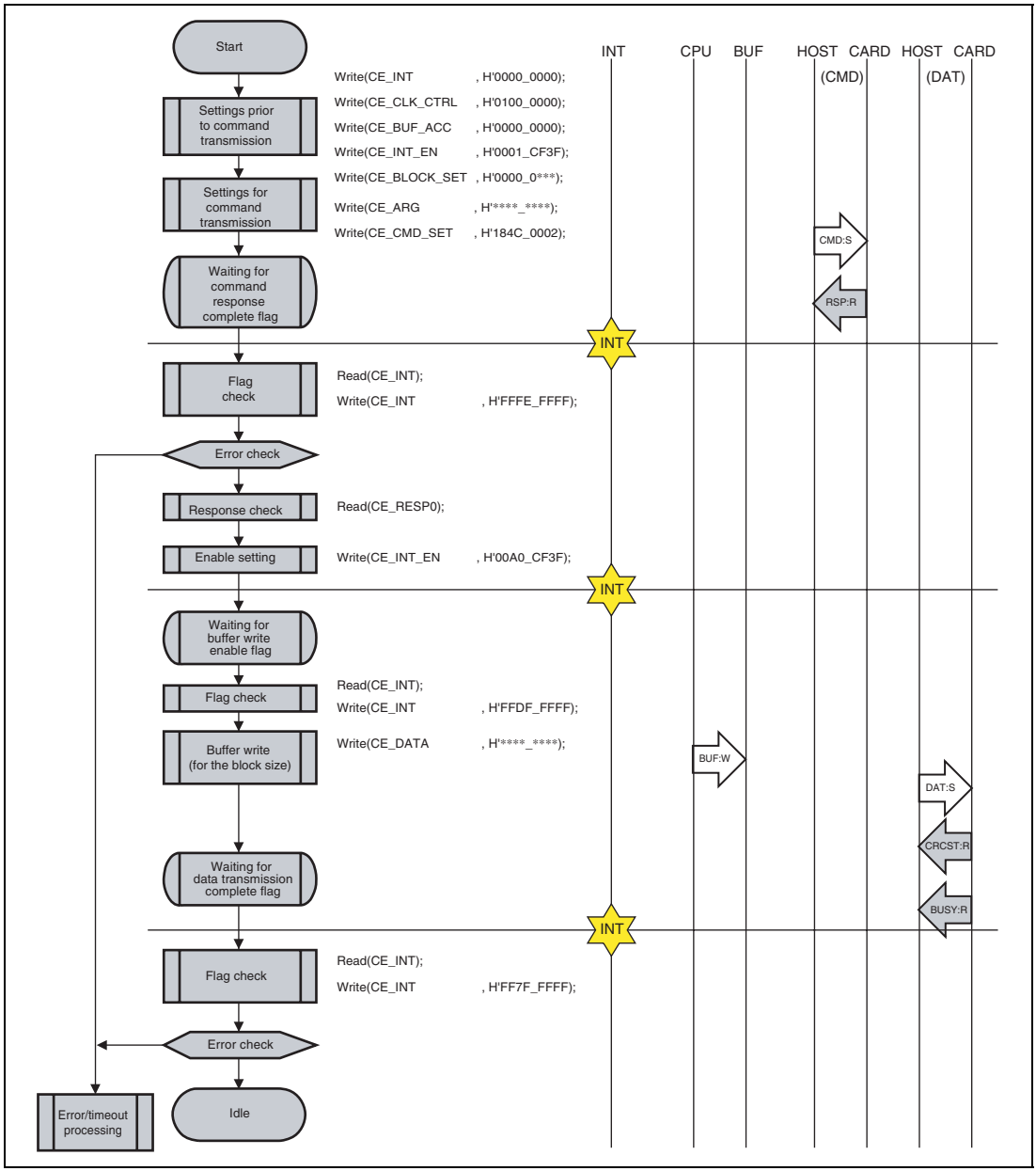


Figure 30.21 Single-Block Write (CMD24)

30.7.9 Multi-Block Write

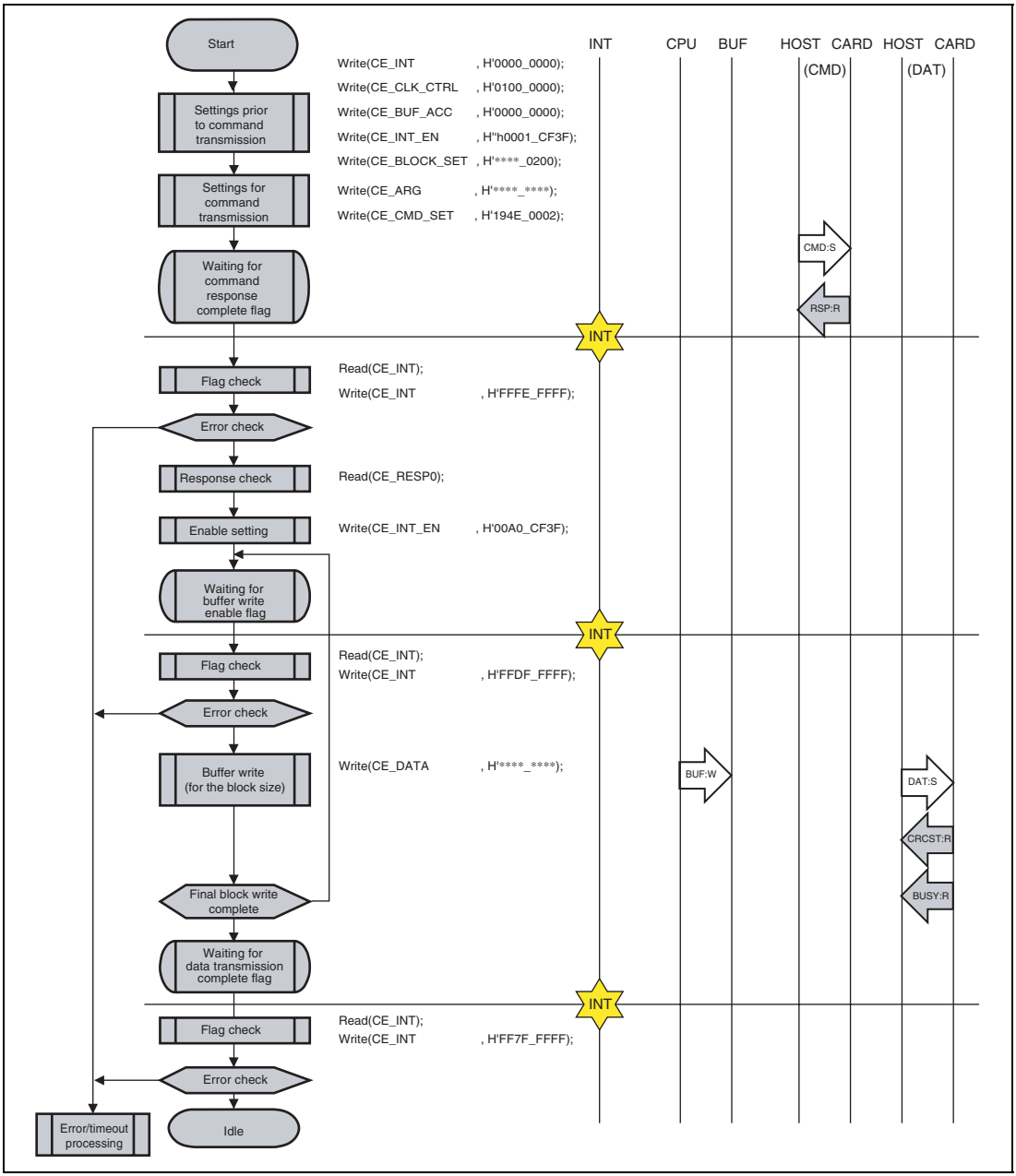


Figure 30.22 Multi-Block Write (CMD25 Pre-Defined)

30.7.10 Multi-Block Write (with Automatic CMD12 Issuance)

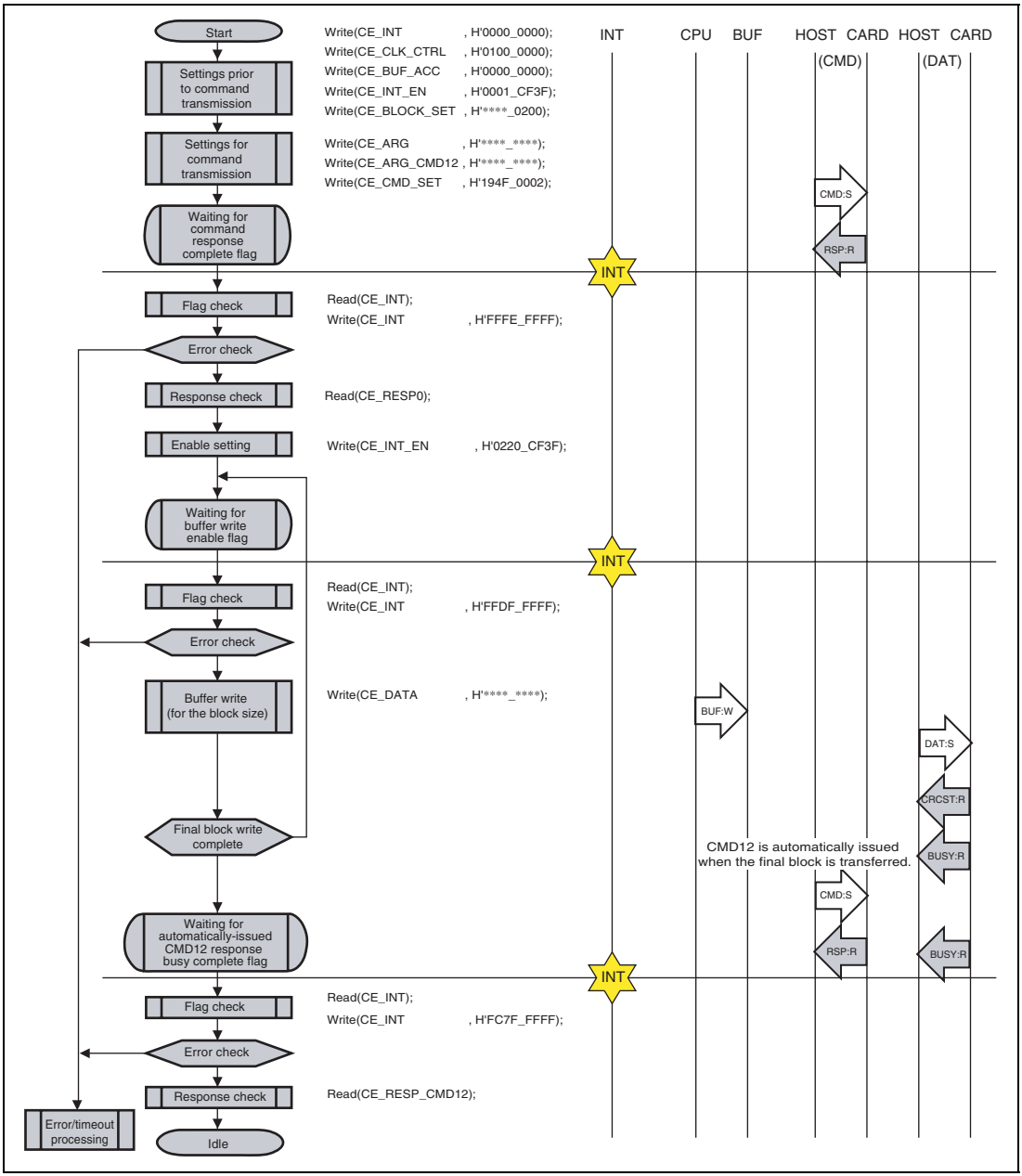


Figure 30.23 Multi-Block Write (with Automatic CMD12 Issuance) (CMD25 Open-Ended)

30.7.11 Boot Operations

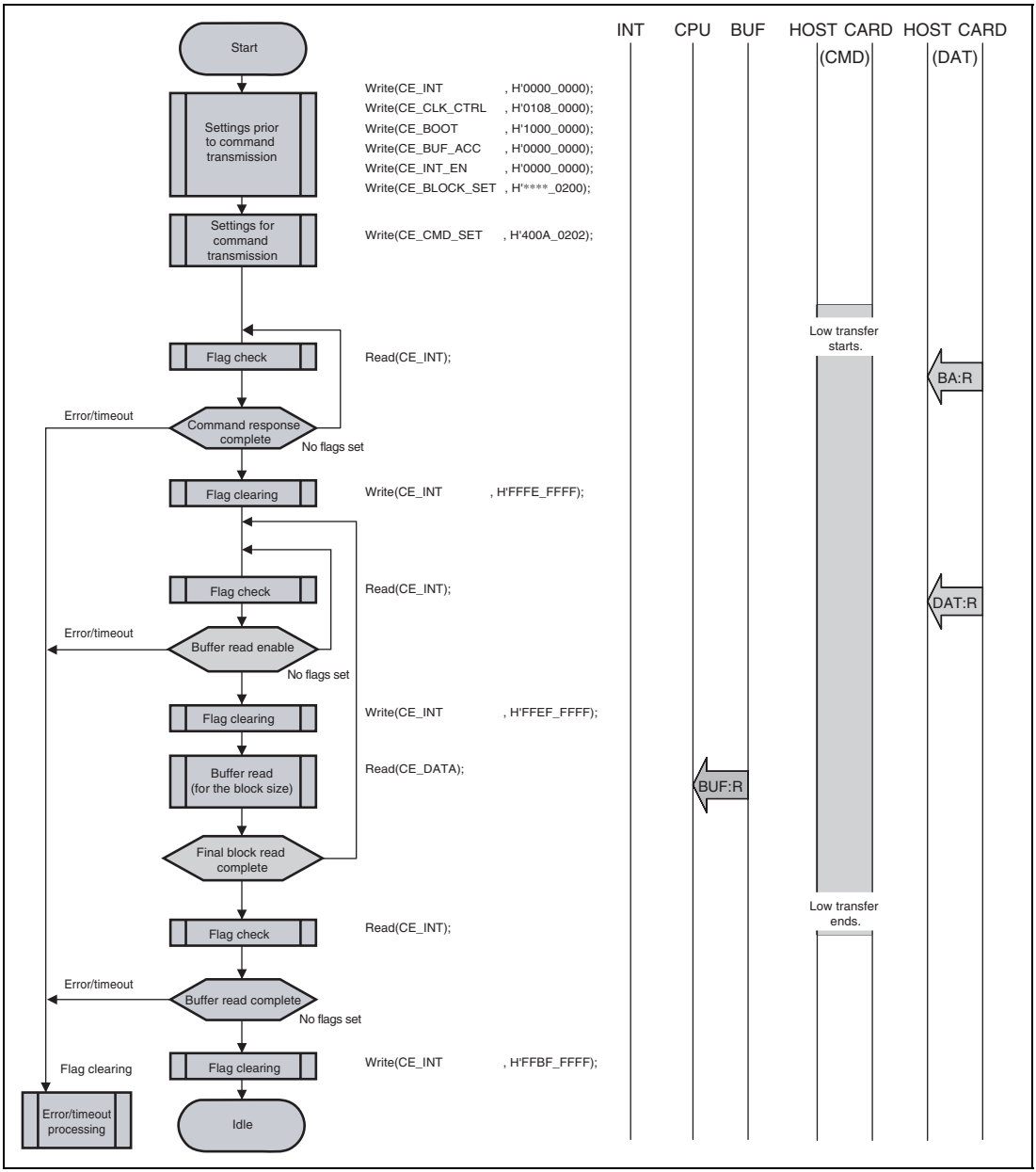


Figure 30.24 Boot Operations (with Boot Acknowledge)

30.7.12 Forcible Termination

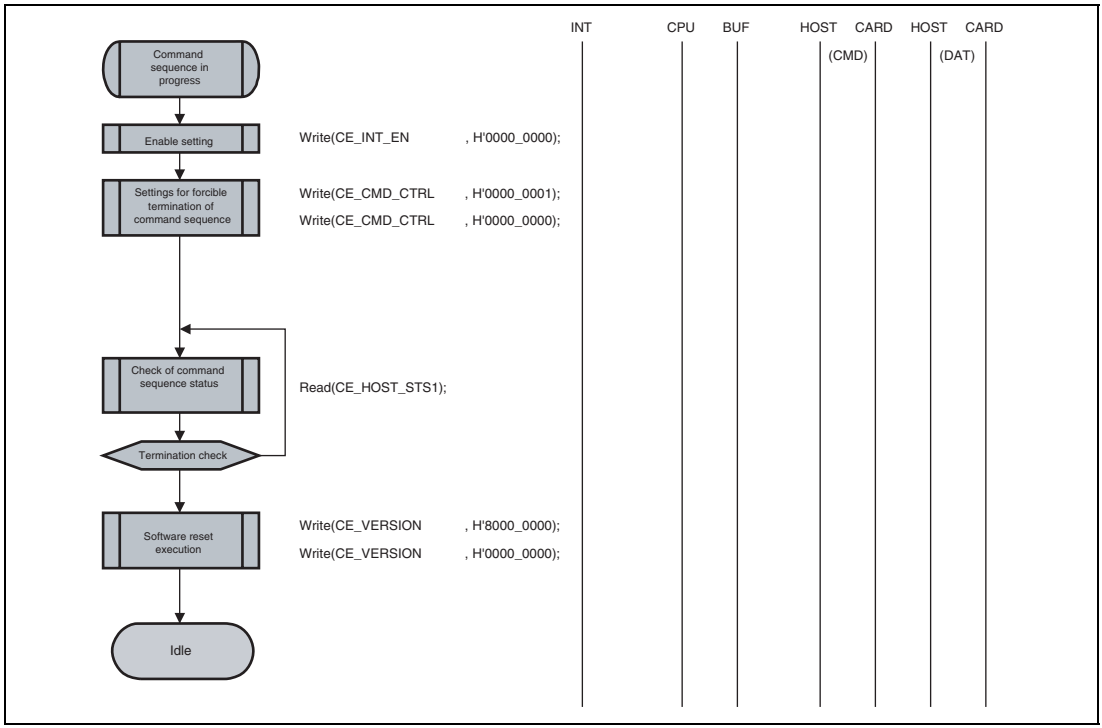


Figure 30.25 Forcible Termination

30.7.13 Command Transmission → Response Reception (with Response Busy and CCS)

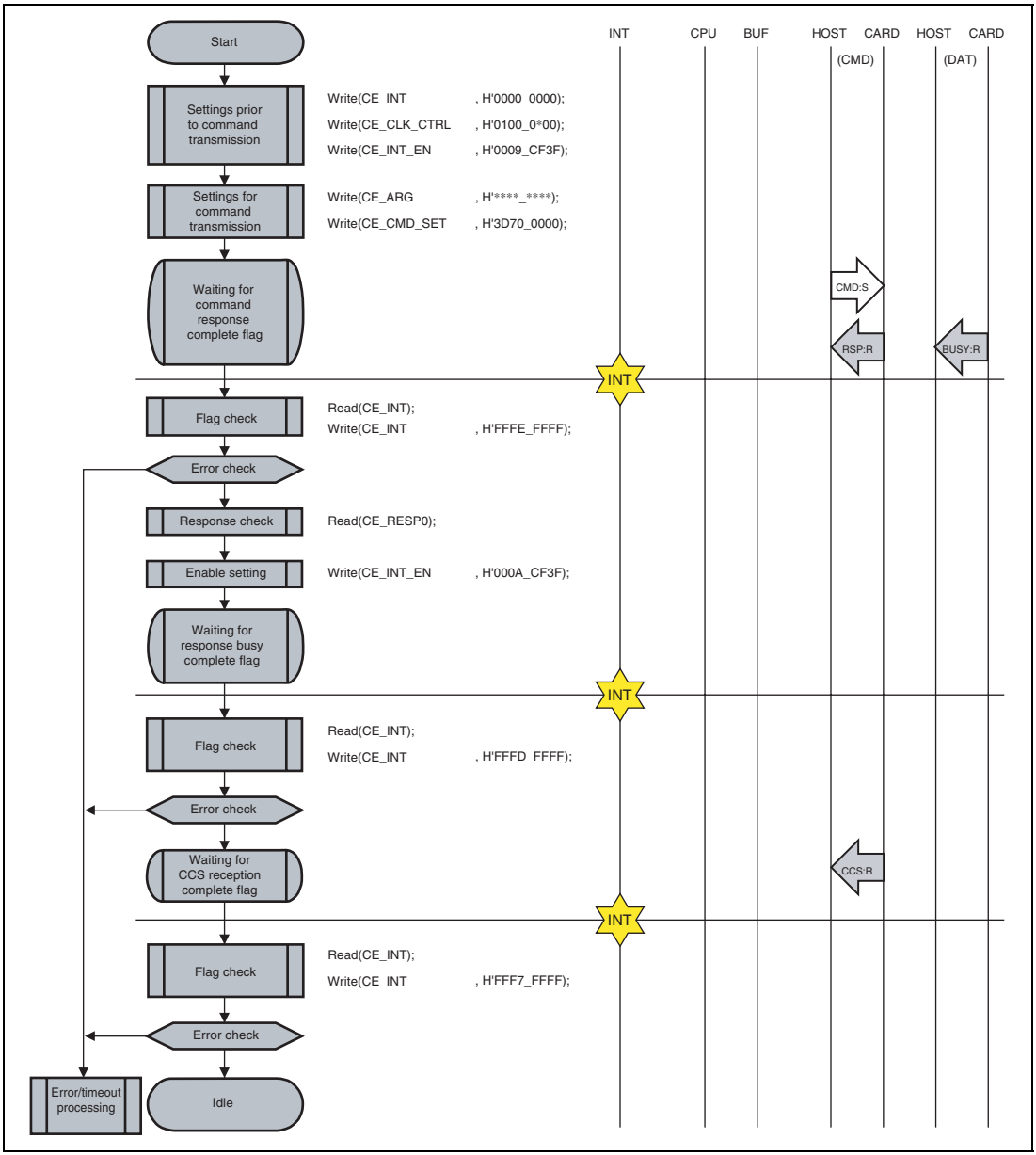


Figure 30.26 Command Transmission → Response Reception (with Response Busy and CCS) (CMD61)

30.7.14 Multi-Block Read (with CCS)

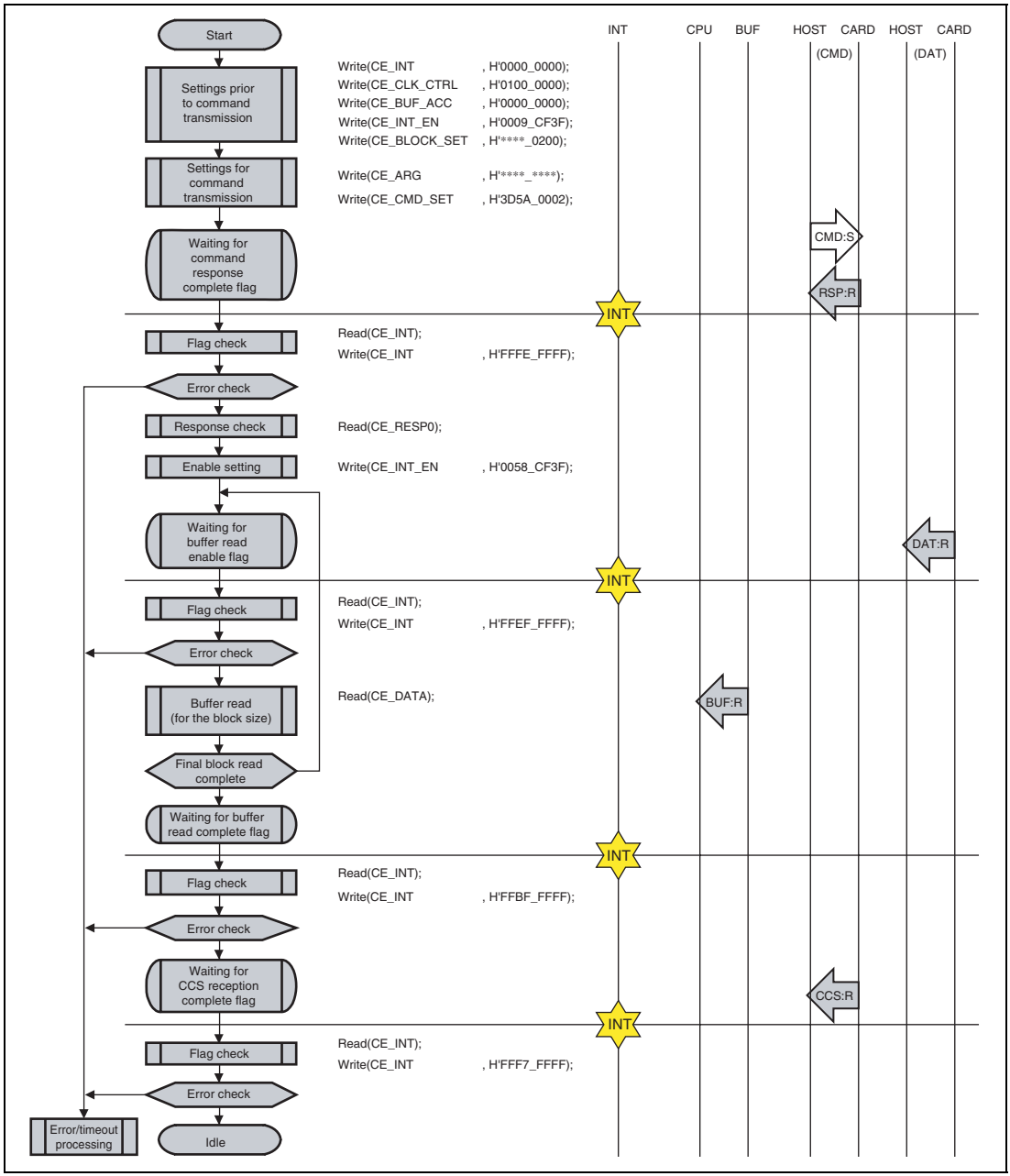


Figure 30.27 Multi-Block Read (with CCS) (CMD61)

30.7.15 Multi-Block Write (with Response Busy and CCS)

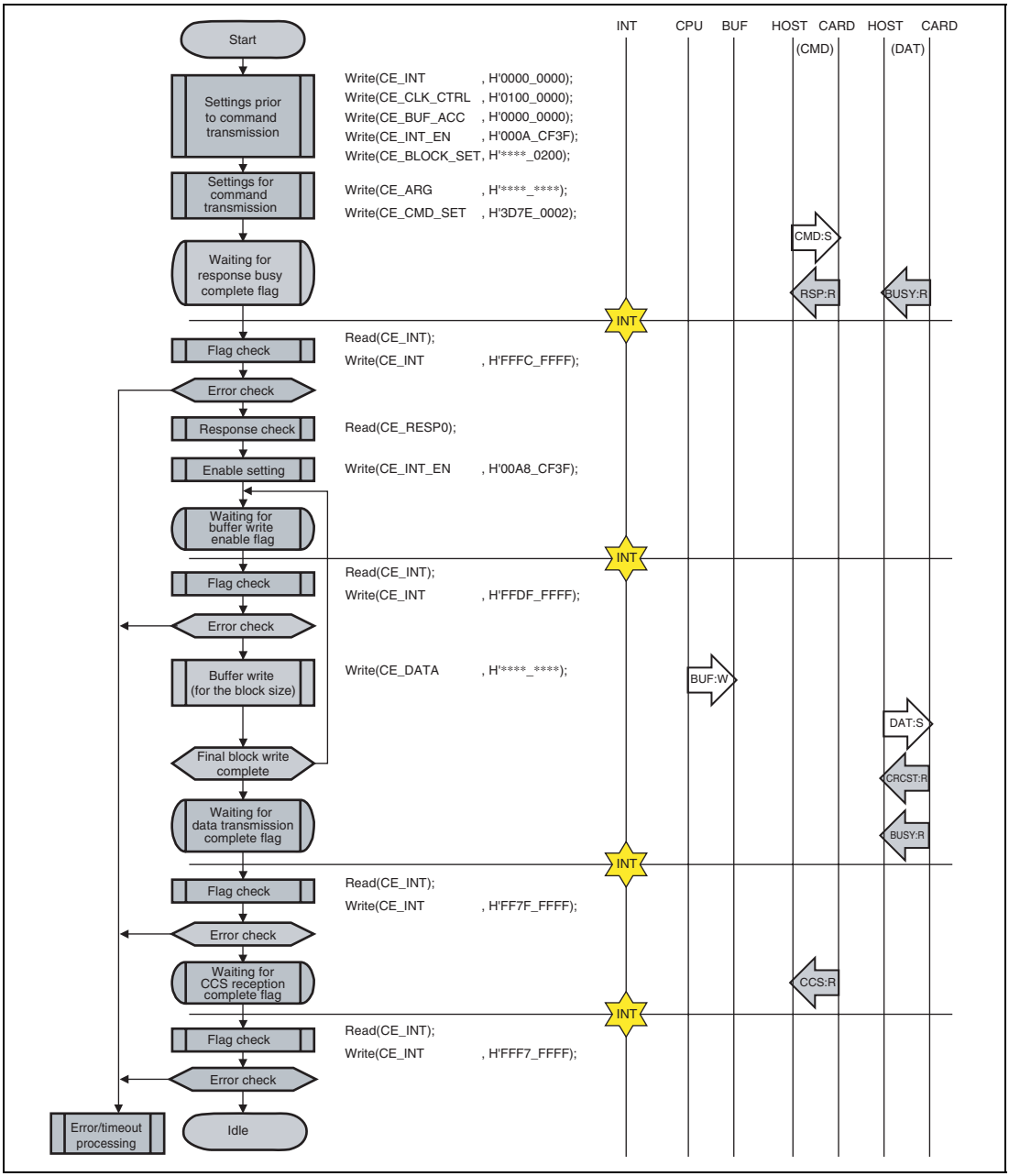


Figure 30.28 Multi-Block Write (with Response Busy and CCS) (CMD61)

30.7.16 Forcible Termination → Transmission of CCSD

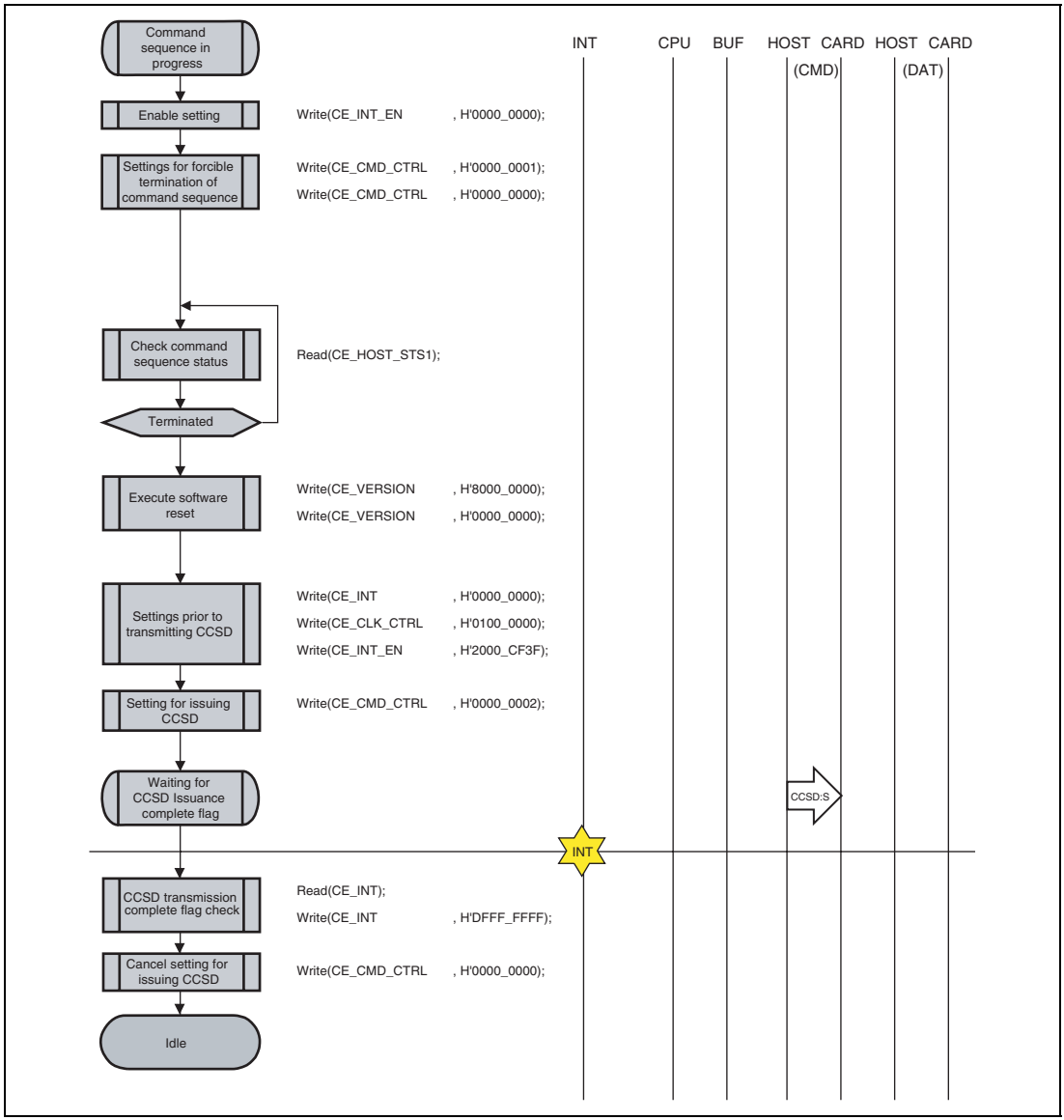


Figure 30.29 Forcible Termination → Transmission of CCSD

30.7.17 Setting Values of CE_CMD_SET

Tables 30.4 to 30.6 list the setting values required to issue commands.

Table 30.4 Setting Values of CE_CMD_SET 1 (Command Sequence other than for MMC and Boot Operations)

Command	Response	CE_CMD_SET																					Remarks
		Reserved	BOOT	CMD[5:0]	RTYP[1:0]	RBSY	CCSEN	WDAT	DWEN	CMLTE	CMD12EN	RIDXC[1:0]	RCRC7C[1:0]	Reserved	CRC16C	BOOTACK	CRCSTE	TBIT	OPDM	CCSH	Reserved[2:0]	DATW[1:0]	
CMD0	—	0	0	000000	00	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	000	00	
CMD1	R3	0	0	000001	01	0	0	0	0	0	0	01	01	0	0	0	0	0	0	0	000	00	
CMD2	R2	0	0	000010	10	0	0	0	0	0	0	01	10	0	0	0	0	0	0	0	000	00	
CMD3	R1	0	0	000011	01	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	000	00	
CMD4	—	0	0	000100	00	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	000	00	
CMD5	R1b	0	0	000101	01	1	0	0	0	0	0	00	00	0	0	0	0	0	0	0	000	00	
CMD6	R1b	0	0	000110	01	1	0	0	0	0	0	00	00	0	0	0	0	0	0	0	000	00	
CMD7	R1	0	0	000111	01	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	000	00	
	R1b	0	0	000111	01	1	0	0	0	0	0	00	00	0	0	0	0	0	0	0	000	00	
CMD8	R1	0	0	001000	01	0	0	1	0	0	0	00	00	0	0	0	0	0	0	0	000	**	
CMD9	R2	0	0	001001	10	0	0	0	0	0	0	01	10	0	0	0	0	0	0	0	000	00	
CMD10	R2	0	0	001010	10	0	0	0	0	0	0	01	10	0	0	0	0	0	0	0	000	00	
CMD12	R1	0	0	001100	01	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	000	00	
	R1b	0	0	001100	01	1	0	0	0	0	0	00	00	0	0	0	0	0	0	0	000	00	
CMD13	R1	0	0	001101	01	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	000	00	
CMD14	R1	0	0	001110	01	0	0	1	0	0	0	00	00	0	1	0	0	0	0	0	000	**	
CMD15	—	0	0	001111	00	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	000	00	
CMD16	R1	0	0	010000	01	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	000	00	
CMD17	R1	0	0	010001	01	0	0	1	0	0	0	00	00	0	0	0	0	0	0	0	000	**	
CMD18	R1	0	0	010010	01	0	0	1	0	1	0	00	00	0	0	0	0	0	0	0	000	**	Pre-defined
	R1	0	0	010010	01	0	0	1	0	1	1	00	00	0	0	0	0	0	0	0	000	**	Open-ended
CMD19	R1	0	0	010011	01	0	0	1	1	0	0	00	00	0	0	0	1	0	0	0	000	**	
CMD23	R1	0	0	010111	01	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	000	00	
CMD24	R1	0	0	011000	01	0	0	1	1	0	0	00	00	0	0	0	0	0	0	0	000	**	
CMD25	R1	0	0	011001	01	0	0	1	1	1	0	00	00	0	0	0	0	0	0	0	000	**	Pre-defined
	R1	0	0	011001	01	0	0	1	1	1	1	00	00	0	0	0	0	0	0	0	000	**	Open-ended

Command	Response	CE_CMD_SET																				Remarks
		Reserved	BOOT	CMD[5:0]	RTYP[1:0]	RBSY	CCSEN	WDAT	DWEN	CMLTE	CMD12EN	RIDXC[1:0]	RCRC7C[1:0]	Reserved	CRC16C	BOOTACK	CRCSTE	TBIT	OPDM	CCSH	Reserved[2:0]	DATW[1:0]	
CMD26	R1	0	0	011010	01	0	0	1	1	0	0	00	00	0	0	0	0	0	0	0	000	**	
CMD27	R1	0	0	011011	01	0	0	1	1	0	0	00	00	0	0	0	0	0	0	0	000	**	
CMD28	R1b	0	0	011100	01	1	0	0	0	0	0	00	00	0	0	0	0	0	0	0	000	00	
CMD29	R1b	0	0	011101	01	1	0	0	0	0	0	00	00	0	0	0	0	0	0	0	000	00	
CMD30	R1	0	0	011110	01	0	0	1	0	0	0	00	00	0	0	0	0	0	0	0	000	**	
CMD35	R1	0	0	100011	01	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	000	00	
CMD36	R1	0	0	100100	01	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	000	00	
CMD38	R1b	0	0	100110	01	1	0	0	0	0	0	00	00	0	0	0	0	0	0	0	000	00	
CMD39	R4	0	0	100111	01	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	000	00	
CMD40	R5	0	0	101000	01	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	000	00	Send CMD
	R5	0	0	101000	01	0	0	0	0	0	0	00	00	0	0	0	0	1	1	0	000	00	Send RSP
CMD42	R1	0	0	101010	01	0	0	1	1	0	0	00	00	0	0	0	0	0	0	0	000	**	
CMD55	R1	0	0	110111	01	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	000	00	
CMD56	R1	0	0	111000	01	0	0	1	0	0	0	00	00	0	0	0	0	0	0	0	000	**	Read
	R1	0	0	111000	01	0	0	1	1	0	0	00	00	0	0	0	0	0	0	0	000	**	Write

Note: This module does not support CMD11 and CMD20.

Table 30.5 Setting Values of CE_CMD_SET 2 (Command Sequence for MMC and Boot Operations)

—	Boot Acknowledge Reception	CE_CMD_SET																				Remarks
		Reserved	BOOT	CMD[5:0]	RTYP[1:0]	RBSY	CCSEN	WDAT	DWEN	CMLTE	CMD12EN	RIDXC[1:0]	RCRC7C[1:0]	Reserved	CRC16C	BOOTACK	CRCSTE	TBIT	OPDM	CCSH	Reserved[2:0]	DATW[1:0]	
—	Enable	0	1	000000	00	0	0	1	0	1	0	00	00	0	0	1	0	0	0	0	000	**	
	Disable	0	1	000000	00	0	0	1	0	1	0	00	00	0	0	0	0	0	0	0	000	**	

Table 30.6 Setting Values of CE_CMD_SET 3 (Command Sequence for CE-ATA)

Command	Response	CE_CMD_SET																					Remarks
		Reserved	BOOT	CMD[5:0]	RTYP[1:0]	RBSY	CCSEN	WDAT	DWEN	CMLTE	CMD12EN	RIDXC[1:0]	RCRC7C[1:0]	Reserved	CRC16C	BOOTACK	CRCSTE	TBIT	OPDM	CCSH	Reserved[2:0]	DATW[1:0]	
CMD60	R1	0	0	111100	01	0	0	1	0	0	0	00	00	0	0	0	0	0	0	0	000	**	Read
	R1b	0	0	111100	01	1	0	1	1	0	0	00	00	0	0	0	0	0	0	0	000	**	Write
CMD61	R1b	0	0	111101	01	1	1	0	0	0	0	00	00	0	0	0	0	0	0	*	000	**	No data
	R1	0	0	111101	01	0	1	1	0	1	0	00	00	0	0	0	0	0	0	*	000	**	Read
	R1b	0	0	111101	01	1	1	1	1	1	0	00	00	0	0	0	0	0	0	*	000	**	Write

30.8 Usage Notes

30.8.1 Timing of Response Busy Output

The MMC standard (MultiMediaCard System Specification) stipulates the timing (Nst) of response busy (R1b) output as two cycles after a command has been issued. However, the timing of response busy output for some devices (cards and embedded flash memory) has been found to be one cycle late.

In accord with the MMC standard, this MMC has been designed to detect response busy output two cycles after a command. Therefore, when the interface is connected with a device as described above, a next command might be issued even though the device is busy (since determination is on completion of response busy). Therefore, the device may be incapable of accepting a command issued at this time.

In order to avoid this problem after the completion of a command sequence that waits until response busy output (CMD12, CMD38, and so on), confirm that the current_state value of the card status becomes the expected state (the device is not busy) by issuing CMD13.

30.8.2 Forcible Termination

Forcible termination of a command sequence may fail depending on the timing set in the CE_CMD_CTRL register for the BREAK bit. Accordingly, follow one of the procedures below if a command sequence is to be forcibly terminated.

1. To forcibly terminate a command sequence, execute a software reset instead of using the BREAK bit in CE_CMD_CTRL.
2. To forcibly terminate a command sequence in the following states after having issued an R1b response command, execute a software reset instead of using the BREAK bit in CE_CMD_CTRL.
   - Response busy timeout
   - Abnormal response value

Section 31 NAND Flash Memory Controller (FLCTL)

The NAND flash memory controller provides interfaces for an external NAND-type flash memory.

31.1 Features

(1) NAND-Type Flash Memory Interface

- Interface directly connectable to NAND-type flash memory
- Read or write in sector units (512 + 16 bytes)
- Read or write in byte units
- Supports large-block (2048 + 64 bytes) flash memory*
- Supports addresses for 2 Gbits and more by extension to 5-byte addresses

Note: * This module handles 512 + 16 bytes as a sector. For products with 2048 + 64 byte-pages, this module divides a page into 512 +16 bytes units (i.e. four sectors per page) for processing.

(2) Access Modes: This module can select one of the following two access modes.

- Command access mode: Performs an access by specifying a command to be issued from this module to flash memory, address, and data size to be input or output.
- Sector access mode: Performs a read or write in sector units by specifying a sector address. By specifying the number of sectors, the continuous physical sectors can be read or written.

(3) Sectors and Control Codes

- A sector is the basic unit of access and comprised of 512-byte data and 16-byte control code fields.
- User information can be written to any part of the control code field.

(4) Data Error

- When a program error or erase error occurs, the error is reflected on the error source flags. Interrupts for each source can be specified.

(5) Data Transfer FIFO and Data Register

- The 224-byte data FIFO register (FLDTFIFO) is incorporated for data transfer of flash memory.
- The 32-byte control code FIFO register (FLECFIFO) is incorporated for data transfer of control code.

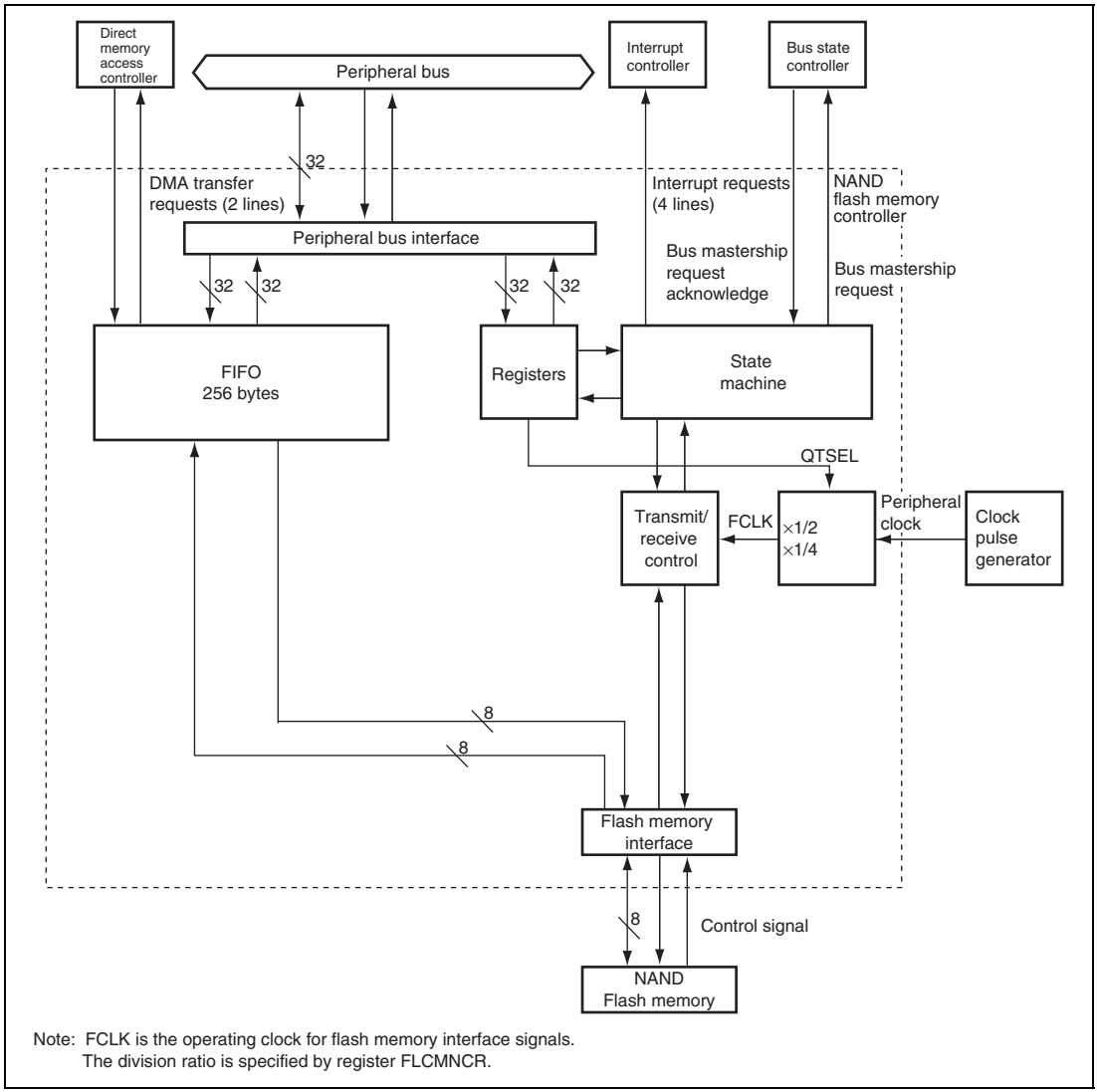
(6) DMA Transfer

- By individually specifying the destinations of data and control code of flash memory to the direct memory access controller, data and control code can be sent to different areas.

(7) Access Time

- The operating clock (FCLK) on the pins for the NAND-type flash memory is generated by dividing the peripheral clock (Pϕ). The division ratio can be specified by the QTSEL bit in the common control register (FLCMNCR).
- Before changing the clock pulse generator configuration, this module must be placed in a module stop state.
- In NAND-type flash memory, the $\overline{\text{FRE}}$ and $\overline{\text{FWE}}$ pins operate at the frequency of FCLK. The operating frequency must be specified within the maximum operating frequency of memory to be connected.

Figure 31.1 shows a block diagram.



Note: FCLK is the operating clock for flash memory interface signals.
The division ratio is specified by register FLCMNCR.

Figure 31.1 Block Diagram

31.2 Input/Output Pins

The pin configuration of is listed in table 31.1.

Table 31.1 Pin Configuration

		Corresponding Flash Memory Pin	
Pin Name	**I/O**	**NAND Type**	**Function**
$\overline{\text{FCE}}$	Output	$\overline{\text{CE}}$	Flash Memory Chip Enable Enables flash memory connected to this LSI.
NAF7 to NAF0	I/O	I/O7 to I/O0	Flash Memory Data I/O pins for command, address, and data.
FCLE	Output	CLE	Flash Memory Command Latch Enable Asserted when a command is output.
FALE	Output	ALE	Flash Memory Address Latch Enable Asserted when an address is output and negated when data is input or output.
$\overline{\text{FRE}}$	Output	$\overline{\text{RE}}$	Flash Memory Read Enable Reads data at the falling edge of $\overline{\text{RE}}$.
$\overline{\text{FWE}}$	Output	$\overline{\text{WE}}$	Flash Memory Write Enable Flash memory latches a command, address, and data at the rising edge of $\overline{\text{WE}}$.
FRB	Input	R/$\overline{\text{B}}$	Flash Memory Ready/Busy Indicates ready state at high level; indicates busy state at low level.
—*	—	$\overline{\text{WP}}$	Write Protect/Reset When this pin goes low, erroneous erasure or programming at power on or off can be prevented.
—*	—	$\overline{\text{SE}}$	Spare Area Enable Used to access spare area. This pin must be fixed at low in sector access mode.

Note: * Not supported in this LSI.

31.3 Register Descriptions

Table 31.2 (1) shows the register configuration.
Do not write to any addresses other than listed below. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than those listed below are undefined.

Table 31.2 (1) Register Configuration

Register Name	Abbreviation	R/W	Address	Access Size
Common control register	FLCMNCR	R/W	H'FFFCA000	32
Command control register	FLCMDCR	R/W	H'FFFCA004	32
Command code register	FLCMCDR	R/W	H'FFFCA008	32
Address register	FLADR	R/W	H'FFFCA00C	32
Address register 2	FLADR2	R/W	H'FFFCA03C	32
Data register	FLDATAR	R/W	H'FFFCA010	32
Data counter register	FLDTCNTR	R/W	H'FFFCA014	32
Interrupt DMA control register	FLINTDMACR	R/W	H'FFFCA018	32
Ready busy timeout setting register	FLBSYTMR	R/W	H'FFFCA01C	32
Ready busy timeout counter	FLBSYCNT	R	H'FFFCA020	32
Data FIFO register	FLDTFIFO	R/W	H'FFFCA050	32
Control code FIFO register	FLECFIFO	R/W	H'FFFCA060	32
Transfer control register	FLTRCR	R/W	H'FFFCA02C	8
MODE Register	FLMODE	R/W	H'FFFCA038	32

Table 31.2 (2) Register State in Each Operating Mode

Abbreviation	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
FLCMNCR	H'00100001	H'00100001	Retained	Retained	Retained	Initialized
FLCMDCR	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
FLCMCDR	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
FLADR	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
FLADR2	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
FLDATAR	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
FLDTCNTR	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
FLINTDMACR	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
FLBSYTMR	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
FLBSYCNT	H'00000000	H'00000000	Retained	Retained	Retained	Initialized
FLDTFIFO	H'xxxxxxxx	H'xxxxxxxx	Retained	Retained	Retained	Initialized
FLECFIFO	H'xxxxxxxx	H'xxxxxxxx	Retained	Retained	Retained	Initialized
FLTRCR	H'00	H'00	Retained	Retained	Retained	Initialized
FLMODE	H'00000000	H'00000000	Retained	Retained	Retained	Initialized

Note: Initialized: The value is the one written in "Power-on Reset".

31.3.1 Common Control Register (FLCMNCR)

FLCMNCR is a 32-bit readable/writable register that specifies access mode, and other items.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	SNAND	QT SEL	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	ACM[1:0]		NAND WF	-	-	-	-	-	CE	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
18	SNAND	0	R/W	Large-Capacity NAND Flash Memory Select This bit is used to specify 1-Gbit or larger NAND flash memory with the page configuration of 2048 + 64 bytes. 0: When flash memory with the page configuration of 512 + 16 bytes is used. 1: When NAND flash memory with the page configuration of 2048 + 64 is used.
17	QTSEL	0	R/W	Select Dividing Rates for Flash Clock Selects the dividing rate of clock FCLK in the flash memory. 0: Divides a clock (Pϕ) provided from the clock pulse generator by two and uses it as FCLK. 1: Divides a clock (Pϕ) provided from the clock pulse generator by four and uses it as FCLK.

Bit	Bit Name	Initial Value	R/W	Description
16 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11, 10	ACM[1:0]	00	R/W	Access Mode Specification 1 and 0 Specify access mode. 00: Command access mode 01: Sector access mode 10: Setting prohibited 11: Setting prohibited
9	NANDWF	0	R/W	NAND Wait Insertion Operation 0: Performs address or data input/output in one FCLK cycle 1: Performs address or data input/output in two FCLK cycles
8 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	CE	0	R/W	Chip Enable 0: Disables the chip (Outputs high level to the $\overline{\text{FCE}}$ pin) 1: Enables the chip (Outputs low level to the $\overline{\text{FCE}}$ pin)
2, 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.

Note: Only the first and second commands are supported when the SNAND bit of the common control register (FLCMNCR) is in use regardless of the setting of the DOCMD1 and DOCMD2 bits of the command control register (FLCMDCR).
Set the SNAND bit to 0 if no commands or only a first command is to be issued.

31.3.2 Command Control Register (FLCMDCR)

FLCMDCR is a 32-bit readable/writable register that issues a command in command access mode, specifies address issue, and specifies source or destination of data transfer. In sector access mode, FLCMDCR specifies the number of sector transfers.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ADR CNT2	SCTCNT[19:16]				ADR MD	CDS RC	DOSR	-	-	SEL RW	DOA DR	ADRCNT[1:0]		DOC MD2	DOC MD1
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SCTCNT[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	ADRCNT2	0	R	Address Issue Byte Count Specification 2 Specifies the number of bytes for the address data to be issued in address stage. This bit is used together with ADRCNT[1:0]. 0: Issue the address of byte count, specified by ADRCNT[1:0]. 1: Issue 5-byte address. ADRCNT[1:0] should be set to 00.
30 to 27	SCTCNT [19:16]	0000	R/W	Sector Transfer Count Specification [19:16] These bits are extended bits of the sector transfer count specification bits (SCTCNT) 15 to 0. SCTCNT[19:16] and SCTCNT[15:0] are used together to operate as SCTCNT[19:0], the 20-bit counter.
26	ADRMD	0	R/W	Sector Access Address Specification This bit is invalid in command access mode. This bit is valid only in sector access mode. 0: The value of the address register is handled as a sector address. Use this value usually in sector access. 1: The value of the address register is output as the address of flash memory. Note: Clear this bit to 0 in continuous sector access.

Bit	Bit Name	Initial Value	R/W	Description
25	CDSRC	0	R/W	Data Buffer Specification Specifies the data buffer to be read from or written to in the data stage in command access mode. 0: Specifies FLDATAR as the data buffer. 1: Specifies FLDTFIFO as the data buffer.
24	DOSR	0	R/W	Status Read Check Specifies whether or not the status read is performed after the second command has been issued in command access mode. 0: Performs no status read 1: Performs status read
23, 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21	SELRW	0	R/W	Data Read/Write Specification Specifies the direction of read or write in data stage. 0: Read 1: Write
20	DOADR	0	R/W	Address Stage Execution Specification Specifies whether or not the address stage is executed in command access mode. 0: Performs no address stage 1: Performs address stage
19, 18	ADRCNT [1:0]	00	R/W	Address Issue Byte Count Specification [1:0] Specify the number of bytes for the address data to be issued in address stage. 00: Issue 1-byte address 01: Issue 2-byte address 10: Issue 3-byte address 11: Issue 4-byte address

Bit	Bit Name	Initial Value	R/W	Description
17	DOCMD2	0	R/W	Second Command Stage Execution Specification Specifies whether or not the second command stage is executed in command access mode. 0: Does not execute the second command stage 1: Executes the second command stage
16	DOCMD1	0	R/W	First Command Stage Execution Specification Specifies whether or not the first command stage is executed in command access mode. 0: Does not execute the first command stage 1: Executes the first command stage
15 to 0	SCTCNT [15:0]	H'0000	R/W	Sector Transfer Count Specification [15:0] Specify the number of sectors to be read continuously in sector access mode. These bits are counted down for each sector transfer end and stop when they reach 0. These bits are used together with SCTCNT[19:16]. In command access mode, these bits are H'0 0001.

31.3.3 Command Code Register (FLCMCDR)

FLCMCDR is a 32-bit readable/writable register that specifies a command to be issued in command access or sector access.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMD2[7:0]								CMD1[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 8	CMD2[7:0]	H'00	R/W	Second Command Data Specify a command code to be issued in the second command stage.
7 to 0	CMD1[7:0]	H'00	R/W	First Command Data Specify a command code to be issued in the first command stage.

31.3.4 Address Register (FLADR)

FLADR is a 32-bit readable/writable register that specifies the value to be output as an address.

The address of the size specified by ADRCNT[1:0] in the command control register is output sequentially from ADR1 in byte units. By the sector access address specification bit (ADRMD) of the command control register, it is possible to specify whether the sector number set in the address data bits is converted into an address to be output to the flash memory.

- When ADRMD = 1

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ADR4[7:0]								ADR3[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADR2[7:0]								ADR1[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	ADR4[7:0]	H'00	R/W	Fourth Address Data Specify 4th data to be output to flash memory as an address when ADRMD = 1.
23 to 16	ADR3[7:0]	H'00	R/W	Third Address Data Specify 3rd data to be output to flash memory as an address when ADRMD = 1.
15 to 8	ADR2[7:0]	H'00	R/W	Second Address Data Specify 2nd data to be output to flash memory as an address when ADRMD = 1.
7 to 0	ADR1[7:0]	H'00	R/W	First Address Data Specify 1st data to be output to flash memory as an address when ADRMD = 1.

- When ADRMD = 0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	ADR[25:16]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADR[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25 to 0	ADR[25:0]	H'0000000	R/W	Sector Address Specification Specify a sector number to be accessed when ADRMD = 0. The sector number is converted into an address and is output to flash memory. When the ADRCNT2 bit in FLCMDCR = 1, the ADR[25:0] bits are valid. When the ADRCNT2 bit in FLCMDCR = 0, the ADR[17:0] bits are valid. See figure 31.11 for details. • Large-block products (2048 + 64 bytes) ADR[25:2] specifies the page address and ADR[1:0] specifies the column address in sector units. ADR[1:0] = 00: 0th byte (sector 0) ADR[1:0] = 01: (512 + 16)th byte (sector 1) ADR[1:0] = 00: (1024 + 32)th byte (sector 2) ADR[1:0] = 00: (1536 + 48)th byte (sector 3) • Small-block products (512 + 16 bytes) Only the page address can be specified.

31.3.5 Address Register 2 (FLADR2)

FLADR2 is a 32-bit readable/writable register, and is valid when the ADRCNT2 bit in FLCMDCR is set to 1. FLADR2 specifies an address to be output in command access mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	ADR5[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	ADR5[7:0]	H'00	R/W	Fifth Address Data Specify 5th data to be output to flash memory as an address when ADRMD = 1.

31.3.6 Data Counter Register (FLDTCNTR)

FLDTCNTR is a 32-bit readable/writable register that specifies the number of bytes to be read or written in command access mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECFLW[7:0]								DTFLW[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	DTCNT[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	ECFLW [7:0]	H'00	R	FLECFIFO Access Count Specify the number of longwords in FLECFIFO to be read or written. These bit values are used when the CPU reads from or writes to FLECFIFO. In FLECFIFO read, these bits specify the number of longwords of the data that can be read from FLECFIFO. In FLECFIFO write, these bits specify the number of longwords of unoccupied area that can be written in FLECFIFO.
23 to 16	DTFLW [7:0]	H'00	R	FLDTFIFO Access Count Specify the number of longwords in FLDTFIFO to be read or written. These bit values are used when the CPU reads from or writes to FLDTFIFO. In FLDTFIFO read, these bits specify the number of longwords of the data that can be read from FLDTFIFO. In FLDTFIFO write, these bits specify the number of longwords of unoccupied area that can be written in FLDTFIFO.
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	DTCNT [11:0]	H'000	R/W	Data Count Specification Specify the number of bytes of data to be read or written in command access mode. (Up to 2048 + 64 bytes can be specified.)

31.3.7 Data Register (FLDATAR)

FLDATAR is a 32-bit readable/writable register. It stores input/output data used when 0 is written to the CDSRC bit in FLCMDCR in command access mode. FLDATAR cannot be used for reading or writing of five or more bytes of contiguous data.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DT4[7:0]								DT3[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DT2[7:0]								DT1[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	DT4[7:0]	H'00	R/W	Fourth Data Specify the 4th data to be input or output via the NAF7 to NAF0 pins. In write: Specify write data In read: Store read data
23 to 16	DT3[7:0]	H'00	R/W	Third Data Specify the 3rd data to be input or output via the NAF7 to NAF0 pins. In write: Specify write data In read: Store read data
15 to 8	DT2[7:0]	H'00	R/W	Second Data Specify the 2nd data to be input or output via the NAF7 to NAF0 pins. In write: Specify write data In read: Store read data
7 to 0	DT1[7:0]	H'00	R/W	First Data Specify the 1st data to be input or output via the NAF7 to NAF0 pins. In write: Specify write data In read: Store read data

31.3.8 Interrupt DMA Control Register (FLINTDMACR)

FLINTDMACR is a 32-bit readable/writable register that enables or disables DMA transfer requests or interrupts. A transfer request from this module to the direct memory access controller is issued after each access mode has been started.

Bits 8 to 5 are the flag bits that indicate various errors occurred in flash memory access and whether there is a transfer request from the FIFO. Only 0 can be written to these bits. To clear a flag, write 0 to the target flag bit and 1 to the other flag bits.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	FIFOTRG [1:0]		AC1 CLR	AC0 CLR	DREQ1 EN	DREQ0 EN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	ST ERB	BTO ERB	TRR EQF1	TRR EQF0	STER INTE	RBER INTE	TE INTE	TR INTE1	TR INTE0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/W	R/W	R/W	R/W	R/W

Note: * Only 0 can be written to these bits.

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
21, 20	FIFOTRG [1:0]	00	R/W	FIFO Trigger Setting Specify the condition (the byte number) for generation of FLDTFIFO and FLECFIFO transfer requests. • In flash-memory read Issue an interrupt to the CPU or issue a DMA transfer request when FLDTFIFO (FLECFIFO) stores the following number of bytes or more: 00: 4 (4) 01: 16 (16) 10: 128 (4) 11: 128 (16) • In flash-memory programming Issue an interrupt to the CPU or issue a DMA transfer request when FLDTFIFO (FLECFIFO) has the following empty area of bytes or more: 00: 4 (4) 01: 16 (16) 10: 128 (4) 11: 128 (16) Note: When DMA transfer is to be performed, setting other than “00” is prohibited.
19	AC1CLR	0	R/W	FLECFIFO Clear Clears FLECFIFO. 0: Retains the FLECFIFO value. In flash-memory access, this bit should be cleared to 0. 1: Clears FLECFIFO. After FLECFIFO has been cleared, this bit should be cleared to 0.
18	AC0CLR	0	R/W	FLDTFIFO Clear Clears FLDTFIFO. 0: Retains the FLDTFIFO value. In flash-memory access, this bit should be cleared to 0. 1: Clears FLDTFIFO. After FLDTFIFO has been cleared, this bit should be cleared to 0.

Bit	Bit Name	Initial Value	R/W	Description
17	DREQ1EN	0	R/W	FLECFIFODMA Request Enable Enables or disables the DMA transfer request issued from FLECFIFO. 0: Disables the DMA transfer request issued from FLECFIFO 1: Enables the DMA transfer request issued from FLECFIFO
16	DREQ0EN	0	R/W	FLDTFIFODMA Request Enable Enables or disables the DMA transfer request issued from FLDTFIFO. 0: Disables the DMA transfer request issued from the FLDTFIFO 1: Enables the DMA transfer request issued from the FLDTFIFO
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	STERB	0	R/(W)*	Status Error Indicates the result of status read. This bit is set to 1 if the specific bit in the bits STAT[7:0] in FLBSYCNT is set to 1 in status read. This bit is a flag. 1 cannot be written to this bit. Only 0 can be written to clear the flag. 0: Indicates that no status error occurs (the specific bit in the bits STAT[7:0] in FLBSYCNT is 0.) 1: Indicates that a status error occurs For details on the specific bit in STAT7 to STAT0 bits, see section 31.4.6, Status Read.
7	BTOERB	0	R/(W)*	R/$\overline{B}$ Timeout Error This bit is set to 1 if an R/$\overline{B}$ timeout error occurs (the bits RBTIMCNT[19:0] in FLBSYCNT are decremented to 0). This bit is a flag. 1 cannot be written to this bit. Only 0 can be written to clear the flag. 0: Indicates that no R/$\overline{B}$ timeout error occurs 1: Indicates that an R/$\overline{B}$ timeout error occurs

Bit	Bit Name	Initial Value	R/W	Description
6	TRREQF1	0	R/(W)*	FLECFIFO Transfer Request Flag Indicates that a transfer request is issued from FLECFIFO. This bit is a flag. 1 cannot be written to this bit. Only 0 can be written to clear the flag. 0: Indicates that no transfer request is issued from FLECFIFO 1: Indicates that a transfer request is issued from FLECFIFO
5	TRREQF0	0	R/(W)*	FLDTFIFO Transfer Request Flag Indicates that a transfer request is issued from FLDTFIFO. This bit is a flag. 1 cannot be written to this bit. Only 0 can be written to clear the flag. 0: Indicates that no transfer request is issued from FLDTFIFO 1: Indicates that a transfer request is issued from FLDTFIFO
4	STERINTE	0	R/W	Interrupt Enable at Status Error Enables or disables an interrupt request to the CPU when a status error has occurred. 0: Disables the interrupt request to the CPU by a status error 1: Enables the interrupt request to the CPU by a status error
3	RBERINTE	0	RW	Interrupt Enable at R/$\overline{B}$ Timeout Error Enables or disables an interrupt request to the CPU when a timeout error has occurred. 0: Disables the interrupt request to the CPU by an R/$\overline{B}$ timeout error 1: Enables the interrupt request to the CPU by an R/$\overline{B}$ timeout error

Bit	Bit Name	Initial Value	R/W	Description
2	TEINTE	0	R/W	Transfer End Interrupt Enable Enables or disables an interrupt request to the CPU when a transfer has been ended (TREND bit in FLTRCR). 0: Disables the transfer end interrupt request to the CPU 1: Enables the transfer end interrupt request to the CPU
1	TRINTE1	0	R/W	FLECFIFO Transfer Request Enable to CPU Enables or disables an interrupt request to the CPU by a transfer request issued from FLECFIFO. 0: Disables an interrupt request to the CPU by a transfer request from FLECFIFO. 1: Enables an interrupt request to the CPU by a transfer request from FLECFIFO. When the DMA transfer is enabled, this bit should be cleared to 0.
0	TRINTE0	0	R/W	FLDTFIFO Transfer Request Enable to CPU Enables or disables an interrupt request to the CPU by a transfer request issued from FLDTFIFO. 0: Disables an interrupt request to the CPU by a transfer request from FLDTFIFO 1: Enables an interrupt request to the CPU by a transfer request from FLDTFIFO When the DMA transfer is enabled, this bit should be cleared to 0.

Note: * Only 0 can be written to these bits.

31.3.9 Ready Busy Timeout Setting Register (FLBSYTMR)

FLBSYTMR is a 32-bit readable/writable register that specifies the timeout time when the FRB pin is busy.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	RBTMOUT[19:16]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RBTMOUT[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
19 to 0	RBTMOUT [19:0]	H'00000	R/W	Ready Busy Timeout Specify timeout time (the number of Pϕ clocks) in busy state. When these bits are set to 0, timeout is not generated.

31.3.10 Ready Busy Timeout Counter (FLBSYCNT)

FLBSYCNT is a 32-bit read-only register.

The status of flash memory obtained by the status read is stored in the bits STAT[7:0].

The timeout time set in the bits RBTMOUT[19:0] in FLBSYTMR is copied to the bits RBTIMCNT[19:0] and counting down is started when the FRB pin is placed in a busy state. When values in the RBTIMCNT[19:0] become 0, 1 is set to the BTOERB bit in FLINTDMACR, thus notifying that a timeout error has occurred. In this case, an FLSTE interrupt request can be issued if an interrupt is enabled by the RBERINTE bit in FLINTDMACR.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	STAT[7:0]								-	-	-	-	RBTIMCNT[19:16]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RBTIMCNT[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	STAT[7:0]	H'00	R	Indicate the flash memory status obtained by the status read.
23 to 20	—	All 0	R	Reserved These bits are always read as 0.
19 to 0	RBTIMCNT[19:0]	H'00000	R	Ready Busy Timeout Counter When the FRB pin is placed in a busy state, the values of the bits RBTMOUT[19:0] in FLBSYTMR are copied to these bits. These bits are counted down while the FRB pin is busy. A timeout error occurs when these bits are decremented to 0.

31.3.11 Data FIFO Register (FLDTFIFO)

FLDTFIFO is used to read or write the data FIFO area.

In DMA transfer, this register must be specified as the destination or source.

Note that the direction of read or write specified by the SELRW bit in FLCMDCR must match that specified in this register. When changing the read/write direction, FLDTFIFO should be cleared by setting the AC0CLR bit in FLINTDMACR before use.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTFO[31:16]															
Initial value:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTFO[15:0]															
Initial value:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DTFO [31:0]	H'xxxxxxxx	R/W	Data FIFO Area Read/Write Data In write: Data in this register is written to the data FIFO area. In read: Data read from the data FIFO area is stored in this register.

31.3.12 Control Code FIFO Register (FLECFIFO)

FLECFIFO is used to read or write the control code FIFO area.

In DMA transfer, data in this register must be specified as the destination (source).

Note that the direction of read or write specified by the SELRW bit in FLCMDCR must match that specified in this register. When changing the read/write direction, FLECFIFO should be cleared by setting the AC1CLR bit in FLINTDMACR before use.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECFO[31:16]															
Initial value:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECFO[15:0]															
Initial value:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ECFO [31:0]	H'xxxxxxxx	R/W	Control Code FIFO Area Read/Write Data In write: Data in this register is written to the control code FIFO area. In read: Data read from the control code FIFO area is stored in this register.

31.3.13 Transfer Control Register (FLTRCR)

Setting the TRSTRT bit to 1 initiates access to flash memory. Access completion can be checked by the TREND bit. During the transfer (from when the TRSTRT bit is set to 1 until the TREND bit is set to 1), the processing should not be forcibly ended (by setting the TRSTRT bit to 0).

When reading from flash memory, TREND is set when reading from flash memory have been finished. However, if there is any read data remaining in the FIFO, the processing should not be forcibly ended until all data has been read from the FIFO. While this module has the external bus mastership and transfer is in progress, the SLEEP instruction should not be executed until the TREND bit is set and transfer is completed.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	TR STAT	TR END	TR STRT
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	TRSTAT	0	R	Transfer State Indicates that this module has acquired the external bus mastership and that transfer is actually being performed. 0: Transfer has not been started. 1: Transfer is in progress or transfer has ended.
1	TREND	0	R/W	Processing End Flag Bit Indicates that the processing performed in the specified access mode has been completed. The write value should always be 0.
0	TRSTRT	0	R/W	Transfer Start By setting this bit from 0 to 1 when the TREND bit is 0, processing in the access mode specified by the access mode specification bits ACM[1:0] is initiated. 0: Stops transfer 1: Starts transfer

31.3.14 MODE Register (FLMODE)

FLMODE specifies the mode of this module.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	MODE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	MODE	0	R/W	Mode Setting 0: Mode 0 1: Mode 1 Note: In this product, be sure to set this bit to 1 before transfer to or from NAND flash memory.

31.4 Operation

31.4.1 Access Sequence

This module performs accesses in several independent stages.

For example, NAND-type flash memory programming consists of the following five stages.

- First command issue stage (program setup command)
- Address issue stage (program address)
- Data stage (output)
- Second command issue stage (program start command)
- Status read stage

NAND-type flash memory programming access is achieved by executing these five stages sequentially. An access to flash memory is completed at the end of the final stage (status read stage).

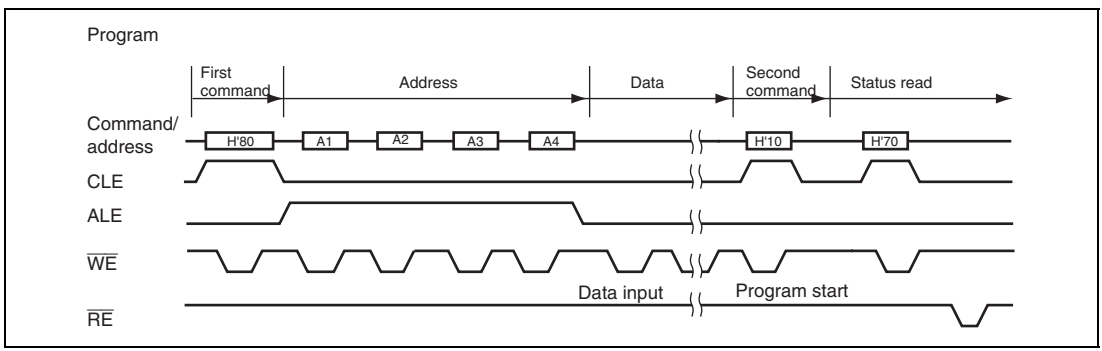


Figure 31.2 Programming Operation for NAND-Type Flash Memory and Stages

For details on NAND-type flash memory read operation, see section 31.4.4, Command Access Mode.

31.4.2 Operating Modes

Two operating modes are supported.

- Command access mode
- Sector access mode

The ECC generation and error check are performed in sector access mode.

31.4.3 Register Setting Procedure

Figure 31.3 shows the register setting flow required for accessing the flash memory.

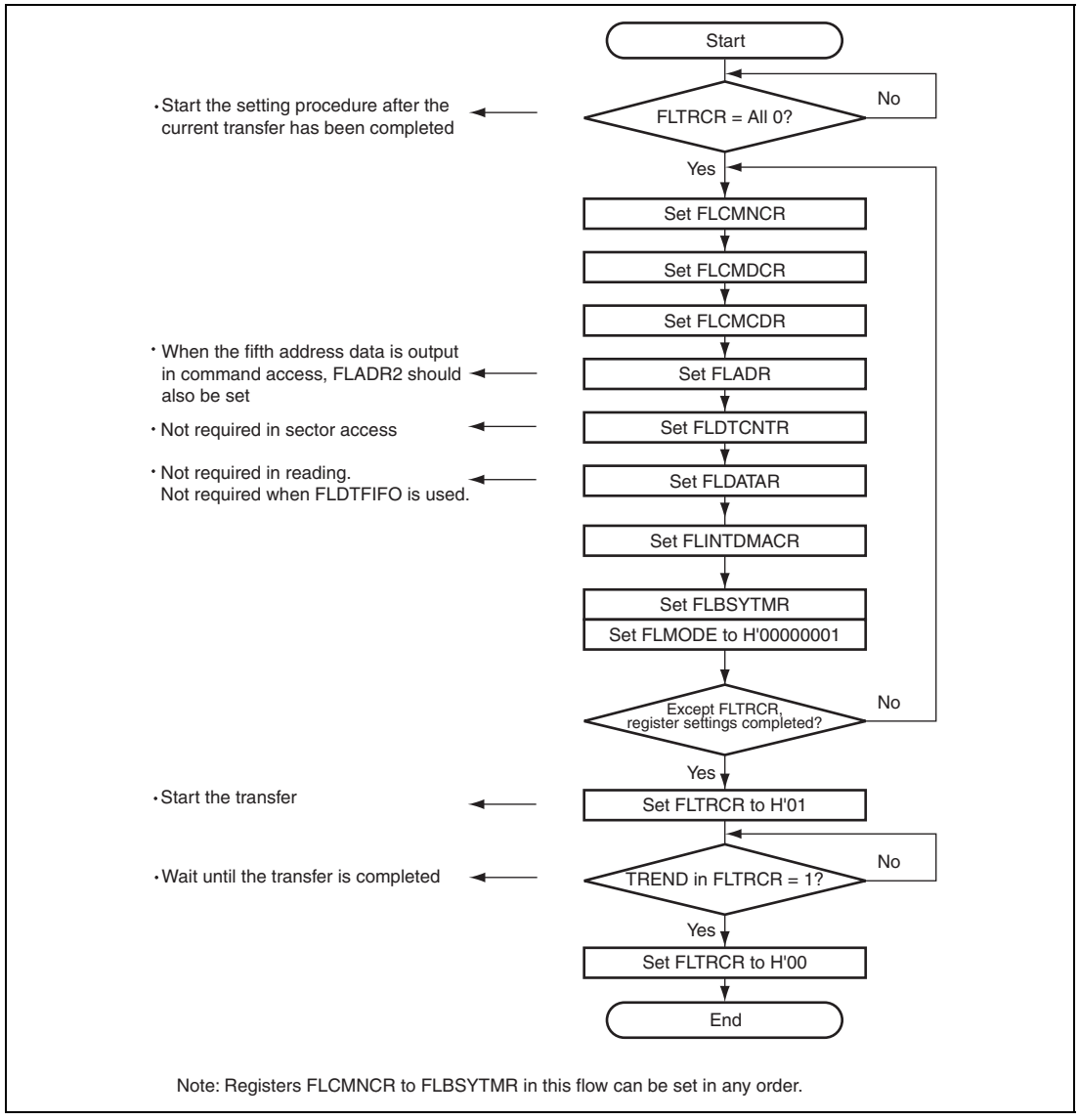


Figure 31.3 Register Setting Flow

31.4.4 Command Access Mode

Command access mode accesses flash memory by specifying a command to be issued to flash memory, address, data, read/write direction, and number of times to the registers. In this mode, I/O data can be transferred by the DMA via FLDTFIFO.

(1) NAND-Type Flash Memory Access

Figure 31.4 shows an example of read operation for NAND-type flash memory. In this example, the first command is specified as H'00, address data length is specified as 3 bytes, and the number of read bytes is specified as 8 bytes in the data counter.

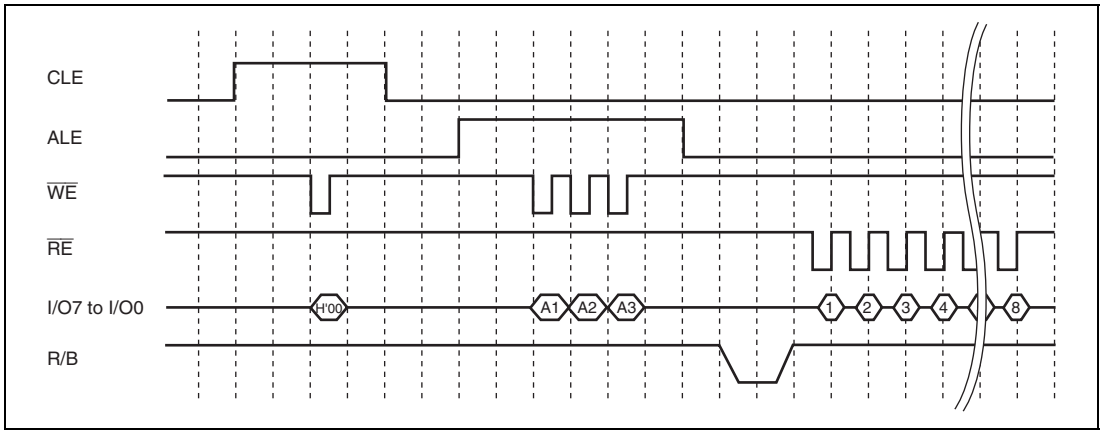


Figure 31.4 Read Operation Timing for NAND-Type Flash Memory

Figures 31.5 and 31.6 show examples of programming operation for NAND-type flash memory.

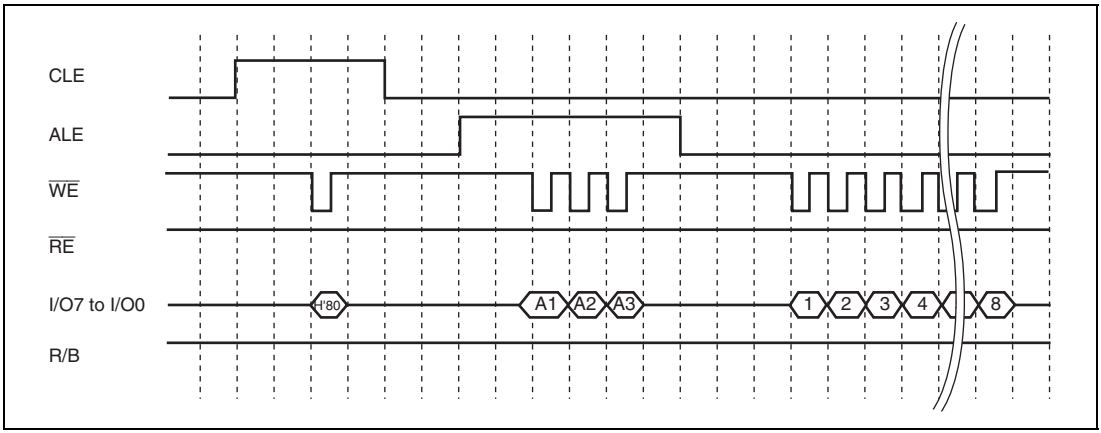


Figure 31.5 Programming Operation Timing for NAND-Type Flash Memory (1)

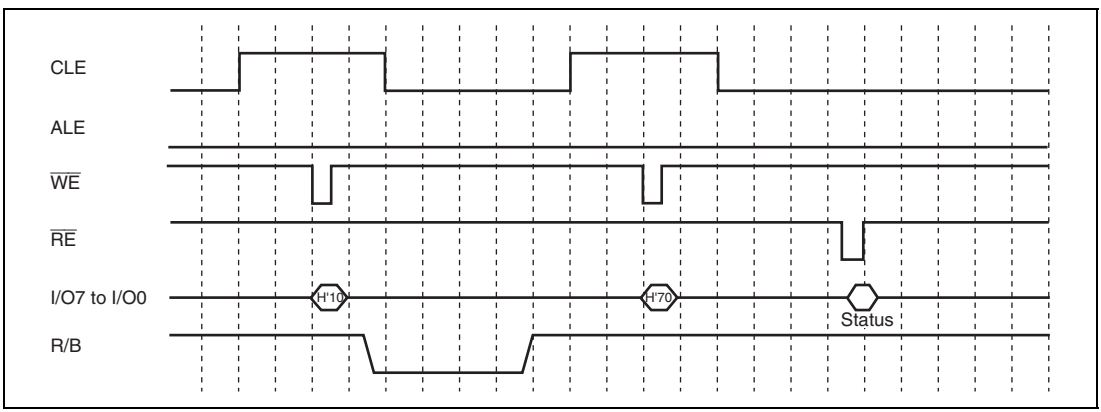


Figure 31.6 Programming Operation Timing for NAND-Type Flash Memory (2)

(2) NAND-Type Flash Memory (2048 + 64 Bytes) Access

Figure 31.7 shows an example of read operation for NAND-type flash memory (2048 + 64 bytes). In this example, the first command is specified as H'00, the second command is specified as H'30, and address data length is specified as 4 bytes. The number of read bytes is specified as 4 bytes in the data counter.

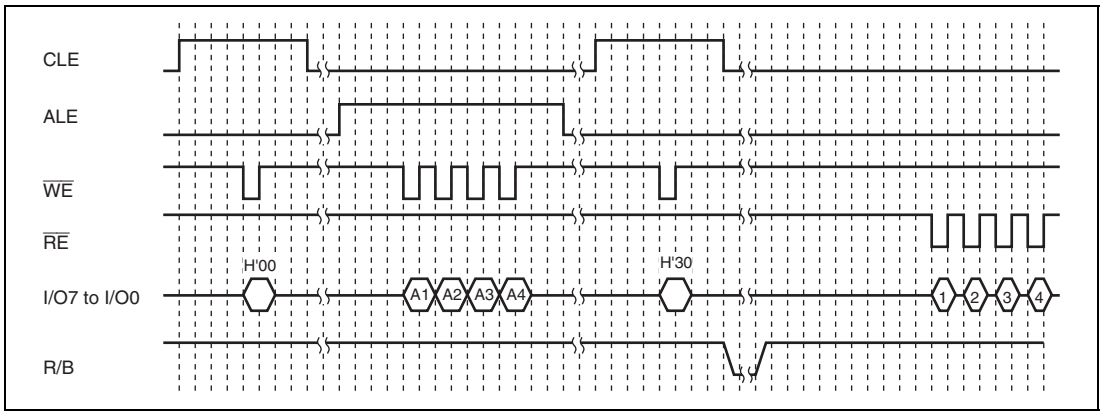


Figure 31.7 Read Operation Timing for NAND-Type Flash Memory

Figures 31.8 and 31.9 show examples of programming operation for NAND-type flash memory (2048 + 64 bytes).

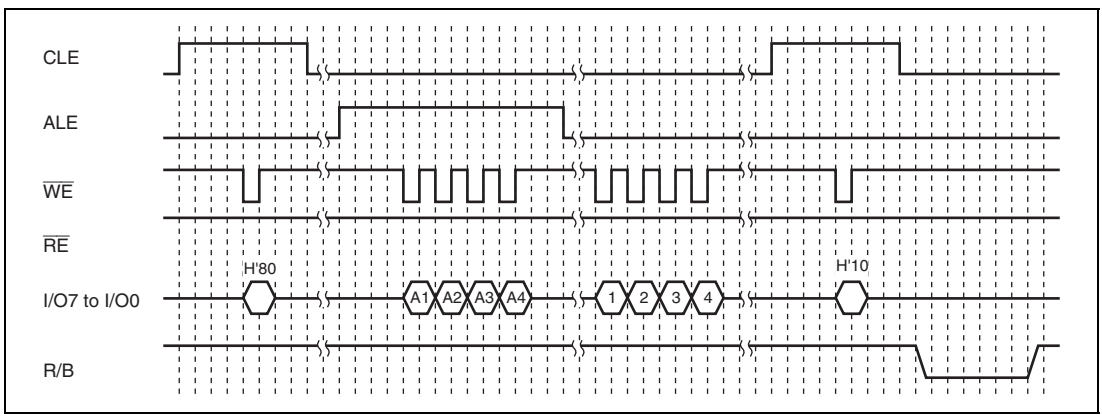


Figure 31.8 Programming Operation Timing for NAND-Type Flash Memory (1)

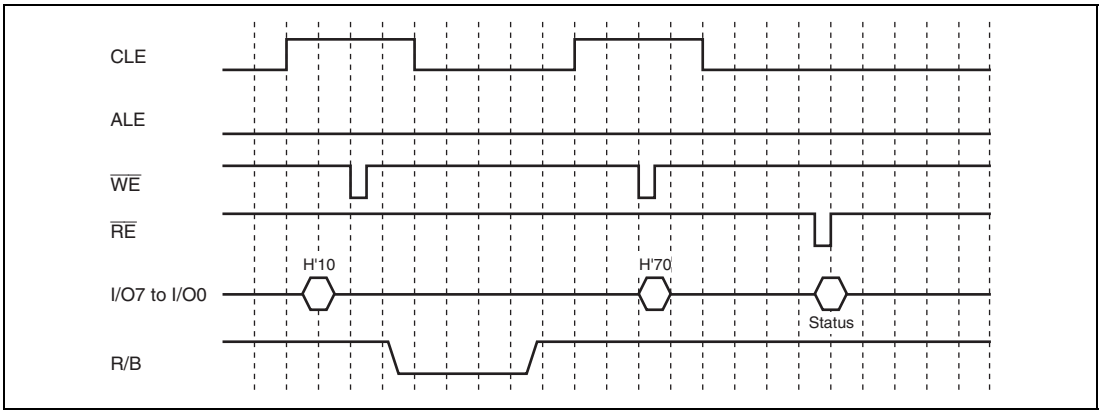


Figure 31.9 Programming Operation Timing for NAND-Type Flash Memory (2)

31.4.5 Sector Access Mode

In sector access mode, flash memory can be read or programmed in sector units by specifying the sector number of the sector to be accessed.

Since 512-byte data is stored in FLDTFIFO and 16-byte control code is stored in FLECFIFO, the DREQ1EN and DREQ0EN bits in FLINTDMACR can be set to transfer by the DMA.

Figure 31.10 shows the relationship of DMA transfer between sectors in flash memory (data and control code) and memory on the address space.

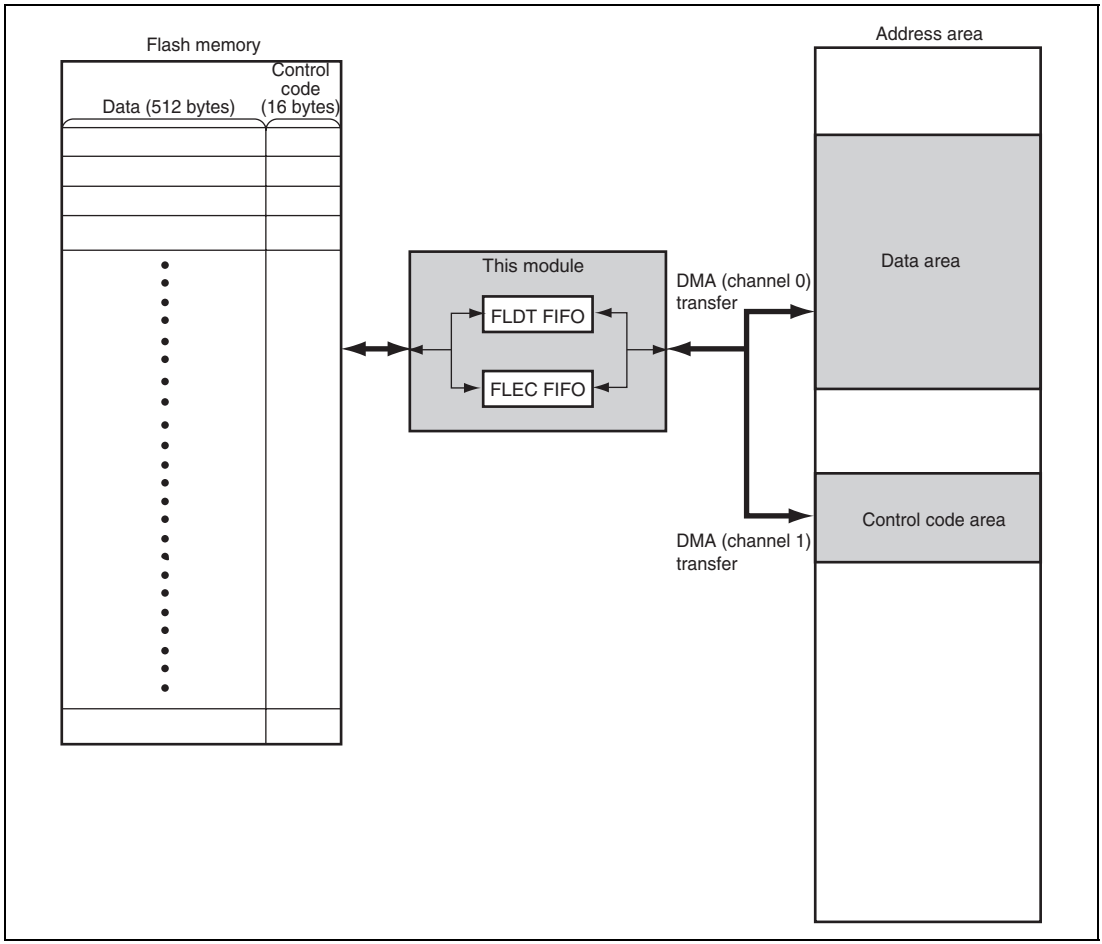


Figure 31.10 Relationship between DMA Transfer and Sector (Data and Control Code), and Memory and DMA Transfer

(1) Sector Address

Figure 31.11 shows the relationship between the physical sector address of NAND-type flash memory and the address of flash memory.

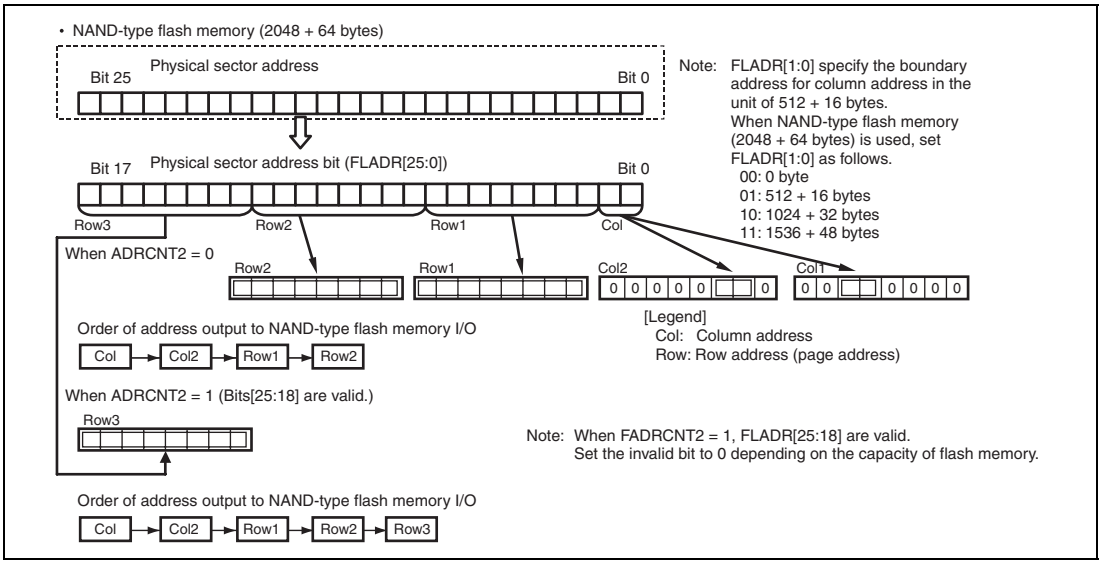


Figure 31.11 Relationship between Sector Number and Address Expansion of NAND-Type Flash Memory

(2) Continuous Sector Access

A series of sectors can be read or written by specifying the start sector address of NAND-type flash memory and the number of sectors to be transferred. Figure 31.12 shows an example of physical sector specification register and transfer count specification register settings when transferring logical sectors 0 to 40, which are not contiguous because of an unusable sector in NAND-type flash memory.

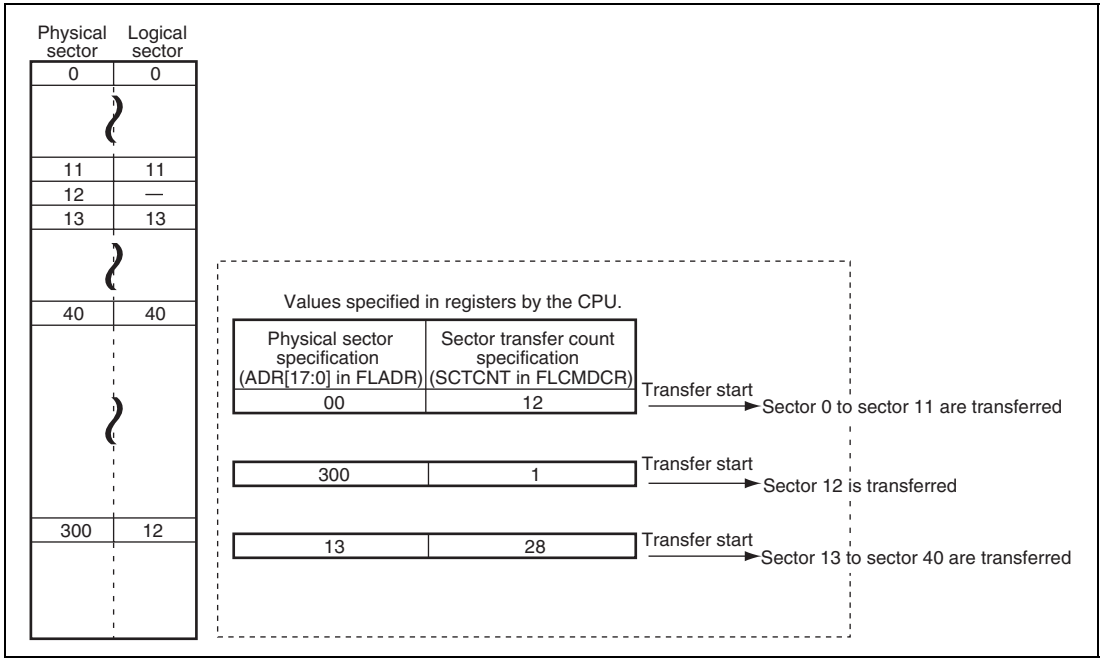


Figure 31.12 Sector Access when Unusable Sector Exists in Continuous Sectors

(3) Flash Memory Access in Sector Access Mode

Figures 31.13 and 31.14 show the timing of writing to and reading from the NAND-type flash memory in sector access mode.

Figure 31.13 shows the timing of writing to the 1-Gbit large-block flash memory. When data straddling multiple pages are being transferred in continuous access to one sector, the data are written to the flash memory with the timing shown in the figure for every page (2048 + 64 bytes).

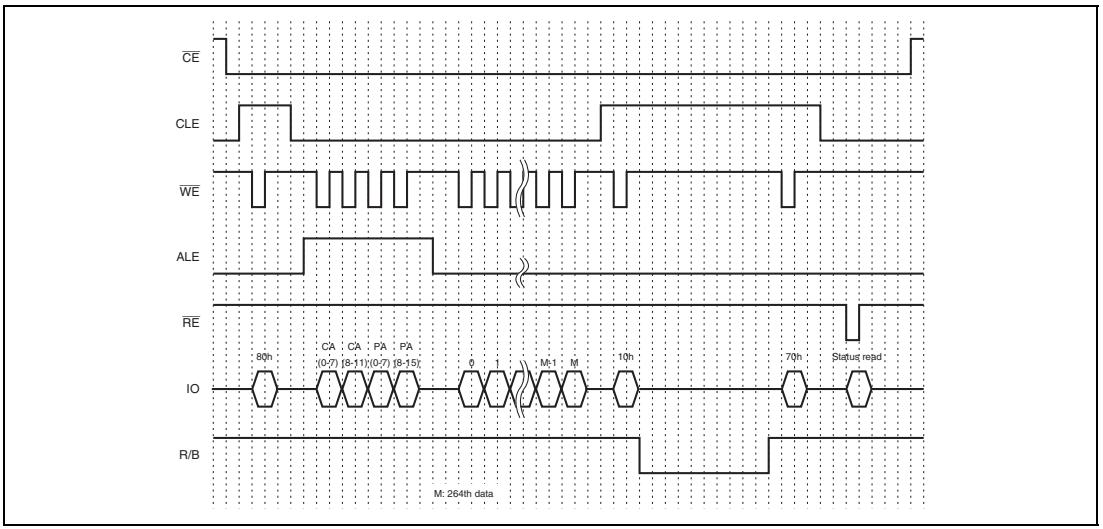


Figure 31.13 Programming Operation Timing for NAND-Type Flash Memory

Figure 31.14 shows the timing of reading from the 1-Gbit large-block flash memory. When data straddling multiple pages are being transferred in continuous access to one sector, the data are read from the flash memory with the timing shown in the figure for every page (2048 + 64 bytes).

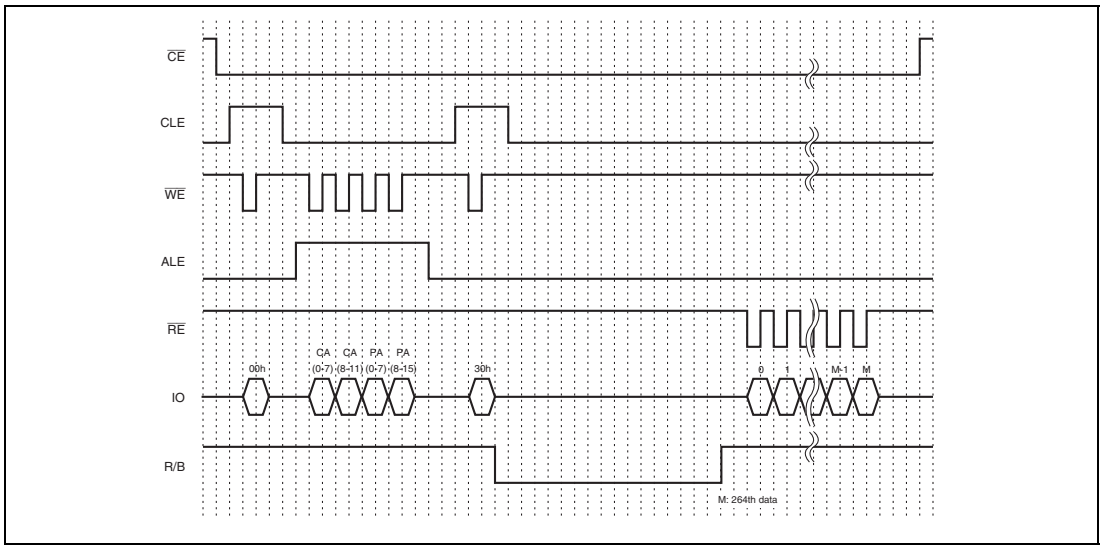


Figure 31.14 Read Timing from NAND-Type Flash Memory (Sector Access Mode)

31.4.6 Status Read

This module can read the status register of a NAND-type flash memory. The data in the status register is input through the I/O7 to I/O0 pins and stored in the bits STAT[7:0] in FLBSYCNT, which can be read by the CPU. If a program error or erase error is detected when the status register value is stored in the bits STAT[7:0] in FLBSYCNT, the STERB bit in FLINTDMACR is set to 1 and generates an interrupt to the CPU if the STERINTE bit in FLINTDMACR is enabled. If a status error occurs in the midst of continuous access to a sector, the TREND bit in FLTRCR is set to 1 and the current processing is ended.

(1) Status Read of NAND-Type Flash Memory

The status register of NAND-type flash memory can be read by inputting command H'70 to NAND-type flash memory. If programming is executed in command access mode or sector access mode while the DOSR bit in FLCMDCR is set to 1, this module automatically inputs command H'70 to NAND-type flash memory and reads the status register of NAND-type flash memory. When the status register of NAND-type flash memory is read, the I/O7 to I/O0 pins indicate the following information as described in table 31.3.

Table 31.3 Status Read of NAND-Type Flash Memory

I/O	Status (definition)	Description
I/O7	Program protection	0: Cannot be programmed 1: Can be programmed
I/O6	Ready/busy	0: Busy state 1: Ready state
I/O5 to I/O1	Reserved	—
I/O0	Program/erase	0: Pass 1: Fail

31.5 Interrupt Sources

This module has five interrupt sources: Status error, ready/busy timeout error, transfer end, FIFO0 transfer request, and FIFO1 transfer request. Each of the interrupt sources has its corresponding interrupt flag and the interrupt can be requested independently to the CPU if the interrupt is enabled by the interrupt enable bit. Note that the status error and ready/busy timeout error use the common FLSTE interrupt to the CPU.

Table 31.4 NAND Flash Memory Controller Interrupt Requests

Interrupt Source	Interrupt Flag	Enable Bit	Description	Priority
FLSTE interrupt	STERB	STERINTE	Status error	High
	BTOERB	RBERINTE	Ready/busy timeout error	
FLTEND interrupt	TREND	TEINTE	Transfer end	
FLTRQ0 interrupt	TRREQF0	TRINTE0	FIFO0 transfer request	
FLTRQ1 interrupt	TRREQF1	TRINTE1	FIFO1 transfer request	Low

31.6 DMA Transfer Specifications

This module can request DMA transfers separately to the data area FLDTFIFO and control code area FLECFIFO. Table 31.5 summarizes DMA transfer enable or disable states in each access mode.

Table 31.5 DMA Transfer Specifications

	Sector Access Mode	Command Access Mode
FLDTFIFO	DMA transfer enabled	DMA transfer enabled
FLECFIFO	DMA transfer enabled	DMA transfer disabled

For details on settings of the direct memory access controller (HPB-DMAC), see section 6A, Direct Memory Access Controller for Local Bus and Peripheral Modules (LBSC-DMAC/HPB-DMAC).

Section 32 High Speed Serial Communication Interface with FIFO (HSCIF)

32.1 Overview

This LSI has a high speed serial communication interface with built-in FIFO buffers (High Speed Serial Communication Interface with FIFO: HSCIF) that handles asynchronous communication and clock synchronous serial communication. The HSCIF has two 128-stage FIFO buffers separately for transmission and reception, which enables fast, efficient, and uninterrupted communication.

32.1.1 Features

The HSCIF has the following features.

- Asynchronous serial communication mode

  The HSCIF performs serial data communication based on a character-by-character asynchronous system. This feature enables serial data communication with standard asynchronous communication chips that support Universal Asynchronous Receiver/Transmitter (UART) or Asynchronous Communication Interface Adapter (ACIA). There is a choice of eight serial data transfer formats.

  — Data length: 7 or 8 bits

  — Stop bit length: 1 or 2 bits

  — Parity: Even/odd/none

  — Receive error detection: Parity, framing, and overrun errors

  — Break detection:

  A break is detected when a framing error lasts for more than 1 frame length at Space 0 (low level).

  When a framing error occurs, a break can also be detected by reading the RX pin level directly from the serial port register (HSSPTR).

  — Sampling rate: Variable (integer number from 8 to 32)

- Full-duplex communication capability

  The HSCIF has an independent transmitter and receiver that enable transmission and reception to performed simultaneously. The transmitter and receiver both have a 128-stage FIFO buffer structure, enabling continuous serial data transmission and reception.

- On-chip baud rate generator, enabling any bit rate to be selected

  The HSCIF enables choice of a clock source for transmission/reception: a clock from the on-chip baud rate generator based on the internal clock or an external clock.
- Eight interrupt sources

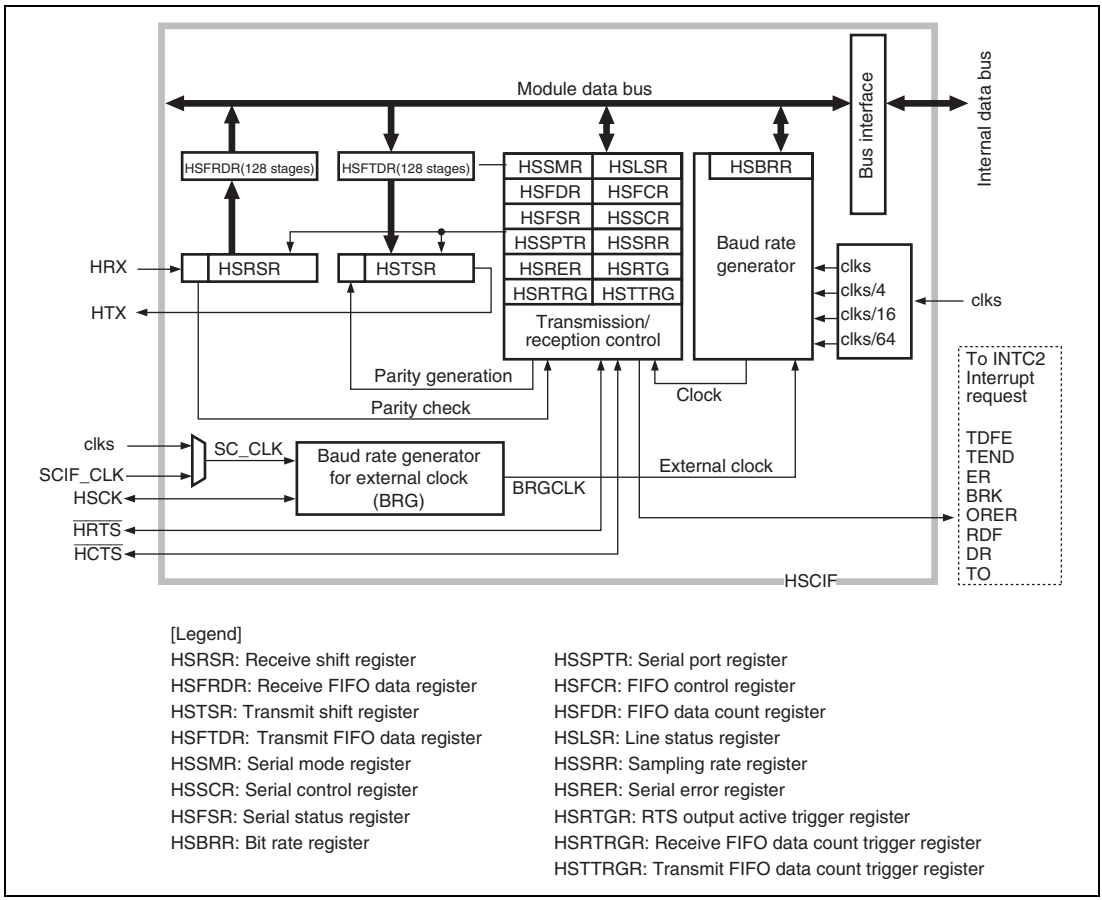
  The HSCIF has eight types of interrupt sources – receive-data-ready, receive-FIFO-data-full, break detection, transmit-FIFO-data-empty, transmit-end, receive-error, overrun-error and time-out and enables any of them to be requested independently.

- DMA data transfer

  When the transmit FIFO register is empty or the receive FIFO register has received data, issuing a DMA transfer request activates the DMA controller (DMAC) to execute a data transfer.
- Modem control functions ($\overline{\text{HRTS}}$ and $\overline{\text{HCTS}}$) are stored.
- The amount of data in the transmit/receive FIFO registers and the number of receive errors in receive data in the receive FIFO register are available.
- In asynchronous mode, a receive data ready (DR) or a timeout error (TO) can be detected during reception.

32.1.2 Block Diagram

Figure 32.1 shows the HSCIF block diagram.



[Legend]

HSRSR: Receive shift register
HSFRDR: Receive FIFO data register
HSTSR: Transmit shift register
HSFTDR: Transmit FIFO data register
HSSMR: Serial mode register
HSSCR: Serial control register
HSFSR: Serial status register
HSBRR: Bit rate register
HSSPTR: Serial port register
HSFCR: FIFO control register
HSFDR: FIFO data count register
HSLSR: Line status register
HSSRR: Sampling rate register
HSRER: Serial error register
HSRTGR: RTS output active trigger register
HSRTRGR: Receive FIFO data count trigger register
HSTTRGR: Transmit FIFO data count trigger register

Figure 32.1 HSCIF Block Diagram

32.1.3 Pin Configuration

Table 32.1 shows the HSCIF pin configuration.

These pins are multiplexed in other functions, so that the usage of the pins are may be restricted depending on the multiplexed pin settings.

Table 32.1 Pin Configuration

Pin Name	Abbreviation	I/O	Descriptions
Serial clock pin	HSCK0	I/O	Clock I/O
Receive data pin	HRX0	Input	Receive data input
Transmit data pin	HTX0	Output	Transmit data output
Modem control pin	$\overline{\text{HCTS0}}$	I/O	Transmission enabled
Modem control pin	$\overline{\text{HRTS0}}$	I/O	Transmission request
Baud rate generation clock pin	SCIF_CLK	Input	Clock for input to the baud rate generator for the external clock

Note: These pins are made to function as serial pins by setting the HSCIF operation using, bits TE, RE, CKE1, and CKE0 in HSSCR, and bit MCE in HSFCR. For break state transmission and detection, HSSPTR can be set in the HSCIF.

32.1.4 Register Configuration

Table 32.2 shows the registers in the HSCIF. Do not write to any addresses other than listed below. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than listed below are undefined.

The base address of HSCIF is as follows;

HSCIF0: H'FFE4_8000

Table 32.2 (1) Register Configuration

Register Name	Abbreviation	R/W	Offset From Base Address	Access Size
Serial mode register	HSSMR	R/W	H'00	16
Bit rate register	HSBRR	R/W	H'04	8
Serial control register	HSSCR	R/W	H'08	16
Transmit FIFO data register	HSFTDR	W	H'0C	8
Serial status register	HSFSR	R/W*1	H'10	16
Receive FIFO data register	HSFRDR	R	H'14	8
FIFO control register	HSFCR	R/W	H'18	16
FIFO data count register	HSFDR	R	H'1C	16
Serial port register	HSSPTR	R/W	H'20	16
Line status register	HSLSR	R/W*2	H'24	16
Sampling rate register	HSSRR	R	H'40	16
Serial error register	HSRER	R/W	H'44	16
RTS output active trigger count register	HSRTGR	R/W	H'50	16
Receive FIFO data count trigger register	HSRTRGR	R/W	H'54	16
Transmit FIFO data count trigger register	HSTTRGR	R/W	H'58	16

Notes: 1. To clear the flags, only 0 can be written. Bits 15 to 8, 3, and 2 are read-only bits and cannot be modified.
2. To clear the flags, only 0 can be written. Bits 15 to 3, and 1 are read-only bits and cannot be modified.

Table 32.2 (2) Register State in Each Operating Mode

Abbreviation	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
HSSMR	Initialized	Initialized	Retained	Retained	Retained	Initialized
HSBRR	Initialized	Initialized	Retained	Retained	Retained	Initialized
HSSCR	Initialized	Initialized	Retained	Retained	Retained	Initialized
HSFTDR	Undefined	Undefined	Retained	Retained	Retained	Undefined
HSFSR	Initialized	Initialized	Retained	Retained	Retained	Initialized
HSFRDR	Undefined	Undefined	Retained	Retained	Retained	Undefined
HSFCR	Initialized	Initialized	Retained	Retained	Retained	Initialized
HSFDR	Initialized	Initialized	Retained	Retained	Retained	Initialized
HSSPTR	Initialized	Initialized	Retained	Retained	Retained	Initialized
HSLSR	Initialized	Initialized	Retained	Retained	Retained	Initialized
HSSRR	Initialized	Initialized	Retained	Retained	Retained	Initialized
HSRER	Initialized	Initialized	Retained	Retained	Retained	Initialized
HSRTGR	Initialized	Initialized	Retained	Retained	Retained	Initialized
HSRTRGR	Initialized	Initialized	Retained	Retained	Retained	Initialized
HSTTRGR	Initialized	Initialized	Retained	Retained	Retained	Initialized

32.2 Register Description

32.2.1 Receive Shift Resister (HSRSR)

HSRSR is a register that receives serial data.

The HSCIF sets serial data input to the HSRSR from the HRX pin in the order received, starting with the LSB (bit 0), and converts it to parallel data. When one byte of data has been received, it is transferred to the receive FIFO register HSFRDR, automatically.

HSRSR cannot be read from and written to by the CPU.

Bit:	7	6	5	4	3	2	1	0
Initial value:	—	—	—	—	—	—	—	—
R/W:	—	—	—	—	—	—	—	—

32.2.2 Receive FIFO Data Register (HSFRDR)

HSFRDR is a 128-stage FIFO register that stores received serial data.

When HSCIF has received one byte of serial data, it transfers the received data from the receive shift resister (HSRSR) to HSFRDR where it is stored, and completes the reception. HSRSR is then ready for reception, and enables up to 128 sets of data consecutively until HSFRDR is full.

HSFRDR is a read-only register and cannot be modified by the CPU. Note that the read value will be undefined while there is no receive data in SCFRDR. When SCFRDR is full of receive data, subsequent serial data is lost.

HSFRDR is read as an undefined value after a power-on reset or manual reset.

Bit:	7	6	5	4	3	2	1	0
Initial value:	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R

32.2.3 Transmit Shift Register (HSTSR)

HSTSR is a register that transmits serial data.

To perform serial data transmission, the HSCIF first transfers transmit data from the transmit FIFO data register (HSFTDR) to HSTSR, then sends the data to the HTX pin starting with the LSB (bit 0). When transmission of one byte is completed, the HSCIF transfers the next transmit data from HSFTDR to HSTSR automatically, then starts transmission.

HSTSR cannot be read from and written to directly by the CPU.

Bit:	7	6	5	4	3	2	1	0
Initial value:	—	—	—	—	—	—	—	—
R/W:	—	—	—	—	—	—	—	—

32.2.4 Transmit FIFO Data Register (HSFTDR)

HSFTDR is an 8-bit FIFO register of 128 stages that stores data for serial transmission.

If HSTSR is empty when transmit data has been written to HSFTDR, the HSCIF transfers the transmit data written to HSFTDR to HSTSR and starts serial transmission.

HSFTDR is a write-only register and cannot be read from by the CPU. HSFTDR disables to write the next data to it when it is filled with 128 bytes of transmit data. The written data is ignored.

HSFTDR is read as an undefined value on a power-on reset or manual reset.

Bit:	7	6	5	4	3	2	1	0
Initial value:	—	—	—	—	—	—	—	—
R/W:	W	W	W	W	W	W	W	W

32.2.5 Serial Mode Register (HSSMR)

HSSMR is a 16-bit register that sets the HSCIF's serial transfer format and selects the baud rate generator clock source.

HSSMR can always be read from and written to by the CPU.

HSSMR is initialized to H'0000 by a power-on reset or a manual reset.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	CHR	PE	O/$\overline{E}$	STOP	—	CKS1	CKS0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6	CHR	0	R/W	Character Length Selects 7 or 8 bits for data length. When the 7-bit data is selected, the MSB (bit 7) in the transmit FIFO data register (HSFTDR) is not transmitted 0: 8 bits 1: 7 bits
5	PE	0	R/W	Parity Enable Determines whether parity bit is added in transmission or not, and parity bit is checked in reception or not. When bit PE is set to 1, the parity (even or odd) specified by bit O/$\overline{E}$ is added to transmit data. In reception, the parity bit is checked for the parity (even or odd) specified by bit O/$\overline{E}$. 0: Disables parity bit addition and check. 1: Enables parity bit addition and check.

Bit	Bit Name	Initial Value	R/W	Description
4	O/$\overline{E}$	0	R/W	Parity Mode Selects either even or odd parity to use in parity addition and check. In asynchronous mode, the O/$\overline{E}$ bit setting is valid only when bit PE is set to 1, enabling parity bit addition and check. 0: Even parity 1: Odd parity When even parity is set, the parity bit is added in transmission so that the total number of 1-bits in the transmit character plus the parity bit is even. In reception, a check is performed to see if the total number of 1-bits in the receive character plus the parity bit is even. When odd parity is set, the parity bit is added in transmission so that the total number of 1-bits in the transmit character plus the parity bit is odd. In reception, a check is performed to see if the total number of 1-bits in the receive character plus the parity bit is odd.
3	STOP	0	R/W	Stop Bit Length Selects 1 bit or 2 bits as the stop bit length. The stop bit setting is valid only in asynchronous mode. In reception, only the first stop bit is checked, regardless of the STOP bit setting. If the second stop bit is 1, it is treated as a stop bit. If it is 0, it is treated as the start bit of the next transmit character. 0: 1 stop bit*1 1: 2 stop bits*2 Notes: 1. In transmission, a single 1-bit (stop bit) is added to the end of a transmit character before it is sent. 2. In transmission, two 1-bits (stop bits) are added to the end of a transmit character before it is sent.
2	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1, 0	CKS[1:0]	00	R/W	Clock Select 1 and 0 These bits select the clock source for the on-chip baud rate generator. The clock source can be selected from clks, clks/4, clks/16, and clks/64, according to the setting of bits CKS1 and CKS0. 00: clks 01: clks/4 10: clks/16 11: clks/64

32.2.6 Serial Control Register (HSSCR)

HSSCR is a register that enables or disables transmission/reception by the HSCIF, enables or disables interrupt requests, and selects transmission/reception clock source for the HSCIF.

HSSCR can always be read from and written to by the CPU.

HSSCR is initialized to H'0000 by a power-on reset or manual reset.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TOT [1:0]		—	—	TEIE	—	—	—	TIE	RIE	TE	RE	REIE	TOIE	CKE1	CKE0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15, 14	TOT[1:0]	00	R/W	Set the time for a data ready (DR) or a timeout (TO) to be set in asynchronous mode. 00: 15 etu* 01: 31 etu 10: 47 etu 11: 63 etu Note: ETU: Elementary Time Unit (time for transfer of one bit) Equivalent to 1.5 frames with an 8-bit, 1-stop-bit format.
13, 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11	TEIE	0	R/W	Transmit End Interrupt Enable When interrupt requests are enabled by the TIE bit, the TEIE bit selects • Setting of the TDFE flag in HSFSR or • Setting of the TEND flag in HSFSR as the source of the requests. 0: The transmit FIFO data empty (TDFE) interrupt request is used. 1: The transmit end (TEND) interrupt request is used.

Bit	Bit Name	Initial Value	R/W	Description
10 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	TIE	0	R/W	Transmit Interrupt Enable Enables or disables a transmit-FIFO-data-empty interrupt (TDFE) request when the TEIE bit in HSSCR is 0, if the following conditions are satisfied: • Serial transmit data has been transferred from HSFTDR to HSTSR, • The number of data bytes in HSFTDR is equal to or less than the transmit trigger count, and • The TDFE flag in HSFSR is set to 1. Enables or disables a transmit-end interrupt (TEND) request when the TEIE bit of HSSCR is set to 1, if the following conditions are satisfied: • Transmission was ended because there is no valid data in HSFTDR when the last bit of the transmit character in HSTSR was transmitted, and • The TEND flag of HSFSR is set to 1. 0: When the TEIE bit is 0, disables transmit-FIFO-data-empty interrupt (TDFE) request. When the TEIE bit is 1, disables transmit-end (TEND) request. 1: When the TEIE bit is 0, enables transmit-FIFO-data-empty interrupt (TDFE) request. When the TEIE bit is 1, enables transmit-end (TEND) request.

Bit	Bit Name	Initial Value	R/W	Description
6	RIE	0	R/W	Receive Interrupt Enable Enables or disables a receive-FIFO-data-full interrupt request when the RDF flag in HSFSR is set to 1, a receive-data-ready interrupt request when the DR flag in HSFSR is set to 1, a receive-error interrupt request when the ER flag in HSFSR is set to 1, a break interrupt request when the BRK flag in HSFSR is set to 1, and an overrun error interrupt request when the ORER flag in HSLSR is set to 1. 0: Disables receive-FIFO-data-full interrupt (RDF) requests, receive-data-ready interrupt (DR) requests, receive-error interrupt (ER) requests, break interrupt (BRK) requests, and overrun error interrupt (ORER) requests. 1: Enables receive-FIFO-data-full interrupt (RDF) requests, receive-data-ready interrupt (DR) requests, receive-error interrupt (ER) requests, break interrupt (BRK) requests, and overrun error interrupt (ORER) requests.
5	TE	0	R/W	Transmit Enable Enables or disables the start of HSCIF serial transmission. The HSCIF starts serial transmission when transmit data is written to the HSFTDR while TE is 1. Before setting TE to 1, set HSSMR and HSFCR to specify the transmission format and reset the transmit FIFO. 0: Disables transmission. 1: Enables transmission.
4	RE	0	R/W	Receive Enable Enables or disables the start of HSCIF serial reception. When RE is 1, the HSCIF starts serial reception by detecting a start bit. Before setting RE to 1, set HSSMR and HSFCR to specify the reception format and reset the receive FIFO. 0: Disables reception.* 1: Enables reception. Note: Even if RE is cleared to 0, the DR, ER, BRK, RDF, FER, PER, TO and ORER flags are not affected, and retain their states.

Bit	Bit Name	Initial Value	R/W	Description
3	REIE	0	R/W	Receive Error Interrupt Enable Enables or Disables generation of receive-error interrupt (ER) requests, break interrupt (BRK) requests and overrun-error interrupt (ORER) requests. 0: Disables receive-error interrupt (ER) requests, break interrupt (BRK) requests and overrun-error interrupt (ORER) requests.* 1: Enables receive-error interrupt (ER) requests, break interrupt (BRK) requests and overrun-error interrupt (ORER) requests. Note: * When REIE is 1, ER, BRK or ORER interrupt requests will occur even if RIE is cleared to 0. This setting is used to notify the interrupt controller of ER, BRK, and ORER interrupt requests during DMAC transfer.
2	TOIE	0	R/W	Timeout Interrupt Enable Enables or disables generation of timeout interrupt (TO) requests when the TO flag in HSLSR is set to 1. 0: Disables timeout interrupts (TO). 1: Enables timeout interrupts (TO).
1, 0	CKE[1:0]	00	R/W	Clock Enable 1 and 0 These bits select the HSCIF clock source and enables or disables the clock output from the HSCK pin. Whether the HSCK pin functions as a serial clock output pin or a serial clock input pin is determined by combination of the CKE1 and CKE0 bit settings.

Table 32.3 Clock Selection

CKE1	CKE0	Clock Source	HSCK Pin
0	0	Internal clock (clks, clks/4, clks/16, and clks/64)	The HSCK pin is not used. The HSCK pin functions as an input pin (Input signals are ignored). (Initial value)
0	1		The HSCK pin outputs the clock (with a bit rate multiplied by the sampling rate).
1	0	Baud rate generator output for external clock or HSCK	When SC_CLK is selected: The HSCK pin is an input pin (Input signals are ignored). Set the SC_CLK frequency so that the frequency of BRGCLK is multiplied by the sampling rate.
			When HSCK is selected: The HSCK pin inputs the clock (with the bit rate multiplied by the sampling rate).
	1	Prohibited	

Note: It is not allowed to set synchronous communication using SC_CLK for input.

32.2.7 Serial Status Register (HSFSR)

HSFSR is a 16-bit register. The lower 8 bits are a status flag that indicates the operating status of the HSCIF, and the upper 8 bits are all reserved.

HSFSR can always be read from and written to by the CPU. However, the flags ER, TEND, TDFE, BRK, RDF, and DR cannot be written by 1. The FER and PER flags are read-only flags and cannot be modified.

HSFSR is initialized to H'0060 by a power-on reset or a manual reset.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ER	TEND	TDFE	BRK	FER	PER	RDF	DR
Initial value:	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W*	R/W*	R/W*	R/W*	R	R	R/W*	R/W*

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
7	ER	0	R/W*	Receive Error Indicates that a framing error or a parity error has occurred in reception.* The ER flag is not affected by an error and retains its previous state when the RE bit is 0 in HSSCR. If a receive error occurs, receive data will be transferred to HSFRDR and reception operation will be continued. Whether there is a receive error in data read from HSFRDR can be determined by the FER and PER bits in HSFSR. 0: Indicates that no framing or parity error has occurred in reception. [Clearing conditions] • A power-on reset or a manual reset is executed. • 0 is written to ER. 1: Indicates that a framing error or a parity error has occurred in reception. [Setting conditions] • The HSCIF checks whether the stop bit at the end of receive data is 1, but the stop bit is 0.*[2] • The number of 1-bits in receive data plus the parity bit does not match the parity setting (even or odd) specified by the O/$\overline{E}$ bit in HSSMR during reception. Note: In the 2-stop-bit mode, only the first stop bit is checked that the value is 1; the second stop bit is not checked.

Bit	Bit Name	Initial Value	R/W	Description
6	TEND	1	R/W*	Transmit End Indicates that transmission has been ended because there was no valid data in HSFTDR when the last bit of the transmit character was transmitted. 0: Indicates that transmission is in progress. [Clearing conditions] • Transmit data is written to HSFTDR, and 0 is written to TEND. • Data is written to HSFTDR by the DMAC. 1: Indicates that transmission has been ended. [Setting conditions] • A power-on reset or a manual reset is executed. • The TE bit in HSSCR is 0. • There is no transmit data in HSFTDR when the last bit of a 1-byte serial transmission character is transmitted.

Bit	Bit Name	Initial Value	R/W	Description
5	TDFE	1	R/W*	Transmit FIFO Data Empty Indicates that the HSCIF has transferred data from HSFTDR to HSTSR, the number of data bytes in HSFTDR becomes equal to or less than the transmit trigger count specified by the TTRG1 and TTRG0 bits in HSFCR, and HSFTDR is ready to be written by new transmit data. HSFTDR is a 128-byte FIFO register. The maximum number of bytes that can be written to when TDFE = 1 is "128 − [the transmit trigger count]". If data exceeding this value is attempted to be written, the data will be ignored. The number of data bytes in HSFTDR is indicated by the upper bits of HSFDR. If the number of data written in HSFTDR is equal to or less than the transmit trigger count, this bit will be set to 1 even if it is cleared to 0 after it is read as 1. 0: Indicates that the number of transmit data written to HSFTDR exceeds the transmit trigger count. [Clearing conditions] • Transmit data exceeding the specified transmit trigger count have been written to HSFTDR, and 0 is written to TDFE. • Transmit data exceeding the specified transmit trigger count have been written to HSFTDR by the DMAC. 1: Indicates that the number of transmit data in HSFTDR is equal to or less than the transmit trigger count. [Setting conditions] • A power-on reset or a manual reset is executed. • The number of transmit data in HSFTDR is equal to or less than the transmit trigger count after transmission.

Bit	Bit Name	Initial Value	R/W	Description
4	BRK	0	R/W*	Break Detect Indicates that a receive data break signal has been detected. If a break signal is detected, receive data (H'00) transfer to HSFRDR is stopped. After the break is canceled and the receive signal returns to 1, the receive data transfer resumes. 0: Indicates that no break signal has been received. [Clearing conditions] • A power-on reset or a manual reset is executed. • 0 is written to BRK. 1: Indicates that a break signal has been received. [Setting condition] • Data with a framing error is received, followed by the space "0" (low level) for at least one frame length.
3	FER	0	R	Framing Error Indicates that a framing error has been found in the data that is to be read next from HSFRDR. 0: Indicates that there is no framing error in the receive data that is to be read from HSFRDR. [Clearing conditions] • A power-on reset or a manual reset is executed. • There is no framing error in the data that is to be read next from HSFRDR. 1: Indicates that there is a framing error in the receive data that is to be read from HSFRDR. [Setting condition] • There is a framing error in the data that is to be read next from HSFRDR.

Bit	Bit Name	Initial Value	R/W	Description
2	PER	0	R	Parity Error This bit indicates that a parity error has been found in the data that is to be read next from HSFRDR. 0: Indicates that there is no parity error in the receive data that is to be read from HSFRDR. [Clearing conditions] • A power-on reset or a manual reset is executed. • There is no parity error in the data that is to be read next from HSFRDR. 1: Indicates that there is a parity error in the receive data that is to be read from HSFRDR. [Setting condition] • There is a parity error in the data that is to be read next from HSFRDR.

Bit	Bit Name	Initial Value	R/W	Description
1	RDF	0	R/W*	Receive FIFO Data Full Indicates that the received data has been transferred from HSRSR to HSFRDR, and the number of receive data bytes in HSFRDR becomes equal to or more than the receive trigger count specified by the RTRG1 and RTRG0 bits in HSFCR. HSFRDR is a 128-byte FIFO register. When RDF = 1, data equal to or more than the number of receive trigger data bytes can be read. When HSFRDR is empty, HSFRDR is read as an undefined value. The number of receive data bytes in HSFRDR is indicated by the lower bits of HSFDR. If the number of data in HSFRDR is equal to or more than the trigger count, this bit will be set to 1 even if it is cleared to 0. At this time, read receive data until the number of data in HSFRDR is less than the trigger count, read RDF as 1, and then clear RDF. 0: Indicates that the number of receive data bytes in HSFRDR is less than the specified receive trigger count. [Clearing conditions] • A power-on reset or a manual reset is executed. • HSFRDR is read until the number of receive data bytes in HSFRDR is less than the receive trigger count, and 0 is written to RDF. • HSFRDR is read by the DMAC until the number of receive data bytes in HSFRDR is less than the receive trigger count. 1: Indicates that the number of receive data bytes in HSFRDR is equal to or more than the specified receive trigger count. [Setting condition] • Received data equal to or more than the receive trigger count have been stored in SDFRDR.

Bit	Bit Name	Initial Value	R/W	Description
0	DR	0	R/W*	Receive Data Ready Indicates that the receive FIFO data register (HSFRDR) contains fewer bytes than the trigger number for reception and no further data have arrived over at least the time corresponding to the setting of HSSCR[15:14]* since the stop bit for the byte to have been received. 0: Indicates that data is being received or has been successfully received, and there is no receive data in HSFRDR. [Clearing conditions] • A power-on reset or a manual reset is executed. • All the receive data in HSFRDR has been read, and 0 is written to DR. • All the receive data in HSFRDR has been read by the DMAC. 1: Indicates that no further receive data has been received. [Setting condition] The receive FIFO data register (HSFRDR) contains fewer bytes than the trigger number for reception and no further data have arrived over at least the time corresponding to the setting of HSSCR[15:14]* since the stop bit for the byte to have been received. Note: When the setting is 00, the time is 15 etu. This is equivalent to 1.5 frames in an 8-bit, 1-stop-bit format. etu: Elementary Time Unit (time for transfer of one bit)

32.2.8 Bit Rate Register (HSBRR)

HSBRR is an 8-bit register that sets the serial transmission/reception bit rate in accordance with the baud rate generator operating clock selected by the CKS1 and CKS0 bits in HSSMR. This baud rate generator is intended for clks, clks/4, clks/16, and clks/64. For details on the baud rate generator for external clock, see section 32.6, Baud Rate Generator for External Clock (BRG).

HSBRR can always be read from and written to by the CPU.

HSBRR is initialized to H'FF by a power-on reset or a manual reset.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—
Initial value:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

The HSBRR setting is determined by the following equation:

[Asynchronous mode]

$$N = \frac{clks}{Sr \times 2^{2n+1} \times B} \times 10^6 - 1$$

B: Bit rate (bit/s)

N: HSBRR setting for the baud rate generator ($0 \leq N \leq 255$) (which satisfies the electrical characteristics)

clks: Peripheral module operating frequency (MHz)

n: Baud rate generator input clock (n = 0, 1, 2, 3)
(See table 32.4 for the relation between n and the clock.)

Sr: Sampling rate (8 to 32)

Table 32.4 HSSMR Settings

	Baud Rate Generator	HSSMR Setting	
n	**Input Clock**	**CKS1**	**CKS0**
0	clks	0	0
1	clks/4	0	1
2	clks/16	1	0
3	clks/64	1	1

The bit rate error in asynchronous mode is determined by the following equation:

$$\text{error (\%)} = \left\{ \frac{\text{clks} \times 10^6}{(N+1) \times B \times Sr \times 2^{2n+1}} - 1 \right\} \times 100$$

32.2.9 FIFO Control Register (HSFCR)

HSFCR is a register that resets data counts and sets the number of trigger data bytes for the transmit and receive FIFO registers. It also has a modem control and a loopback test enable bit.

HSFCR can always be read from and written to by the CPU.

HSFCR is initialized to H'0000 by a power-on reset or a manual reset.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	MCE	TFRST	RFRST	LOOP
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
3	MCE	0	R/W	Modem Control Enable Enables or disables modem control signals $\overline{\text{HCTS}}$ and $\overline{\text{HRTS}}$. 0: Disables modem signals.* 1: Enables modem signals. Note: $\overline{\text{HCTS}}$ and $\overline{\text{HRTS}}$ control ports.
2	TFRST	0	R/W	Transmit FIFO Data Register Reset Enables or disables a transmit FIFO data register reset that empties the register. 0: Disables the reset.* 1: Enables the reset. Note: The register is reset by a power-on reset or a manual reset.
1	RFRST	0	R/W	Receive FIFO Data Register Reset Enables or disables a receive FIFO data register (HSFRDR) reset that empties the register. 0: Disables the reset.* 1: Enables the reset. Note: The register is reset by a power-on reset or a manual reset.
0	LOOP	0	R/W	Loopback Test Enables or disables the loopback test by internally connecting the transmit output pin (HTX) and receive input pin (HRX), and the $\overline{\text{HRTS}}$ pin and $\overline{\text{HCTS}}$ pin. 0: Disables the loopback test. 1: Enables the loopback test.

32.2.10 FIFO Data Count Register (HSFDR)

HSFDR is a 16-bit register that indicates the number of data bytes stored in HSFTDR and that in HSFRDR.

The upper 8 bits indicate the number of transmit data bytes in HSFTDR, and the lower 8 bits indicates the number of receive data bytes in HSFRDR.

HSFDR can always be read by the CPU.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	T[7:0]								R[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	T[7:0]	H'00	R	Indicates the number of untransmitted data in HSFTDR. H'00 indicates that there is no transmit data in HSFTDR, and H'80 indicates that HSFTDR is full of transmit data.
7 to 0	R[7:0]	H'00	R	Indicates the number of receive data stored in HSFRDR in bytes. H'00 indicates that there is no receive data in HSFRDR, and H'80 indicates that HSFRDR is full of receive data

32.2.11 Serial Port Register (HSSPTR)

HSSPTR controls input/output of the data into/from the ports of the high speed serial communication interface (HSCIF) ports.

Bits 1 and 0: control breaks in serial transmission/reception by reading input data from the HRX pin and writing output data to the HTX pin.

Bits 3 and 2: read input data and write output data to the HSCK pin.

Bits 5 and 4: read input data and write output data to the $\overline{\text{HCTS}}$ pin.

Bits 7 and 6: read input data and write output data to the $\overline{\text{HRTS}}$ pin.

HSSPTR is a 16-bit register that can always be read from and written to by the CPU.

All HSSPTR bits except bits 6, 4, 2, and 0 are initialized to 0 by a power-on reset or a manual reset. The values of bits 6, 4, 2, and 0 are undefined.

Note: Whether modem control can be selected or not depends on the channel.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RTSIO	RTSDT	CTSIO	CTSDT	SCKIO	SCKDT	SPB2IO	SPB2DT
Initial value:	0	0	0	0	0	0	0	0	0	—	0	—	0	—	0	—
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	RTSIO	0	R/W	Serial Port - RTS Port Input/Output Specifies input or output for the serial port $\overline{\text{HRTS}}$ pin. To actually set the $\overline{\text{HRTS}}$ pin as a port output pin to output the value set by the RTSDT bit, the MCE bit in HSFCR should be cleared to 0. 0: Indicates that this bit does not output the value of the RTSDT bit to the $\overline{\text{HRTS}}$ pin. 1: Indicates that this bit outputs the value of the RTSDT bit to the $\overline{\text{HRTS}}$ pin.

Bit	Bit Name	Initial Value	R/W	Description
6	RTSDT	Undefined	R/W	Serial Port - RTS Port Data Specifies the input/output data level of the serial port $\overline{\text{HRTS}}$ pin. Whether the pin is set for input or output is determined by the RTSIO bit. When the pin is set for output, the value of the RTSDT bit is output to the $\overline{\text{HRTS}}$ pin. Regardless of the value of the RTSIO bit, the value of the $\overline{\text{HRTS}}$ pin is read from the RTSDT bit. The initial value of this bit is undefined after a power-on reset or a manual reset. 0: Indicates that the input/output data is low level. 1: Indicates that the input/output data is high level.
5	CTSIO	0	R/W	Serial Port - CTS Port Input/Output Specifies input or output for the serial port $\overline{\text{HCTS}}$ pin. To actually set the $\overline{\text{HCTS}}$ pin as a port output pin to output the value set by the CTSDT bit, the MCE bit in HSFCR should be cleared to 0. 0: Indicates that the CTSDT bit value is not output to the $\overline{\text{HCTS}}$ pin. 1: Indicates that the CTSDT bit value is output to the $\overline{\text{HCTS}}$ pin.
4	CTSDT	Undefined	R/W	Serial Port - CTS Port Data Specifies the input/output data level of the serial port $\overline{\text{HCTS}}$ pin. Whether the pin is set for input or output is determined by the CTSIO bit. When the pin is set for output, the value of the CTSDT bit is output to the $\overline{\text{HCTS}}$ pin. Regardless of the value of the CTSIO bit, the value of the $\overline{\text{HCTS}}$ pin is read from the CTSDT bit. The initial value of this bit is undefined after a power-on reset or a manual reset. 0: Indicates that the input/output data is low level. 1: Indicates that the input/output data is high level.

Bit	Bit Name	Initial Value	R/W	Description
3	SCKIO	0	R/W	Serial Port – Clock Port Input/Output Specifies input or output for the serial port HSCK pin. To actually set the HSCK pin as a port output pin to output the value set by the SCKDT bit, the CKE1 and CKE0 bits in HSSCR should be cleared to 0. 0: Indicates that the SCKDT bit value is not output to the HSCK pin. 1: Indicates that the SCKDT bit value is output to the HSCK pin.
2	SCKDT	Undefined	R/W	Serial Port – Clock Port Data Specifies the input/output data level of the serial port HSCK pin. Whether the pin is set for input or output is determined by the SCKIO bit. When the pin is set for output, the value of the SCKDT bit is output to the HSCK pin. Regardless of the value of the SCKIO bit, the value of the HSCK pin is read from the SCKDT bit. The initial value of this bit is undefined after a power-on reset or a manual reset. 0: Indicates that the input/output data is low level. 1: Indicates that the input/output data is high level.
1	SPB2IO	0	R/W	Serial Port – Break Input/Output Specifies the output condition of the serial port HTX pin. To actually set the HTX pin as a port output pin to output the value set by the SPB2DT bit, the TE bit in HSSCR should be cleared to 0. 0: Indicates that the SPB2DT bit value is not output to the HTX pin. 1: Indicates that the SPB2DT bit value is output to the HTX pin.

Bit	Bit Name	Initial Value	R/W	Description
0	SPB2DT	Undefined	R/W	Serial Port – Break Data Specifies the input level of the serial port HRX pin and the output level of the HTX pin. The HTX pin output conditions are determined by the SPB2IO bit. When the HTX pin is set for output, the value of the SPB2DT bit is output to the HTX pin. Regardless of the value of the SPB2IO bit, the value of the HRX pin is read from the SPB2DT bit. The initial value of this bit is undefined after a power-on reset or a manual reset. 0: Indicates that the input/output data is low level. 1: Indicates that the input/output data is high level.

32.2.12 Line Status Register (HSLSR)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	TO	—	ORER
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W*	R	R/W*

Bit	Bit Name	Initial Value	R/W	Description
15 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	TO	0	R/(W)*	Timeout Indicates that the receive FIFO data register (HSFRDR) contains fewer bytes than the trigger number for reception and no further data have arrived over at least the time corresponding to the setting of HSSCR[15:14]* since the stop bit for the byte to have been received. 0: Indicates that data is being received or has been successfully received and that there is no receive data in HSFRDR. [Clearing conditions] • A power-on reset or a manual reset is executed. • All the receive data in the HSFRDR has been read, and 0 is written to TO. 1: Indicates that no further receive data has been received (receive timeout). [Setting condition] • The receive FIFO data register (HSFRDR) contains fewer bytes than the trigger number for reception and no further data have arrived over at least the time corresponding to the setting of HSSCR[15:14]* since the stop bit for the byte to have been received.* Note: When the setting is 00, the time is 15 etu. This is equivalent to 1.5 frames in an 8-bit, 1-stop-bit format. etu: Elementary Time Unit (time for transfer of one bit)
1	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
0	ORER	0	R/(W)*	Overrun Error Indicates that an overrun error has occurred in reception and abnormal termination is caused. If an overrun error occurs, the receive data prior to the overrun error is retained in HSFRDR and the data received subsequently is discarded. Any subsequent serial reception is disabled while the ORER flag is 1. To resume data reception after clearing the ORER flag, be sure to first read (or clear) data in the receive FIFO and handle the error, then clear the ORER flag. 0: Indicates that data is being received or has been successfully received. [Clearing conditions] • A power-on reset or a manual reset is executed. • 0 is written to ORER. 1: Indicates that an overrun error has occurred in reception. [Setting condition] • The next serial reception has been completed while HSFRDR is full of 128-byte data. Note: When bit RE in HSSCR is cleared to 0, the ORER flag is not affected and its previous state is retained.

32.2.13 Sampling Rate Register (HSSRR)

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SRE	SRDE	—	—	SRHP[3:0]				—	—	—	SRCYC[4:0]				
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W :	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	SRE	0	R/W	Sampling Rate Register Enable (SRE) 0: When this bit is set to 0, set the SRCYC4 to SRCYC0 bits to 15 (initial value). 1: Validates the setting of the SRCYC4 to SRCYC0 bits.
14	SRDE	0	R/W	Sampling Point Register Enable (SRDE) 0: Invalidates the setting of the SRHP3 to SRHP0 bits and the sampling point will be (S + 1)/2 for an odd sampling rate (S) and S/2 for an even sampling rate. 1: Validates the setting of the SRHP3 to SRHP0 bits.
13, 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 8	SRHP[3:0]	0000	R/W	11 to 8: Sampling Point Register (SRHP) The sampling point can be moved by setting the SDRE bit to 1 and setting a value in these bits. Normally, the sampling point is the point of S/2 or (S + 1)/2 for a sampling rate of S. By setting a signed 4-bit integer in these bits, the sampling point can be shifted by the amount of the specified sampling clock cycles. This will improve the receive margin. When setting a value in these bits, take notice that the sampling point does not become a negative value or it does not exceed the sampling rate. The shifted sampling point must satisfy the setup margin and hold margin.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
4 to 0	SRCYC[4:0]	01111	R/W	Bits 4 to 0: Sampling Rate Register (SRCYC) Data transfer at a desired sampling rate can be enabled by setting the SRE bit to 1 and setting a value in these bits. Set a value of "S - 1" in these bits for a sampling rate of S. Note that the sampling rate must be from 8 to 32 (a value from 7 to 31 can be set in these bits). When the SRE bit is set to 0, set 15 (initial value) in these bits.

32.2.14 Serial Error Register (HSRER)

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PER[6:0]							—	FER[6:0]						
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14 to 8	PER[6:0]	H'00	R	Parity Error Count These bits indicate the number of data items in which a parity error occurred in the receive data stored in the receive FIFO data register (HSFRDR). After the ER bit in HSFSR is set, the value in bits 14 to 8 will be the number of data items in which a parity error occurred. If all 128 bytes of receive data in HSFRDR have parity errors, bits PER6 to PER0 will have the value 0.
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
6 to 0	FER[6:0]	H'00	R	Framing Error Count These bits indicate the number of data items in which a framing error occurred in the receive data stored in the receive FIFO data register (HSFRDR). After the ER bit in HSFSR is set, the value in bits 6 to 0 will be the number of data items in which a framing error occurred. If all 128 bytes of receive data in HSFRDR have framing errors, bits FER6 to FER0 will have the value 0.

32.2.15 RTS Output Active Trigger Register (HSRTGR)

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	RSTRG[6:0]						
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W :	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6 to 0	RSTRG [6:0]	H'0F	R/W	RTS Output Active Trigger Count The $\overline{\text{HRTS}}$ signal goes high when the number of receive data items stored in the receive FIFO data register (HSFRDR) exceeds the value set in these bits. The initial value is 15.

32.2.16 Receive FIFO Data Count Trigger Register (HSRTRGR)

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	RTRG[6:0]						
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W :	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6 to 0	RTRG[6:0]	H'01	R/W	Receive FIFO Data Count Trigger These bits set the receive data item count at which the receive data full (RDF) flag in the serial status register (HSFSR) is set. The RDF flag is set when the number of receive data items stored in the receive FIFO data register (HSFRDR) equals to or exceeds the value set in these bits. The initial value is 1.

32.2.17 Transmit FIFO Data Count Trigger Register (HSTTRGR)

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	TTRG[6:0]						
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6 to 0	TTRG[6:0]	H'08	R/W	Transmit FIFO Data Count Trigger These bits set the untransmitted data item count at which the transmit FIFO data register empty (TDFE) flag in the serial status register (HSFSR) is set. The TDFE flag is set when the number of transmit data items in the transmit FIFO data register (HSFTDR) falls under the value set in these bits due to transmit operations. The initial value is 8.

32.3 Operation

32.3.1 Operation in Asynchronous Serial Communication Mode

In asynchronous serial communication mode, the HSCIF performs serial communication, in which data is transmitted/received in character units using the attached start bit indicating the start of communication and stop bit indicating the end of communication.

Figure 32.2 shows the general format for asynchronous serial communication.

In asynchronous serial communication, the transmission line is usually kept marked (high level). The HSCIF monitors the transmission line, and when it finds a space (low level), it regards the space as a start bit and starts serial communication.

One character in serial communication consists of a start bit (low level), followed by transmit/receive data (LSB-first; from the lowest bit), a parity bit (high or low level), and a stop bit (high level).

During reception in asynchronous mode, the HSCIF performs synchronization at the falling edge of the start bit. Communication data is acquired at the center of each bit because the HSCIF samples data at the S/second pulse of the clock which has a frequency of S times the bit rate, when the sampling rate is S which is set at sampling rate register (HSSRR). (If the sampling rate is odd number, the data is sampled at the (S+1)/second pulse.)

In addition, when the setting of the SRDE bit makes the setting of the sampling point bits effective, the point where the bits are latched can be intentionally moved from the centers of each bit.

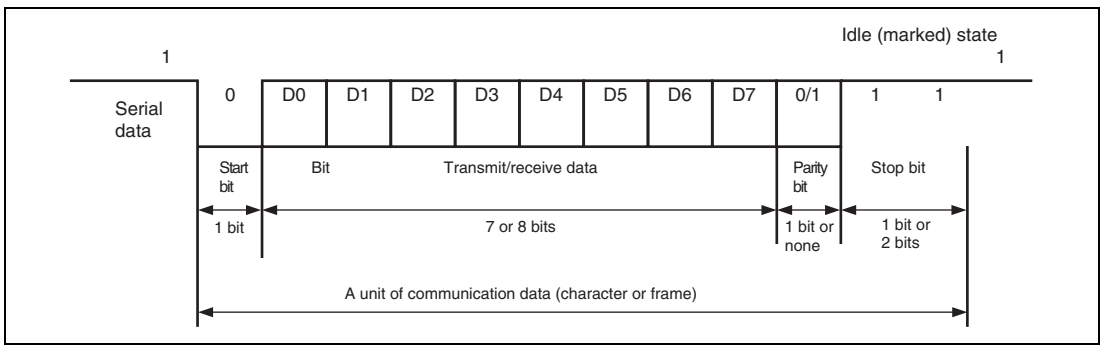


Figure 32.2 Data Format in Asynchronous Mode (Example of 8-Bit Data with Parity and Two Stop Bits)

(1) Transmission/Reception Format and Clock

Table 32.5 shows available data transfer formats. The HSCIF supports 8 transfer formats, which can be specified by HSSMR.

The transfer clock can be selected from internal clock generated by the on-chip baud rate generator or external clock generated by an external clock baud rate generator by using the CKE1 and CKE0 bits in HSSCR.

Table 32.5 Serial Transmission/Reception Formats (Asynchronous Mode)

HSSMR settings			Serial transmission/reception format and frame length											
CHR	PE	STOP	1	2	3	4	5	6	7	8	9	10	11	12
0	0	0	S	8-bit data								STOP		
0	0	1	S	8-bit data								STOP	STOP	
0	1	0	S	8-bit data								P	STOP	
0	1	1	S	8-bit data								P	STOP	STOP
1	0	0	S	7-bit data							STOP			
1	0	1	S	7-bit data							STOP	STOP		
1	1	0	S	7-bit data							P	STOP		
1	1	1	S	7-bit data							P	STOP	STOP	

[Legend]
S : Start bit
STOP : Stop bit
P : Parity bit

(2) Data Transmission/Reception

(a) Initialization of HSCIF (Asynchronous Mode)

Before transmitting/receiving data or changing the operating mode or communication format, the HSCIF should be initialized using the sample flowchart for HSCIF initialization shown in figure 32.3.

[Notes]

Clearing the TE bit to 0 initializes HSTSR. However, HSFSR, HSFTDR, and HSFRDR contents are retained even if the TE and RE bits are cleared to 0.

The TE bit should be cleared to 0 after all transmit data has been sent and the TEND flag has been set in HSFSR. The TE bit can be cleared to 0 during transmission, but the data being transmitted will enter the marked state after clearing. In addition, before setting the TE bit to 1 to restart the transmission, set the TFRST bit to 1 in HSFCR to reset HSFTDR.

When an external clock is used, do not stop the clock during operation or initialization. If stopped, the operation will be unreliable. Furthermore, when the baud rate generator for external clock is also to be used, be sure to make settings for it before starting initialization of the HSCIF.

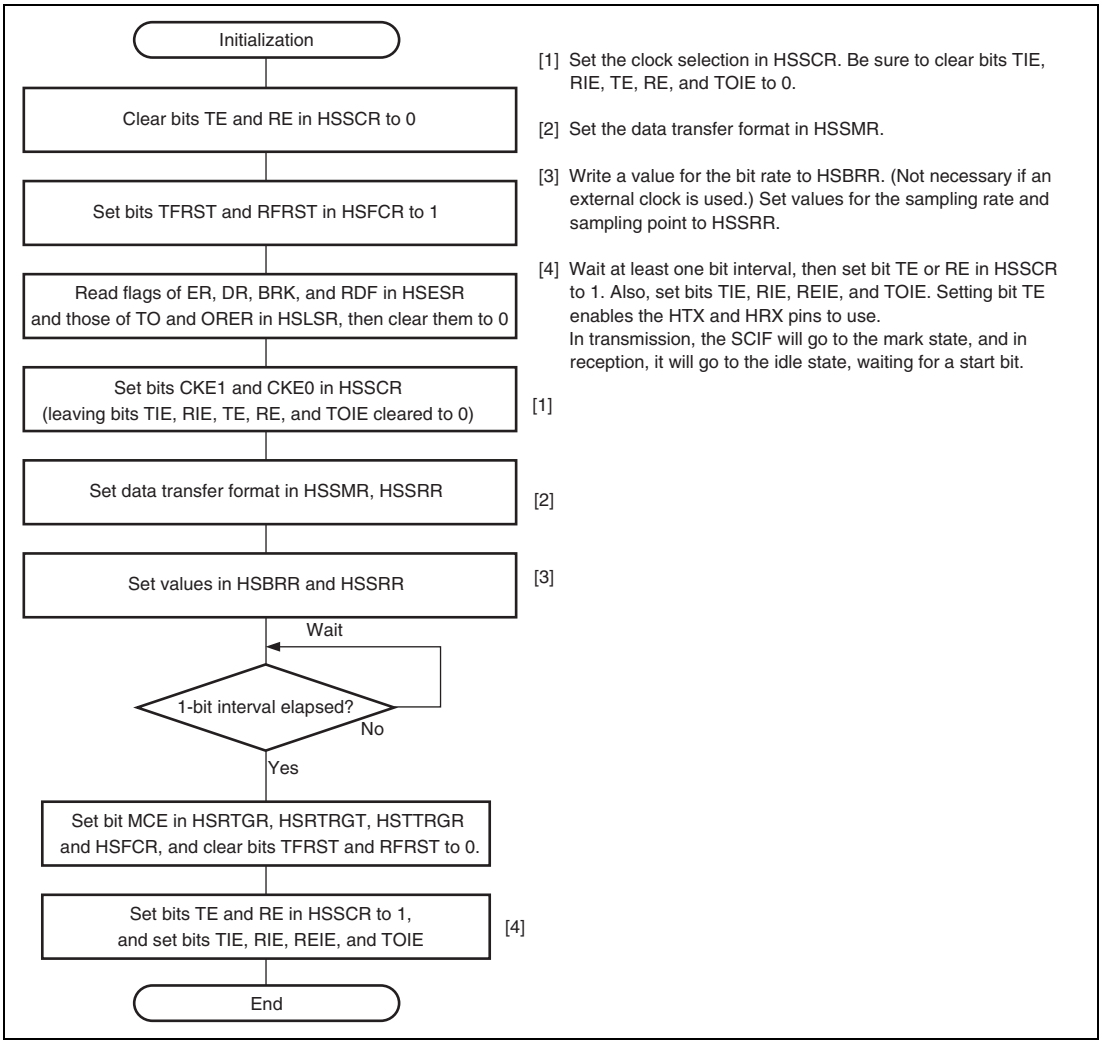


Figure 32.3 Sample Flowchart for Initializing the HSCIF

(b) Serial Data Transmission (Asynchronous Mode)

Figure 32.4 shows a sample flowchart for serial transmission.

After the HSCIF transmission operation is enabled, serial data transmission can be performed using the following procedure:

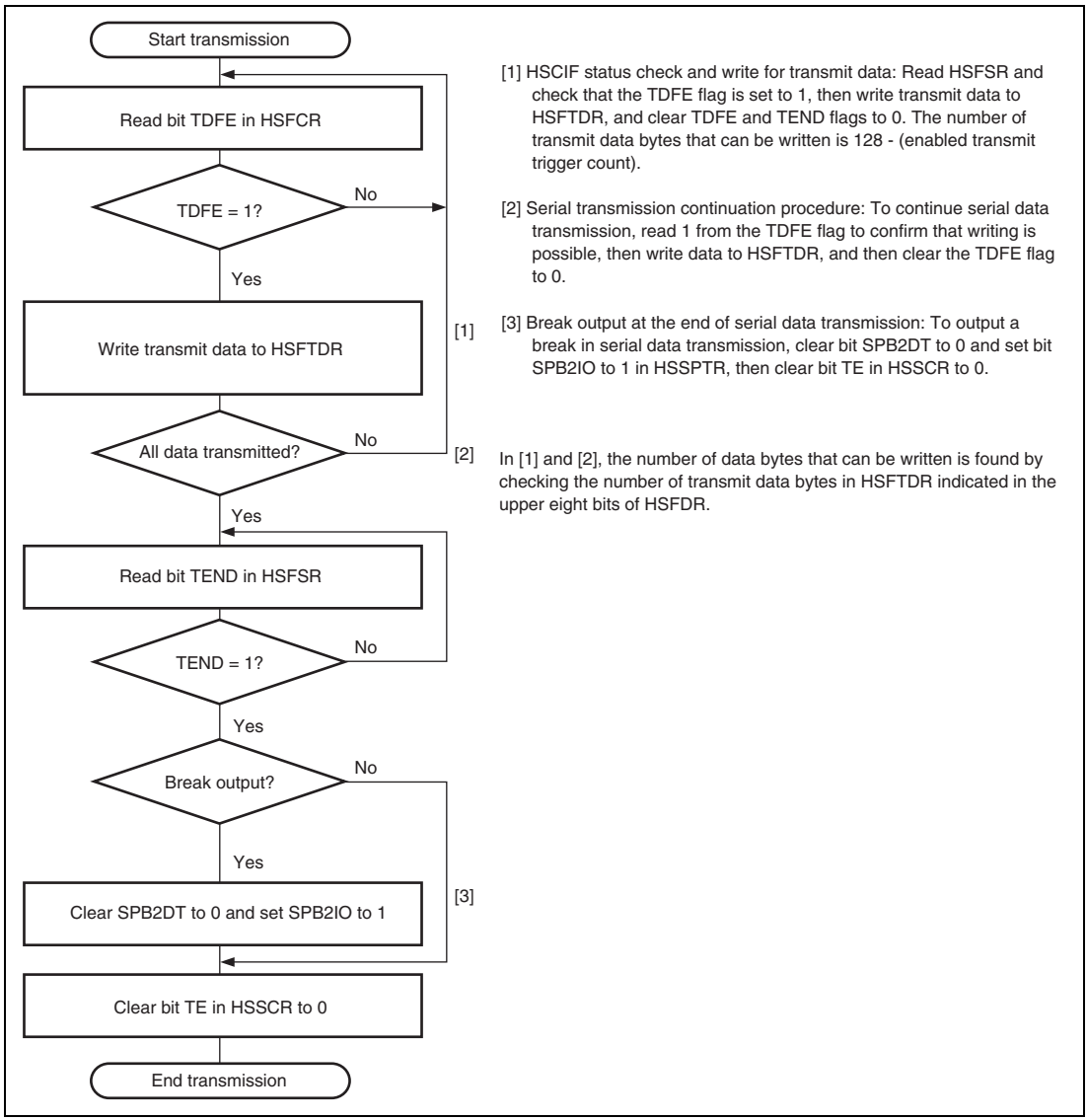


Figure 32.4 Sample Flowchart for Serial Transmission

In serial transmission, the HSCIF operates as follows:

1. When data is written to HSFTDR, the HSCIF transfers the data from HSFTDR to HSTSR and starts transmitting. Confirm that the TDFE flag in HSFSR is set to 1 before writing transmit data to HSFTDR. The number of data bytes that can be written is at least 128 - (transmit trigger count).
2. When data is transferred from HSFTDR to HSTSR and the HSCIF starts transmission, consecutive transmission is performed until there is no transmit data left in HSFTDR. When the number of transmit data bytes in HSFTDR is equal to or less than the transmit trigger count specified in HSTTRGR, the TDFE flag is set. If the TIE and TEIE bits in HSSCR are set to 1 and 0, respectively at this time, a transmit-FIFO-data-empty interrupt (TDFE) request occurs. The serial transmit data is sent from the HTX pin in the following order:
   A. Start bit: One 0-bit is output.
   B. Transmit data: 8- or 7-bit data is output in LSB-first order.
   C. Parity bit: One parity bit (even or odd parity) is output.
      A format that does not output a parity bit can also be selected.
   D. Stop bit(s): One or two 1-bits (stop bits) are output.
   E. Marked state: 1 is output continuously until the start bit that starts the next transmission is sent.
3. The HSCIF checks transmit data in HSFTDR when sending the stop bit. If there is data in it, the HSCIF transfers the data from HSFTDR to HSTSR, sends the stop bit, and then starts serial transmission of the next frame. If there is no transmit data, the HSCIF sets the TEND flag to 1 in HSFSR and sends out the stop bit, and then the marked state is entered to output 1 continuously. At this time, if the TIE and TEIE bits in HSSCR are set to 1 and 0, respectively, a transmit-end interrupt (TEND) request occurs.

Figure 32.5 shows an example of transmission in asynchronous mode.

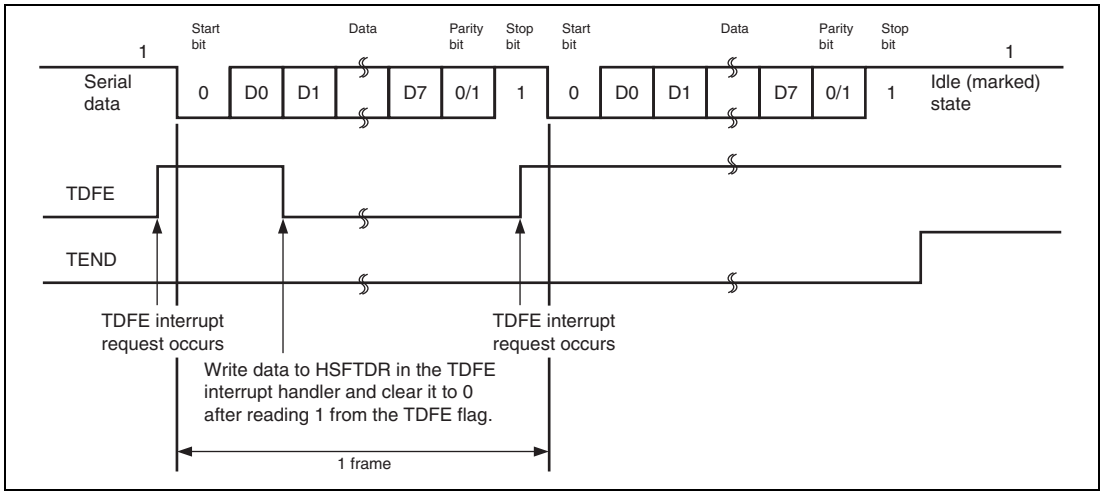


Figure 32.5 Sample HSCIF Transmission Operation (Example of 8-Bit Data with Parity and One Stop Bit)

4. When modem control is enabled, transmission can be stopped or resumed in accordance with the $\overline{\text{HCTS}}$ input value. When $\overline{\text{HCTS}}$ is set to 1 during transmission, the marked state is entered after one frame of data transmission is ended. Setting $\overline{\text{HCTS}}$ to 0 restarts outputting the next transmit data from the start bit. Figure 32.6 shows an example of the operation with modem control enabled.

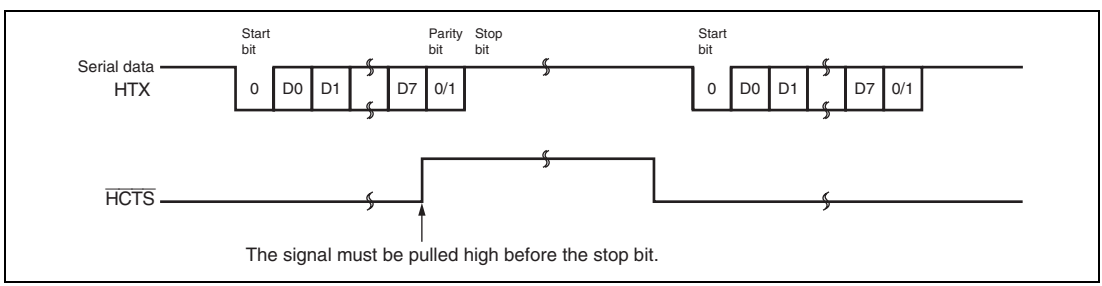


Figure 32.6 Sample Operation with Modem Control Enabled ($\overline{\text{HCTS}}$)

(c) Serial Data Reception (Asynchronous Mode)

Figures 32.7 and 32.8 show sample flowcharts for serial reception.

After the HSCIF reception operation is enabled, serial data reception can be performed using the following procedure:

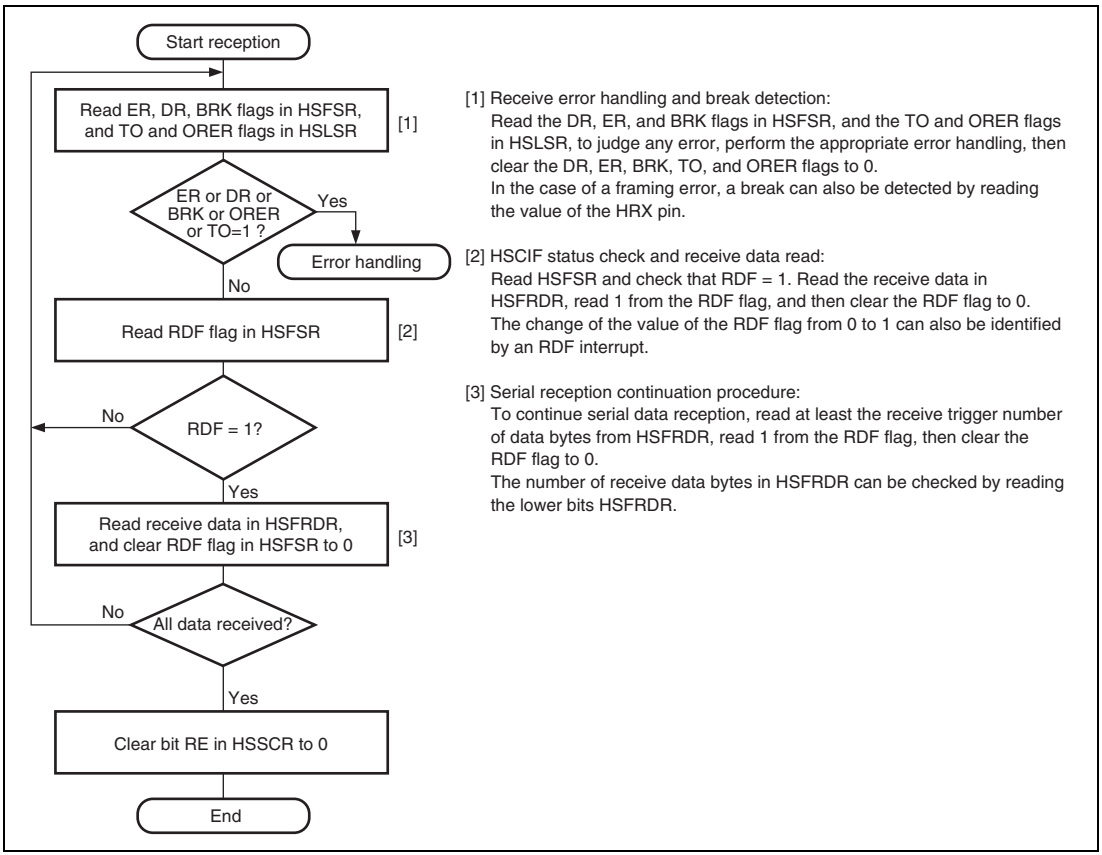
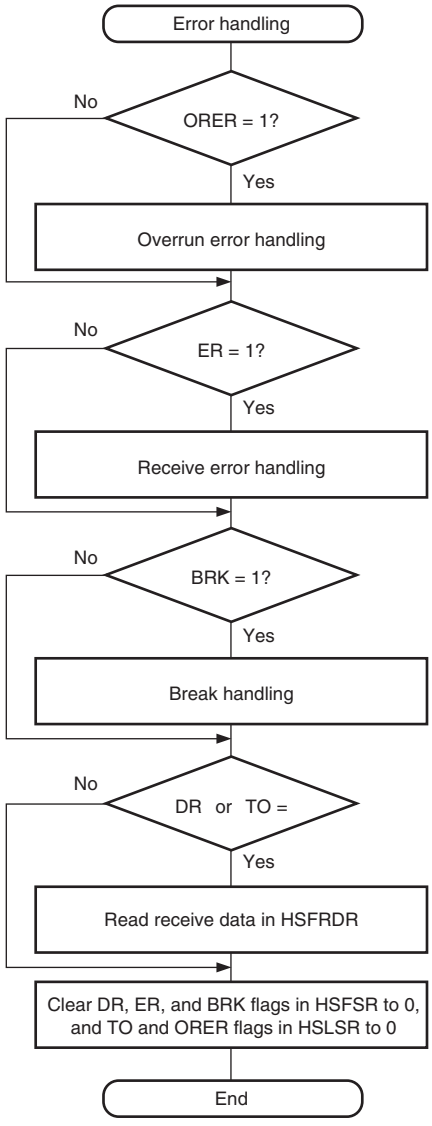


Figure 32.7 Sample Flowchart for Serial Reception (1)



Whether a framing error or parity error has occurred in the receive data that is to be read from HSFRDR can be determined by checking HSRER.

Figure 32.8 Sample Flowchart for Serial Reception (2)

In serial reception, the HSCIF operates as follows:

1. The HSCIF monitors the transmission line, and when detecting the start bit 0, it performs internal synchronization and starts reception.
2. The HSCIF stores the received data in HSRSR in LSB-to-MSB order.
3. The HSCIF receives the parity bit and stop bit.

   After receiving these bits, the HSCIF performs the following checks. If the HSCIF can confirm the conditions of (b), (c), and (d), it stores the receive data in HSFRDR.

Note: The HSCIF continues to receive data even when a parity error or a framing error occurs.

   A. Stop bit: The HSCIF checks whether the stop bit is 1.

      If there are two stop bits, it checks only the first stop bit.

   B. Receive data: The HSCIF checks that receive data can be transferred from the receive shift register (HSRSR) to HSFRDR.

   C. Overrun error: The HSCIF checks that the ORER flag is 0, indicating that no overrun error has occurred.

   D. Break state: The HSCIF checks that the BRK flag is 0, indicating that the break state is not set.

4. If the RD flag changes to 1 while the RIE bit in HSSCR is 1, a receive-FIFO-data-full interrupt (RDF) request occurs.

   If the DR flag changes to 1 while the RIE bit in HSSCR is 1, a receive-data-ready interrupt (DR) request occurs.

   If the TO flag changes to 1 while the TOIE bit in HSSCR is 1, a timeout interrupt (TO) request occurs.

   If the ER flag changes to 1 while the RIE or REIE bit in HSSCR is 1, a receive-error interrupt (ER) request occurs.

   If the BRK flag changes to 1 while the RIE or REIE bit in HSSCR is 1, a break interrupt (BRK) request occurs.

   If the ORER flag changes to 1 while the RIE or REIE bit in HSSCR is 1, an overrun-error interrupt (ORER) request occurs.

Figure 32.9 shows an example of reception in asynchronous mode.

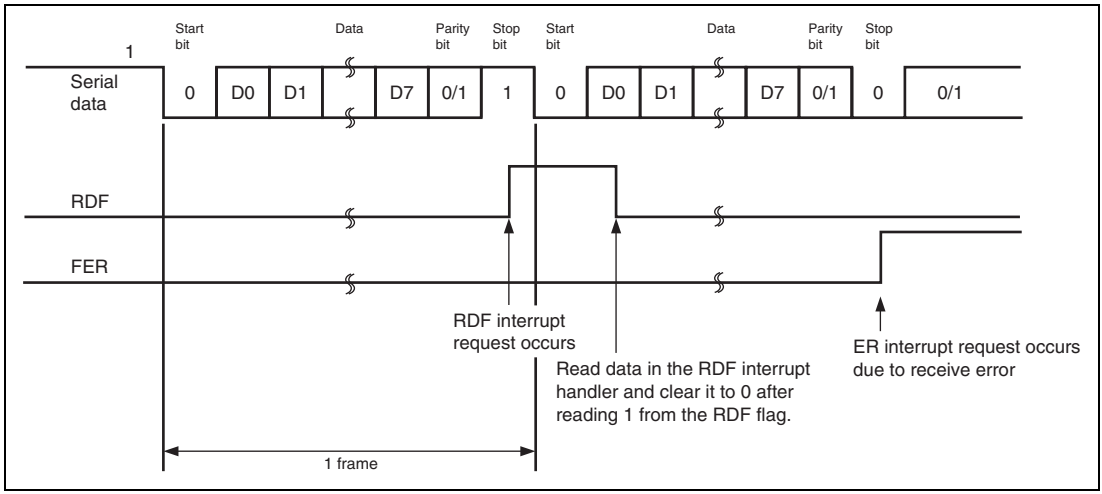


Figure 32.9 Sample HSCIF Receive Operation (Example of 8-Bit Data with Parity and One Stop Bit)

5. When modem control is enabled, the $\overline{\text{HRTS}}$ signal is output when HSFRDR is empty. When $\overline{\text{HRTS}}$ is 0, data can be received. When $\overline{\text{HRTS}}$ is set to 1, the number of data bytes in HSFRDR is equal to or more than the $\overline{\text{HRTS}}$ output active trigger count.

Figure 32.10 shows an example of the operation with modem control enabled.

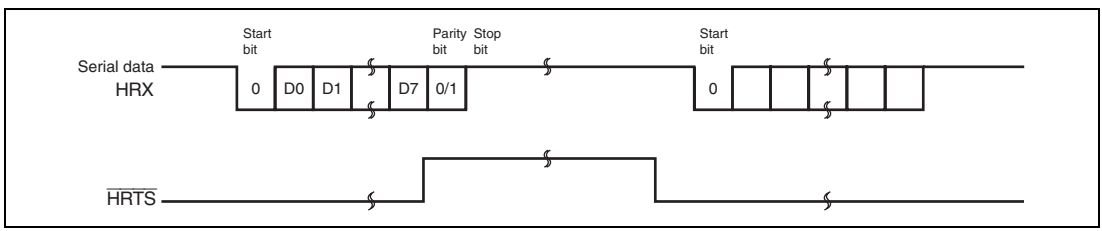


Figure 32.10 Example of the Operation with Modem Control Enabled ($\overline{\text{HRTS}}$)

32.4 HSCIF Interrupt Sources and the DMAC

If the DMAC is used for transmission/reception, set and enable the DMAC before setting the HSCIF.

Transmission Interrupts and DMA Transfer:

If the TDFE/TEND flag in HSFSR is set to 1 when the TDFE/TEND interrupt is enabled by the TIE bit, a TDFE/TEND interrupt request and a transmit-FIFO-data-empty DMA transfer request will occur. If the TDFE/TEND flag is set to 1 when TDFE/TEND interrupt is disabled by the TIE bit, only the transmit-FIFO-data-empty DMA transfer request will occur. (A transmit-FIFO-data-empty DMA transfer request is generated when the TDFE flag is set while TEIE is 0, or when the TEND flag is set while TEIE is 1. DMA transfer requests are not affected by the TEIE bit.)

When TDFE/TEND interrupt requests are enabled, the interrupt requests are cleared by the DMAC regardless of the interrupt handling program.

Reception Interrupts and DMA Transfer:

If the RDF/DR flag in HSFSR is set to 1 when RDF/DR interrupt is enabled by the RIE bit, an RDF/DR interrupt request occurs. If the RDF/DR flag is set to 1, a receive-FIFO-data-full DMA transfer request occurs. If the RDF/DR flag is set to 1 when RDF/DR interrupt is disabled by the RIE bit, and only a receive-FIFO-data-full DMA transfer request occurs and DMAC can be activated to perform data transfer.

Setting the RIE bit in SDCSCR to 0 and the REIE bit to 1 generates the ER/BRK/ORER interrupt requests without generating RDF/DR interrupt requests. When the BRK flag in HSFSR or the ORER flag in HSLSR is set to 1, BRK/ORER interrupt requests occur.

If the TO flag is set to 1 in HSLSR when TO interrupts are enabled by the TOIE bit, TO interrupt requests occur.

When DR/TO interrupt requests are enabled to be issued, interrupt requests generated by the DR flag are cleared by the DMAC regardless of the interrupt handling program, however, those generated by the TO flag are not cleared by the DMAC. Therefore, the TO flag interrupt requests need to be cleared with the interrupt handling program. (The DR and TO flags are set at the same time, but cleared separately.)

Table 32.6 HSCIF Interrupt Sources

Interrupt Source	DMAC Activation	Priority on Reset
Interrupts generated by receive error flag (ER)	Not possible	High
Interrupts generated by receive-FIFO-data-full (RDF), receive-data-ready (DR) or timeout (TO)	Possible	
Interrupts generated by break (BRK) or overrun error (ORER)	Not possible	
Interrupts generated by transmit FIFO data empty (TDFE)	Possible	Low

32.5 Usage Notes

Note the following on use of the HSCIF.

(1) Break Detection and Operation

Break signals can also be detected by reading the HRX pin value directly when a framing error (FER) is detected. In the break state, the input values from the HRX pin are all 0s. So, the parity error flag (PER) may be set after the FER flag is set to 1.

Although the HSCIF stops receive data transfer to HSFRDR after detecting a break, it continues data reception.

(2) Sending a Break Signal

The input/output condition and level of the HTX pin are determined by the SPB2IO and SPB2DT bits in HSSPTR. This enables to send a break signal.

The pin does not function as the HTX pin from the initialization of the serial transmitter to setting of the TE bit (enabling transmission). In this period, the marked state is substituted by the value of the SPB2DT bit. Therefore, the SPB2IO and SPB2DT bits should be set to 1 (designating output and high level) beforehand.

To send a break signal during serial transmission, clear the SPB2DT bit to 0 (low level), and then clear the TE bit to 0 (halting transmission). When the TE bit is cleared, the transmitter is initialized regardless of the current transmission state, and the HTX pin outputs 0.

(3) Data Sampling Timing and Reception Margin in Asynchronous Mode

The HSCIF operates on the base clock with a frequency multiplied by the number which is set as a sampling rate for the bit rate.

In reception, the HSCIF performs the internal synchronization by sampling the falling edge of the start bit using the base clock. In addition, the HSCIF takes receive data at the rising edge of the S/2 pulse (when S is an even number) or (S+1)/2 pulse (when S is an odd number) on the base clock when sampling rate is S.

Figure 32.11 shows the timing of this operation when S = 16.

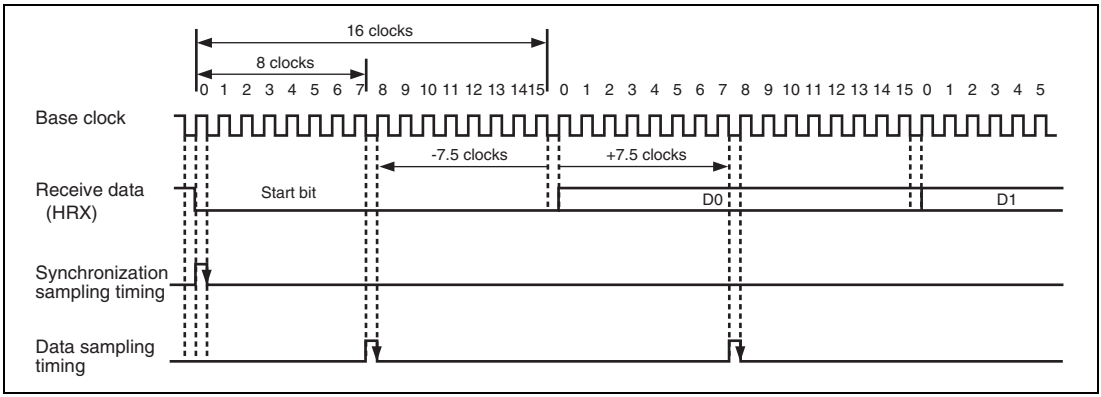


Figure 32.11 Timing Chart of Receive Data Sampling

The reception margin in asynchronous mode is given by equation (1).

$$M = \left| (0.5 - \frac{1}{2N}) - (L - 0.5)\,F - \frac{|D - 0.5|}{N}\,(1 + F) \right| \times 100\% \quad \cdots\cdots \text{Equation (1)}$$

M: Receive margin (%)

N: Ratio of bit rate to clock (N = sampling rate)

D: Clock duty (D = 0 to 1.0)

L: Frame length (L = 9 to 12)

F: Absolute value of clock frequency deviation

Assuming F = 0, D = 0.5 and sampling rate = 16 for equation (1), the reception margin obtained with equation (2) is 46.875% as shown below:

Assuming D = 0.5 and F = 0

$M = (0.5 - 1/(2 \times 16)) \times 100\%$

$= 46.875\%$ …equation (2)

Note this is a theoretical value. A reasonable margin allowed in system designs is 20% to 30%.

(4) Reception Margin and Baud Rate Error

The value of 46.875% obtained by the above equation (2) indicates the reception margin when the baud rate error is 0 (F = 0). If there is no error in the reception and transmission baud rates, reception is possible even with misalignment of approx. 1/2 bit. If there is an error in the reception and transmission baud rates, the errors are accumulated up to the stop bit reception, which reduces the reception margin. The allowable baud rate error can be obtained by modifying the F in expression (1). When D= 0.5:

$F = \{(15/32 - M)/(L - 0.5)\} \times 100\ (\%)$…equation (3)

By using equation (3), the relationship between the allowable error and reception margin with the frame length = 12 can be summarized as follows:

Allowed Error (%)	Reception Margin (%)
4.07	0
3.64	5
3.20	10
2.33	20
1.46	30

(5) Usage Method of SRHP Bits

The method for using the SRHP bits in the sampling rate register is described below.

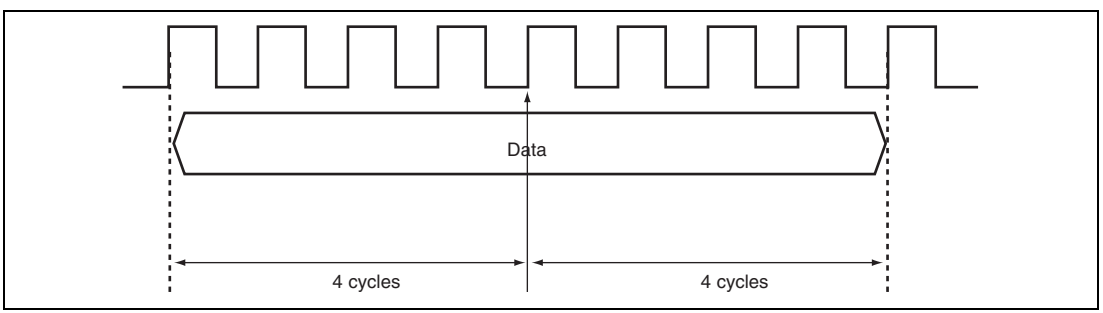


Figure 32.12 Sampling with Invalid SRHP Field

Figure 32.12 shows a sampling example in which the SRHP bits are invalid (sampling rate = 8). In this case, the HSCIF samples data at half of the sampling rate, which is at the rising edge of the fourth pulse of the clock. This allows a setup margin and hold margin of 50% each to be provided.

However, if the ratio between the baud rate and sampling clock is not 1:1, either the setup margin or hold margin is omitted in data reception of a single frame. If the setup margin is omitted, the hold margin increases. On the other hand, if the hold margin is omitted, the setup margin increases. Figures 32.13 and 32.14 show examples.

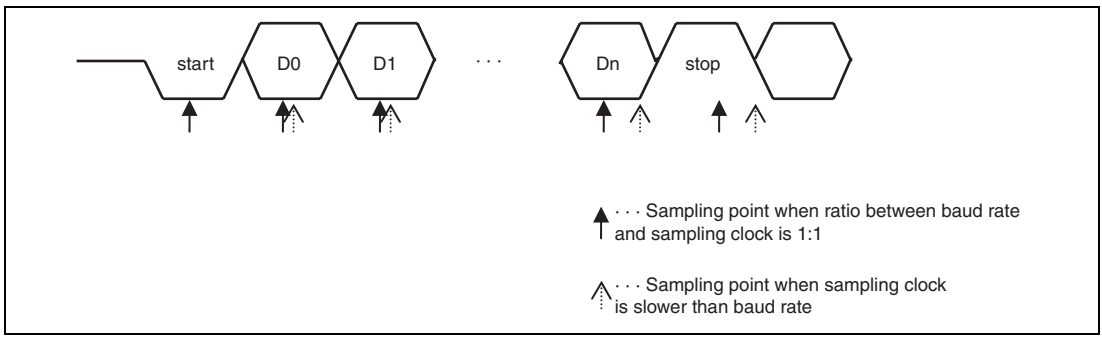


Figure 32.13 Sampling Point with Sampling Clock Slower Than Baud Rate

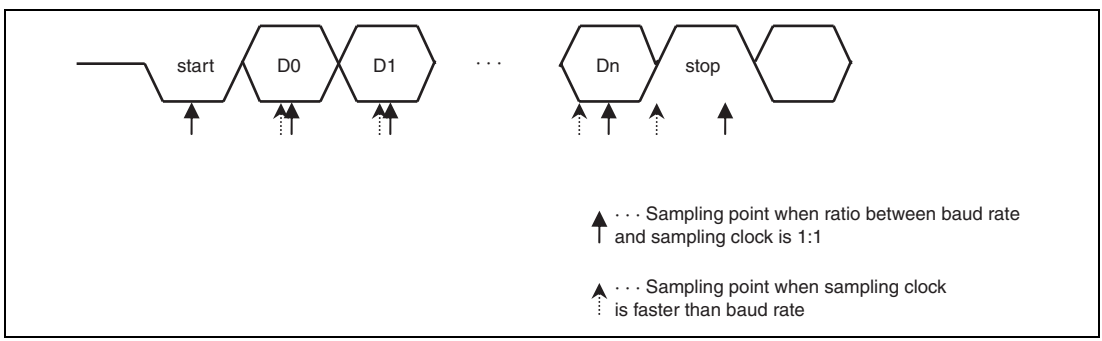


Figure 32.14 Sampling Point with Sampling Clock Faster Than Baud Rate

The ratio between the baud rate and sampling clock can be used to determine which margin among the setup margin and hold margin is to be omitted and which is to be increased. Based on this, the margin within a single frame can be increased by intentionally increasing the margin to be omitted in advance. The sampling point can be moved by setting a value in the SRHP bits. Figure 32.15 shows an example.

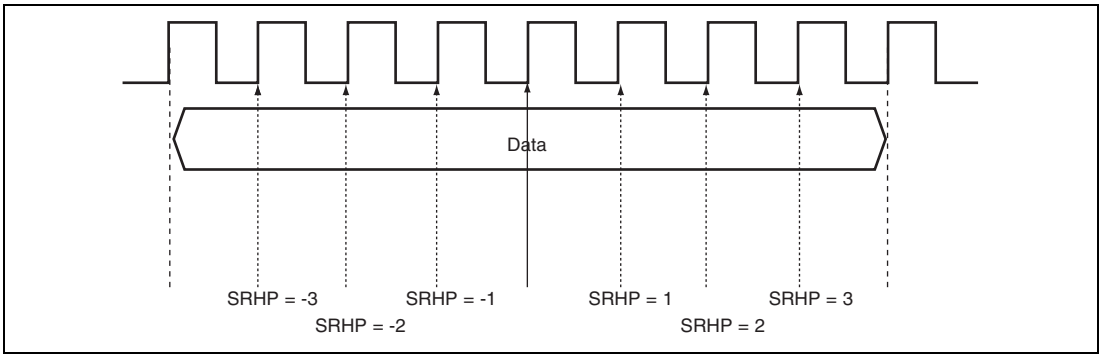


Figure 32.15 Sampling Point Moved by SRHP Bits

(6) Sampling Rate and Bit Rate Settings and Respective Margin

To set the baud rate in the HSCIF, in addition to setting the clock division ratio, the two registers, bit rate register (HSBRR) and sampling rate register (HSSRR), need to be set. These registers must be set so that the margin calculated by equation (1) in section 32.5 (3), Data Sampling Timing and Reception Margin in Asynchronous Mode, will be a sufficient value.

The bit rate error is small when the value obtained by dividing the divided clock frequency used in the HSCIF with the sampling rate is near the desired baud rate. If there are many combinations of bit rate and sampling rate with the same bit rate error, choosing a combination with a large sampling rate provides a large margin. This is because when the bit rate error is constant in equation (1) in section 32.5 (3), Data Sampling Timing and Reception Margin in Asynchronous Mode, that is, when the "absolute value of clock frequency deviation (F)" is constant, the margin (M) increases in accordance with the sampling rate (N).

If the sampling rate is made larger in a combination of bit rate and sampling rate with different bit rate errors, the bit rate error is increased. Accordingly, the "absolute value of clock frequency deviation (F)" on the right side in equation (1) in section 32.5 (3), Data Sampling Timing and Reception Margin in Asynchronous Mode, gets larger and the margin (M) smaller.

Refer to the following procedure when selecting the sampling rate and bit rate settings.

1. Obtain the bit rate with the smallest bit rate error for sampling rates from 8 to 32.
2. Using equation (1) in section 32.5 (3), Data Sampling Timing and Reception Margin in Asynchronous Mode, calculate the margin for each combination of sampling rate and bit rate which was obtained in step 1.
3. Select the combination with the largest margin among the combinations of sampling rate and bit rate.

32.6 Baud Rate Generator for External Clock (BRG)

32.6.1 Overview

The HSCIF incorporates a baud rate generator for external clock (abbreviated as BRG, hereafter). The BRG supplies a sampling clock (BRGCLK) to the HSCIF core by dividing the external clock SC_CLK (selectable between HSCIF_CLK and clks) by 1 to $2^{16} - 1$. In addition, the BRG switches the output between the external clock HSCK and divided clock.

32.6.2 Block Description

Figure 32.16 shows a block diagram of the BRG.

(1) Reset Controller:

The reset controller handles resetting the control register, base counter, and trigger generator.

(2) Control Register:

The control register has the frequency division register and clock select register.

(3) Base Counter:

The base counter is a 16-bit CLK synchronization counter that is used to determine the timing for generating a frequency divided clock.

(4) Trigger Generator:

The trigger generator generates rising-edge/falling-edge triggers for a frequency divided clock with the timing according to values of the frequency division register and base counter. The triggers are used to generate the frequency divided clock. In addition, the trigger generator switches the output between the HSCK (external clock input) and frequency divided clock.

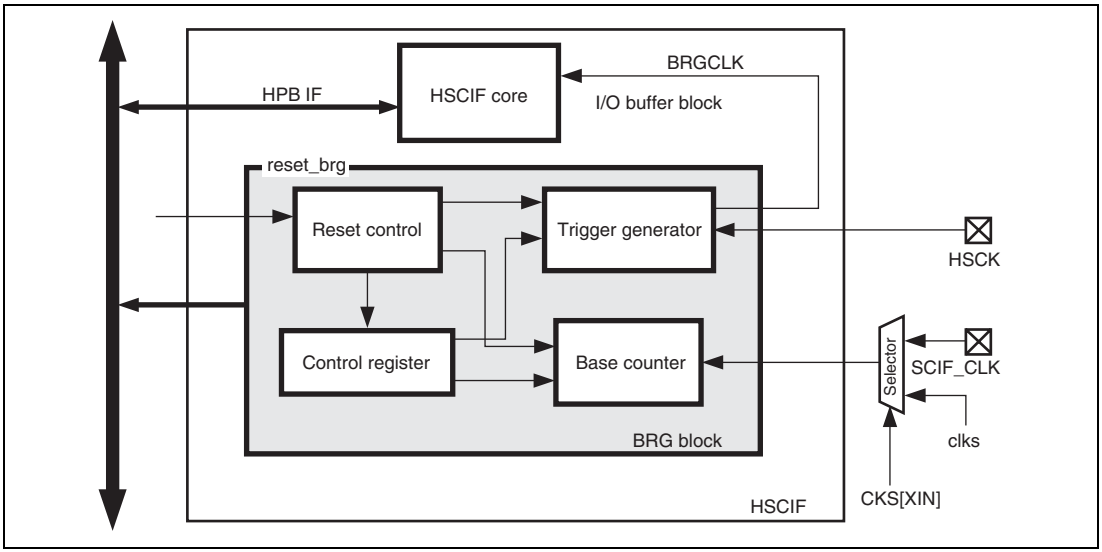


Figure 32.16 BRG Block Diagram

32.6.3 Register Configuration

Table 32.7 shows the registers in the BRG block.

Table 32.7 List of Registers

Name	Code	R/W	Initial Value	Offset From Base Address	Access Size
Frequency division register 0	DL	R/W	H'00	H'30	16
Clock select register 0	CKS	R/W	H'00	H'34	16

Note: For the base address, see section 32.1.4, Register Configuration.

(1) Frequency Division Register (DL)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DL15	DL14	DL13	DL12	DL11	DL10	DL9	DL8	DL7	DL6	DL5	DL4	DL3	DL2	DL1	DL0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	DL[15:0]	H'0000	R/W	Specifies a division value of frequency clock generated in BRG. These bits support a 16-bit binary format that allows specifying a value in the range of 1 to 65535. Setting H'0000 in these bits makes the BRG output the frequency divided clock at the low level. The value of frequency division is given by the following formula: The value of frequency division = (clock input frequency)/(required baud rate × sampling rate) Table 32.8 and table 32.9 show how to use the baud rate generator with a 3.6864-MHz crystal resonator, and a 26-MHz crystal resonator.

(2) Clock Select Register (CKS)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CKS	XIN	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	CKS	0	R/W	This bit switches the output between the frequency divided clock (SC_CLK) and external clock (HSCK). 0: Selects the frequency divided clock. 1: Selects the external clock.
14	XIN	0	R/W	Selects the clock source for the baud rate generator for external clock from SCIF_CLK or clks. 0: Selects the external clock (SCIF_CLK). 1: Selects the internal clock (clks).
13 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Table 32.8 Baud Rate (3.6864-MHz clock)

Baud Rate	Value of Frequency Division	Sampling Rate	Error Rate (*)
50	4608	16	—
75	3072	16	—
110	4189	8	–0.0022
134.5	1713	16	0.001
150	1536	16	—
300	768	16	—
600	384	16	—
1200	192	16	—
1800	128	16	—
2000	123	15	0.098
2400	96	16	—
3600	64	16	—
4800	48	16	—
7200	32	16	—
9600	24	16	—
14400	16	16	—
19200	12	16	—
38400	6	16	—
76800	3	16	—
115200	2	16	—

Note: * —: Indicates that the error rate is 0.

Table 32.9 Baud Rate (26-MHz clock)

Baud Rate	Value of Frequency Division	Sampling Rate	Error Rate (*)
9600	129	21	0.025
19200	52	26	–0.160
38400	26	26	–0.160
57600	15	30	–0.309
115200	9	25	–0.309
230400	4	28	–0.756
460800	2	28	–0.756
921600	1	28	–0.756
1843200	1	14	–0.756
3250000	1	8	—

Note: * —: Indicates that the error rate is 0.

32.6.4 Notes On Setting Frequency Division Register

1. For the initial setting of the frequency division register after a reset, at least one bit of waiting period is required to secure the clock stabilization time.

   (Example) One bit period when DL = 2

   3.68 (MHz) × 1/2 × 1/16 = 0.115 (MHz) → 8695 (ns)

2. For modifying the register value after the setting stated in <1> above, at least one bit of waiting period at the maximum bit rate (DL = 65535) is required.

   The HSCIF registers and BRG registers should be set as the following table:

- Asynchronous mode (SC_CLK external input)

HSCIF	Register Name	Setting Value	BRG	Register Name	Setting Value
	HSSCR.CKE1, CKE0	10		CKS	H'0000
				DL	H'1 to H'FFFF

- Asynchronous mode (SCK external input)

HSCIF	Register Name	Setting Value	BRG	Register Name	Setting Value
	HSSCR.CKE1, CKE0	10		CKS	H'8000
				DL	Don't care

3. The register settings for the baud rate generator for external clock should be made before starting initialization of the HSCIF.

Section 33 Sampling Rate Converter (SRC)

The sampling rate converter (SRC) converts the sampling rate for data produced by decoders such as WMA, MP3, or AAC.

33.1 Features

- Data size: 16 bits (stereo/monaural)
- Sampling rates

  Input: Either 8 kHz, 11.025 kHz, 12 kHz, 16 kHz, 22.05 kHz, 24 kHz, 32 kHz, 44.1 kHz, or 48 kHz is selectable.

  Output: Either 32 kHz, 44.1 kHz or 48 kHz is selectable.
- Processing capacity: A maximum of 10 μs sample output interval (Pch = 51 MHz)
- SNR: 93 db or higher
- Three interrupt sources: Input data FIFO empty, output data FIFO full, and output data FIFO overwrite
- Two DMA transfer sources: Input data FIFO empty and output data FIFO full
- Module standby mode

  Power consumption can be reduced by stopping clock supply to the SRC when not used.

Figure 33.1 shows a block diagram of the SRC.

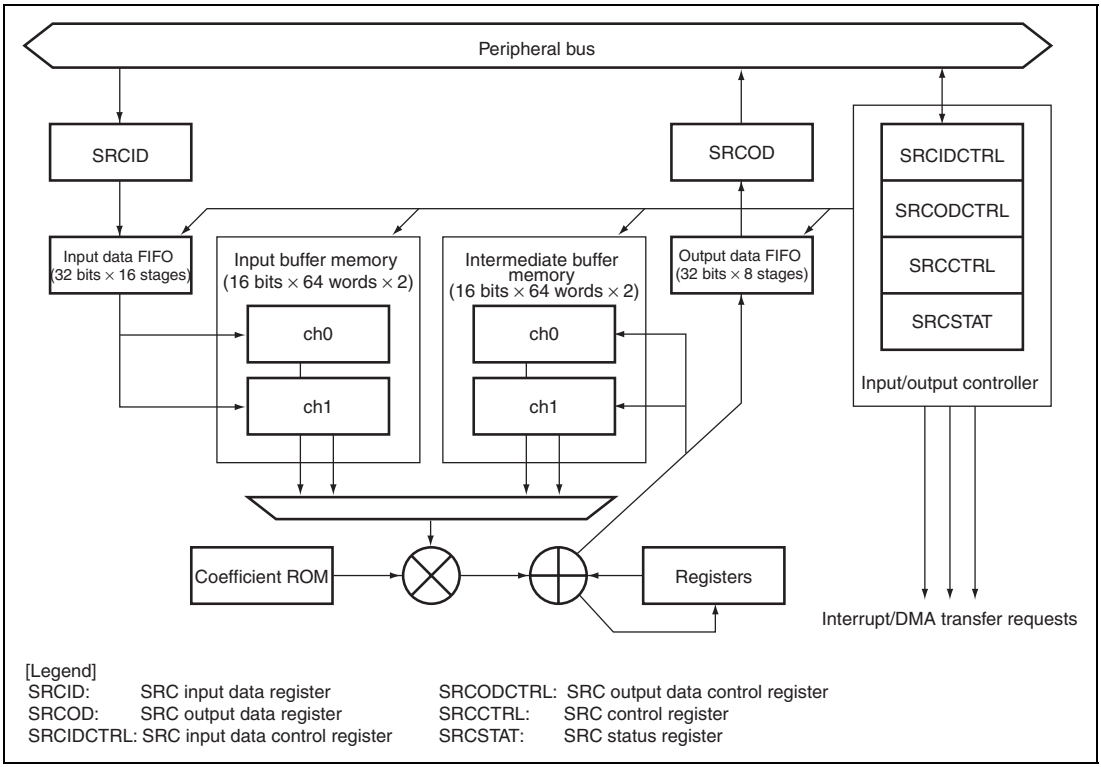


Figure 33.1 Block Diagram of SRC

33.2 Register Descriptions

The SRC has the following registers:

Table 33.1 Register Configuration

Channel	Register Name	Abbreviation	R/W	P4 Address	Area 7 Address	Access Size
0	SRC input data register	SRCID_0	R/W	H'FFF2 0000	H'1FF2 0000	16, 32
	SRC output data register	SRCOD_0	R	H'FFF2 0004	H'1FF2 0004	16, 32
	SRC input data control register	SRCIDCTRL_0	R/W	H'FFF2 0008	H'1FF2 0008	16
	SRC output data control register	SRCODCTRL_0	R/W	H'FFF2 000A	H'1FF2 000A	16
	SRC control register	SRCCTRL_0	R/W	H'FFF2 000C	H'1FF2 000C	16
	SRC status register	SRCSTAT_0	R/(W)*	H'FFF2 000E	H'1FF2 000E	16
1	SRC input data register	SRCID_1	R/W	H'FFF3 0000	H'1FF3 0000	16, 32
	SRC output data register	SRCOD_1	R	H'FFF3 0004	H'1FF3 0004	16, 32
	SRC input data control register	SRCIDCTRL_1	R/W	H'FFF3 0008	H'1FF3 0008	16
	SRC output data control register	SRCODCTRL_1	R/W	H'FFF3 000A	H'1FF3 000A	16
	SRC control register	SRCCTRL_1	R/W	H'FFF3 000C	H'1FF3 000C	16
	SRC status register	SRCSTAT_1	R/(W)*	H'FFF3 000E	H'1FF3 000E	16

Note: * Bits 15 to 3 are read-only. Only 0 can be written to bits 2 to 0 after having read as 1. When bit 2 is not cleared, always write 1 to the bit. Writing 1 to bits 2 to 0 does not change their respective values.

Table 33.2 State of Registers in Each Operating Mode (Common to each channel)

Abbreviation	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
SRCID	H'0000 0000	H'0000 0000	Retained	Retained	Retained	H'0000 0000
SRCOD	H'0000 0000	H'0000 0000	Retained	Retained	Retained	H'0000 0000
SRCIDCTRL	H'0000	H'0000	Retained	Retained	Retained	H'0000
SRCODCTRL	H'0000	H'0000	Retained	Retained	Retained	H'0000
SRCCTRL	H'0000	H'0000	Retained	Retained	Retained	H'0000
SRCSTAT	H'0002	H'0002	Retained	Retained	Retained	H'0002

33.2.1 SRC Input Data Register (SRCID)

SRCID is a 32-bit readable/writable register that is used to input the data before sampling rate conversion. All the bits are read as 0. The data input to SRCID is stored in the 16-stage input data FIFO. When the number of data in input data FIFO is 16, writing to SRCID is invalid. For stereo data, bits 31 to 16 are for ch 0 data, and bits 15 to 0 are for ch 1 data. For monaural data, data in bits 31 to 16 is valid, and data in bits 15 to 0 is invalid.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

The data subject to sampling rate conversion is aligned differently depending on the IED bit setting in SRCIDCTRL. Table 33.3 shows the relationship between the IED bit setting and data alignment.

Table 33.3 Alignment of Data before Sampling Rate Conversion

IED	ch0[15:8]	ch0[7:0]	ch1[15:8]	ch1[7:0]
0	SRCID[31:24]	SRCID[23:16]	SRCID[15:8]	SRCID[7:0]
1	SRCID[23:16]	SRCID[31:24]	SRCID[7:0]	SRCID[15:8]

33.2.2 SRC Output Data Register (SRCOD)

SRCOD is a 32-bit read-only register used to output the data after sampling rate conversion. The data in 8-stage output data FIFO is read through SRCOD.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Initial value:	0*3	0*3	0*3	0*3	0*3	0*3	0*3	0*3	0*3	0*3	0*3	0*3	0*3	0*3	0*3	0*3
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial value:	0*3	0*3	0*3	0*3	0*3	0*3	0*3	0*3	0*3	0*3	0*3	0*3	0*3	0*3	0*3	0*3
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

The data in SRCOD is aligned differently depending on the OCH and OED bit setting in SRCODCTRL. Table 33.4 shows the correspondence between the OCH and OED bit setting and data alignment in SRCOD.

Table 33.4 Alignment of Data in SRCOD

OCH	OED	SRCOD[31:24]	SRCOD[23:16]	SRCOD[15:8]	SRCOD[7:0]
0	0	ch0[15:8]	ch0[7:0]	ch1[15:8]*2	ch1[7:0]*2
	1	ch0[7:0]	ch0[15:8]	ch1[7:0]*2	ch1[15:8]*2
1*1	0	ch1[15:8]	ch1[7:0]	ch0[15:8]	ch0[7:0]
	1	ch1[7:0]	ch1[15:8]	ch0[7:0]	ch0[15:8]

Notes: 1. When processing monaural data, do not set the bit to 1.
2. When processing monaural data, the data in these bits is invalid.
3. If the CL bit in the SRCCTRL register is read after 1 is written to it, it is read as 0. If the CL bit is read before 1 is written to it, the read value cannot be guaranteed.

33.2.3 SRC Input Data Control Register (SRCIDCTRL)

SRCIDCTRL is a 16-bit readable/writable register that specifies the endian format of input data, enables/disables the interrupt requests, and specifies the triggering number of data units.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	IED	IEN	-	-	-	-	-	-	IFTRG[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	IED	0	R/W	Input Data Endian Specifies the endian format of the input data. 0: Big endian 1: Little endian
8	IEN	0	R/W	Input Data FIFO Empty Interrupt Enable Enables/disables the input data FIFO empty interrupt request to be issued when the number of data units in the input FIFO becomes equal to or smaller than the triggering number specified by the IFTRG1 and IFTRG0 bits, thus resulting in the IINT bit in the SRC status register (SRCSTAT) being set to 1. 0: Input data FIFO empty interrupt is disabled. 1: Input data FIFO empty interrupt is enabled.
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1, 0	IFTRG[1:0]	00	R/W	Input FIFO Data Triggering Number Specifies the condition in terms of the number on which the IINT bit in the SRC status register (SRCSTAT) is set to 1. When the number of data units in the input FIFO becomes equal to or smaller than the triggering number listed below, the IINT bit is set to 1. 00: 0 01: 4 10: 8 11: 12

33.2.4 SRC Output Data Control Register (SRCODCTRL)

SRCODCTRL is a 16-bit readable/writable register that specifies whether to exchange the channels for the output data, specifies the endian format of output data, enables/disables the interrupt requests, and specifies the triggering number of data units.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	OCH	OED	OEN	-	-	-	-	-	-	OFTRG[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10	OCH	0	R/W	Output Data Channel Exchange Specifies whether to exchange the channels for the SRC output data register (SRCOD). When processing monaural data, do not set this bit to 1. 0: Does not exchange the channels (the same order as data input) 1: Exchanges the channels (the opposite order from data input)

Bit	Bit Name	Initial Value	R/W	Description
9	OED	0	R/W	Output Data Endian Specifies the endian format of the output data. 0: Big endian 1: Little endian
8	OEN	0	R/W	Output Data FIFO Full Interrupt Enable Enables/disables the output data FIFO full interrupt request to be issued when the number of data units in the output FIFO becomes equal to or greater than the number specified by the OFTRG1 and OFTRG0 bits, thus resulting in the OINT bit in SRC status register (SRCSTAT) being set to 1. 0: Output data FIFO full interrupt is disabled. 1: Output data FIFO full interrupt is enabled.
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	OFTRG[1:0]	00	R/W	Output FIFO Data Trigger Number Specifies the condition in terms of the number on which the OINT bit in the SRC status register (SRCSTAT) is set to 1. When the number of data units in the output FIFO becomes equal to or greater than the number listed below, the OINT bit is set to 1. 00: 1 01: 2 10: 4 11: 6

33.2.5 SRC Control Register (SRCCTRL)

SRCCTRL is a 16-bit readable/writable register that enables/disables the SRC module operation, enables/disables the interrupt requests, and specifies flush processing, clear processing of the internal work memory, and the input and output sampling rates.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	SRCEN	-	EEN	FL	CL	IFS[3:0]				-	-	OFS[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	SRCEN	0	R/W	SRC Module Enable Enables/disables the SRC module operation. 0: Disables the SRC module operation. 1: Enables the SRC module operation. Note: When the SRCEN bit is 1, do not modify the following bits:
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Register Name	Bit	Bit Name
SRCIDCTRL	9	IED
SRCODCTRL	9, 10	OCH, OED
SRCCTRL	7 to 4, 0	IFS[3:0], OFS

Bit	Bit Name	Initial Value	R/W	Description
10	EEN	0	R/W	Output Data FIFO Overwrite Interrupt Enable Enables/disables the output data FIFO overwrite interrupt request to be issued when the data in the output FIFO has been overwritten before being read thus setting the OVF bit in SRC status register (SRCSTAT) to 1. 0: Output data FIFO overwrite interrupt is disabled. 1: Output data FIFO overwrite interrupt is enabled.
9	FL	0	R/W	Internal Work Memory Flush Writing 1 to this bit starts converting the sampling rate of all the data in the input FIFO, input buffer memory, and intermediate memory (i.e., flush processing). This bit is always read as 0. When SRCEN = 0, writing 1 to this bit does not trigger flush processing. If this bit is set to 1 while the number of data units in the input buffer memory is less than 64, the flush processing is not performed because valid output data cannot be obtained.
8	CL	0	R/W	Internal Work Memory Clear Writing 1 to this bit clears the input FIFO, output FIFO, input buffer memory, intermediate memory, and accumulator. This bit is always read as 0. Before operating the SRC, the SRC should be internally cleared by writing 1 to this bit. To perform the clearing processing correctly, wait 32 cycles of peripheral bus clock after wring 1 to this bit and then perform the next processing. In addition, when this bit is set to 1, IFS[3:0] and OFS should also be set.

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	IFS[3:0]	0000	R/W	Input Sampling Rate Specifies the input sampling rate. 0000: 8.0 kHz 0001: 11.025 kHz 0010: 12.0 kHz 0011: Setting prohibited 0100: 16.0 kHz 0101: 22.05 kHz 0110: 24.0 kHz 0111: Setting prohibited 1000: 32.0 kHz 1001: 44.1 kHz 1010: 48.0 kHz 1011: Setting prohibited 1100: Setting prohibited 1101: Setting prohibited 1110: Setting prohibited 1111: Setting prohibited
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	OFS[1:0]	00	R/W	Output Sampling Rate Specifies the output sampling rate. 00: 44.1 kHz 01: 48.0 kHz 10: 32.0 kHz 11: Setting prohibited

The number of output data units obtained as conversion result can be calculated by using the following expression (A) or (B). Table 33.5 shows the relationship of setting value and applicable formula between IFS and OFS[1:0].

$$\text{Number of output data} = \text{Number of input data} \times \frac{\text{Output sampling rate}}{\text{Input sampling rate}} \quad \text{... (A)}$$

$$\text{Number of output data} = \text{Number of input data} \times \frac{\text{Output sampling rate}}{\text{Input sampling rate}} - 1 \quad \text{... (B)}$$

Table 33.5 Relationship between Sampling Rate Setting and Number of Output Data

OFS[1:0]	IFS [3:0]Setting (Input Sampling Rate [kHz])								
Value (Output Sampling Rate (kHz))	0000 (8.0)	0001 (11.025)	0010 (12.0)	0100 (16.0)	0101 (22.05)	0110 (24.0)	1000 (32.0)	1001 (44.1)	1010 (48.0)
00 (44.1)	B	A	A	B	A	A	B	—	A
01 (48.0)	B	B	A	B	B	A	B	B	—
10 (32.0)	A	B	B	A	B	A	—	B	A

33.2.6 SRC Status Register (SRCSTAT)

SRCSTAT is a 16-bit readable/writable register that indicates the number of data units in the input and output data FIFOs, whether the various interrupt sources have been generated or not, and the flush processing status.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OFDN[3:0]				IFDN[4:0]					-	-	FLF	-	OVF	IINT	OINT
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/(W)*	R/(W)*	R/(W)*

Note: * Only 0 can be written after having read as 1.

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	OFDN[3:0]	0000	R	Output FIFO Data Count Indicates the number of data units in the output FIFO.

Bit	Bit Name	Initial Value	R/W	Description
11 to 7	IFDN[4:0]	00000	R	Input FIFO Data Count Indicates the number of data units in the input FIFO.
6, 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	FLF	0	R	Flush Processing Status Flag Indicates whether flush processing is in progress or not. [Clearing conditions] • When flush processing has been completed. • When 1 has been written to the CL bit in SRCCTRL. [Setting condition] • When 1 has been written to the FL bit in SRCCTRL.
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	OVF	0	R/(W)*	Output Data FIFO Overwrite Interrupt Request Flag Indicates that the sampling rate conversion for the next data has been completed when there are eight units of data in the output FIFO. The sampling rate conversion stops until the output data FIFO becomes not full after the SRC output data register (SRCOD) has been read. [Clearing condition] • When 0 has been written to the OVF bit after reading OVF = 1 (however, when the OVF bit is not cleared, always write 1 to this bit. Writing 1 to this bit does not change its value). • When 1 has been written to the CL bit in SRCCTRL. [Setting condition] • When the sampling rate conversion for the next data has been completed when there are eight units of data in the output FIFO.

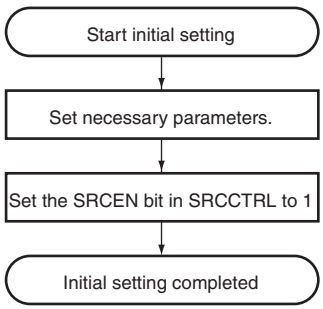
Bit	Bit Name	Initial Value	R/W	Description
1	IINT	1	R/(W)*	Input Data FIFO Empty Interrupt Request Flag Indicates that the number of data units in the input FIFO has become equal to or smaller than the triggering number specified by the IFTRG1 and IFTRG0 bits in the SRC input data control register (SRCIDCTRL). [Clearing conditions] • When 0 has been written to the IINT bit after reading IINT = 1. • When the DMAC has transferred data to the input FIFO resulting in the number of data units in the FIFO exceeding that of the specified triggering number. [Setting condition] • When the number of data units in the input FIFO has become equal to or smaller than the specified triggering number. • When 1 has been written to the CL bit in SRCCTRL.
0	OINT	0	R/(W)*	Output Data FIFO Full Interrupt Request Flag Indicates that the number of data units in the output FIFO has become equal to or greater than the triggering number specified by the OFTRG[1:0] bits in the SRC output data control register (SRCODCTRL). [Clearing conditions] • When 0 has been written to the OINT bit after reading OINT = 1. • When the DMAC has transferred data from the output FIFO resulting in the number of data units in the FIFO being less than the specified triggering number. [Setting condition] • When the number of data units in the output FIFO has become equal to or greater than the specified triggering number.

Note: * Only 0 can be written after having read as 1.

33.3 Operation

33.3.1 Initial Setting

Figure 33.2 shows a sample flowchart for initial setting.



Register	Bit	Items to be Set
SRCCTRL	EEN	Enabling/disabling of the OVF interrupt
	IFS[3:0]	Input sampling rate
	OFS	Output sampling rate
SRCIDCTRL	IED	Input data endian
	IEN	Enabling/disabling of the IDE interrupt
	IFTRG[1:0]	Input data FIFO triggering number
SRCODCTRL	OCH	Exchanging of output data channels
	OED	Output data endian
	OEN	Enabling/disabling of the ODF interrupt
	OFTRG[1:0]	Output data FIFO triggering number

Figure 33.2 Sample Flowchart for Initial Setting

33.3.2 Data Input

Figure 33.3 is a sample flowchart for data input.

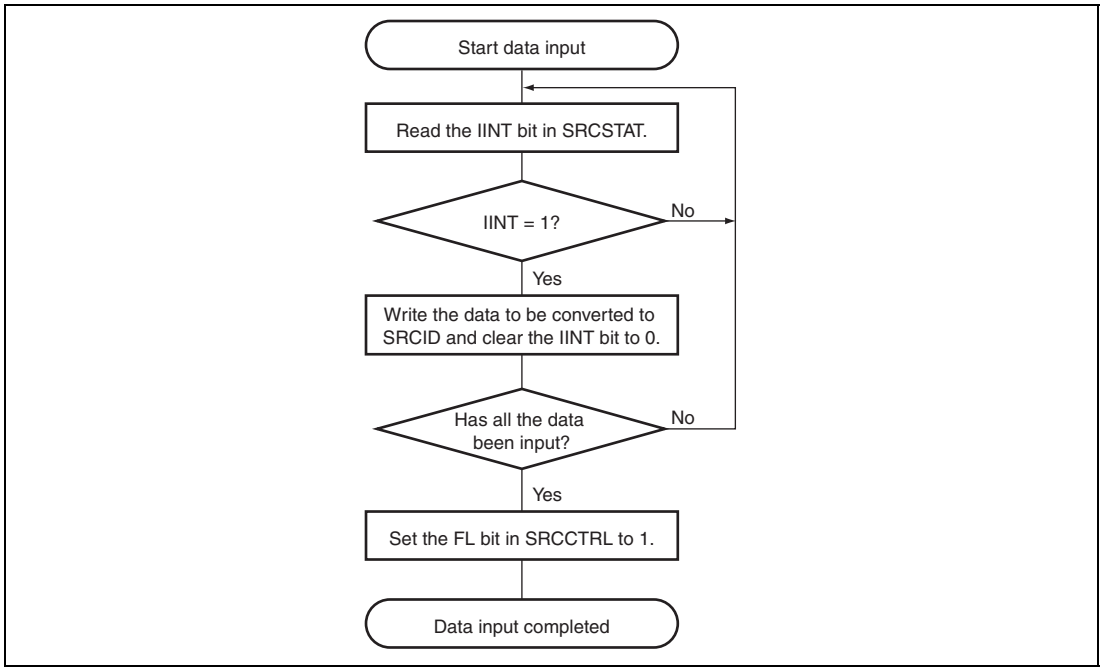


Figure 33.3 Sample Flowchart for Data Input

(1) When Interrupts are Issued to CPU

1. Set the IEN bit in SRCIDCTRL to 1.
2. Set the interrupt controller.
3. When the IINT bit in SRCSTAT is set to 1, the IDE interrupt request is issued. In the interrupt processing routine, read the IINT bit and confirm that it is 1, write data to SRCID, and write 0 to the IINT bit. Then return from the interrupt processing routine.
4. Repeat step 3 until all the data has been input, and write 1 to the FL bit in SRCCTRL.

(2) When Interrupts are Used to Activate DMAC

1. Assign IDEI of the SRC to one channel of the DMAC.
2. Set the IEN bit in SRCIDCTRL to 1.
3. When the IINT bit in SRCSTAT is set to 1, the IDE interrupt request is issued thus activating the DMAC. When the DMAC has written data to the SRCID thus resulting in the number of data units in the input data FIFO exceeding that of the triggering number specified by the IFTRG1 and IFTRG 0 bits in SRCIDCTRL, the IINT bit is cleared to 0.
4. Repeat step 3 until all the data has been input, and write 1 to the FL bit in SRCCTRL.

33.3.3 Data Output

Figure 33.4 is a sample flowchart for data output.

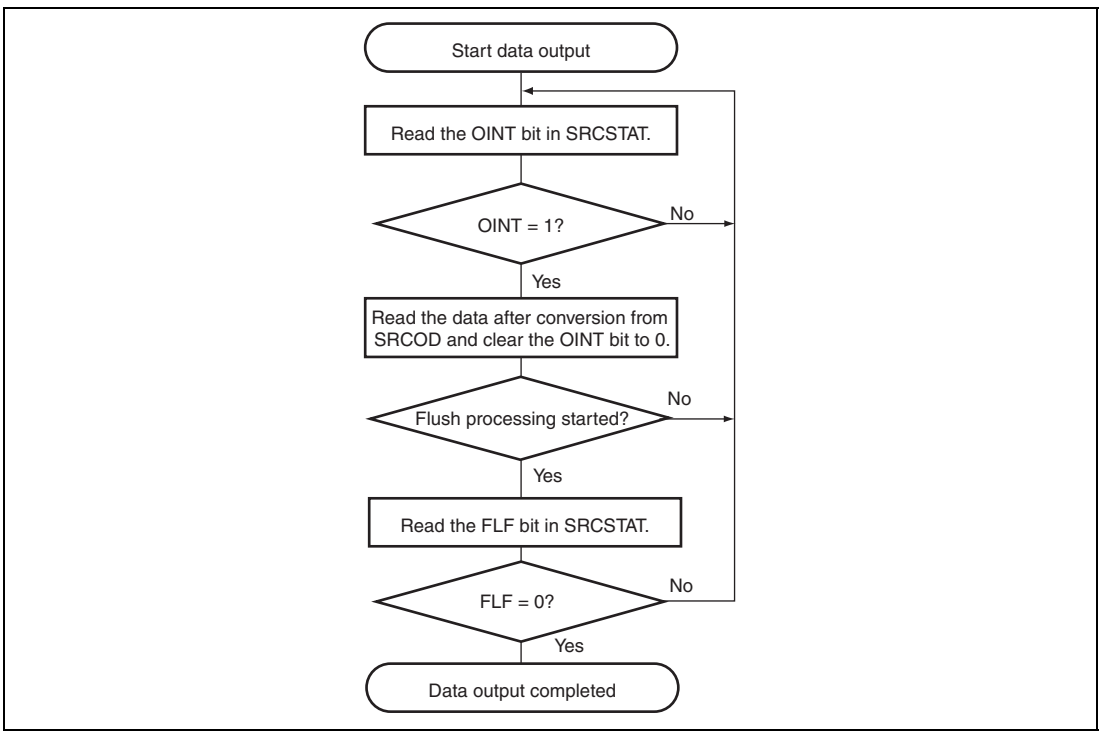


Figure 33.4 Sample Flowchart for Data Output

(1) When Interrupts are Issued to CPU

1. Set the OEN bit in SRCODCTRL to 1.
2. Set the interrupt controller.
3. When the OINT bit in SRCSTAT is set to 1, the ODF interrupt request is issued. In the interrupt processing routine, read the OINT bit and confirm that it is 1, read data from SRCOD, and write 0 to the OINT bit. Then return from the interrupt processing routine.
4. After flush processing starts, repeat step 3 until the FLF bit in SRCSTAT is read as 0.

(2) When Interrupts are Used to Activate DMAC

1. Assign ODFI of the SRC to one channel of the DMAC.
2. Set the OEN bit in SRCODCTRL to 1.
3. When the OINT bit in SRCSTAT is set to 1, the ODF interrupt request is issued thus activating the DMAC. When the DMAC has read data from SRCOD thus resulting in the number of data units in the output data FIFO being less than the triggering number specified by the OFTRG1 and OFTRG0 bits in SRCODCTRL, the OINT bit is cleared to 0.
4. After flush processing starts, repeat step 3 until the FLF bit in SRCSTAT is read as 0.

33.4 Interrupts

The SRC has three interrupt sources: input data FIFO empty (IDEI), output data FIFO full (ODFI), and output data FIFO overwrite (OVF). Table 33.6 summarizes the interrupts.

Table 33.6 Interrupt Requests and Generation Conditions

Interrupt Request	Abbreviation	Interrupt Condition	DMAC Activation
Input data FIFO empty	IDEI	IINT = 1, IEN = 1, and SRCEN = 1	Possible
Output data FIFO full	ODFI	OINT = 1, OEN = 1, and SRCEN = 1	Possible
Output data FIFO overwrite	OVF	OVF = 1, EEN = 1, and SRCEN = 1	Not possible

When the interrupt condition is satisfied, the CPU executes the interrupt exception handling routine. The interrupt source flags should be cleared in the routine.

The IDEI and ODFI interrupts can activate the DMAC when the DMAC is set to allow this. When the DMAC has written data to SRCID resulting in the number of data units in the input data FIFO exceeding that of the specified triggering number, the IINT bit is cleared to 0. Similarly, when the DMAC has read data from SRCOD resulting in the number of data units in the output data FIFO being less than the specified triggering number, the OINT bit is cleared to 0.

33.5 Usage Note

33.5.1 Note on Access Register

After the FL bit in SRCCTRL is set to 1, it takes 3 cycles of peripheral bus clock until the FLF bit in SRCSTAT is set to 1. While the CPU executes the next instruction without waiting the register write completion. Accordingly, the FLF set status cannot be read by the instruction immediately. To check the execution status of flush processing, dummy read the SRCCTRL or SRCSTAT after following the SRCCTRL write instruction and wait until the FLF bit is set.

33.5.2 Note on Flush Processing

After set 1 to the FL bit in SRCCTRL, the SRC continues exchange processing with setting to 0 after following the destination of the data that has already input. Flush processing allowed to be executed only under the condition that the destination bit of audio data has input completely and no following data exists.

In a case that implement the exchange processing after the flush processing, clear the internal work memories with using either way listed as follows.

- Set 1 to the CL bit in SRCCTRL.
- Set 0 to the SRCEN bit in SRCCTRL and back to 1.

33.5.3 Clearing OVF Flag Bit

When the CPU reads the OVF bit in SRC status register (SRCSTAT) on the timing of the bit is set to 1, even though 0 has been read, the SRC internal logic may recognize that the CPU reads 1. Then, therefore, if the CPU writes 0 to the flag, the flag may be cleared illegally because of the satisfaction of the condition that 0 is written after 1 has been read.

To avoid the malfunction, when the SRCSTAT is written, in which the OVF flag is not intended to be cleared, please write 1 to the flag bit (writing 1 to it does not affect the flag). Only when the SRCSTAT is written, in which the OVF flag is intended to be cleared, please write 0 to the flag bit.

Section 34 Stream Interface (STIF)

For details, refer to the separate STIF-related manual.

Section 35 Video Engine Unit (VEU3F)

The video engine unit (VEU3F; abbreviated as VEU, hereafter) is a module used connected to the buses via bus bridge modules. The VEU reads an image from a specified memory area, and writes it back to a specified address.

35.1 Features

- Format conversion using the RGB ⇔ YCbCr conversion function
- Scaling of an image using the filter function
- Tone reduction (quantization) to pack RGB data in 32-bit units
- Dithering for tone reduction of RGB data
- Removal of high-frequency components using the low-pass filter function
- Low-pass filter is applied to only the boundary of the blocks using the deblocking filter function
- Median filter function
- Edge enhancement of an image (enhancer function)

35.2 Functional Overview of VEU

The functional overview of the VEU is shown in table 35.1. Some functions of the VEU cannot be used at the same time unless the VEU is re-activated. Table 35.2 shows which functions can/cannot be used simultaneously during one VEU activation.

Note that when the instructions at other places in this manual differ from instructions in the following table, the instructions in section 35.4, Usage Notes for VEU, are given priority.

Table 35.1 Functional Overview of VEU

Item	Function	Description	Note
Input format	YCbCr (4:4:4/4:2:2/4:2:0)		
	RGB pack		
Output format	YCbCr (4:4:4/4:2:2/4:2:0)		
	RGB pack		
Read mode	Normal read	8 to 960 lines can be set as the number of lines in bundle read	
	Bundle read		
Low-pass filter	Removal of high-frequency components		
Deblocking filter	Removal of high-frequency components only at the boundary of blocks		
Enhancer	Enhancement of image		
Median filter	Removal of shot noise		
Rotation/ inversion of image	Vertical or horizontal inversion	Can be specified independently	A combination of both functions can realize rotation by 180 degrees
	Rotation by 90 or 270 degrees	Rotated clockwise	
Scale-up, scale-down, or no scaling	Scale-up or scale-down of memory display	Any scaling factor from $\times$ 1/16 to $\times$ 16	
Format conversion	YCbCr $\Leftrightarrow$ RGB conversion	Bidirectional conversion between YCbCr format and RGB format	

Item	Function	Description	Note
Dithering (tone reduction)	24 bpp	Full colors (16,700,000 colors)	Dithering not possible
	18 bpp	260,000 colors	Dithering not possible
	16 bpp	High colors (65,536 colors)	
	12 bpp	4,096 colors	
	8 bpp	256 colors	
Maximum image size	16 M pixels	4092 pixels × 4092 lines	
Minimum image size	16 × 16 pixels	16 pixels × 16 lines	

Note: The scaling factor of the filter can be set between 1/16 and 16. For details, see section 35.3.11, VEU Resize Filter Control Register (VRFCR), and section 35.3.12, VEU Resize Filter Size Clip Register (VRFSR).

Table 35.2 Simultaneous Usage of Functions in One VEU Activation

	Bundle Mode	Color Conversion	Low-Pass Filter	Deblocking Filter	Median Filter	Enhancer	Vertical/ Horizontal Inversion	90°/270° Rotation	Scale-Up/ Scale-Down
Bundle Mode	—	√	×	×	×	√	√	×	√
Color Conversion	√	—	√	√	√	√	√	√	√
Low-Pass Filter	×	√	—	×	×	×	√	√	×
Deblocking Filter	×	√	×	—	√*	×	√	√	×
Median Filter	×	√	×	×*	—	×	√	√	×
Enhancer	√	√	×	×	×	—	√	×	√
Vertical/ Horizontal Inversion	√	√	√	√	√	√	—	√	√
90°/270° Rotation	×	√	√	√	√	×	√	—	×
Scale-Up/ Scale-Down	√	√	×	×	×	√	√	×	—

[Legend]

√: Possible

×: Not possible

Notes: The register values corresponding to the items in this table are shown below.

Bundle mode:	VESTR.VBE
Color conversion:	VTRCR.TE
Low-pass filter:	VFMCR.LPHV
Deblocking filter:	VFMCR.LPHV && VFMCR.DBLK
Median filter:	VFMCR.MED
Enhancer:	VENHR.ENHH || VENHR.ENHV
Vertical/horizontal inversion:	VFMCR.VMRR/VFMCR.HMRR
90°/270° rotation:	VFMCR.POTR/VFMCR.POTL
Scale-up/scale-down:	VRFCR ! = 0

* The order of deblocking filter → median filter is allowed. However, a reversed order is not allowed.

Figure 35.1 shows a block diagram of the VEU.

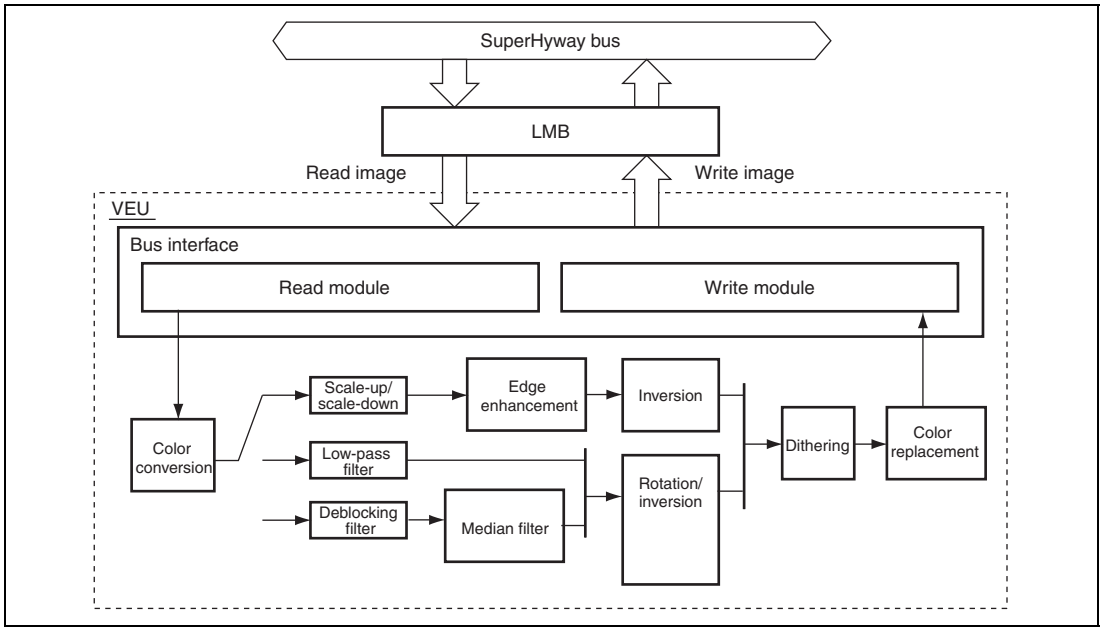


Figure 35.1 Block Diagram of VEU

Note: In figure 35.1, inversion indicates horizontal or vertical inversion, and rotation indicates 90° or 270° rotation.
In the inversion circuit after scale-up/scale-down, horizontal inversion, vertical inversion, or 180° rotation, which is a combination of the two types of inversion, can be performed.
In the rotation/inversion circuit after the median filter, in addition to the possible operations in the inversion circuit after scale-up/scale-down, 90° rotation, 270° rotation, a combination of 90° rotation and horizontal inversion, or a combination of 90° rotation and vertical inversion can be performed.

- Operation

  Figure 35.2 shows the sequence of the VEU operation processing. Though the VEU has various functions, some functions cannot be used simultaneously during one VEU activation, as shown in table 35.2. Either flow 1 or flow 2 shown in figure 35.2 is possible. Accordingly, processing in different flows cannot be performed together.

  1. Flow 2 is selected when the following condition is satisfied.

     VESTR.VBE || (VENHR.ENHH || VENHR.ENHV) || (VRFCR != 0) and VFMCR.FLTPI = 0

  2. Flow 1 is selected in cases other than above.

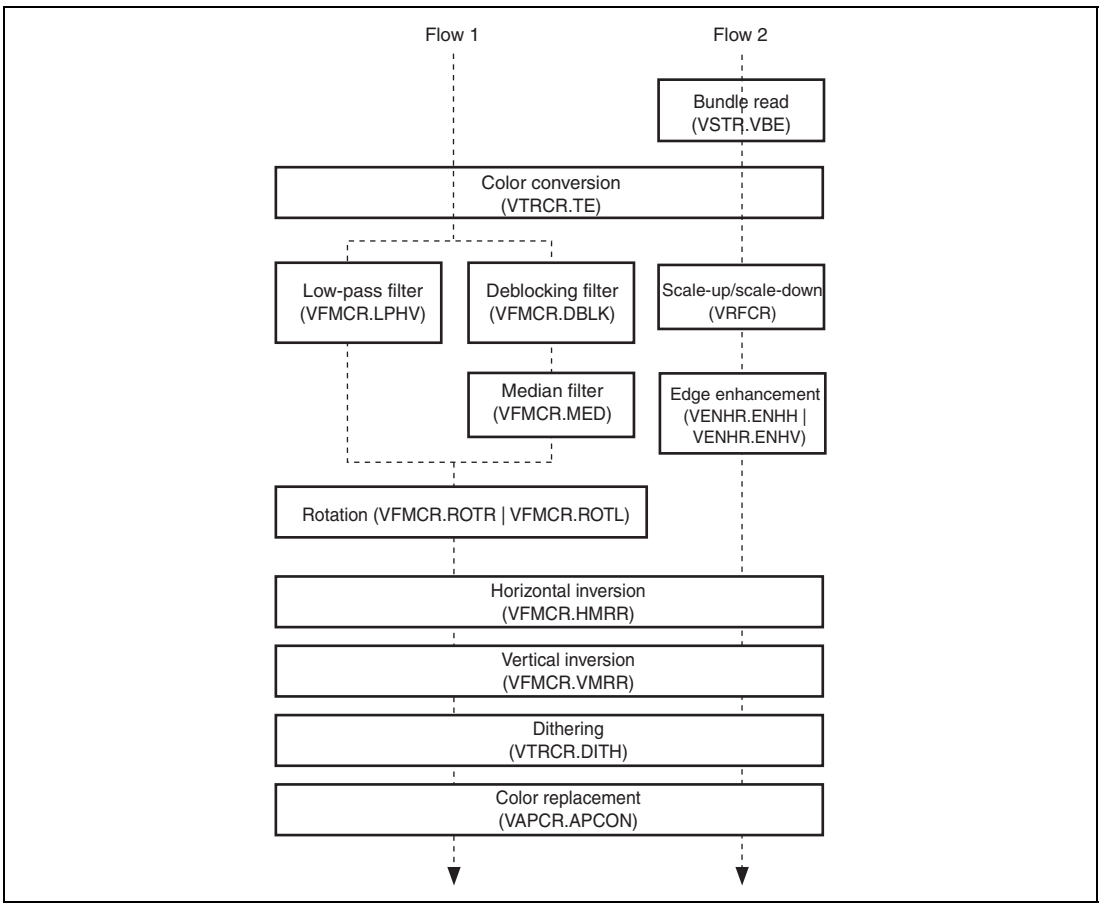


Figure 35.2 Register Settings, Processing Flows, and Simultaneously Executable Functions of VEU

35.3 Register Descriptions of VEU

(1) R/W Restrictions on VEU Registers

The read/write restrictions on the VEU registers are listed below. If the following register handling is not guaranteed, a malfunction may occur.

1. For the read-only bits in all VEU registers and the reserved areas, writing 1 is prohibited. Do not specify an unspecified value.
2. Though a value other than 0 may be read from a read-only bit, do not modify this bit.
3. For registers (bits) to which writing during operation is prohibited, do not modify them during operation (reading is possible). The VEU operating state can be confirmed by reading the VE bit in the VEU start register (VESTR). Modify the above registers when this bit is 0.
4. Allocate all VEU registers in a non-cacheable area.

(2) Terms and Abbreviations Used in This Section

The terms used in this section are described below.

1. "software reset" indicates that the VEU processing was halted and the current processing is stopped. The image processing result of the frame in which a software reset occurs is not guaranteed.
2. "module reset" indicates the VEU internal circuit was forcibly reset. In a module reset, reset processing is performed with no consideration to the hardware state. Therefore, if a module reset is generated when the VEU is operating normally, the VEU peripheral hardware may become unable to operate.
3. "during operation" in this section indicates a state in which the VE bit in the VEU start register (VESTR) is set to 1.
4. When a bit name in a register is referred to in this section, it is indicated in the format of (register name).(bit name).
   Example: VESTR.VE

(3) Point for Caution

Addresses of registers used by the VEU module are given in table 35.3. Operation is not guaranteed in the case of access to other addresses.

(4) List of Registers

The register configuration of the VEU is shown in table 35.3. The register states in each processing mode are shown in table 35.4.

Table 35.3 Register Configuration of VEU

Register Name	Abbr.	R/W	Address	Access Size
VEU start register	VESTR	R/W	H'FD12 0000	32
VEU source memory width register	VESWR	R/W	H'FD12 0010	32
VEU source size register	VESSR	R/W	H'FD12 0014	32
VEU source address Y register	VSAYR	R/W	H'FD12 0018	32
VEU source address C register	VSACR	R/W	H'FD12 001C	32
VEU bundle source size register	VBSSR	R/W	H'FD12 0020	32
VEU destination memory width register	VEDWR	R/W	H'FD12 0030	32
VEU destination address Y register	VDAYR	R/W	H'FD12 0034	32
VEU destination address C register	VDACR	R/W	H'FD12 0038	32
VEU transform control register	VTRCR	R/W	H'FD12 0050	32
VEU resize filter control register	VRFCR	R/W	H'FD12 0054	32
VEU resize filter size clip register	VRFSR	R/W	H'FD12 0058	32
VEU enhance register	VENHR	R/W	H'FD12 005C	32
VEU resize filter sub control register	VRSCR	R/W	H'FD12 0064	32
VEU resize filter size clip offset register	VRSOR	R/W	H'FD12 0068	32
VEU filter mode control register	VFMCR	R/W	H'FD12 0070	32
VEU vertical tap coefficient register	VVTCR	R/W	H'FD12 0074	32
VEU horizontal tap coefficient register	VHTCR	R/W	H'FD12 0078	32
VEU designated color register	VAPCR	R/W	H'FD12 0080	32
VEU conversion color register	VECCR	R/W	H'FD12 0084	32
VEU fill color specification register	VFLCR	RW	H'FD12 0088	32
VEU address fixed register	VAFXR	R/W	H'FD12 0090	32
VEU swapping register	VSWPR	R/W	H'FD12 0094	32
VEU event interrupt enable register	VEIER	R/W	H'FD12 00A0	32
VEU event register	VEVTR	R/W	H'FD12 00A4	32
VEU status register	VSTAR	R	H'FD12 00B0	32

Register Name	Abbr.	R/W	Address	Access Size
VEU module reset register	VBSRR	R/W	H'FD12 00B4	32
VEU resize passband register	VRPBR	R/W	H'FD12 00C8	32

Table 35.4 VEU Register States in Each Processing Mode

Register Abbreviation	Power-On Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
VESTR	Initialized	Initialized	Retained	Retained	Retained	Initialized
VESWR	Initialized	Initialized	Retained	Retained	Retained	Initialized
VESSR	Initialized	Initialized	Retained	Retained	Retained	Initialized
VSAYR	Initialized	Initialized	Retained	Retained	Retained	Initialized
VSACR	Initialized	Initialized	Retained	Retained	Retained	Initialized
VBSSR	Initialized	Initialized	Retained	Retained	Retained	Initialized
VEDWR	Initialized	Initialized	Retained	Retained	Retained	Initialized
VDAYR	Initialized	Initialized	Retained	Retained	Retained	Initialized
VDACR	Initialized	Initialized	Retained	Retained	Retained	Initialized
VTRCR	Initialized	Initialized	Retained	Retained	Retained	Initialized
VRFCR	Initialized	Initialized	Retained	Retained	Retained	Initialized
VRFSR	Initialized	Initialized	Retained	Retained	Retained	Initialized
VENHR	Initialized	Initialized	Retained	Retained	Retained	Initialized
VFMCR	Initialized	Initialized	Retained	Retained	Retained	Initialized
VVTCR	Initialized	Initialized	Retained	Retained	Retained	Initialized
VHTCR	Initialized	Initialized	Retained	Retained	Retained	Initialized
VAPCR	Initialized	Initialized	Retained	Retained	Retained	Initialized
VECCR	Initialized	Initialized	Retained	Retained	Retained	Initialized
VFLCR	Initialized	Initialized	Retained	Retained	Retained	Initialized
VAFXR	Initialized	Initialized	Retained	Retained	Retained	Initialized
VSWPR	Initialized	Initialized	Retained	Retained	Retained	Initialized
VEIER	Initialized	Initialized	Retained	Retained	Retained	Initialized
VEVTR	Initialized	Initialized	Retained	Retained	Retained	Initialized
VSTAR	Initialized	Initialized	Retained	Retained	Retained	Initialized
VBSRR	Initialized	Initialized	Retained	Retained	Retained	Initialized
VRPBR	Initialized	Initialized	Retained	Retained	Retained	Initialized

35.3.1 VEU Start Register (VESTR)

VESTR controls activation of the VEU and software reset of VEU processing. Before starting VEU processing by VESTR, all registers related to the VEU must be set.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	VBE	—	—	—	—	—	—	—	VE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	VBE	0	R/W	Sets the bundle read mode (N-line read mode). In bundle read mode (N-line read mode), the VEU divides a one-frame image at the number of lines set by VBSSR and reads data for that number of lines. When the VEU is activated in bundle read mode (N-line read mode), rotation processing, deblocking filter processing, low-pass filter processing, and median filter processing cannot be executed (horizontal inversion, vertical inversion, and horizontal-vertical inversion are possible). 0: Normal operation 1: Bundle read (N-line read) mode
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
0	VE	0	R/W	Controls the VEU processing start and software reset of processing (processing halt). If this bit is set to 1 when the VEU is halted, the VEU starts processing images in the memory according to the register settings, and this bit is read as 1. In addition, if 0 is written to this bit when it is 1, the current VEU processing is immediately terminated. The end of software reset processing can be confirmed when this bit becomes 0 after a software reset has been issued to the VEU. Confirm that this bit is 0 before restarting the VEU after a software reset. 0: NOP (software reset processing when 0 is written to this bit when it is 1) 1: VEU processing starts

If B'1 is written to the VE bit when the VBE bit is B'1 at VEU activation, bundle read mode (N-line read mode) is selected. In bundle read mode, data is read from two memory areas alternately for the number of lines set by the VBSS bits in VBSSR. If reading for the entire frame is not finished when the VEU has completed reading for the number of lines set by VBSSR, a bundle end interrupt (VEVTR.VBEND) occurs and the VEU waits for reading to be restarted. To restart reading, first clear the bundle end interrupt source, and then write B'1 to the VE bit again (write with the VBE bit still set to B'1). After repeating this until reading for the entire frame finishes, a one-frame end interrupt (VEVTR.VEEND) occurs. The image of processing in bundle read mode is shown in figure 35.5. The status of normal image processing, read processing in bundle read mode, and waiting of read to be restarted in bundle read mode can be confirmed by reading VSTAR.

Notes on software reset:

1. Though the VEU end interrupt flag (VEVTR.VEEND) may become B'1 depending on the timing to issue a VEU software reset, the processed image of the frame in which a software reset was issued is not guaranteed.
2. Even when the VEEND flag does not become B'1 after a VEU software reset, the VEEND flag must always be cleared to B'0.
3. When issuing a software reset in bundle read mode, issue it before reading is restarted subsequent to a bundle end interrupt.

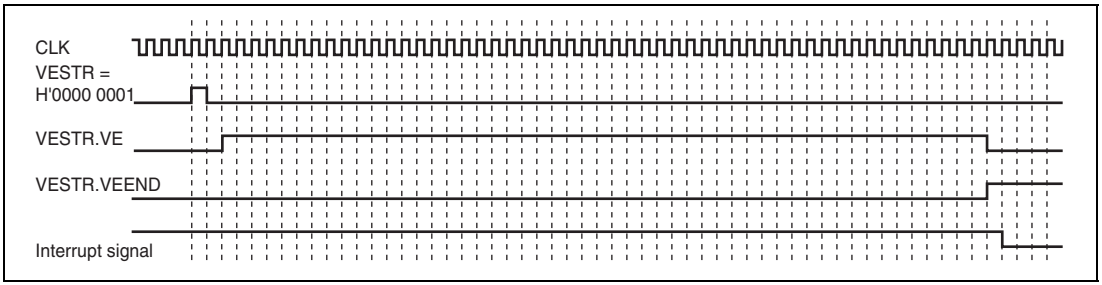


Figure 35.3 Operation Timing of VE Bit and Each Status

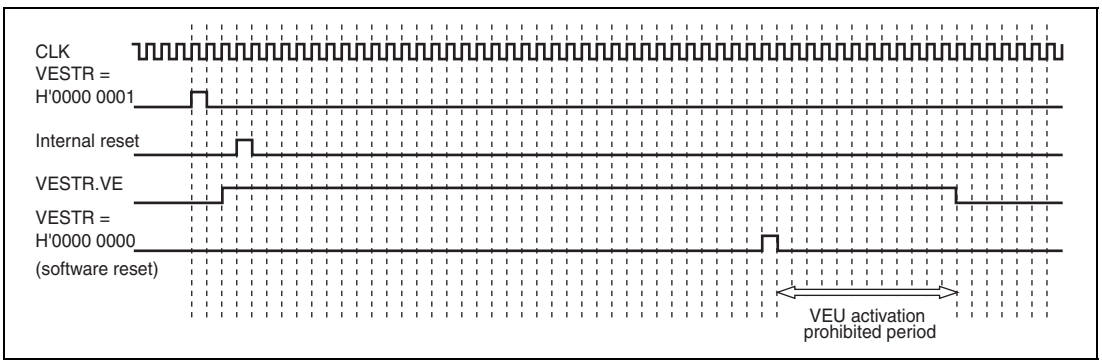


Figure 35.4 Operation Timing of Software Reset and Each Status

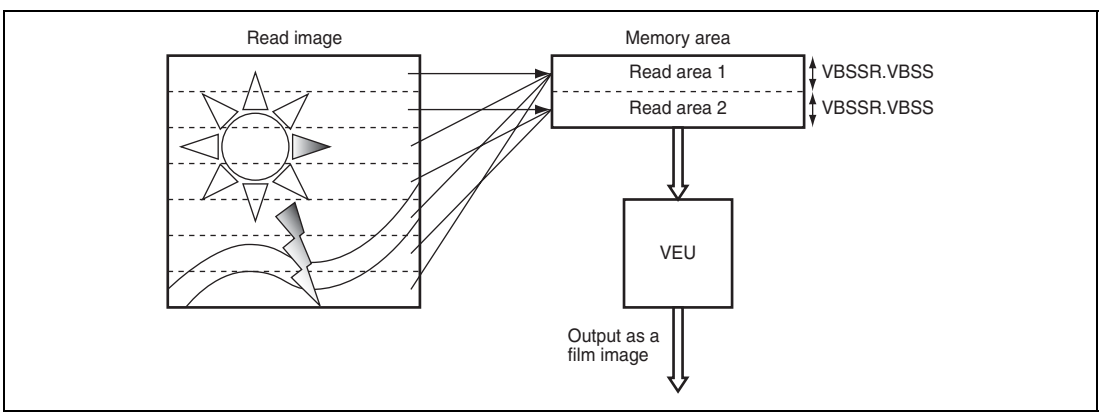


Figure 35.5 Image of Bundle Read Mode Processing

35.3.2 VEU Source Memory Width Register (VESWR)

VESWR sets the memory width of the source memory area of the VEU.

Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VSW[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 2	VSW[15:2]	H'0000	R/W	These bits set the width of the source memory area of the VEU (4-pixel units). This setting should be made in 2-pixel units when the RPKF bits in VTRCR are set to 0,1,3,7, or 13.
1, 0	VSW[1:0]		R	

In the VSW bits, set the horizontal width of the source memory area where the source image is located in the VEU processing as a number of bytes. This setting must be made in byte units that correspond to four pixels of the source image. If the read image is in the YCbCr format, make a setting matching the Y component. For reading data in the YCbCr 4:2:0, YCbCr 4:2:2, and YCbCr 4:4:4 formats, the VSW setting, VSW setting, and (VSW setting × 2) are applied respectively to the horizontal width of the source memory area for the C component. When the RPKF bits in VTRCR are set to 0,1,3,7, or 13 (RGB 2 bytes/pixel or RGB 4 bytes/pixel), a setting in byte units that corresponds to two pixels of the source image is possible. As shown in (A) in figure 35.6, when an image is clipped from the memory area, the right-end address of a line's image is discontinuous from the left-end address of the next line's image. Compared to this, (B) in figure 35.6 shows an example where the horizontal size of an image input to the VEU matches the horizontal length of the memory. In this case, the right-end address of a line's image and the left-end address of the next line's image have continuous values.

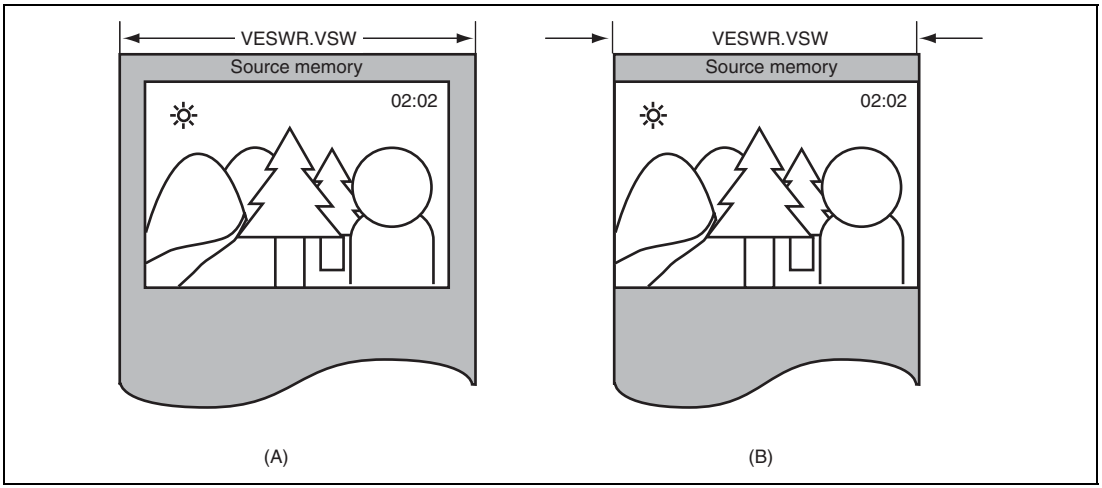


Figure 35.6 Relationship between Image Clipping Size and VESWR

35.3.3 VEU Source Size Register (VESSR)

VESSR sets the vertical and horizontal sizes (number of pixels) of the image read by the VEU.

Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	VVSS[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	VHSS[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 16	VVSS[11:0]	H'000	R/W	These bits set the number of vertical pixels read by the VEU. Set the number of vertical read pixels of the source image read by the VEU in these bits (see figure 35.7). • Scale-up/down, enhancer, or bundle read is not performed (see figure 35.7): A setting in 2-pixel units is possible for the following cases: Reading and writing are both in RGB format. VTRCR.RPKF is either 3 (RGB 2 bytes/pixel), 0, 1, 7, or 13 (RGB 4 bytes/pixel) and VTRCR.WPKF is either 1, 2, 6 (RGB 2 bytes/pixel), 14, 19, 20, 22, or 23 (RGB 4 bytes/pixel). Other than those settings, specify a setting in 4-pixel units. • Scale-up/down, enhancer, or bundle read is performed, a setting in 1-pixel unit is possible (see figure 35.7). However, the data should be allocated in the memory to specify 2-pixel units when the input format is YCbCr4:2:0. The maximum value to be set is 4092 pixels.

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	VHSS[11:0]	H'000	R/W	These bits set the number of horizontal pixels read by the VEU. Set the number of horizontal read pixels of the source image read by the VEU in these bits (see figure 35.7). • Scale-up/down, enhancer, or bundle read is not performed (see figure 35.7): A setting in 2-pixel units is possible for the following cases: Reading and writing are both in RGB format. VTRCR.RPKF is either 3 (RGB 2 bytes/pixel), 0, 1, 7, 13 (RGB 4 bytes/pixel) and VTRCR.WPKF is either 1, 2, 6 (RGB 2 bytes/pixel), 14, 19, 20, 22, 23 (RGB 4 bytes/pixel). Other than those settings, specify a setting in 4-pixel units. • Scale-up/down, enhancer, or bundle read is performed, a setting in 1-pixel unit is possible (see figure 35.7). However, set in the VESWR.VSW[15:0] a value (a value that VHSS[11:0] specify in 4-pixel units) × 4 / (P_density_y in use format) or more. Further, the data should be allocated in the memory to specify in 2-pixel units when the input format is YCbCr4:2:0. The maximum value to be set is 4092 pixels.

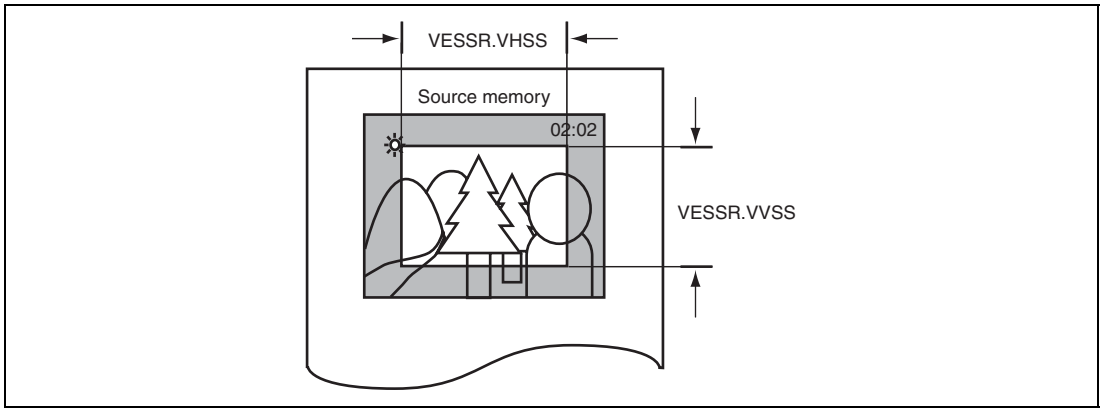


Figure 35.7 Relationship between Read Size and VESSR

35.3.4 VEU Source Address Y Register (VSAYR)

VSAYR sets the start address for the Y/RGB plane of the image read by the VEU.

Modifying this register during operation is prohibited. However, this register can be modified in the restart wait state (when VSTAR = H'0000 1001) in N-line read mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	VSAY[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VSAY[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	VSAY[31:2]	H'0000 0000	R/W	These bits set the start address for reading the Y/RGB plane by the VEU (longword units).
1, 0	VSAY[1:0]		R	

Set the start address for the Y/RGB plane of the source image of the VEU in the VSAY bits, as shown in figure 35.8. When reading a YCbCr image, specify the start address of the Y plane. When reading an RGB image, specify the start address (top-left address in the read image area) of the RGB plane. Figure 35.8 shows the input format for the Y plane of the YCbCr format. For the input format of the RGB format, refer to table 35.8.

Note: In bundle read mode, set two addresses alternately for each VEVTR.VBEND interrupt.

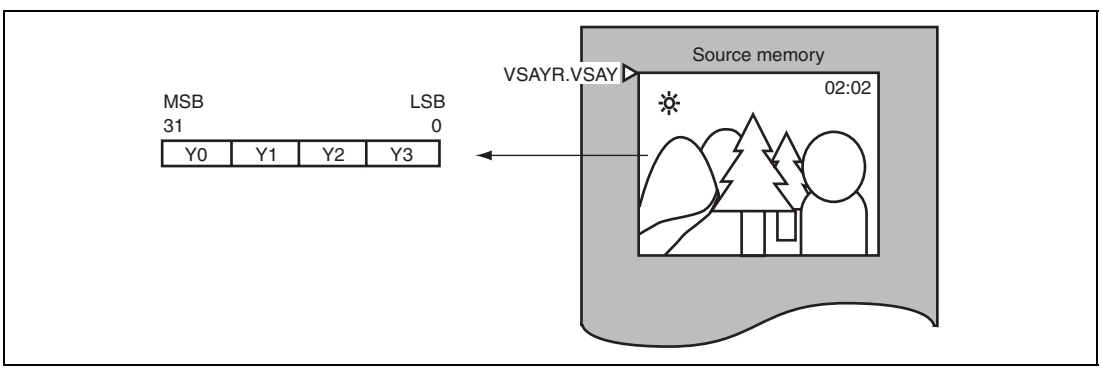


Figure 35.8 Address Set in VSAYR and Y-Plane Format

35.3.5 VEU Source Address C Register (VSACR)

VSACR sets the start address for the C plane of the image read by the VEU.

Modifying this register during operation is prohibited. However, this register can be modified in the restart wait state (when VSTAR = H'0000 1001) in N-line read mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	VSAC[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VSAC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	VSAC[31:2]	H'0000 0000	R/W	These bits set the start address for reading the C plane by the VEU. (1-longword units. 2-longword units for the YCbCr 4:4:4 format.)
1, 0	VSAC[1:0]		R	

Set the start address for the C plane of the source image of the VEU in the VSAC bits, as shown in figure 35.9. When reading a YCbCr image, specify the start address (top-left address in the read image area) of the C plane. Figure 35.9 shows the input format for the C plane. When reading an RGB image, VSACR is not used. VSACR should be set in 1-longword units for the YCbCr 4:2:0 or YCbCr 4:2:2 format, and in 2-longword units for the YCbCr 4:4:4 format.

Note: In bundle read mode, set two addresses alternately for each VEVTR.VBEND interrupt.

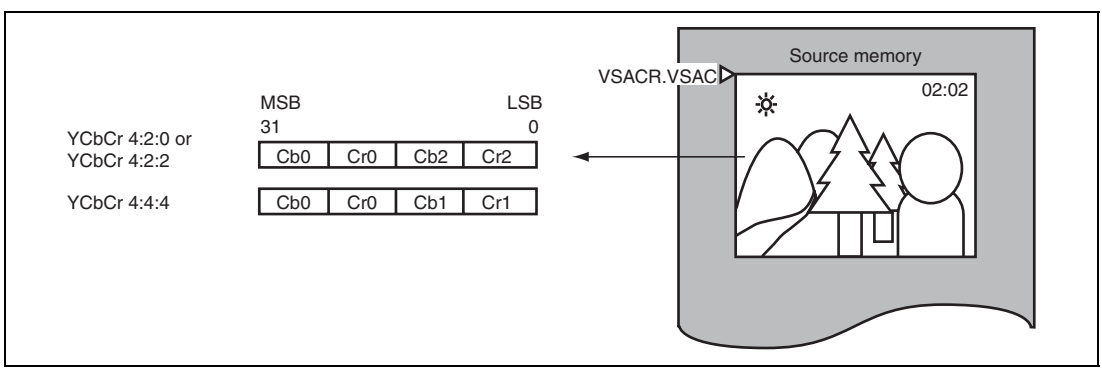


Figure 35.9 Address Set in VSACR and C-Plane Format

35.3.6 VEU Bundle Source Size Register (VBSSR)

VBSSR sets the number of lines (number of pixels) to be read when the VEU is activated once in bundle read mode (N-line read mode). VBSSR needs to be set only in bundle read mode.

Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	VBSS[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 3	VBSS[11:3]	H'000	R/W	These bits set the number of lines to be read when the VEU is activated once in bundle read mode (N-line read mode) (8-line units).
2 to 0	VBSS[2:0]		R	

Set the number of lines of the source image to be read by the VEU in one processing in bundle read mode in the VBSS bits. The VBSS bits should be set in 8-line units, and the maximum value to be set is 960 lines.

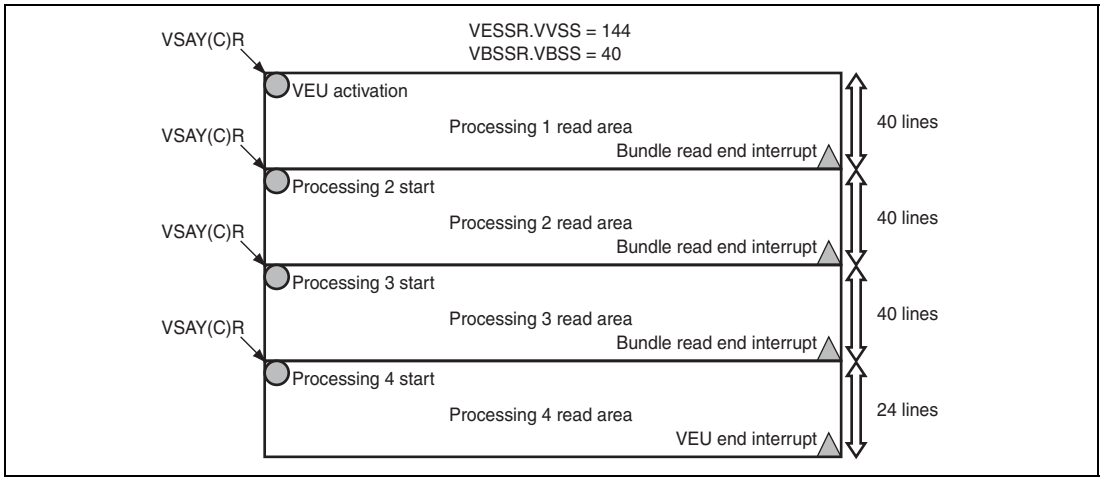


Figure 35.10 Relationship between Read Image in Bundle Read Mode and Read Image in Each Processing

35.3.7 VEU Destination Memory Width Register (VEDWR)

VEDWR sets the memory width of the destination memory area of the VEU.

Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VDW[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 2	VDW[15:2]	H'0000	R/W	These bits set the width of the destination memory area of the VEU (longword units).
1, 0	VDW[1:0]		R	

Set the horizontal width of the destination memory area where the destination image is located as a number of bytes in the VDW bits. This setting must be made in byte units that correspond to four pixels of the destination image. If the drawing image is in the YCbCr format, make a setting matching the Y component. For writing data in the YCbCr 4:2:0, YCbCr 4:2:2, and YCbCr 4:4:4 formats, the VDW setting, VDW setting, and (VDW setting × 2) are applied respectively to the horizontal width of the destination memory area for the C component. When the WPKF bits in VTRCR are set to 1, 2, 6 (RGB 2 bytes/pixel), 8 to 14, 19, 20, 22, 23 (RGB 4 bytes/pixel), a setting in byte units that corresponds to two pixels of the destination image is possible. As shown in (A) in figure 35.11, when an image is pasted to the back display or the like in the memory area, the right-end address of a line's image is discontinuous from the left-end address of the next line's image. Compared to this, (B) in figure 35.11 shows an example where the horizontal size of an image output from the VEU matches the horizontal length of the memory. In this case, the right-end address of a line's image and the left-end address of the next line's image have continuous values.

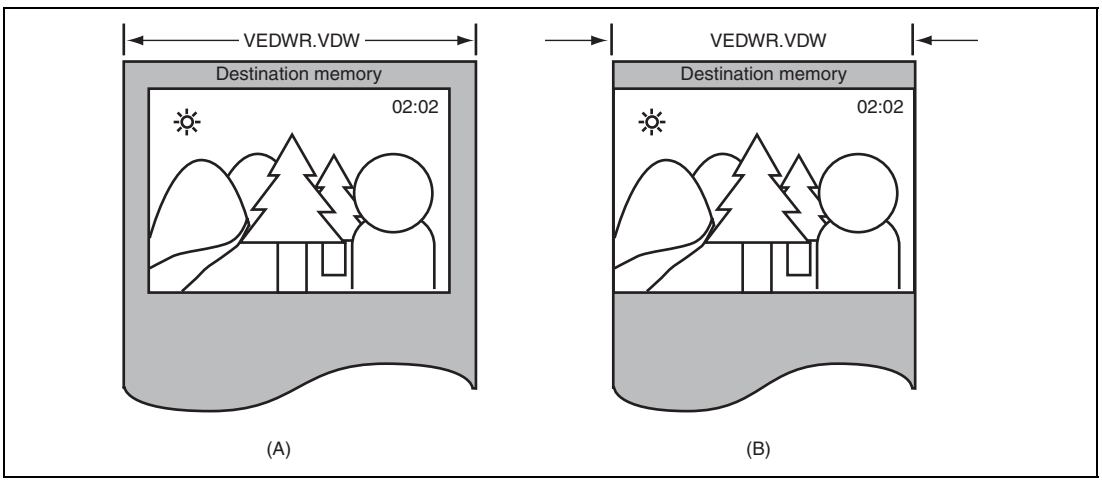


Figure 35.11 Relationship between Image Pasting Size and VEDWR

35.3.8 VEU Destination Address Y Register (VDAYR)

VDAYR sets the start address for the Y/RGB plane of the image drawn by the VEU.

Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	VDAY[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VDAY[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	VDAY[31:2]	H'0000 0000	R/W	These bits set the start address for drawing the Y/RGB plane by the VEU (longword units).
1, 0	VDAY[1:0]		R	

Set the start address of the destination image (top-left address in the destination image area) of the VEU in the VDAY bits, as shown in figure 35.12. When drawing a YCbCr image, specify the start address of the Y plane. When drawing an RGB image, specify the start address of the RGB plane.

When VTRCR.WPKF = 6 (RGB 16-bpp), a setting in word units is possible.

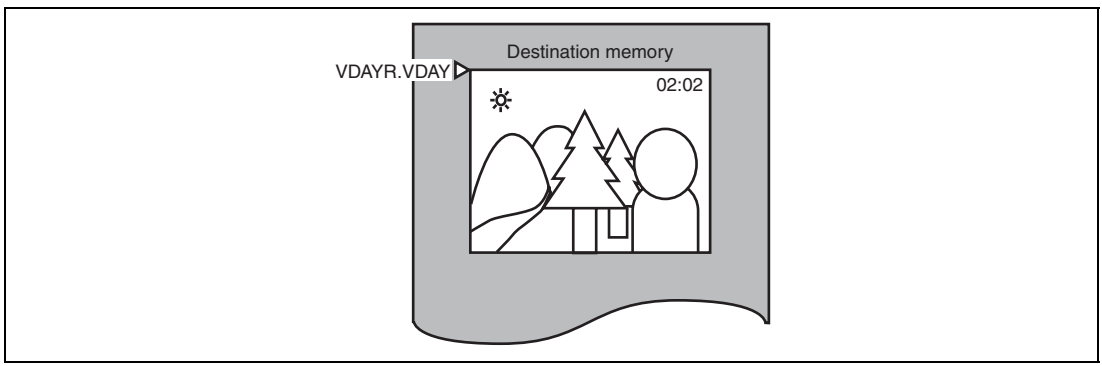


Figure 35.12 Address Set in VDAYR

The point specified as the address to be set in VDAYR changes when the image is rotated or inverted. This specification point of the address for each processing is shown in table 35.5. Figure 35.13 illustrates the address specification point in block processing, and figure 35.14 illustrates that in line processing. The gray quadrangle is assumed as the destination image and the ☆ marks as the address specification points.

The address specification point if neither scale-up/down, enhancer, bundle read, nor VFMCR.FLTPI = 0 is performed is the top-left address for the block located in one of the four corners with the destination image divided into blocks of 16 × 16 pixels. In a case that scale-up/down, enhance, and bundle read are not used and even VFMCR.FLTPI = 1, or either scale-up/down, enhance, or bundle read is used, each address specification point becomes one of the four corners in the destination image.

Table 35.5 Address Specification Point for Setting VDAYR

	No Rotation/ Inversion	90° Rotation	270° Rotation	90° Rotation + Horizontal Inversion	90° Rotation + Vertical Inversion	Horizontal Inversion	Vertical Inversion	Horizontal Inversion + Vertical Inversion
Block processing: No scale-up/down, enhancer, or bundle read and even FLTPI = 0.	—	D	E	F	G	A	B	C
Line processing: Scale-up/down, enhancer, or bundle read performed and even VFMCR.FLTPI = 1.	—	x	x	x	x	H	J	K

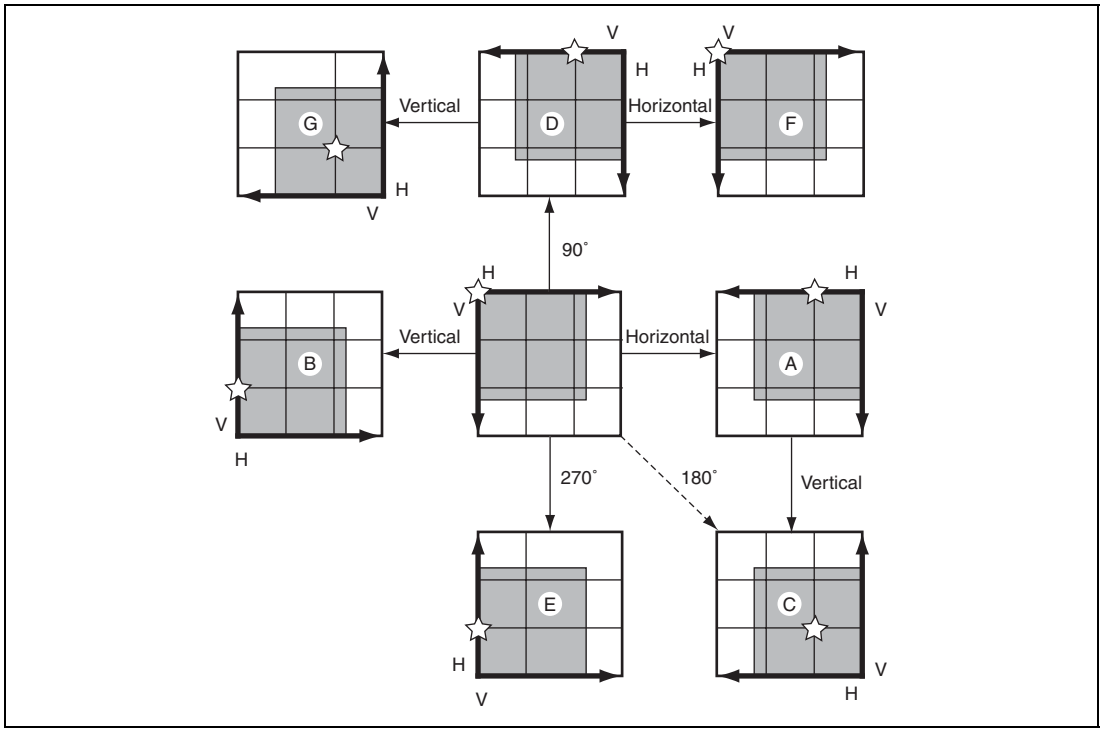


Figure 35.13 Location of Address Specification Point for VDAYR (Block Processing)

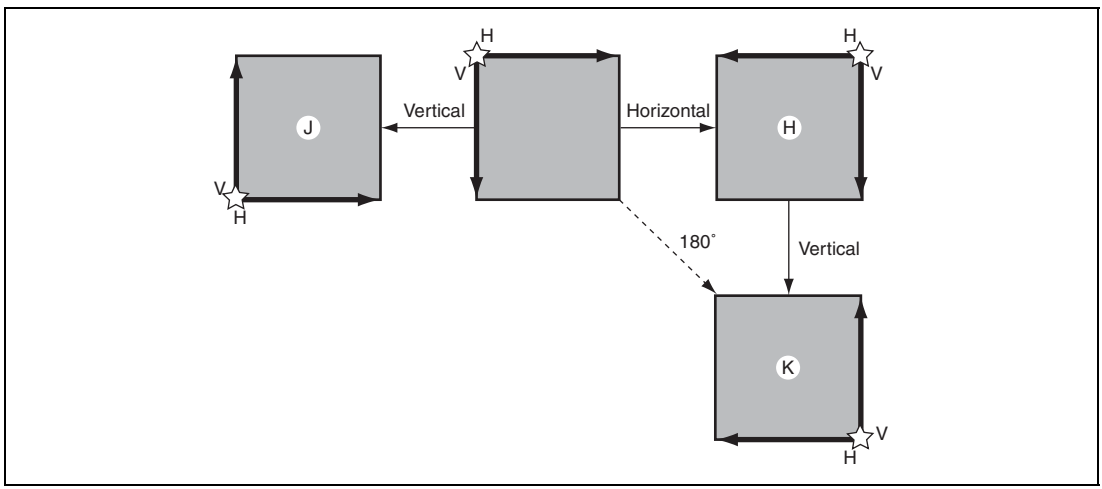


Figure 35.14 Location of Address Specification Point for VDAYR (Line Processing)

When drawing a YCbCr image, specify an address of the Y plane. When drawing an RGB image, specify an address of the RGB plane.

The formulas for setting the addresses are shown below.

Offset_ad	Becomes the address location of the top-left corner in the destination image	
yuv420out	Drawing format is YCbCr 4:2:0	1: VTRCR. CHDS = 0 0: VTRCR. CHDS = 1 or 2
clip_hsize	Horizontal size of line processing output	VRFSR.VHCLP
clip_vsize	Vertical size of line processing output	VRFSR.VVCLP
clip_vsize_c	Vertical size (C) of line processing output	(clip_vsize + 1)/2: VTRCR. CHDS = 0 (round-down) clip_vsize: VTRCR. CHDS = 1 or 2
dest_width	Destination memory width	VEDWR.VDW
dest_width_c	Destination memory width (C)	dest_width: VTRCR.CHDS = 0 or 1 dest_width × 2: VTRCR.CHDS = 2
src_hblk	Horizontal block count	VESSR.VHSS/16 (round-up)
src_vblk	Vertical block count	VESSR.VVSS/16 (round-up)
src_sideh	Horizontal size of right-end block	(VESSR.VHSS + 15)%16 + 1
src_sidev	Vertical size of bottom-end block	(VESSR.VHSS + 15)%16 + 1
< >	Operator changing a negative value to 0	0 when the value in < > is 0 or lower, otherwise the value in < >

For P_density_y of an RGB output, see table 35.6. For P_density_y and P_density_c of a YCbCr output, see table 35.7.

[Address for Y component output]

0 VDAYR = offset_ad

F VDAYR = offset_ad

D VDAYR = offset_ad + <((src_vblk − 2) × 16 + src_sidev)> × (4/P_density_y)

A VDAYR = offset_ad + <((src_hblk − 2) × 16 + src_sideh)> × (4/P_density_y)

E VDAYR = offset_ad + <((src_hblk − 2) × 16 + src_sideh)> × dest_width

B VDAYR = offset_ad + <((src_vblk − 2) × 16 + src_sidev)> × dest_width

G VDAYR = offset_ad + <((src_hblk − 2) × 16 + src_sideh)> × dest_width + <((src_vblk − 2) × 16 + src_sidev)> × (4/P_density_y)

C VDAYR = offset_ad + <((src_vblk − 2) × 16 + src_sidev)> × dest_width + <((src_hblk − 2) × 16 + src_sideh)> × (4/P_density_y)

H VDAYR = offset_ad + clip_hsize × (4/P_density_y)

J VDAYR = offset_ad + (clip_vsize − 1) × dest_width

K VDAYR = offset_ad + (clip_vsize − 1) × dest_width + clip_hsize × (4/P_density_y)

[Address for C component output]

0 VDACR = offset_ad

F VDACR = offset_ad

D VDACR = offset_ad + <((src_vblk − 2) × 16 + src_sidev)> × (4/P_density_c)

A VDACR = offset_ad + <((src_hblk − 2) × 16 + src_sideh)> × (4/P_density_c)

E VDACR = offset_ad + <((src_hblk − 2) × 16 + src_sideh)>/(1 + yuv420out) × dest_width_c

B VDACR = offset_ad + <((src_vblk − 2) × 16 + src_sidev)>/(1 + yuv420out) × dest_width_c

G VDACR = offset_ad + <((src_hblk − 2) × 16 + src_sideh)>/(1 + yuv420out) × dest_width_c + <((src_vblk − 2) × 16 + src_sidev)> × (4/P_density_c)

C VDACR = offset_ad + <((src_vblk − 2) × 16 + src_sidev)>/(1 + yuv420out) × dest_width_c + <((src_hblk − 2) × 16 + src_sideh)> × (4/P_density_c)

H VDACR = offset_ad + clip_hsize × (4/P_density_c)

J VDACR = offset_ad + (clip_vsize_c − 1) × dest_width_c

K VDACR = offset_ad + (clip_vsize_c − 1) × dest_width_c + clip_hsize × (4/P_density_c)

35.3.9 VEU Destination Address C Register (VDACR)

VDACR specifies the start address for the C plane of the image drawn by the VEU.

Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	VDAC[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VDAC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	VDAC[31:2]	H'0000 0000	R/W	These bits set the start address for drawing the C plane by the VEU. (1-longword units. 2-longword units for the YCbCr 4:4:4 format.)
1, 0	VDAC[1:0]		R	

Set the start address of the destination image (top-left address in the destination image area) of the VEU in the VDAC bits, as shown in figure 35.15. When drawing a YCbCr image, specify the start address of the C plane. When drawing an RGB image, VDACR is not used.

The point specified as the address to be set in VDACR changes when the image is rotated or inverted. As this specification point is similar to that for VDAYR, see figure 35.13 and table 35.5 for details. However, when an image is drawn in the YCbCr 4:2:0 format, the number of lines output in the vertical direction for the C component is half of the number of lines for the Y component or an RGB output. When calculating the address in the vertical direction (memory width × number of vertical lines), take notice that the number of lines will become half. (See the description in section 35.3.8, VEU Destination Address Y Register (VDAYR).)

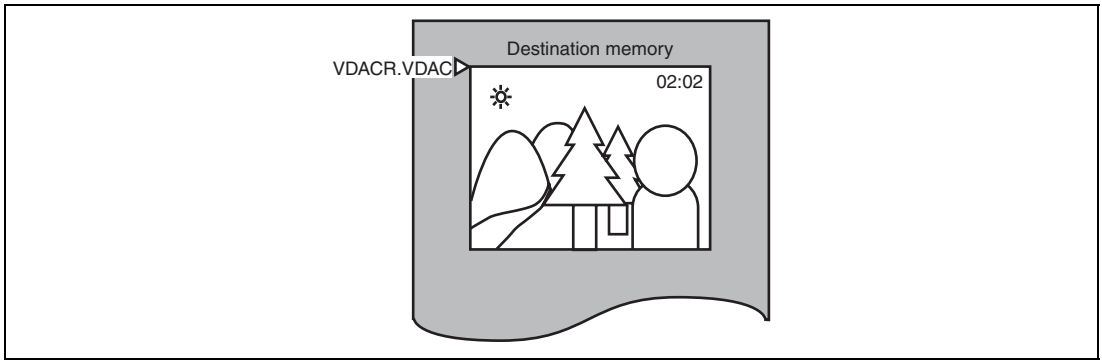


Figure 35.15 Address Set in VDACR

35.3.10 VEU Transform Control Register (VTRCR)

VTRCR mainly sets the processing contents related to color conversion by the VEU and the input/output data format.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PAD[7:0]								CHDS[1:0]		—	WPKF[4:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CHRR[1:0]		—	—	RPKF[3:0]				—	—	—	DITH	TM2	TM1	TE	RY
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	PAD[7:0]	H'00	R/W	These bits set the PAD value to be stuffed into the output data when the VEU output data is in the RGB format. Some forms of RGB output data packs (see table 35.6) include a PAD field as shown in figure 35.16. In the PAD field, any 8-bit data can be added. Specify the value to be added to the PAD field to these bits.

Bit	Bit Name	Initial Value	R/W	Description
23, 22	CHDS[1:0]	00	R/W	These bits select the output format when the VEU output data is in the YCbCr format. For the YCbCr format, data is output in one of the pack patterns shown in table 35.7. Conversion from YCbCr 4:4:4 to YCbCr 4:2:2 or YCbCr 4:2:0 is performed by taking the average of two pixels for the horizontal direction, and simply skipping pixels for the vertical direction. The planes for outputting the Y and C data to memory are separate. Therefore, the ratio of the Y:C memory areas is 2:1 for YCbCr 4:2:0, 2:2 for YCbCr 4:2:2, and 2:4 for YCbCr 4:4:4. 00: YCbCr 4:2:0 output for YCbCr-format output 01: YCbCr 4:2:2 output for YCbCr-format output 10: YCbCr 4:4:4 output for YCbCr-format output 11: Setting prohibited
21	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
20 to 16	WPKF[4:0]	00000	R/W	These bits set the output data pack form when the VEU output data is in the RGB format. The VEU packs output data into a 32-bit pack. When data is output in the RGB format and YCbCr format, the data is packed into one of the patterns shown in tables 35.6 and 35.7, respectively. For RGB-format output, set a value in the WPKF column in table 35.6 these bits. When the RGB stuffing output pack is used (WPKF = H'10, H'11, H'12, or H'15), the start line must have the pack form of phase 0 shown in table 35.6.

Bit	Bit Name	Initial Value	R/W	Description
15, 14	CHRR[1:0]	00	R/W	These bits set the source image form for the VEU. When a YCbCr image is input, output is done in one of the pack patterns shown in table 35.7. 00: Source image is read in the YCbCr 4:2:0 format 01: Source image is read in the YCbCr 4:2:2 format 10: Source image is read in the YCbCr 4:4:4 format 11: Setting prohibited
13, 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 8	RPKF[3:0]	0000	R/W	These bits set the RGB input data pack form when the VEU input data is in the RGB format. Table 35.8 shows the relationship between the RGB format selected by these bits and the input form. When the VEU input data is in the RGB format, set a value in the RPKF column in table 35.8 in these bits.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	DITH	0	R/W	Selects whether dithering is applied or not when performing tone reduction of an RGB image. The VEU has a tone reduction function for RGB images. The number of tones after this processing is determined by the WPKF bit setting. Tone reduction processing for RGB images uses quantization which causes the resultant image to contain false contours. This problem becomes more serious as the number of tones is reduced, and it generally degrades the image quality. The VEU can perform dithering to lessen this degradation caused by tone reduction. 0: Does not apply dithering at tone reduction for the VEU destination image in the RGB format 1: Applies dithering at tone reduction for the VEU destination image in the RGB format

Bit	Bit Name	Initial Value	R/W	Description
3	TM2	0	R/W	The TM1 bit sets the color converting equation for the color-conversion circuit to that specified in ITU-R BT.601 or BT.709. The TM2 bit sets the color conversion range of 8-bit digital value between R/G/B and Y/Cb/Cr. 00: Use color conversion of full-range RGB[0,255] ⇔ Y[16,235], CbCr[16,240] conforms to ITU-R BT.601. 01: Use color conversion of full range RGB[0,255] ⇔ YCbCr[0,255] conforms to ITU-R BT.601. 10: Use color conversion of full-range RGB[0,255] ⇔ Y[16,235], CbCr[16,240] conforms to ITU-R BT.709. 11: Use color conversion of full-range RGB[16,235] ⇔ Y[16,235], CbCr[16,240] conforms to ITU-R BT.709.
2	TM1	0	R/W	
1	TE	0	R/W	The TE bit turns on or off the circuit for RGB ⇔ YCbCr conversion. The RY bit specifies whether the image input is in the YCbCr format or RGB format. Table 35.9 shows the relationship between the TE and RY bit settings and the input/output data format. 00: Reads a YCbCr source image and does not perform YCbCr → RGB conversion 01: Reads an RGB source image and does not perform RGB → YCbCr conversion 10: Reads a YCbCr source image and performs YCbCr → RGB conversion 11: Reads an RGB source image and performs RGB → YCbCr conversion
0	RY	0	R/W	

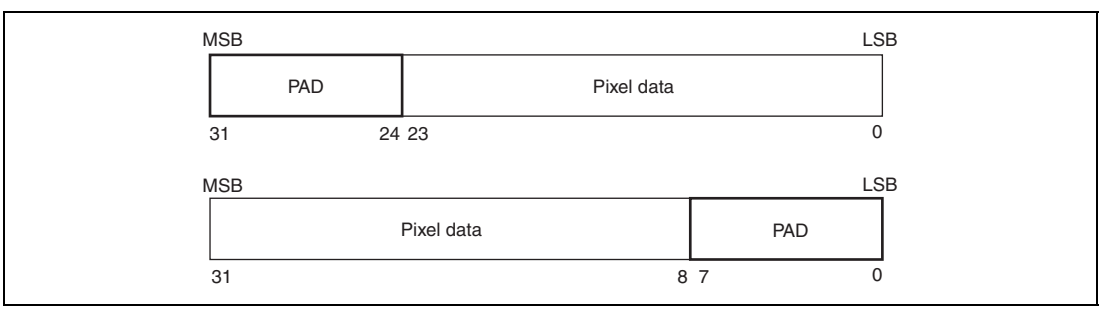


Figure 35.16 Data Pack Forms Including PAD

Table 35.6 RGB Data Output Pack Forms

No.	WPKF	Bit Rate [bpp]	P_density_y [pix/LW]	Phase	Bit																																			
					31 to 24								23 to 16								15 to 8								7 to 0											
0	B'00000	8	4	—	R0	R0	R0	G0	G0	G0	B0	B0	R1	R1	R1	G1	G1	G1	B1	B1	R2	R2	R2	G2	G2	G2	B2	B2	R3	R3	R3	R3	R3	R3	B3	B3
1	B'00001	12	2	—	0	0	0	0	R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0	0	0	0	0	R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1
2	B'00010				R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0	0	0	0	0	R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1	0	0	0	0
6	B'00110	16	2	—	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1
8	B'01000	18	1	—	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	0	0	0	0	0	0	PAD							
10	B'01010				PAD								R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	0	0	0	0	0	0
13	B'01101				PAD								R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	0	0
14	B'01110				R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	0	0	PAD							
16	B'10000		4/3	0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	0	0	0	0	0	0	R1	R1	R1	R1	R1	R1	G1	G1
				1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1	0	0	0	0	0	0	R2	R2	R2	R2	R2	R2	G2	G2	G2	G2	G2	G2	B2	B2	B2	B2
				2	B2	B2	0	0	0	0	0	0	R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3	B3	0	0	0	0	0	0
17	B'10001		1	0	0	0	R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	0	0	R1	R1	R1	R1	R1	R1
				1	0	0	G1	G1	G1	G1	G1	G1	0	0	B1	B1	B1	B1	B1	B1	0	0	R2	R2	R2	R2	R2	R2	0	0	G2	G2	G2	G2	G2	G2
				2	0	0	B2	B2	B2	B2	B2	B2	0	0	R3	R3	R3	R3	R3	R3	0	0	G3	G3	G3	G3	G3	G3	0	0	B3	B3	B3	B3	B3	B3
18	B'10010		4/3	0	R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	0	0	R1	R1	R1	R1	R1	R1	0	0
				1	G1	G1	G1	G1	G1	G1	0	0	B1	B1	B1	B1	B1	B1	0	0	R2	R2	R2	R2	R2	R2	0	0	G2	G2	G2	G2	G2	G2	0	0
				2	B2	B2	B2	B2	B2	B2	0	0	R3	R3	R3	R3	R3	R3	0	0	G3	G3	G3	G3	G3	G3	0	0	B3	B3	B3	B3	B3	B3	0	0
19	B'10011	24	1	—	PAD								R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0
20	B'10100				R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	PAD							
21	B'10101		4/3	0	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	R1	R1	R1	R1	R1	R1	R1	R1
				1	G1	G1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1	B1	B1	R2	R2	R2	R2	R2	R2	R2	R2	G2	G2	G2	G2	G2	G2	G2	G2
				2	B2	B2	B2	B2	B2	B2	B2	B2	R3	R3	R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3	B3	B3
22	B'10110	18	1	—	0	0	0	0	0	0	0	R0	R0	R0	R0	R0	R0	G0	G0	G0	0	0	0	0	0	0	0	G0	G0	G0	G0	G0	G0	G0	G0	G0
23	B'10111				0	0	0	0	0	0	0	0	0	0	0	0	0	0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	G0	G0	G0	G0

Table 35.7 YCbCr Data Pack Forms

No.	Compo-nent	Bit Rate [bpp]	P_density_y, P_density_c [pix/LW]	Bit																																				
				31 to 24									23 to 16								15 to 8								7 to 0											
0	Y data	16	4	Y0	Y0	Y0	Y0	Y0	Y0	Y0	Y0	Y1	Y1	Y1	Y1	Y1	Y1	Y1	Y1	Y2	Y2	Y2	Y2	Y2	Y2	Y2	Y2	Y3	Y3	Y3	Y3	Y3	Y3	Y3	Y3
	C data		4	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cb2	Cb2	Cb2	Cb2	Cb2	Cb2	Cb2	Cb2	Cr2	Cr2	Cr2	Cr2	Cr2	Cr2	Cr2	Cr2
1	Y data	16	4	Y0	Y0	Y0	Y0	Y0	Y0	Y0	Y0	Y1	Y1	Y1	Y1	Y1	Y1	Y1	Y1	Y2	Y2	Y2	Y2	Y2	Y2	Y2	Y2	Y3	Y3	Y3	Y3	Y3	Y3	Y3	Y3
	C data		4	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cb2	Cb2	Cb2	Cb2	Cb2	Cb2	Cb2	Cb2	Cr2	Cr2	Cr2	Cr2	Cr2	Cr2	Cr2	Cr2
2	Y data	24	4	Y0	Y0	Y0	Y0	Y0	Y0	Y0	Y0	Y1	Y1	Y1	Y1	Y1	Y1	Y1	Y1	Y2	Y2	Y2	Y2	Y2	Y2	Y2	Y2	Y3	Y3	Y3	Y3	Y3	Y3	Y3	Y3
	C data		2	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cb1	Cb1	Cb2	Cb1	Cb1	Cb1	Cb1	Cb1	Cr1	Cr1	Cr1	Cr1	Cr1	Cr1	Cr1	Cr1
			2	Cb2	Cb2	Cb2	Cb2	Cb2	Cb2	Cb2	Cb2	Cr2	Cr2	Cr2	Cr2	Cr2	Cr2	Cr2	Cr2	Cb3	Cb3	Cb3	Cb3	Cb3	Cb3	Cb3	Cb3	Cr3	Cr3	Cr3	Cr3	Cr3	Cr3	Cr3	Cr3

Table 35.8 RGB Data Input Pack Forms

No.	RPKF	Bit Rate [bpp]	P Density-y [pix/LW]	Phase	Bit																															
					31 to 24								23 to 16								15 to 8								7 to 0							
0	B'00000	24	1	—	X	X	X	X	X	X	X	X	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0
1	B'00001			—	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	X	X	X	X	X	X	X	X
2	B'00010		4/3	0	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	R1	R1	R1	R1	R1	R1	R1	R1
				1	G1	G1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1	B1	B1	R2	R2	R2	R2	R2	R2	R2	R2	G2	G2	G2	G2	G2	G2	G2	G2
				2	B2	B2	B2	B2	B2	B2	B2	B2	R3	R3	R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3	B3	B3
3	B'00011	16	2	—	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1
7	B'00111	18	1	—	X	X	X	X	X	X	X	X	X	X	X	X	X	X	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0
13	B'01101	16	1	—	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0

x : Don't care

Table 35.9 Relationship between TE and RY Bits in VTRCR and Input/Output Data Format

TE Bit	RY Bit	Input	Output
0	0	YCbCr format	YCbCr format
0	1	RGB format	RGB format
1	0	YCbCr format	RGB format
1	1	RGB format	YCbCr format

35.3.11 VEU Resize Filter Control Register (VRFCR)

VRFCR sets the scale-up/down factor for the image scaling filter. When performing scale-up/down processing, some other processings cannot be performed simultaneously (see table 35.2).

Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	VVMNT[3:0]				VVFRC[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VHMNT[3:0]				VHFRC[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	VVMNT[3:0]	H'0	R/W	Mantissa Part of Vertical Scaling Factor These bits set the vertical scaling factor. A value from H'0 to H'F can be set. When the VVMNT bits are set to H'0 and the VVFRC bits to H'000, the output is the same size.
27 to 16	VVFRC[11:0]	H'000	R/W	Fraction Part of Vertical Scaling Factor These bits set the vertical scaling factor. A value from H'000 to H'FFF can be set.
15 to 12	VHMNT[3:0]	H'0	R/W	Mantissa Part of Horizontal Scaling Factor These bits set the horizontal scaling factor. A value from H'0 to H'F can be set. When the VHMNT bits are set to H'0 and the VHFRC bits to H'000, the output is the same size.
11 to 0	VHFRC[11:0]	H'000	R/W	Fraction Part of Horizontal Scaling Factor These bits set the horizontal scaling factor. A value from H'000 to H'FFF can be set.

The VEU has an image scaling filter that can scale the image up and down, as shown in figure 35.17. The scaling filter cannot be operated simultaneously with the median filter, deblocking filter, or low-pass filter, and rotation is also prohibited. Accordingly, set B'0 to the bits in VFMCR, except the HMRR (horizontal inversion) and VMRR (vertical inversion) bits, when setting VRFCR.

The scaling factor for scale-up/down can be set between 1/16 to 16.

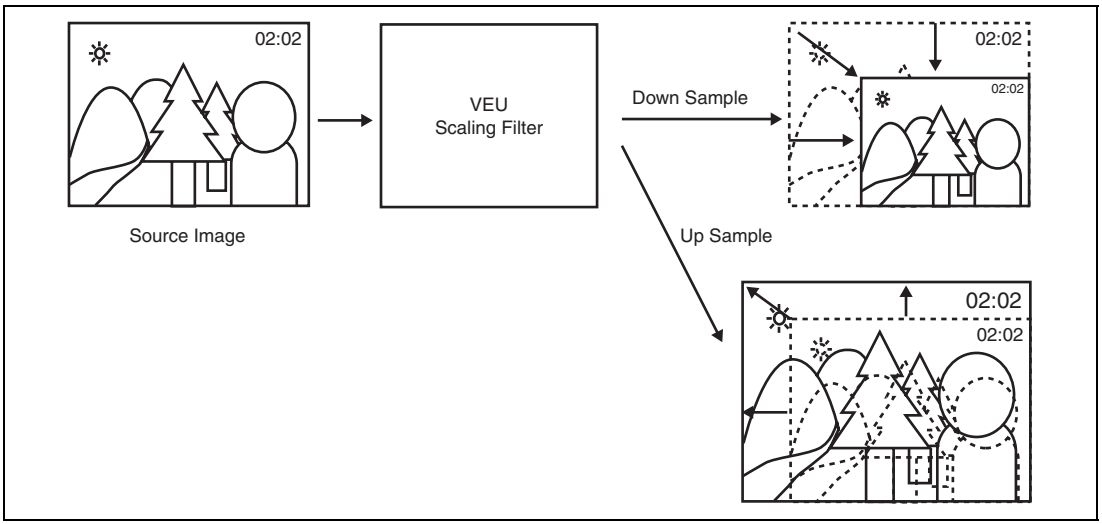


Figure 35.17 Scaling of Image by VEU

The formulas for obtaining the output pixel count of the scale-up/-down from the scaling ratio and the input pixel count are defined as follows:

$$\alpha = MANT \times 4096 + FRAC \quad \ldots \text{Formula 1}$$

$$SCL \text{ (scaling factor)} = \frac{4096}{\alpha} \quad \ldots \text{Formula 2}$$

In a formula listed below, decimal place of a calculation result of <A> is rounded off.

Here, an input image size is indicated as Sin, and a output image size as SIZE.

Note: Horizontal: Sin = VESSR.VHSS
Vertical: Sin = VESSR.VVSS

[Scale-down]

$$SIZE =< 1 + (< 1 + (Sin - 1) / MANTpre > - 1) \times MANTpre \times SCL > [1/16 < SCL \leq 1]$$
…Formula 3

MANTpre = 1 [1 ≤ MANT > 4]
= 2 [4 ≤ MANT > 8]
= 4 [8 ≤ MANT > 16]

[Scale-up]

$$SIZE =< 1 + (Sin - 1) \times SCL > [1 < SCL \leq 16, VRSCR.AMD = 0]$$ …Formula 4

$$SIZE =< Sin \times SCL > [1 < SCL \leq 16, VRSCR.AMD = 1]$$ …Formula 5

Note: See section 35.3.14, VEU Resize Filter Control Register (VRSCR), for details on the AMD bit.

Example: Scale up 88 pixels to 352 pixels (VRSCR.AMD = 0)

SCL = 352/88 = 4, suppose SCL = 4, and the preliminary settings of MANT = 0 and FRAC = 1024 are made based on formulas 1 and 2. Substituting these into formula 5 349 as the number of output pixels. Since this calculated value is smaller than the desired number of output pixels (352), set the SCL (formula 1) to the smallest value that is also greater than this value (MANT = 0, FRAC = H'3FF). At this point, the number of output pixels is 349, which is still not the desired number, so the calculation must be repeated. Setting to a value FRAC = H'3F7, the output pixel count becomes 352 which is the desired output pixel count. So set MANT = 0, FRAC = H'3F7 for the scaling up 88 pixels to 352.

Table 35.10 gives examples of settings for this register in the case of compression and expansion by the VEU module.

Table 35.10 Settings for Each Horizontal Scaling Factor of Scaling Filter

		FRAC		
Scaling Factor	**AMD**	**Decimal**	**Hexadecimal**	**MANT**
8	0	508	H'1FC	0
4	0	1017	H'3F9	0
2	0	2039	H'7F7	0
1.5	0	2723	H'AA3	0
8	1	512	H'200	0
4	1	1024	H'400	0
2	1	2048	H'800	0
1.5	1	2730	H'AAA	0
7/8	0	585	H'249	1
3/4	0	1365	H'555	1
5/8	0	2457	H'999	1
1/2	0	0	H'0	2
1/4	0	0	H'0	4
1/8	0	0	H'0	8

35.3.12 VEU Resize Filter Size Clip Register (VRFSR)

VRFSR sets the clipping (discarded) size of fractions in pixels output from the filter, and must be set in combination with VRFCR.

Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	VVCLP[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	VHCLP[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 17	VVCLP[11:1]	H'000	R/W	These bits set the vertical clipping size after scale-up/down in pixel units (1-pixel units). This setting should be made in 2-pixel units when the YCbCr format is 4:2:0.
16	VVCLP[0]	H'0	R	
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 1	VHCLP[11:1]	H'000	R/W	These bits set the horizontal clipping size after scale-up/down in pixel units (4-pixel units). This setting should be made in 2-pixel units when the WPKF bits (write pack) in VTRCR are set to 1, 2, 6, 8 to 14, 19, 20, 22, or 23.
0	VHCLP[0]	H'0	R	

The VVCLP and VHCLP bits in the VRFSR register set the output size when line processing (with scale-up/down, enhancer, or bundle read) is performed. The size can be specified in 4-pixel units horizontally and in 1-pixel (2-pixel for YCbCr4:2:0 format) units vertically in VRFSR. However, a setting in 2-pixel units horizontally is possible for RGB 2 bytes/pixel (VTRCR.WPKF = 1, 2, 6) and RGB 4 bytes/pixel (VTRCR.WPKF = 8 to 14, 19, 20, 22, 23).

The maximum value to be set in the VVCLP and VHCLP bits in the VRFSR register is 4092 pixels. The minimum value for them is 16 pixels. Note that the values should be such that VRFSR.VHCLP + VRSOR.VHCLOFS ≤ 4092.

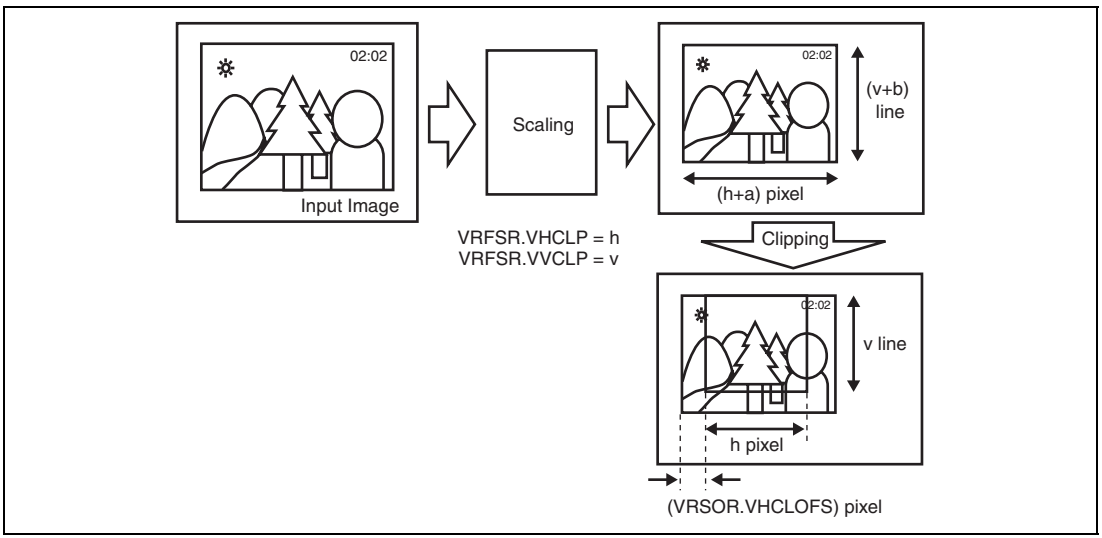


Figure 35.18 Clipping of Image Output from VEU Scaling Filter

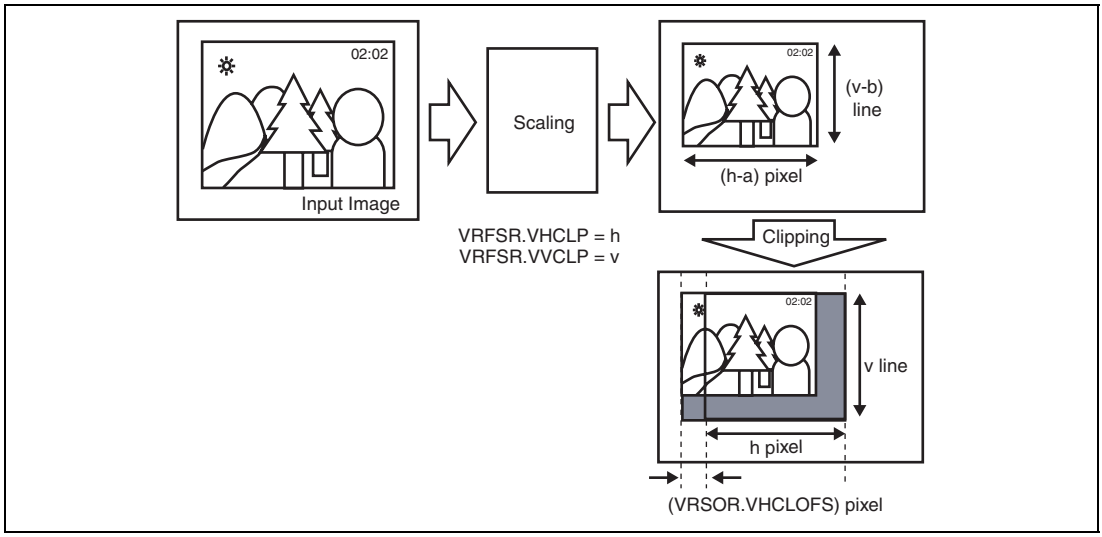


Figure 35.19 Shifting of Image Output from VEU Scaling Filter

If the values are smaller than the filter output pixel count, output images are clipped as shown in figure 35.18. The number of output pixels is counted from the position shifted to the right by the offset specified in VRSOR.VHCLOFS. The pixels that are located at the right of the specified pixel count or below the specified line count are discarded by the clipping function.

If the values are larger than the filter output pixel count, images are output by complementing pixels up to the specified pixel count according to VRSCR.FMD, where the pixels are counted from the position shifted to the right by the offset specified in VRSOR.VHCLOFS register (see figure 35.19).

However, if the horizontal number of pixels specified in VRFSR is larger than the number of the scaling filter output pixels rounded up to the nearest multiple of 16, the complementary function does not work. In this case, the VEU may be hung up, so the VRFSR value should be a value that is obtained by calculating by the formulas 1 to 6 and rounding up to a multiple of 16.

If the filter output pixel count obtained by the formulas 1 to 6 is equal to 16M +N (N = 1 to 16), VRFSR.VV(H)CLP ≤ 16M+16.

When scale-up/down is not performed, set VRFSR.VHCLP = VESSR.VHSS, VRFSR.VVCLP = VESSR.VVSS.

35.3.13 VEU Enhance Register (VENHR)

VENHR sets up the filter (enhancer) for enhancing the edges of images. Some other processings cannot be performed simultaneously with the edge enhancement processing (see table 35.2). Modifying this register during operation is prohibited.

The enhancer cannot be operated simultaneously with the median filter, deblocking filter, or low-pass filter, and rotation is also prohibited. Accordingly, set B'0 to the bits in VFMCR, except the HMRR (horizontal inversion) and VMRR (vertical inversion) bits, when setting VENHR.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ENSCL[2:0]			—	—	—	—	—	—	ENHV	ENHH
Initial value:	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 8	ENSCL[2:0]	010	R/W	These bits set the factor for edge enhancement. 001: Enhancement level 1 (max) 010: Enhancement level 2 011: Enhancement level 3 100: Enhancement level 4 (min) Other than above: Setting prohibited
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1	ENHV	0	R/W	Enables or disables edge enhancement in the vertical direction. 0: Vertical edge enhancement is not performed 1: Vertical edge enhancement is performed
0	ENHH	0	R/W	Enables or disables edge enhancement in the horizontal direction. 0: Horizontal edge enhancement is not performed 1: Horizontal edge enhancement is performed

35.3.14 VEU Resize Filter Control Register (VRSCR)

VRSCR performs scaling filter adjustments for line processing (with scale-up/down, enhancer, or bundle read). For block processing, set all of the initial values of VRSCR to 0 (without scale-up/down, enhancer, or bundle read).

Writing 1 to the read-only bits is prohibited. If 1 is written to any of these bits a malfunction may occur. Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	AMD	FMD	LC[1:0]		—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	AMD	0	R/W	This bit specifies the pixel count generated when the scaling filter performs a scale-up. When AMD = 1 is specified, int (n* scale-up factor) pixels are created by complementing horizontal and vertical pixels in the destination image for AMD = 0. 0: The pixel count obtained at a scale-up is 1+ int ((n –1)* scale-up factor). 1: The pixel count obtained at a scale-up is int (n* scale-up factor).
30	FMD	0	R/W	This bit specifies the pixel complementing method for the scaling filter. When the scaling filter outputs an image that does not fill the clip size, the pixels are complemented up to the clip size. 0: Copy right (lower) pixels and provide complementary pixels. 1: Provide complementary right(lower) pixels of a color specified by VFLCR.
29, 28	LC[1:0]	00	R/W	These bits specify the number of pixels for source image horizontal-left clipping. From 0 to 3 can be specified. Because the source address registers VSAYR and VSCAR are both in longword units, if the input format is YCbCr, the starting position for horizontal clipping from the background image is in four-pixel units. Therefore, if you perform background image clipping from pixel position at the left end 4M+LC (M; integer, LC: 0-3), 4M is adjusted by the VSAYR read start address, and the lower two bits are adjusted by this bit (LC). When the value of this bit has been set, input to the scaling filter is specified by VESSR.VHSS – VRSCR.LC for the horizontal direction and by VESSR.VVSS for the vertical direction.
27 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

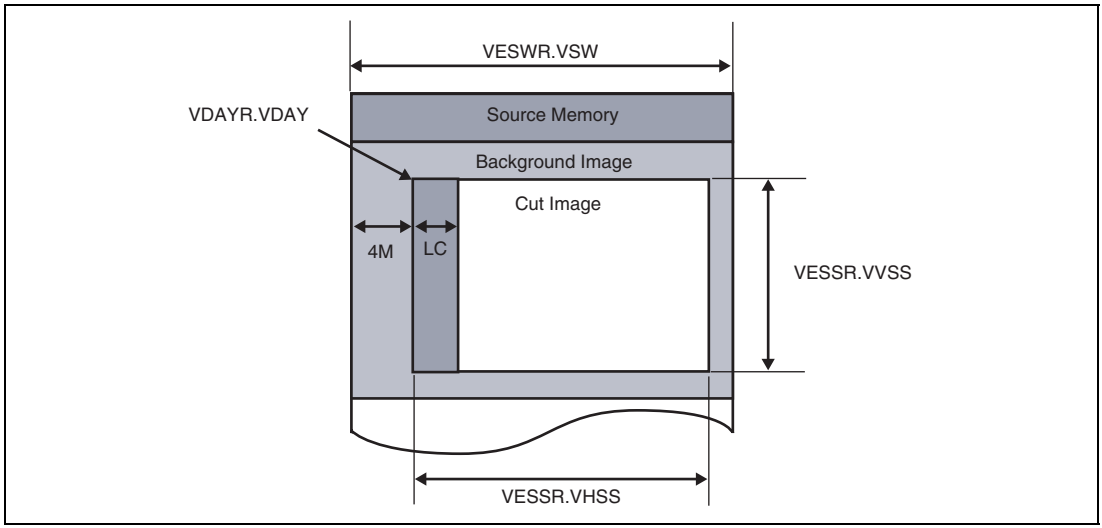


Figure 35.20 Relations between the Background Image and Clipping Pixel Location

35.3.15 VEU Resize Filter Size Clip Offset Register (VRSOR)

VRSOR sets the left clip offset when images are clipped bases on the filter output pixel count. For details on the offset positions, see figures 35.18 and 35.19. When following Flow 2 in figure 35.2, set an offset that is smaller than the horizontal number of pixels of the output image after scaling down.

Also, make sure such that VRFSR.VHCLP + VRSOR.VHCLOF ≤ 4092.

When this register is set, the VRFCR and VRFSR registers should also be set. Since the register setting is used as the output size when Flow 2 describe in section 35.2, Functional Overview of VEU (Scale-up/down, enhancer, or bundle read is performed), this register should be set even if scale-up/down is not performed.

Writing to this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	VHCLOFS[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	VHCLOFS [7:0]	H'00	R/W	Specifies the lateral clip offset by the number of pixel (one-pixel unit) after scale-up/down.

35.3.16 VEU Filter Mode Control Register (VFMCR)

VFMCR sets the operating mode for filter processing.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	MED FST	FLTPI	—	—	—	—	—	—	—	MED
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	TPN	—	—	DBLK	LPHV	—	—	VMRR	HMRR	—	—	ROTL	ROTR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25	MEDFST	0	R/W	Changes the order of applying the median filter and deblocking filter when the FLTPI bit is set to 1. 0: Applies the LPF/deblocking filer and median filter in this order. 1: Applies the median filer and LPF/deblocking filter in this order.

Bit	Bit Name	Initial Value	R/W	Description
24	FLTPI	0	R/W	Controls fast processing of LPF, deblocking filer, and median filter. 0: Disables fast processing of LPF, deblocking filer, and median filter. 1: Enables fast processing of LPF, deblocking filer, and median filter. When the FLTPI is set to 1, the possible settings are restricted to the followings: (1) VFMCR.ROTR = 0 (2) VFMCR.ROTL = 0 (3) VESTR.VBE = 0 (4) VRSOR.VHCLOFS = 0 (5) VRFSR.VVCLP = VSSR.VVSS (6) VRFSR.VHCLP = VSSR.VHSS
23 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	MED	0	R/W	Enables or disables usage of the median filter. When the median filter is applied to an image, scale-up/down processing and edge enhance processing cannot be performed at the same time. In bundle read mode, the median filter cannot be used. Therefore, when this bit is set to 1, set VRFCR, VENHR, and the VBE bit in VESTR to 0. (Interoperating with the scale-up/down processing, edge enhance processing, and bundle read mode is prohibited.) The relationship between this bit and the filter operation is shown in table 35.11. 0: Median filter is not used 1: Median filter is applied to the source image

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	TPN	0	R/W	Sets the number of taps of the low-pass filter (LPF). 0: LPF tap count is set to 3 taps 1: LPF tap count is set to 5 taps
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	DBLK	0	R/W	Enables or disables usage of the deblocking filter. When the deblocking filter is applied to an image in bundle read mode, scale-up/down processing and edge enhance processing cannot be performed at the same time. When this bit is set to 1, set VRFCR, VENHR, and the VBE bit in VESTR to 0. (Interoperating with the scale-up/down processing, edge enhance processing, and bundle read mode is prohibited.) The relationship between this bit and the filter operation is shown in table 35.11. 0: LPF is applied to the entire source image (normal mode) 1: LPF is applied to only the boundary of the 8 × 8 pixel blocks of the source image (deblocking mode)
8	LPHV	0	R/W	Enables or disables usage of the low-pass filter. In bundle read mode, the low-pass filter cannot be used. When the low-pass filter is applied to an image, scale-up/down processing and edge enhance processing cannot be performed at the same time. When this bit is set to 1, set VRFCR, VENHR, and the VBE bit in VESTR to 0. (Interoperating with the scale-up/down processing, edge enhance processing, and bundle read mode is prohibited.) The relationship between this bit and the filter operation is shown in table 35.11. 0: NOP 1: LPF is applied to the source image

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	VMRR	0	R/W	Enables or disables usage of the vertical inversion (symmetric with regard to horizontal axis) filter. The relationship between this bit and the rotate/invert operation is shown in table 35.12. 0: NOP 1: Vertical inversion (symmetric with regard to horizontal axis) filter is applied to the source image
4	HMRR	0	R/W	Enables or disables usage of the horizontal inversion (symmetric with regard to vertical axis) filter. The relationship between this bit and the rotate/invert operation is shown in table 35.12. 0: NOP 1: Horizontal inversion (symmetric with regard to vertical axis) filter is applied to the source image
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	ROTL	0	R/W	Enables or disables usage of the 270-degree rotation (clockwise) filter. In bundle read mode, rotation cannot be performed. When rotation processing is performed for an image, scale-up/down processing and edge enhance processing cannot be performed at the same time. When this bit is set to 1, set VRFCR, VENHR, and the VBE bit in VESTR to 0. (Interoperating with the scale-up/down processing, edge enhance processing, and bundle read mode is prohibited.) The relationship between this bit and the rotate/invert operation is shown in table 35.12. 0: NOP 1: 270-degree rotation (clockwise) filter is applied to the source image

Bit	Bit Name	Initial Value	R/W	Description
0	ROTR	0	R/W	Enables or disables usage of the 90-degree rotation (clockwise) filter. In bundle read mode, rotation cannot be performed. When rotation processing is performed for an image, scale-up/down processing and edge enhance processing cannot be performed at the same time. When this bit is set to 1, set VRFCR, VENHR, and the VBE bit in VESTR to 0. (Interoperating with the scale-up/down processing, edge enhance processing, and bundle read mode is prohibited.) The relationship between this bit and the rotate/invert operation is shown in table 35.12. 0: NOP 1: 90-degree rotation (clockwise) filter is applied to the source image

Table 35.11 Relationship between MED, DBLK, and LPHV Bits and Filter Operation

MED Bit	DBLK Bit	LPHV Bit	Filter Operation
0	0	0	Passed through
0	0	1	Low-pass filter
0	1	1	Deblocking filter
1	1	1	Deblocking filter + median filter*
1	0	0	Median filter
Other than above			Setting prohibited

Note: * The order of deblocking filter → median filter cannot be changed.

Table 35.12 Relationship between VMRR, HMRR, ROTL, and ROTR Bits and Rotate/Invert Operation

VMRR Bit	HMRR Bit	ROTL Bit	ROTR Bit	Rotate/Invert Operation
0	0	0	0	No rotation/inversion
0	0	0	1	Rotated by 90 degrees clockwise
0	0	1	0	Rotated by 270 degrees clockwise
0	1	0	1	Rotated by 90 degrees clockwise and then inverted in the horizontal direction
1	0	0	1	Rotated by 90 degrees clockwise and then inverted in the vertical direction
0	1	0	0	Inverted in the horizontal direction
1	0	0	0	Inverted in the vertical direction
1	1	0	0	Rotated by 180 degrees
Other than above				Setting prohibited

To rotate the image 270 degrees and then invert it in the horizontal direction, rotate the image 90 degrees and then invert it in the vertical direction. Likewise, to rotate the image 270 degrees and then invert it in the vertical direction, rotate the image 90 degrees and then invert it in the horizontal direction.

Figures 35.21 and 35.22 show the relationship between the rotated/inverted images and raw image.

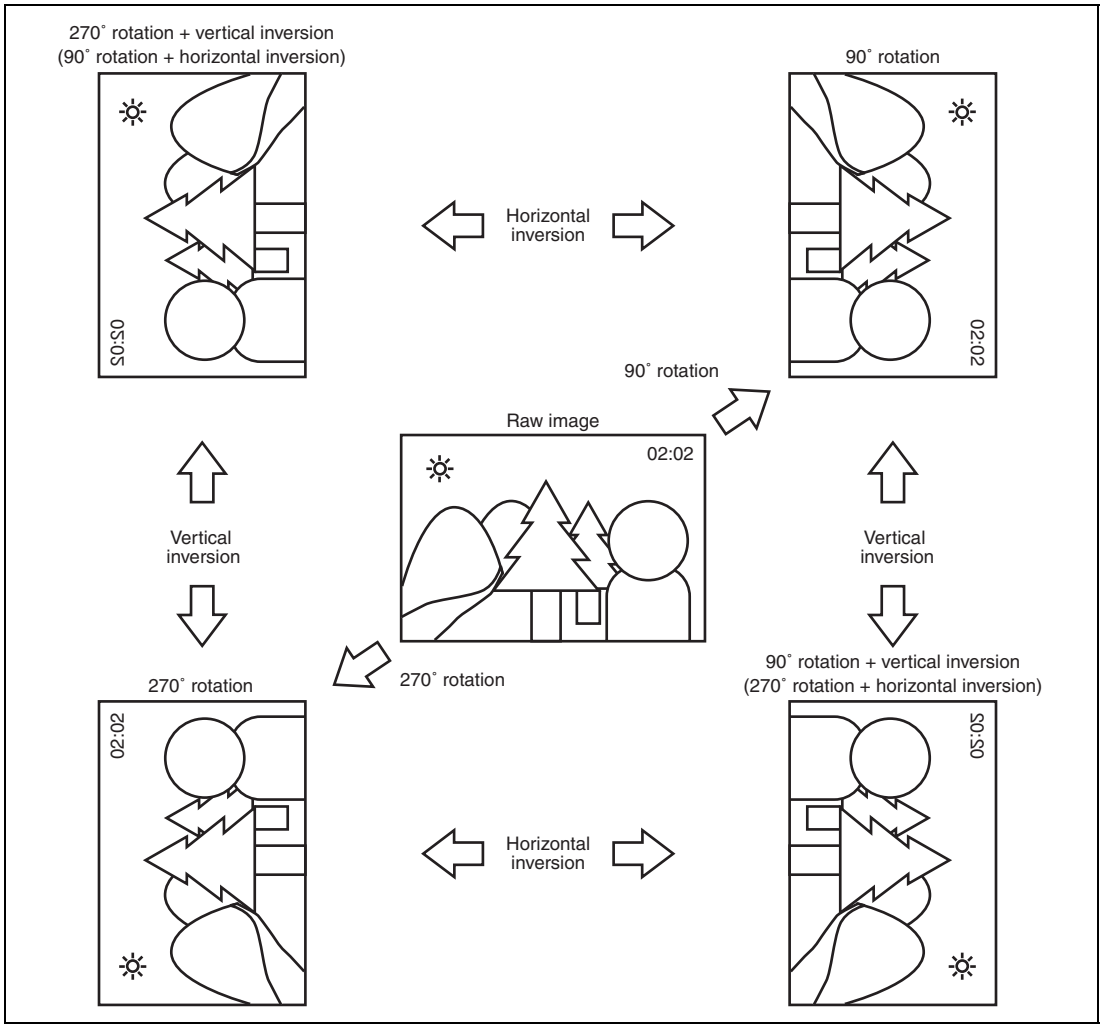


Figure 35.21 Relationship between Rotated/Inverted Images and Raw Image

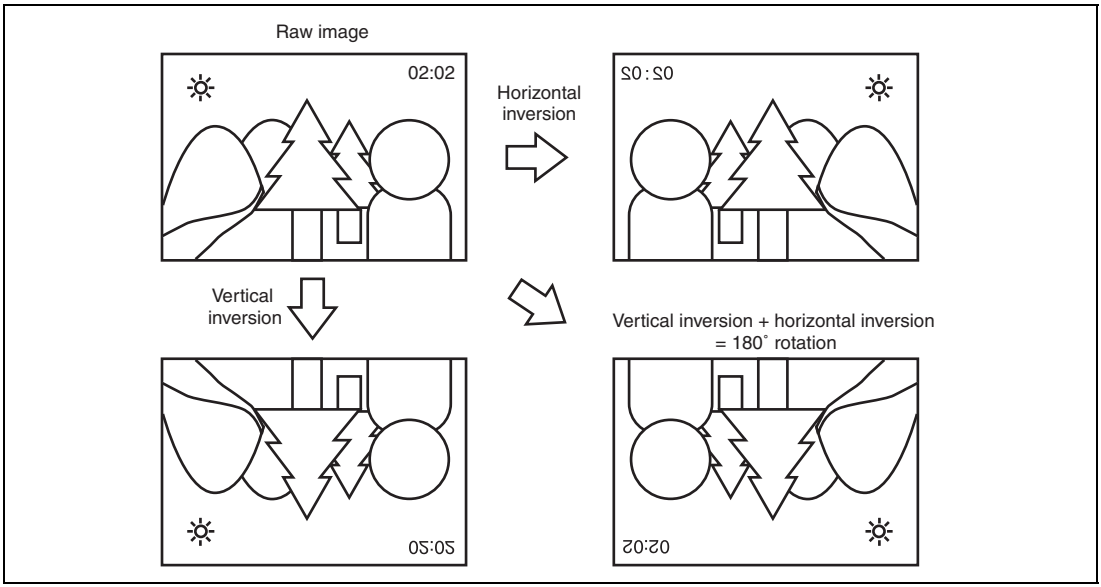


Figure 35.22 Relations between Inverted Images and Raw Image

35.3.17 VEU Vertical Tap Coefficient Register (VVTCR)

VVTCR sets the vertical tap coefficients when applying the low-pass filter.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	VSHFT[2:0]			VTPC4[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VTPC3[3:0]				VTPC2[3:0]				VTPC1[3:0]				VTPC0[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
22 to 20	VSHFT[2:0]	000	R/W	These bits specify the shift capacity after vertical tap calculation. Set how many pixels the location is to be shifted right after adding each vertical tap coefficient. This setting must correspond to the sum of the VTPC0 to VTPC4 bits. See table 35.13 for details on the values corresponding to the total value of VVTCR.VTPC.
19 to 16	VTPC4[3:0]	H'0	R/W	These bits set vertical tap coefficient 4 (clear these bits to 0 for a 3-tap filter). Set the component of the pixel two pixels right of that location to be given when applying a 5-tap low-pass filter. Clear these bits to 0 when applying a 3-tap low-pass filter.
15 to 12	VTPC3[3:0]	H'0	R/W	These bits set vertical tap coefficient 3. Set the component of the pixel on the right of that location to be given when applying a low-pass filter.

Bit	Bit Name	Initial Value	R/W	Description
11 to 8	VTPC2[3:0]	H'0	R/W	These bits set vertical tap coefficient 2. Set the component of that location to be given when applying a low-pass filter.
7 to 4	VTPC1[3:0]	H'0	R/W	These bits set vertical tap coefficient 1. Set the component of the pixel on the left of that location to be given when applying a low-pass filter
3 to 0	VTPC0[3:0]	H'0	R/W	These bits set vertical tap coefficient 0 (clear these bits to 0 for a 3-tap filter). Set the component of the pixel two pixels left of that location to be given when applying a 5-tap low-pass filter. Clear these bits to 0 when applying a 3-tap low-pass filter.

Table 35.13 Shifting Value Corresponds to Sum of VTPC4 to VTPC0

ΣVVTCR.VTPC	VVTCR.VSHFT
4	2
8	3
16	4
32	5
64	6

Examples of setting VVTCR are shown below. When the VTPC4 bits are H'0 and the VTPC0 bits are H'0, set the TPN bit in VFMCR to B'0 because a 3-tap low-pass filter is to be used. In all other cases, set the TPN bit to B'1 because a 5-tap low-pass filter is to be used.

Table 35.14 Values Set in VVTCR

Bit Name	VSHFT	VTPC4	VTPC3	VTPC2	VTPC1	VTPC0
Values to be set	2	0	1	2	1	0
	3	0	1	6	1	0
	3	1	2	2	2	1
	3	1	1	4	1	1
	4	0	1	14	1	0
	4	0	3	10	3	0
	4	0	5	6	5	0
	4	1	1	12	1	1
	4	1	2	10	2	1
	4	1	3	8	3	1
	4	1	4	6	4	1
	4	2	3	6	3	2
	4	3	3	4	3	3
	5	0	9	14	9	0
	5	1	8	14	8	1
	5	1	9	12	9	1
	5	1	10	10	10	1
	5	2	7	14	7	2
	5	2	9	10	9	2
	5	3	6	14	6	3
	5	3	7	12	7	3
	5	3	8	10	8	3
	5	4	5	14	5	4
	5	4	7	10	7	4
	5	5	5	12	5	5
	5	5	6	10	6	5
	5	5	7	8	7	5
	5	5	5	12	5	5
	5	5	6	10	6	5
	5	5	7	8	7	5
	6	11	14	14	14	11
	6	12	13	14	13	12

Note: Horizontal and vertical taps should be set in the same size.

35.3.18 VEU Horizontal Tap Coefficient Register (VHTCR)

VHTCR sets the horizontal tap coefficients when applying the low-pass filter.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	HSHFT[2:0]			HTPC4[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HTPC3[3:0]				HTPC2[3:0]				HTPC1[3:0]				HTPC0[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
22 to 20	HSHFT[2:0]	000	R/W	These bits specify the shift capacity after horizontal tap calculation. Set how many pixels the location is to be shifted right after adding each horizontal tap coefficient. For details on the setting, see the description of the VSHFT bits in VVTCR. See table 35.15 for details on the values corresponding to the total value of VHTCR.HTPC.
19 to 16	HTPC4[3:0]	H'0	R/W	These bits set horizontal tap coefficient 4. Set the component of the pixel two pixels right of that location to be given when applying a 5-tap low-pass filter. Clear these bits to 0 when applying a 3-tap low-pass filter.
15 to 12	HTPC3[3:0]	H'0	R/W	These bits set horizontal tap coefficient 3. Set the component of the pixel on the right of that location to be given when applying a low-pass filter.
11 to 8	HTPC2[3:0]	H'0	R/W	These bits set horizontal tap coefficient 2. Set the component of that location to be given when applying a low-pass filter.
7 to 4	HTPC1[3:0]	H'0	R/W	These bits set horizontal tap coefficient 1. Set the component of the pixel on the left of that location to be given when applying a low-pass filter

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	HTPC0[3:0]	H'0	R/W	These bits set horizontal tap coefficient 0. Set the component of the pixel two pixels left of that location to be given when applying a 5-tap low-pass filter. Clear these bits to 0 when applying a 3-tap low-pass filter.

Table 35.15 Shifting Value Corresponds to Sum of HTPC4 to HTPC0

VHTCR.HTPC	VHTCR.HSHFT
4	2
8	3
16	4
32	5
64	6

Examples of setting VHTCR are shown below. When both the HTPC4 bits and HTPC0 bits are H'0, set the TPN bit in VFMCR to B'0 because a 3-tap low-pass filter is to be used. In all other cases, set the TPN bit to B'1 because a 5-tap low-pass filter is to be used.

Table 35.16 Values Set in VHTCR

Bit Name	HSHFT	HTPC4	HTPC3	HTPC2	HTPC1	HTPC0
Values to be set	2	0	1	2	1	0
	3	0	1	6	1	0
	3	1	2	2	2	1
	3	1	1	4	1	1
	4	0	1	14	1	0
	4	0	3	10	3	0
	4	0	5	6	5	0
	4	1	1	12	1	1
	4	1	2	10	2	1
	4	1	3	8	3	1
	4	1	4	6	4	1
	4	2	3	6	3	2
	4	3	3	4	3	3
	5	0	9	14	9	0
	5	1	8	14	8	1
	5	1	9	12	9	1
	5	1	10	10	10	1
	5	2	7	14	7	2
	5	2	9	10	9	2
	5	3	6	14	6	3
	5	3	7	12	7	3
	5	3	8	10	8	3
	5	4	5	14	5	4
	5	4	7	10	7	4
	5	5	5	12	5	5
	5	5	6	10	6	5
	5	5	7	8	7	5
	5	5	5	12	5	5
	5	5	6	10	6	5
	5	5	7	8	7	5
	6	11	14	14	14	11
	6	12	13	14	13	12

Note: Horizontal and vertical taps should be set in the same size.

35.3.19 VEU Designated Color Register (VAPCR)

VAPCR specifies the designated color.

When any pixel in an image to be processed by the VEU is the same as the designated color set by this register, that pixel is replaced with the conversion color set by VCCR. For an RGB output, clear the lower bits except the valid bits to 0. For a YCbCr output, only the YCbCr 4:4:4 format is supported.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	APCON	RAPC[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAPC[7:0]								BAPC[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	APCON	0	R/W	Enables or disables replacement of the designated color. 0: VEU processed image is output as it is 1: VEU processed image is output after pixels corresponding to the designated color have been replaced with the conversion color
23 to 16	RAPC[7:0]	H'00	R/W	These bits specify the R (Cb) component of the designated color.
15 to 8	GAPC[7:0]	H'00	R/W	These bits specify the G (Y) component of the designated color.
7 to 0	BAPC[7:0]	H'00	R/W	These bits specify the B (Cr) component of the designated color.

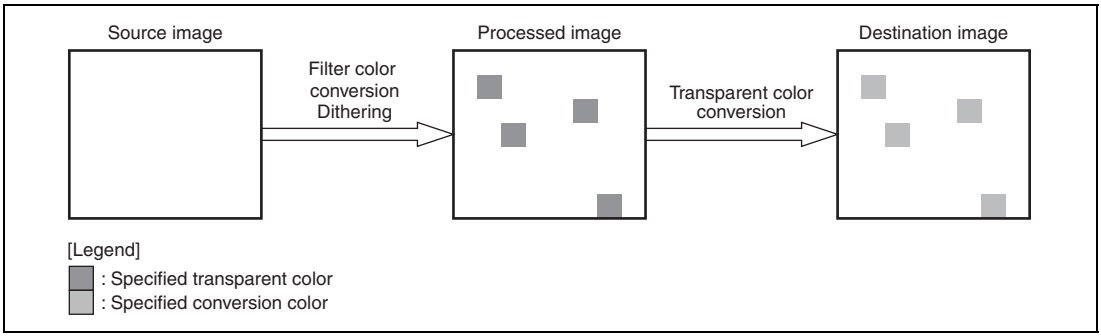


Figure 35.23 Replacement of Designated – Color Pixels in VEU Processed Image with Conversion – Color Pixels

35.3.20 VEU Conversion Color Register (VECCR)

VECCR specifies the conversion color.

When any pixel in an image to be processed by the VEU is the same as the designated color set by VAPCR, that pixel is replaced with the conversion color set by this register. For an RGB output, clear the lower bits except for the valid bits to 0. For a YCbCr output, only the YCbCr 4:4:4 format is supported.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	RCHGC[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GCHGC[7:0]								BCHGC[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	RCHGC[7:0]	H'00	R/W	These bits specify the R (Cb) component of the conversion color.
15 to 8	GCHGC[7:0]	H'00	R/W	These bits specify the G (Y) component of the conversion color.
7 to 0	BCHGC[7:0]	H'00	R/W	These bits specify the B (Cr) component of the conversion color.

35.3.21 VEU Fill Color Specification Register (VFLCR)

VFLCR is a fill color specification register.

When VRSCR.FMD = 1, VFLCR specifies the fill color used to complement pixels when the pixel area created by the VEU scaling filter is smaller than the clip area specified by the clip size register. For RGB output, specify 0 for the lower bits other than the valid bits.
RGB

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	RFILC[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GFILC[7:0]								BFILC[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	RFILC[7:0]	H'00	R/W	These bits specify the R (Cb) component of the fill color.
15 to 8	GFILC[7:0]	H'00	R/W	These bits specify the G (Y) component of the fill color.
7 to 0	BFILC[7:0]	H'00	R/W	These bits specify the B (Cr) component of the fill color.

35.3.22 VEU Address Fixed Register (VAFXR)

VAFXR specifies fixed mode for the addresses of the data output from the VEU.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VAFIX
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	VAFIX	0	R/W	Sets fixed mode for output addresses. When this bit is 1, the addresses to draw the image output from the VEU is fixed to the values set in VDAYR. When this bit is 0, the output address is incremented according to the data and VEDWR. Fixed address mode is only available when the VEU destination image is in the RGB format. For YCbCr destination images, set this bit to 0. When the fixed address mode is in use, set all of bits MED, LPHV, ROTL, and ROTR in VFMCR (which prohibit median filter, low-pass filter, and rotation) to 0, set the VRFCR register (same-size output) to 0, and set the three lower-order bits in VDAYR and VRFSR (which disable transfer other than burst transfer) to 0. 0: Output addresses are not in fixed address mode 1: Output addresses are in fixed address mode

35.3.23 VEU Swapping Register (VSWPR)

VSWPR sets swapping within the 64-bit data at the data input/output section of the VEU.

When dividing addresses into 64 bits, make the following setting for each case.

- When swapping data in longword units for all inputs/outputs: H'0000 0044
- When swapping data in word units for all inputs/outputs: H'0000 0022
- When swapping data in byte units for all inputs/outputs: H'0000 0011
- When swapping data in byte units from the MSB to the LSB: H'0000 0077

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	VEOLS	VEOWS	VEOBS	—	VEILS	VEIWS	VEIBS
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6	VEOLS	0	R/W	Sets swapping in longword units for output data. In longword swapping for output data, data is swapped in longword units within the 32 bits on the MSB side and within the 32 bits on the LSB side for 64-bit data in the VEU output section (figure 35.24). 0: Output data is not swapped in longword units 1: Output data is swapped in longword units

Bit	Bit Name	Initial Value	R/W	Description
5	VEOWS	0	R/W	Sets swapping data in word units for output data. In word swapping for output data, data is swapped in word units within the 32 bits on the MSB side and within the 32 bits on the LSB side for 64-bit data in the VEU output section (figure 35.25). 0: Output data is not swapped in word units 1: Output data is swapped in word units
4	VEOBS	0	R/W	Sets swapping data in byte units for output data. In byte swapping for output data, data is swapped in byte units within each 16 bits for 64-bit data in the VEU output section (figure 35.26). 0: Output data is not swapped in byte units 1: Output data is swapped in byte units
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	VEILS	0	R/W	Sets swapping data in longword units for input data. In longword swapping for input data, data is swapped in longword units within the 32 bits on the MSB side and within the 32 bits on the LSB side for 64-bit data in the VEU input section (figure 35.24). 0: Input data is not swapped in longword units 1: Input data is swapped in longword units
1	VEIWS	0	R/W	Sets swapping data in word units for input data. In word swapping for input data, data is swapped in word units within the 32 bits on the MSB side and within the 32 bits on the LSB side for 64-bit data in the VEU input section (figure 35.25). 0: Input data is not swapped in word units 1: Input data is swapped in word units

Bit	Bit Name	Initial Value	R/W	Description
0	VEIBS	0	R/W	Sets swapping data in byte units for input data. In byte swapping for input data, data is swapped in byte units within each 16 bits for 64-bit data in the VEU input section (figure 35.26). 0: Input data is not swapped in byte units 1: Input data is swapped in byte units

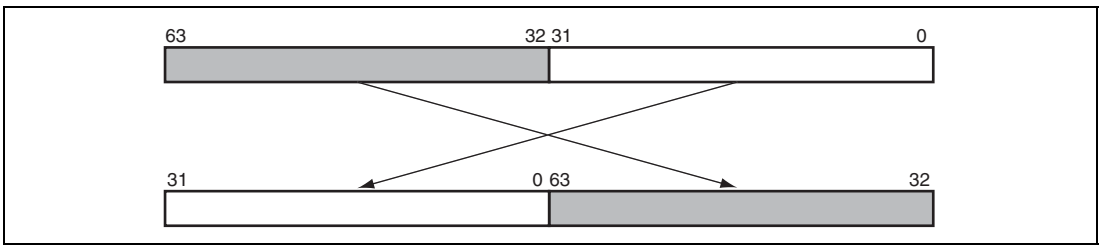


Figure 35.24 Relations before and after Data Swapping in Longword Units

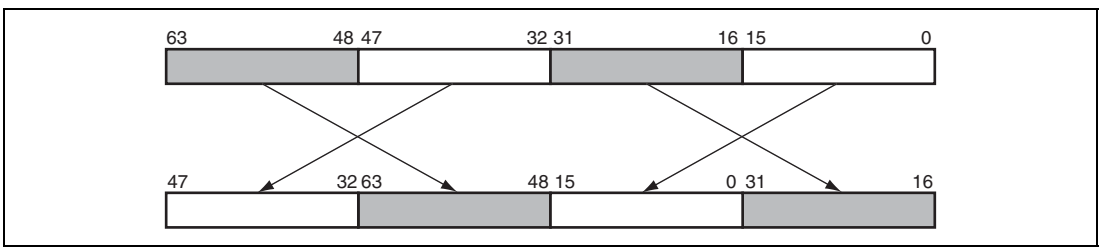


Figure 35.25 Relations before and after Data Swapping in Word Units

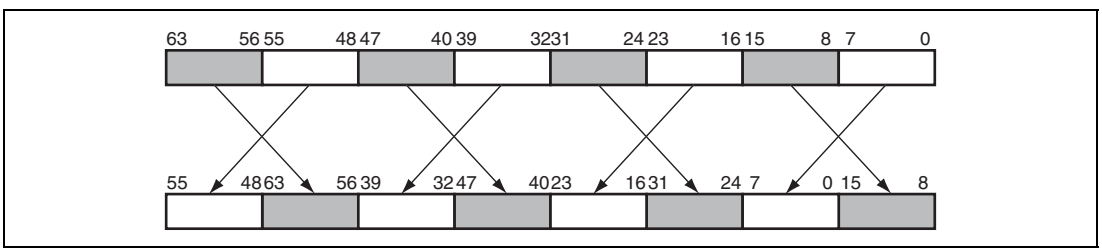


Figure 35.26 Relations before and after Data Swapping in Byte Units

35.3.24 VEU Event Interrupt Enable Register (VEIER)

VEIER enables or disables output of the interrupt signal of a VEVTR flag.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	VBENDE	—	—	—	—	—	—	—	VEENDE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	VBENDE	0	R/W	Enables or disables the interrupt signal of VEVTR.VBEND to be output. When VEIER.VBENDE = 1, the interrupt signal is output if VEVTR.VBENDE becomes 1. When VEIER.VEENDE = 0, the interrupt signal is not output even if VEVTR.VBEND becomes 1. 0: Disables output of the VEVTR.VBEND interrupt signal 1: Enables output of the VEVTR.VBEND interrupt signal
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
0	VEENDE	0	R/W	Enables or disables the interrupt signal of VEVTR.VEEND to be output. When VEIER.VEENDE = 1, the interrupt signal is output if VEVTR.VEENDE becomes 1. When VEIER.VEENDE = 0, the interrupt signal is not output even if VEVTR.VEEND becomes 1. 0: Disables output of the VEVTR.VEEND interrupt signal 1: Enables output of the VEVTR.VEEND interrupt signal

35.3.25 VEU Event Register (VEVTR)

VEVTR indicates the interrupt source when an internal interrupt occurs in the VEU. Whether output of the interrupt signal of each source in VEVTR is enabled or disabled is set by VEIER.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	VBEND	—	—	—	—	—	—	—	VEEND
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	VBEND	0	R/W	This flag is set to 1 when VEU processing has finished for a single read (reading of the number of lines set in VBSSR) in bundle read mode (N-line read mode). The VEU enters the read restart wait state when this flag is set. Therefore, after clearing the interrupt source, switch the address registers, and start reading again (VESTR = H'0000 0011). [Reading] 0: Indicates in progress of reading in bundle read processing, or this flag has already been cleared 1: Indicates that restart is being waited in bundle read processing [Writing] 0: Clears this flag by writing 0 to it 1: Holds the current value
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	VEEND	0	R/W	This flag is set to 1 when VEU processing has totally finished. This flag is not cleared unless 0 is written to clear the bit after VEU processing ends. Therefore, 0 must be written to this bit to clear it before the VEU is activated again. [Reading] 0: Indicates that VEU processing has not finished, or this flag has already been cleared 1: Indicates that VEU processing has finished [Writing] 0: Clears this flag by writing 0 to it 1: Holds the current value

35.3.26 VEU Status Register (VSTAR)

VSTAR indicates the internal status of the VEU and the internal signal states.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	INTL	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	VB PRC	—	—	—	VB READ	—	—	—	—	—	—	—	VE PRC
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	INTL	0	R	Indicates the assert state of the VEU interrupt signal to the CPU. 0: Indicates that no interrupt signal is asserted at the VEU interrupt signal port 1: Indicates that an interrupt signal is asserted at the VEU interrupt signal port
23 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	VBPRC	0	R	Indicates the VEU status in bundle read mode. For details, see table 35.17. 0: Indicates that the VEU is not operating in bundle read (N-line read) mode 1: Indicates that the VEU is operating in bundle read (N-line read) mode
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
8	VBREAD	0	R	For details, see table 35.18. 0: Indicates that the VEU is waiting to be restarted in bundle read (N-line read) mode 1: Indicates that the VEU is reading in bundle read (N-line read) mode
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	VEPRC	0	R	Indicates the same value as the VE bit in VESTR. For details, see table 35.18. 0: VEU is ready (halted) 1: VEU is busy (operating state)

The VEU operates in either normal operating mode or bundle read mode. In normal operating mode, the registers are not to be modified between activation and processing end. In bundle read mode, there are two operating states: read processing state and the state waiting for the VEU to be restarted by software. The values when reading VSTAR in each state are shown in table 35.17. Figure 35.27 shows the VSTAR state transitions in normal operating mode, and figure 35.27 shows the VSTAR state transitions in bundle read mode.

Table 35.17 VSTAR Value Read in Each State

Operating Mode		VBPRC Bit	VBREAD Bit	VEPRC Bit
Normal operation	Halted	0	0	0
	Operating	0	0	1
N-line read mode	Halted	0	0	0
	During read processing	1	1	1
	Waiting for restart	1	0	1

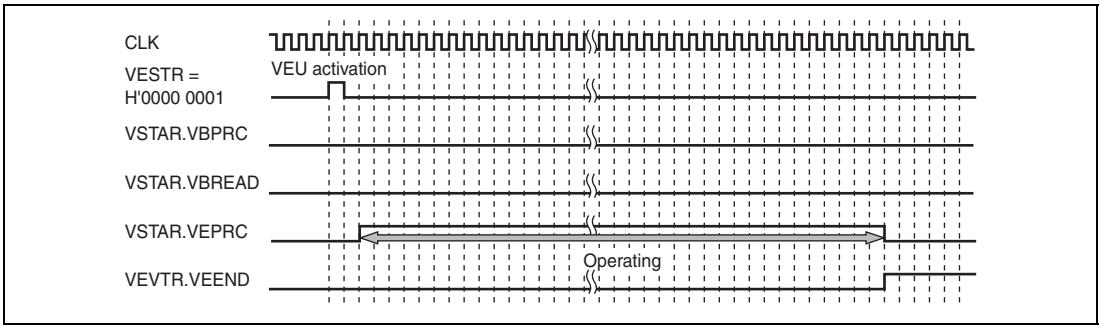


Figure 35.27 VSTAR State Transitions in Normal Operating Mode

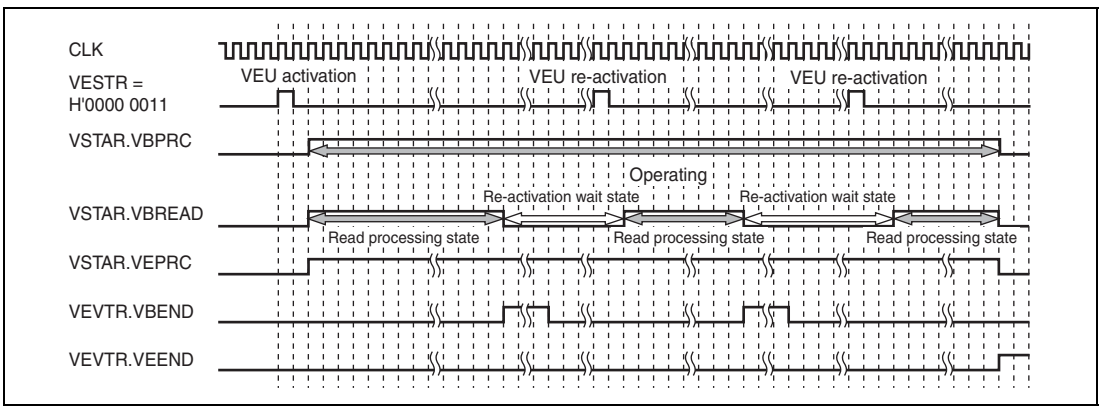


Figure 35.28 VSTAR State Transitions in Bundle Read Mode

35.3.27 VEU Module Reset Register (VBSRR)

VBSRR executes the module reset of the VEU.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying this register during operation is prohibited.

If a module reset is performed during operation, the logic circuits handshaking with the CPU bus are forcibly reset, and malfunction may also affect modules other than the VEU. To correctly break the handshake with the CPU bus before VEU termination, see section 35.3.1, VEU Start Register (VESTR).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ALL RST	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	ALLRST	0	W	Module Reset When 1 is written to this bit, all internal control signals of the VEU are reset. 0: Setting prohibited 1: Internal reset of the VEU
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

35.3.28 VEU Resize Passband Register (VRPBR)

VRPBR sets the signal passbands when scale-up/down is performed. When this register is set, the VRFCR and VRFSR registers should be also set. Writing this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	VBW[6:0]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	HBW[6:0]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
22 to 16	VBW[6:0]	H'00	R/W	Sets the vertical signal passband when scaling-up/down.
15 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6 to 0	HBW[6:0]	H'00	R/W	Sets the horizontal signal passband when scaling-up/down.

When scaling up vertically (VRFCR.VMNT = 0), set 64 in VRPBR.VBW.

When scaling down vertically (other than VRFCR.VMNT = 0), set a value obtained by the following formula in VRPBR.VBW:

$$VSPBR.VBW = \left(64 \times \frac{4096 \times VMANTpre}{4096 \times VRFCR.VMNT + VRFCR.VFRC} \right)$$

Where VMANTpre is

4(8 ≤ VRFCR.VMNT<16)
2(4 ≤ VRFCR.VMNT<8)
1(1 ≤ VRFCR.VMNT<4)

Note that <A> is an operation that rounds value A down to the nearest integer.

When scaling up horizontally (VRFCR.HMNT = 0), set 64 in VRPBR.HBW.

When scaling down horizontally (other than VRFCR.HMNT = 0), set a value obtained by the following formula in VRPBR.HBW:

$$VSPBR.HBW = \left(64 \times \frac{4096 \times HMANTpre}{4096 \times VRFCR.HMNT + VRFCR.HFRC} \right)$$

Where HMANTpre is

4(8 ≤ VRFCR.HMNT<16)
2(4 ≤ VRFCR.HMNT<8)
1(1 ≤ VRFCR.HMNT<4)

Note that <A> is an operation that rounds value A down to the nearest integer.

35.4 Usage Notes for VEU

(1) Restrictions during Operation

Do not halt the clock or module during VEU operation.

(2) Restrictions on Input/Output Functions

Note that when restrictions at other places in this manual differ from restrictions in the following table, restrictions in the following table are given priority.

Table 35.18 Restrictions on VEU Input/Output Functions

Item	Pack Type	Restrictions
Input	RGB565, 4 bytes/pixel	• The start address for input must be specified in longword units. • The horizontal size of the raw image (memory) must be specified in byte units that correspond to two pixels of the source image.
	Others	• The start address for input must be specified in longword units. However, when the RGB stuffing pack is input, the address must be specified so that reading starts from the pack of phase 0. • The horizontal size of the raw image (memory) must be specified in byte units that correspond to four pixels of the destination image.
Output	RGB565 4 bytes/pixel	• The output address must be specified in longword units. • The horizontal size of the destination image (memory) must be specified in byte units that correspond to two pixels of the destination image.
	Others	• The output address must be specified in longword units. • The horizontal size of the destination image (memory) must be specified in byte units that correspond to four pixels of the destination image.

Section 36 GPIO

36.1 Overview

This LSI incorporates six GPIO blocks, each of which is a functional block that supports a maximum of 32-channel ports for general input/output and interrupt input. (A maximum of 171 ports in total can be used. Twelve out of 171 channels are dedicated for general input and interrupt input. Note that the port pins are multiplexed.) When the relevant register is written to, a signal is output via the corresponding general output port pin. When a signal is input via the general input port pin, the corresponding register indicates the value of the input signal; specifically, when an interrupt is input via a general port pin, the relevant register indicates that it is currently receiving an interrupt input, and an interrupt is also requested to the SH4A core via the interrupt control block. The functions (modes) can be assigned to each port channel as desired by so setting the corresponding registers. It is also possible to select the signal polarity (positive or negative logic) and the interrupt detection condition (edge or level) for each port. Particularly, a filtering function to prevent external chattering is also available for channels 0 to 3 in input modes.

36.1.1 GPIO Block Diagram

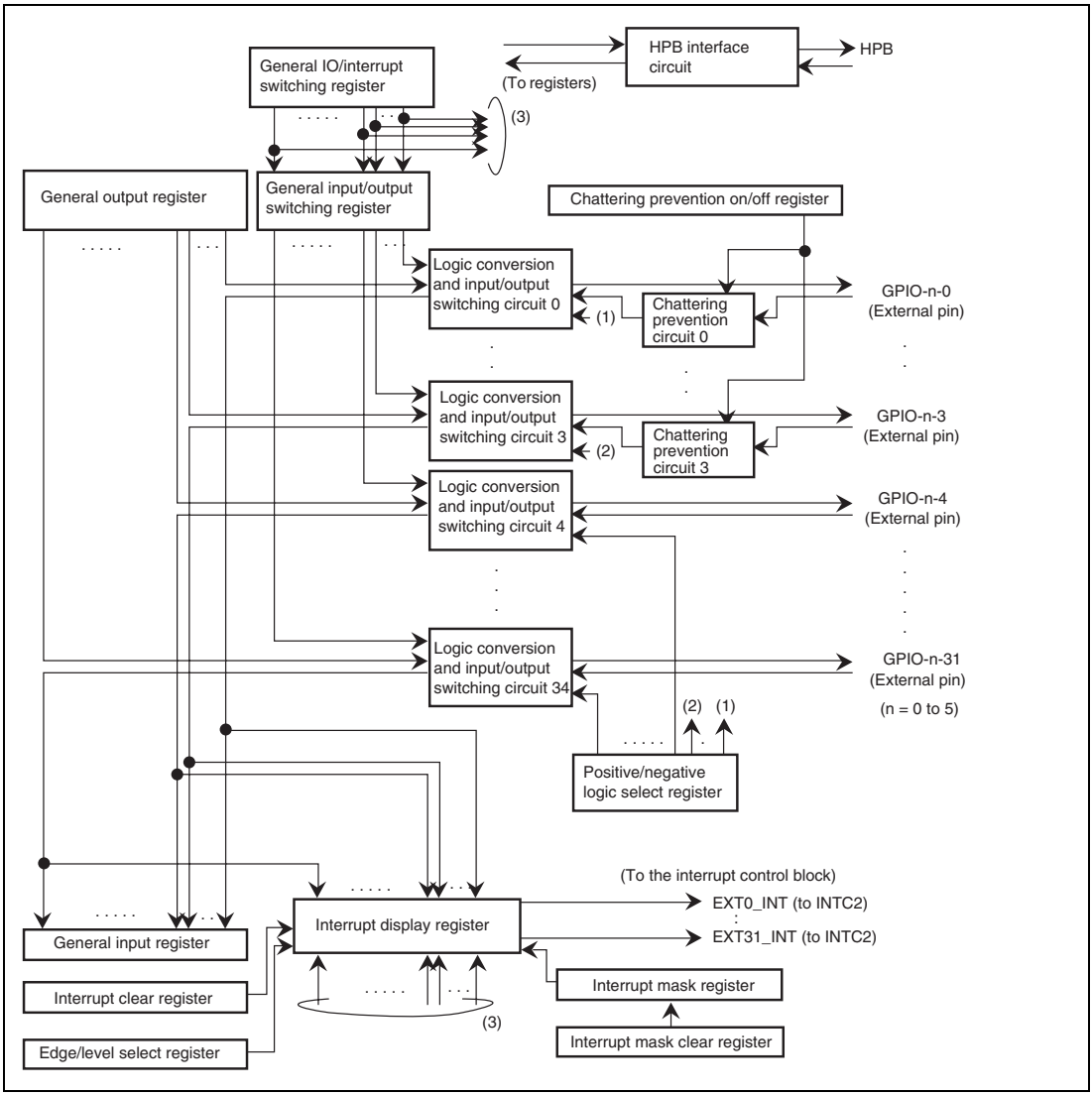


Figure 36.1 GPIO Block Configuration

36.1.2 I/O Pins

Table 36.1 shows the pin configuration of the GPIO blocks.

Table 36.1 Pin Configuration

Signal Name	Pin Name	I/O	Descriptions
IO/interrupt input ports	GP-0-0 to GP-5-31	I/O	General input/output port and interrupt input

36.1.3 Mode Switching

Two registers are used to switch modes of the general IO/interrupt input pins (GPIO-n-0 to GPIO-n-31) of the GPIO blocks. Each register is provided with 32 bits each controlling one of the 32-channel pins. The general IO/interrupt switching register is first used to select either general input/output mode or interrupt input mode for each channel. When general input/output mode is selected, the setting of the relevant bit in the second register, i.e., the general input/output switching register, is used. Specifically, when a bit in the general input/output switching register is set for general output mode, the corresponding port pin is turned to the output direction and the route is formed so that the set value in the corresponding bit in the general output register should be output via the pin. Likewise, when set for the general input mode, the corresponding port pin is turned to the input direction and the route is formed so that the value received via the pin should be indicated by the corresponding bit in the general input register. When interrupt input mode is selected, the corresponding port pin is turned to the input direction and the route is formed so that the reception of the signal input via the pin should be indicated by the interrupt display register. Here, the setting of the second register, i.e., the general input/output switching register, is invalid.

36.2 Port Pin Specifications

Each GPIO block is provided with 32-channel pins for general input/output and external interrupt input ports. Table 36.2 specifies these pins.

Table 36.2 Port Pin Specifications (1/6)

Block	Number	Abbreviation	Name	Descriptions
GPIO-0 Applicable registers: IOINTSEL0 INOUTSEL0 OUTDT0 INDT0 INTDT0 INTCLR0 INTMSK0 MSKCLR0 POSNEG0 EDGLEVEL0 FILONOFF0	1	GP-0-0	IO/interrupt input port A0	• Either general input/output mode or interrupt input mode can be set for each port. • In general input mode, the polarity of input signals can be set for each port. • In general output mode, the polarity of output signals can be set for each port. • In interrupt input mode, the polarity of interrupt signal can be set for each port. • In interrupt input mode, the detection conditions of interrupt signals can be set for each port.
	2	GP-0-1	IO/interrupt input port A1	
	3	GP-0-2	IO/interrupt input port A2	
	4	GP-0-3	IO/interrupt input port A3	
	5	GP-0-4	IO/interrupt input port A4	
	6	GP-0-5	IO/interrupt input port A5	
	7	GP-0-6	IO/interrupt input port A6	
	8	GP-0-7	IO/interrupt input port A7	
	9	GP-0-8	IO/interrupt input port A8	
	10	GP-0-9	IO/interrupt input port A9	
	11	GP-0-10	IO/interrupt input port A10	
	12	GP-0-11	IO/interrupt input port A11	
	13	GP-0-12	IO/interrupt input port A12	
	14	GP-0-13	IO/interrupt input port A13	
	15	GP-0-14	IO/interrupt input port A14	
	16	GP-0-15	IO/interrupt input port A15	
	17	GP-0-16	IO/interrupt input port A16	
	18	GP-0-17	IO/interrupt input port A17	
	19	GP-0-18	IO/interrupt input port A18	
	20	GP-0-19	IO/interrupt input port A19	
	21	GP-0-20	IO/interrupt input port A20	
	22	GP-0-21	IO/interrupt input port A21	
	23	GP-0-22	IO/interrupt input port A22	
	24	GP-0-23	IO/interrupt input port A23	
	25	GP-0-24	IO/interrupt input port A24	
	26	GP-0-25	IO/interrupt input port A25	
	27	GP-0-26	IO/interrupt input port A26	
	28	GP-0-27	IO/interrupt input port A27	
	29	GP-0-28	IO/interrupt input port A28	
	30	GP-0-29	IO/interrupt input port A29	
	31	GP-0-30	IO/interrupt input port A30	
	32	GP-0-31	IO/interrupt input port A31	

Table 36.2 Port Pin Specifications (2/6)

Block	Number	Abbreviation	Name	Descriptions
GPIO-1 Applicable registers: IOINTSEL1 INOUTSEL1 OUTDT1 INDT1 INTDT1 INTCLR1 INTMSK1 MSKCLR1 POSNEG1 EDGLEVEL1 FILONOFF1	33	GP-1-0	IO/interrupt input port A0	• Either general input/output mode or interrupt input mode can be set for each port. • In general input mode, the polarity of input signals can be set for each port. • In general output mode, the polarity of output signals can be set for each port. • In interrupt input mode, the polarity of interrupt signal can be set for each port. • In interrupt input mode, the detection conditions of interrupt signals can be set for each port.
	34	GP-1-1	IO/interrupt input port A1	
	35	GP-1-2	IO/interrupt input port A2	
	36	GP-1-3	IO/interrupt input port A3	
	37	GP-1-4	IO/interrupt input port A4	
	38	GP-1-5	IO/interrupt input port A5	
	39	GP-1-6	IO/interrupt input port A6	
	40	GP-1-7	IO/interrupt input port A7	
	41	GP-1-8	IO/interrupt input port A8	
	42	GP-1-9	IO/interrupt input port A9	
	43	GP-1-10	IO/interrupt input port A10	
	44	GP-1-11	IO/interrupt input port A11	
	45	GP-1-12	IO/interrupt input port A12	
	46	GP-1-13	IO/interrupt input port A13	
	47	GP-1-14	IO/interrupt input port A14	
	48	GP-1-15	IO/interrupt input port A15	
	49	GP-1-16	IO/interrupt input port A16	
	50	GP-1-17	IO/interrupt input port A17	
	51	GP-1-18	IO/interrupt input port A18	
	52	GP-1-19	IO/interrupt input port A19	
	53	GP-1-20	IO/interrupt input port A20	
	54	GP-1-21	IO/interrupt input port A21	
	55	GP-1-22	IO/interrupt input port A22	
	56	GP-1-23	IO/interrupt input port A23	
	57	GP-1-24	IO/interrupt input port A24	
	58	GP-1-25	IO/interrupt input port A25	
	59	GP-1-26	IO/interrupt input port A26	
	60	GP-1-27	IO/interrupt input port A27	
	61	GP-1-28	IO/interrupt input port A28	
	62	GP-1-29	IO/interrupt input port A29	
	63	GP-1-30	IO/interrupt input port A30	
	64	GP-1-31	IO/interrupt input port A31	

Table 36.2 Port Pin Specifications (3/6)

Block	Number	Abbreviation	Name	Descriptions
GPIO-2	65	GP-2-0	IO/interrupt input port A0	• Either general input/output mode or interrupt input mode can be set for each port. • In general input mode, the polarity of input signals can be set for each port. • In general output mode, the polarity of output signals can be set for each port. • In interrupt input mode, the polarity of interrupt signal can be set for each port. • In interrupt input mode, the detection conditions of interrupt signals can be set for each port.
Applicable registers: IOINTSEL2 INOUTSEL2 OUTDT2 INDT2 INTDT2 INTCLR2 INTMSK2 MSKCLR2 POSNEG2 EDGLEVEL2 FILONOFF2	66	GP-2-1	IO/interrupt input port A1	
	67	GP-2-2	IO/interrupt input port A2	
	68	GP-2-3	IO/interrupt input port A3	
	69	GP-2-4	IO/interrupt input port A4	
	70	GP-2-5	IO/interrupt input port A5	
	71	GP-2-6	IO/interrupt input port A6	
	72	GP-2-7	IO/interrupt input port A7	
	73	GP-2-8	IO/interrupt input port A8	
	74	GP-2-9	IO/interrupt input port A9	
	75	GP-2-10	IO/interrupt input port A10	
	76	GP-2-11	IO/interrupt input port A11	
	77	GP-2-12	IO/interrupt input port A12	
	78	GP-2-13	IO/interrupt input port A13	
	79	GP-2-14	IO/interrupt input port A14	
	80	GP-2-15	IO/interrupt input port A15	
	81	GP-2-16	IO/interrupt input port A16	
	82	GP-2-17	IO/interrupt input port A17	
	83	GP-2-18	IO/interrupt input port A18	
	84	GP-2-19	IO/interrupt input port A19	
	85	GP-2-20	IO/interrupt input port A20	
	86	GP-2-21	IO/interrupt input port A21	
	87	GP-2-22	IO/interrupt input port A22	
	88	GP-2-23	IO/interrupt input port A23	
	89	GP-2-24	IO/interrupt input port A24	
	90	GP-2-25	IO/interrupt input port A25	
	91	GP-2-26	IO/interrupt input port A26	
	92	GP-2-27	IO/interrupt input port A27	
	93	GP-2-28	IO/interrupt input port A28	
	94	GP-2-29	IO/interrupt input port A29	
	95	GP-2-30	IO/interrupt input port A30	
	96	GP-2-31	IO/interrupt input port A31	

Table 36.2 Port Pin Specifications (4/6)

Block	Number	Abbreviation	Name	Descriptions
GPIO-3	97	GP-3-0	IO/interrupt input port A0	• Either general input/output mode or interrupt input mode can be set for each port. • In general input mode, the polarity of input signals can be set for each port. • In general output mode, the polarity of output signals can be set for each port. • In interrupt input mode, the polarity of interrupt signal can be set for each port. • In interrupt input mode, the detection conditions of interrupt signals can be set for each port.
	98	GP-3-1	IO/interrupt input port A1	
Applicable registers: IOINTSEL3 INOUTSEL3 OUTDT3 INDT3 INTDT3 INTCLR3 INTMSK3 MSKCLR3 POSNEG3 EDGLEVEL3 FILONOFF3	99	GP-3-2	IO/interrupt input port A2	
	100	GP-3-3	IO/interrupt input port A3	
	101	GP-3-4	IO/interrupt input port A4	
	102	GP-3-5	IO/interrupt input port A5	
	103	GP-3-6	IO/interrupt input port A6	
	104	GP-3-7	IO/interrupt input port A7	
	105	GP-3-8	IO/interrupt input port A8	
	106	GP-3-9	IO/interrupt input port A9	
	107	GP-3-10	IO/interrupt input port A10	
	108	GP-3-11	IO/interrupt input port A11	
	109	GP-3-12	IO/interrupt input port A12	
	110	GP-3-13	IO/interrupt input port A13	
	111	GP-3-14	IO/interrupt input port A14	
	112	GP-3-15	IO/interrupt input port A15	
	113	GP-3-16	IO/interrupt input port A16	
	114	GP-3-17	IO/interrupt input port A17	
	115	GP-3-18	IO/interrupt input port A18	
	116	GP-3-19	IO/interrupt input port A19	
	117	GP-3-20	IO/interrupt input port A20	
	118	GP-3-21	IO/interrupt input port A21	
	119	GP-3-22	IO/interrupt input port A22	
	120	GP-3-23	IO/interrupt input port A23	
	121	GP-3-24	IO/interrupt input port A24	
	122	GP-3-25	IO/interrupt input port A25	
	123	GP-3-26	IO/interrupt input port A26	
	124	GP-3-27	IO/interrupt input port A27	
	125	GP-3-28	IO/interrupt input port A28	
	126	GP-3-29	IO/interrupt input port A29	
	127	GP-3-30	IO/interrupt input port A30	
	128	GP-3-31	IO/interrupt input port A31	

Table 36.2 Port Pin Specifications (5/6)

Block	Number	Abbreviation	Name	Descriptions
GPIO-4 Applicable registers: IOINTSEL4 INOUTSEL4 OUTDT4 INDT4 INTDT4 INTCLR4 INTMSK4 MSKCLR4 POSNEG4 EDGLEVEL4 FILONOFF4	129	GP-4-0	IO/interrupt input port A0	• Either general input/output mode or interrupt input mode can be set for each port. • In general input mode, the polarity of input signals can be set for each port. • In general output mode, the polarity of output signals can be set for each port. • In interrupt input mode, the polarity of interrupt signal can be set for each port. • In interrupt input mode, the detection conditions of interrupt signals can be set for each port.
	130	GP-4-1	IO/interrupt input port A1	
	131	GP-4-2	IO/interrupt input port A2	
	132	GP-4-3	IO/interrupt input port A3	
	133	GP-4-4	IO/interrupt input port A4	
	134	GP-4-5	IO/interrupt input port A5	
	135	GP-4-6	IO/interrupt input port A6	
	136	GP-4-7	IO/interrupt input port A7	
	137	GP-4-8	IO/interrupt input port A8	
	138	GP-4-9	IO/interrupt input port A9	
	139	GP-4-10	IO/interrupt input port A10	
	140	GP-4-11	IO/interrupt input port A11	
	141	GP-4-12	IO/interrupt input port A12	
	142	GP-4-13	IO/interrupt input port A13	
	143	GP-4-14	IO/interrupt input port A14	
	144	GP-4-15	IO/interrupt input port A15	
	145	GP-4-16	IO/interrupt input port A16	
	146	GP-4-17	IO/interrupt input port A17	
	147	GP-4-18	IO/interrupt input port A18	
	148	GP-4-19	IO/interrupt input port A19	
	149	GP-4-20	IO/interrupt input port A20	
	150	GP-4-21	IO/interrupt input port A21	
	151	GP-4-22	Input/interrupt input port A22	
	152	GP-4-23	Input/interrupt input port A23	
	153	GP-4-24	Input/interrupt input port A24	
	154	GP-4-25	Input/interrupt input port A25	
	155	GP-4-26	IO/interrupt input port A26	
	156	GP-4-27	IO/interrupt input port A27	
	157	GP-4-28	IO/interrupt input port A28	
	158	GP-4-29	IO/interrupt input port A29	
	159	GP-4-30	IO/interrupt input port A30	
	160	GP-4-31	IO/interrupt input port A31	

Table 36.2 Port Pin Specifications (6/6)

Block	Number	Abbreviation	Name	Descriptions
GPIO-5	161	GP-5-0	IO/interrupt input port A0	• Either general input/output mode or interrupt input mode can be set for each port. • In general input mode, the polarity of input signals can be set for each port. • In general output mode, the polarity of output signals can be set for each port. • In interrupt input mode, the polarity of interrupt signal can be set for each port. • In interrupt input mode, the detection conditions of interrupt signals can be set for each port.
	162	—	—	
Applicable registers: IOINTSEL5 INOUTSEL5 OUTDT5 INDT5 INTDT5 INTCLR5 INTMSK5 MSKCLR5 POSNEG5 EDGLEVEL5 FILONOFF5	163	GP-5-2	Input/interrupt input port A2	
	164	GP-5-3	Input/interrupt input port A3	
	165	GP-5-4	Input/interrupt input port A4	
	166	GP-5-5	Input/interrupt input port A5	
	167	GP-5-6	Input/interrupt input port A6	
	168	GP-5-7	Input/interrupt input port A7	
	169	GP-5-8	Input/interrupt input port A8	
	170	GP-5-9	Input/interrupt input port A9	
	171	GP-5-10	IO/interrupt input port A10	
	172	GP-5-11	IO/interrupt input port A11	
	173	—	—	
	174	—	—	
	175	—	—	
	176	—	—	
	177	—	—	
	178	—	—	
	179	—	—	
	180	—	—	
	181	—	—	
	182	—	—	
	183	—	—	
	184	—	—	
	185	—	—	
	186	—	—	
	187	—	—	
	188	—	—	
	189	—	—	
	190	—	—	
	191	—	—	
	192	—	—	

36.3 Operations in Each Mode

36.3.1 General Input/Output Mode

When a port is set for general input/output mode using the corresponding bit in the general IO/interrupt switching register, the corresponding port serves as a general input/output pin. In general input/output mode, either mode can be selected using the corresponding bit in the general input/output switching register. When a port is set for general output mode, the port outputs the value set in the corresponding bit in the general output register. Here, the polarity of the actual output signal is determined by the setting of the corresponding bit in the positive/negative logic select register. When a port is set for general input mode, the polarity of the input signal is also determined by the setting of the corresponding bit in the positive/negative logic select register. The general input register indicates the value accordingly. Note that the general input register does not hold the input signal using the FF.

36.3.2 Interrupt Input Mode

When a port is set for interrupt input mode using the corresponding bit in the general IO/interrupt switching register, the corresponding port serves as an interrupt input pin. In interrupt input mode, when the port receives an external interrupt, the corresponding bit in the interrupt display register indicates the input of an interrupt signal on the corresponding port pin, and an interrupt signal is output to the interrupt control block. In this mode, the polarity and detection conditions (edge or level) of the external input signal can be set for each port. The corresponding bits in the positive/negative logic select register and edge/level select register should be used to set the polarity and detection conditions, respectively.

If a port is set for edge detection using the corresponding bit in the edge/level select register, even when an external pulse interrupt signal is input, the corresponding bit in the interrupt display register holds the input using the FF and allows the level interrupt signal to be output to the interrupt control block. To stop all the interrupt signal outputs, all the bits in the interrupt clear register corresponding to the bits in the interrupt display register currently indicating the reception of the corresponding interrupt signals should be cleared to 0. Note that if a port is set for level detection using the corresponding bit in the edge/level select register and an external level interrupt signal is input, the corresponding bit in the interrupt display register does not use the FF to hold the input.

Interrupts indicated by the interrupt display register can be separately masked using the corresponding bits in the interrupt mask register. When all the bits currently indicating the reception of the interrupt signals are masked, no interrupt signals are output to the interrupt control block. Masks can be canceled by writing 1 to the corresponding bits in the interrupt mask clear register.

36.4 Registers in GPIO Blocks

Each GPIO block incorporates eleven 32-bit registers. These registers can be accessed via the HPB interface. Table 36.3 describes all the GPIO block registers.

Do not write to any addresses other than listed below. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than those listed below are undefined.

Table 36.3 Register Configuration

- GPIO-0

Address	Register Name	Abbreviation
H'FFC40000	GPIO-0 General IO/interrupt switching register	IOINTSEL0
H'FFC40004	GPIO-0 General input/output switching register	INOUTSEL0
H'FFC40008	GPIO-0 General output register	OUTDT0
H'FFC4000C	GPIO-0 General input register	INDT0
H'FFC40010	GPIO-0 Interrupt display register	INTDT0
H'FFC40014	GPIO-0 Interrupt clear register	INTCLR0
H'FFC40018	GPIO-0 Interrupt mask register	INTMSK0
H'FFC4001C	GPIO-0 Interrupt mask clear register	MSKCLR0
H'FFC40020	GPIO-0 Positive/negative logic select register	POSNEG0
H'FFC40024	GPIO-0 Edge/level select register	EDGLEVEL0
H'FFC40028	GPIO-0 Chattering prevention on/off register	FILONOFF0

- GPIO-1

Address	Register Name	Abbreviation
H'FFC41000	GPIO-1 General IO/interrupt switching register	IOINTSEL1
H'FFC41004	GPIO-1 General input/output switching register	INOUTSEL1
H'FFC41008	GPIO-1 General output register	OUTDT1
H'FFC4100C	GPIO-1 General input register	INDT1
H'FFC41010	GPIO-1 Interrupt display register	INTDT1
H'FFC41014	GPIO-1 Interrupt clear register	INTCLR1
H'FFC41018	GPIO-1 Interrupt mask register	INTMSK1
H'FFC4101C	GPIO-1 Interrupt mask clear register	MSKCLR1
H'FFC41020	GPIO-1 Positive/negative logic select register	POSNEG1
H'FFC41024	GPIO-1 Edge/level select register	EDGLEVEL1
H'FFC41028	GPIO-1 Chattering prevention on/off register	FILONOFF1

- GPIO-2

Address	Register Name	Abbreviation
H'FFC42000	GPIO-2 General IO/interrupt switching register	IOINTSEL2
H'FFC42004	GPIO-2 General input/output switching register	INOUTSEL2
H'FFC42008	GPIO-2 General output register	OUTDT2
H'FFC4200C	GPIO-2 General input register	INDT2
H'FFC42010	GPIO-2 Interrupt display register	INTDT2
H'FFC42014	GPIO-2 Interrupt clear register	INTCLR2
H'FFC42018	GPIO-2 Interrupt mask register	INTMSK2
H'FFC4201C	GPIO-2 Interrupt mask clear register	MSKCLR2
H'FFC42020	GPIO-2 Positive/negative logic select register	POSNEG2
H'FFC42024	GPIO-2 Edge/level select register	EDGLEVEL2
H'FFC42028	GPIO-2 Chattering prevention on/off register	FILONOFF2

- GPIO-3

Address	Register Name	Abbreviation
H'FFC43000	GPIO-3 General IO/interrupt switching register	IOINTSEL3
H'FFC43004	GPIO-3 General input/output switching register	INOUTSEL3
H'FFC43008	GPIO-3 General output register	OUTDT3
H'FFC4300C	GPIO-3 General input register	INDT3
H'FFC43010	GPIO-3 Interrupt display register	INTDT3
H'FFC43014	GPIO-3 Interrupt clear register	INTCLR3
H'FFC43018	GPIO-3 Interrupt mask register	INTMSK3
H'FFC4301C	GPIO-3 Interrupt mask clear register	MSKCLR3
H'FFC43020	GPIO-3 Positive/negative logic select register	POSNEG3
H'FFC43024	GPIO-3 Edge/level select register	EDGLEVEL3
H'FFC43028	GPIO-3 Chattering prevention on/off register	FILONOFF3

- GPIO-4

Address	Register Name	Abbreviation
H'FFC44000	GPIO-4 General IO/interrupt switching register	IOINTSEL4
H'FFC44004	GPIO-4 General input/output switching register	INOUTSEL4
H'FFC44008	GPIO-4 General output register	OUTDT4
H'FFC4400C	GPIO-4 General input register	INDT4
H'FFC44010	GPIO-4 Interrupt display register	INTDT4
H'FFC44014	GPIO-4 Interrupt clear register	INTCLR4
H'FFC44018	GPIO-4 Interrupt mask register	INTMSK4
H'FFC4401C	GPIO-4 Interrupt mask clear register	MSKCLR4
H'FFC44020	GPIO-4 Positive/negative logic select register	POSNEG4
H'FFC44024	GPIO-4 Edge/level select register	EDGLEVEL4
H'FFC44028	GPIO-4 Chattering prevention on/off register	FILONOFF4

- GPIO-5

Address	Register Name	Abbreviation
H'FFC45000	GPIO-5 General IO/interrupt switching register	IOINTSEL5
H'FFC45004	GPIO-5 General input/output switching register	INOUTSEL5
H'FFC45008	GPIO-5 General output register	OUTDT5
H'FFC4500C	GPIO-5 General input register	INDT5
H'FFC45010	GPIO-5 Interrupt display register	INTDT5
H'FFC45014	GPIO-5 Interrupt clear register	INTCLR5
H'FFC45018	GPIO-5 Interrupt mask register	INTMSK5
H'FFC4501C	GPIO-5 Interrupt mask clear register	MSKCLR5
H'FFC45020	GPIO-5 Positive/negative logic select register	POSNEG5
H'FFC45024	GPIO-5 Edge/level select register	EDGLEVEL5
H'FFC45028	GPIO-5 Chattering prevention on/off register	FILONOFF5

Note: Do not write to any addresses other than listed above.
Operation cannot be guaranteed if writing is attempted. Values read from addresses other than those listed above are undefined.

- GPIO-0 Register State in Each Operating Mode

Abbreviation	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
IOINTSEL0	H'0000 0000	Retained	Retained	Retained	—	Initialized
INOUTSEL0	H'0000 0000	Retained	Retained	Retained	—	Initialized
OUTDT0	H'0000 0000	Retained	Retained	Retained	—	Initialized
INDT0	State of the port pins	Retained	Retained	Retained	—	Initialized
INTDT0	H'0000 0000	Retained	Retained	Retained	—	Initialized
INTCLR0	H'0000 0000	Retained	Retained	Retained	—	Initialized
INTMSK0	H'0000 0000	Retained	Retained	Retained	—	Initialized
MSKCLR0	H'0000 0000	Retained	Retained	Retained	—	Initialized
POSNEG0	H'0000 0000	Retained	Retained	Retained	—	Initialized
EDGLEVEL0	H'0000 0000	Retained	Retained	Retained	—	Initialized
FILONOFF0	H'0000 0000	Retained	Retained	Retained	—	Initialized

- GPIO-1 Register State in Each Operating Mode

Abbreviation	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
IOINTSEL1	H'0000 0000	Retained	Retained	Retained	—	Initialized
INOUTSEL1	H'0000 0000	Retained	Retained	Retained	—	Initialized
OUTDT1	H'0000 0000	Retained	Retained	Retained	—	Initialized
INDT1	State of the port pins	Retained	Retained	Retained	—	Initialized
INTDT1	H'0000 0000	Retained	Retained	Retained	—	Initialized
INTCLR1	H'0000 0000	Retained	Retained	Retained	—	Initialized
INTMSK1	H'0000 0000	Retained	Retained	Retained	—	Initialized
MSKCLR1	H'0000 0000	Retained	Retained	Retained	—	Initialized
POSNEG1	H'0000 0000	Retained	Retained	Retained	—	Initialized
EDGLEVEL1	H'0000 0000	Retained	Retained	Retained	—	Initialized
FILONOFF1	H'0000 0000	Retained	Retained	Retained	—	Initialized

- GPIO-2 Register State in Each Operating Mode

Abbreviation	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
IOINTSEL2	H'0000 0000	Retained	Retained	Retained	—	Initialized
INOUTSEL2	H'0000 0000	Retained	Retained	Retained	—	Initialized
OUTDT2	H'0000 0000	Retained	Retained	Retained	—	Initialized
INDT2	State of the port pins	Retained	Retained	Retained	—	Initialized
INTDT2	H'0000 0000	Retained	Retained	Retained	—	Initialized
INTCLR2	H'0000 0000	Retained	Retained	Retained	—	Initialized
INTMSK2	H'0000 0000	Retained	Retained	Retained	—	Initialized
MSKCLR2	H'0000 0000	Retained	Retained	Retained	—	Initialized
POSNEG2	H'0000 0000	Retained	Retained	Retained	—	Initialized
EDGLEVEL2	H'0000 0000	Retained	Retained	Retained	—	Initialized
FILONOFF2	H'0000 0000	Retained	Retained	Retained	—	Initialized

- GPIO-3 Register State in Each Operating Mode

Abbreviation	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
IOINTSEL3	H'0000 0000	Retained	Retained	Retained	—	Initialized
INOUTSEL3	H'0000 0000	Retained	Retained	Retained	—	Initialized
OUTDT3	H'0000 0000	Retained	Retained	Retained	—	Initialized
INDT3	State of the port pins	Retained	Retained	Retained	—	Initialized
INTDT3	H'0000 0000	Retained	Retained	Retained	—	Initialized
INTCLR3	H'0000 0000	Retained	Retained	Retained	—	Initialized
INTMSK3	H'0000 0000	Retained	Retained	Retained	—	Initialized
MSKCLR3	H'0000 0000	Retained	Retained	Retained	—	Initialized
POSNEG3	H'0000 0000	Retained	Retained	Retained	—	Initialized
EDGLEVEL3	H'0000 0000	Retained	Retained	Retained	—	Initialized
FILONOFF3	H'0000 0000	Retained	Retained	Retained	—	Initialized

- GPIO-4 Register State in Each Operating Mode

Abbreviation	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
IOINTSEL4	H'0000 0000	Retained	Retained	Retained	—	Initialized
INOUTSEL4	H'0000 0000	Retained	Retained	Retained	—	Initialized
OUTDT4	H'0000 0000	Retained	Retained	Retained	—	Initialized
INDT4	State of the port pins	Retained	Retained	Retained	—	Initialized
INTDT4	H'0000 0000	Retained	Retained	Retained	—	Initialized
INTCLR4	H'0000 0000	Retained	Retained	Retained	—	Initialized
INTMSK4	H'0000 0000	Retained	Retained	Retained	—	Initialized
MSKCLR4	H'0000 0000	Retained	Retained	Retained	—	Initialized
POSNEG4	H'0000 0000	Retained	Retained	Retained	—	Initialized
EDGLEVEL4	H'0000 0000	Retained	Retained	Retained	—	Initialized
FILONOFF4	H'0000 0000	Retained	Retained	Retained	—	Initialized

- GPIO-5 Register State in Each Operating Mode

Abbreviation	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
IOINTSEL5	H'0000 0000	Retained	Retained	Retained	—	Initialized
INOUTSEL5	H'0000 0000	Retained	Retained	Retained	—	Initialized
OUTDT5	H'0000 0000	Retained	Retained	Retained	—	Initialized
INDT5	State of the port pins	Retained	Retained	Retained	—	Initialized
INTDT5	H'0000 0000	Retained	Retained	Retained	—	Initialized
INTCLR5	H'0000 0000	Retained	Retained	Retained	—	Initialized
INTMSK5	H'0000 0000	Retained	Retained	Retained	—	Initialized
MSKCLR5	H'0000 0000	Retained	Retained	Retained	—	Initialized
POSNEG5	H'0000 0000	Retained	Retained	Retained	—	Initialized
EDGLEVEL5	H'0000 0000	Retained	Retained	Retained	—	Initialized
FILONOFF5	H'0000 0000	Retained	Retained	Retained	—	Initialized

Note: Initialized: The value is the one written in "Power-on Reset".

36.4.1 General IO/Interrupt Switching Register n (IOINTSEL0 to IOINTSEL5)

IOINTSEL selects either general input/output mode or interrupt input mode for each of the 32-channel ports of the GPIO block. When general input/output mode is selected for a port, it is also necessary to select either input or output mode for the port using the corresponding bit in the general input/output switching register. When interrupt input mode is selected for a port, the setting of the general input/output switching register for the port is ignored.

[Hardware default value: H'00000000 = general input/output mode is selected for all the ports.]

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IOINTSEL[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IOINTSEL[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	IOINTSEL [31:0]	H'0000 0000	R/W	Each bit reflects the value received through the corresponding port pin. 0: Input is 0. (assuming positive logic) 1: Input is 1. (assuming positive logic)

Note: Unused bits should be set to the initial values.

36.4.2 General Input/Output Switching Register n (INOUTSEL0 to INOUTSEL5)

INOUTSEL is valid only for the ports for which general input/output mode is selected by the general IO/interrupt switching register. Specifically, INOUTSEL selects either general input or general output mode for a port using the bit corresponding to the port number. The INOUTSEL bits can be written to only when the corresponding bits in the general IO/interrupt switching register are 0. Note that after general input/output mode is changed to interrupt input mode, INOUTSEL retains the setting but is read as 0.

[Hardware default value: H'00000000 = general input mode is selected for all the ports.]

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	INOUTSEL[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	INOUTSEL[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	INOUTSEL [31:0]	H'0000 0000	R/W	Selects either general input mode or general output mode for each port using the bits corresponding to the port numbers. 0: General input mode 1: General output mode

Note: Unused bits should be set to the initial values.

36.4.3 General Output Register n (OUTDT0 to OUTDT5)

OUTDT is valid only for the ports for which general input/output mode is selected by the general IO/interrupt switching register and then general output mode is selected by the general input/output switching register. Specifically, the value of the bit in OUTDT corresponding to the port number is inverted or not inverted depending on the setting of the positive/negative logic select register before being output from the corresponding port pin. Note that the polarity of the output signal should previously be set using the corresponding bit in the positive/negative logic select register.

[Hardware default value: H'00000000 = 0 is output from all the ports.]

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OUTDT[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OUTDT[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	OUTDT [31:0]	H'0000 0000	R/W	Allows the port to output the value set in the bit corresponding to the port number when the port is appropriately set by IOINTSEL0 to IOINTSEL3 and INOUTSEL0 to INOUTSEL3. 0: 0 is output. 1: 1 is output.

Note: The values set in OUTDT are not directly output from the GPIO pins; the above set values are processed according to the settings of the positive/negative logic select register before being output.
Unused bits should be set to the initial values.

36.4.4 General Input Register n (INDT0 to INDT5)

INDT is valid only for the ports for which general input/output mode is selected by the general IO/interrupt switching register and then general input mode is selected by the general input/output switching register. Each bit reflects the value received through the corresponding port pin.

Note that when a bit in the positive/negative logic select register is 1, the corresponding bit in INDT indicates the inverted value of the input signal.

[Hardware default value: state of the signals input to the port pins.]

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	INDT[31:16]															
Initial value:	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	INDT[15:0]															
Initial value:	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	INDT[31:0]	*	R	Each bit reflects the value received through the corresponding port pin. 0: Input is 0. (assuming positive logic) 1: Input is 1. (assuming positive logic)

Notes: Unused bits should be set to the initial values.
* State of the signals input to the port pins.

36.4.5 Interrupt Display Register n (INTDT0 to INTDT5)

INTDT is valid only when interrupt input mode is selected by the general IO/interrupt switching register. Specifically, when an interrupt is input via a port pin when INTDT is valid, the bit in INTDT corresponding to the port indicates whether the port has received an interrupt input or not. In interrupt input mode, the polarity and detection conditions (edge or level) of the external input signal can be set for each port pin. Before using a port pin for interrupt input, the corresponding bits in the positive/negative logic select register and edge/level select register should be set, respectively.

If a port is set for edge detection using the corresponding bit in the edge/level select register, even when an external pulse interrupt signal is input, the corresponding bit in INTDT holds the input using the FF and allows the level interrupt signal to be output to the interrupt control block.
To stop all the interrupt signal outputs, all the bits in the interrupt clear register corresponding to the bits in INTDT currently indicating the reception of the corresponding interrupt signals should be cleared to 0. Note that if a port is set for level detection using the corresponding bit in the edge/level select register and an external level interrupt signal is input, the corresponding bit in INTDT does not use the FF to hold the input. Therefore, when an external input signal is stopped, the corresponding bit in INTDT is cleared automatically. When all the bits in INTDT are turned off (= 0), the GPIO stops outputting all the interrupt signals.

[Hardware default value: H'00000000 = no interrupt signals are input from ports.]

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	INTDT[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	INTDT[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	INTDT [31:0]	H'0000 0000	R	Each bit indicates the input of an interrupt signal on the corresponding port pin. 0: No interrupt signal has been input. 1: Interrupt signal has been input.

Note: Unused bits should be set to the initial values.

Conditions of Indicating Interrupt Input:

1. For level-sensitive interrupt input (EDGLEVEL = 0)
   — External input signals are constantly monitored and indicated. (When the negative logic is selected, the inverted value of the external input signal is indicated.)
2. For edge-sensitive interrupt input (EDGLEVEL = 1)
   — Clearing condition: When the interrupt clear register is cleared, indication is cleared regardless of the positive/negative logic select register.
   — Setting condition: With the positive logic (POSNEG = 0), when the rising edge of an external interrupt signal is detected, the interrupt input is indicated. With the negative logic (POSNEG = 1), when the falling edge is detected, the interrupt input is indicated.

36.4.6 Interrupt Clear Register n (INTCLR0 to INTCLR5)

When the interrupt display register is currently indicates the reception of the interrupt input on the port for which the edge detection is selected by the edge/level select register in interrupt input mode, INTCLR clears the indication. Specifically, writing 1 to the bits in INTCLR corresponding to port numbers can clear the corresponding bits in the interrupt display register. However, when the interrupt display register is currently indicates the reception of the interrupt input on the port for which the level detection is selected by the edge/level select register, writing 1 to the corresponding bits in INTCLR cannot clear the corresponding bits in the interrupt display register. Only writing 1 to INTCLR is effective; INTCLR is always read as 0.

[Hardware default value: H'00000000 = interrupt indication is cleared for no ports.]

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	INTCLR[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	INTCLR[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	INTCLR [31:0]	H'0000 0000	R/W	Writing 1 to bits corresponding to port numbers clears the corresponding bits in the interrupt display register. 0: No effect 1: Interrupt display register bit is cleared.

Note: Unused bits should be set to the initial values.

36.4.7 Interrupt Mask Register n (INTMSK0 to INTMSK5)

INTMSK masks the interrupt requests indicated by the interrupt display register. Interrupts can be separately masked using the corresponding bits in INTMSK. When all the bits currently indicating the reception of the interrupt signals are masked, no interrupt signals are output to the interrupt control block. Masks can be canceled by writing 1 to the corresponding bits in the interrupt mask clear register. Only writing 0 to this register is effective.

[Hardware default value: H'00000000 = all the ports are masked.]

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	INTMSK[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	INTMSK[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	INTMSK [31:0]	H'0000 0000	R/W	Setting a mask to the bit disables the corresponding interrupt signal to be output to the interrupt control block. 0: Interrupt is masked. 1: Interrupt is not masked.

Note: Unused bits should be set to the initial values.

36.4.8 Interrupt Mask Clear Register n (MSKCLR0 to MSKCLR5)

MSKCLR cancels masks that are set by the interrupt mask register. Each mask can be canceled (cleared) by writing 1 to the corresponding bit in MSKCLR. Only writing 1 to MSKCLR is effective; MSKCLR is always read as 0.

[Hardware default value: H'00000000 = interrupt masks are cleared for no ports.]

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MSKCLR[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MSKCLR[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	MSKCLR [31:0]	H'0000 0000	R/W	Setting a mask to the bit disables the corresponding interrupt signal to be output to the interrupt control block. 0: No effect 1: Interrupt is not masked.

Note: Unused bits should be set to the initial values. (When GPIO is not selected by the pin multiplex settings, do not cancel the interrupt mask.)

36.4.9 Positive/Negative Logic Select Register n (POSNEG0 to POSNEG5)

POSNEG selects the polarity (positive or negative logic) of each port pin in general input mode, general output mode, or interrupt input mode. POSNEG should be set before mode selection.

[Hardware default value: H'00000000 = positive logic is selected for all the ports.]

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	POSNEG[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	POSNEG[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	POSNEG [31:0]	H'0000 0000	R/W	Selects the polarity (positive or negative logic) of each port pin. 0: Positive logic 1: Negative logic

Note: Unused bits should be set to the initial values.

36.4.10 Edge/level Select Register n (EDGLEVEL0 to EDGLEVEL5)

EDGLEVEL is valid only for the ports for which interrupt input mode is selected by the general IO/interrupt switching register. Specifically, EDGLEVEL selects the detection conditions (edge or level) of the interrupt input signal on each port pin for which interrupt input mode is selected. EDGLEVEL should be set before selection of interrupt input mode.

[Hardware default value: H'00000000 = level detection is selected for all the ports.]

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	EDGLEVEL[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EDGLEVEL[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	EDGLEVEL [31:0]	H'0000 0000	R/W	Selects the level or edge as detection conditions of the interrupt input signal on each port pin for which interrupt input mode is selected. 0: Level 1: Edge

Note: Unused bits should be set to the initial values.

36.4.11 Chattering Prevention On/Off Register n (FILONOFF0 to FILONOFF5)

FILONOFF prevents chattering input to the port pins of channels 0 to 3. For details, refer to section 36.5, Handling of Input Signals on Port Pins.

[Hardware default value: H'0 = chattering prevention function is turned off for all the ports.]

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	FILONOFF[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	—	—	Reserved
3 to 0	FILONOFF [3:0]	0000	R/W	Enables or disables the chattering prevention function. 0: Chattering prevention function is disabled. 1: Chattering prevention function is enabled.

Note: Unused bits should be set to the initial values.

36.5 Handling of Input Signals on Port Pins

36.5.1 Chattering

In general input mode and interrupt input modes, a filtering function can be used for the channels 0 to 3 port pins to prevent external chattering input. Specifically, when a bit in the chattering prevention on/off register is set to use the function, the external input to the corresponding port pin is sampled four consecutive times based on the 600-µs clock signal, which is internally generated by the GPIO. The external input is canceled except when the active input is detected four consecutive times. Therefore, when a filtering function is used, input to the channels 0 to 3 port pins need to be at least four 500-µs sampling clock cycles long (when the peripheral clock (clkp) frequency is 50.0 MHz, the sampling clock cycle is 500 µs).

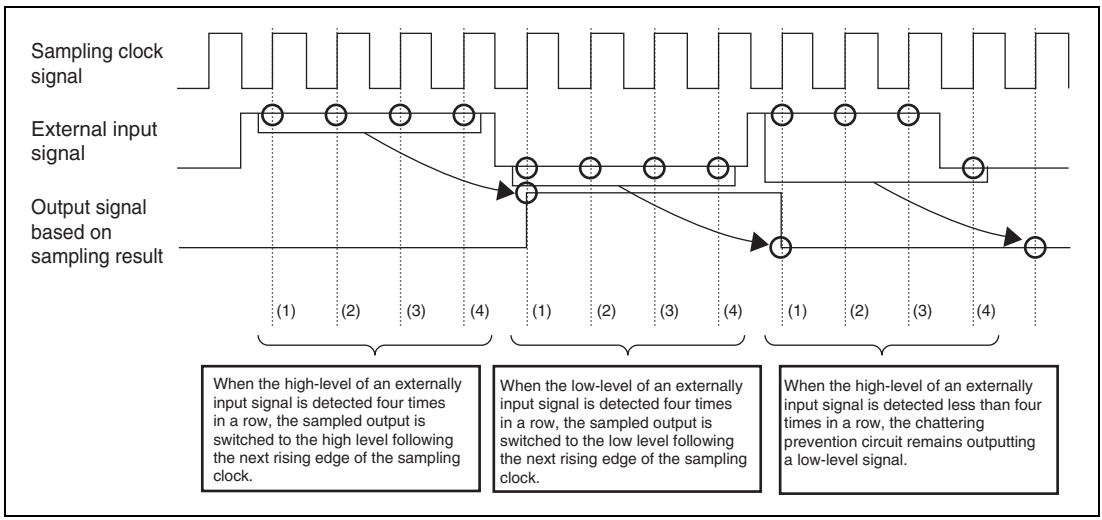


Figure 36.2 Sampling Timing Chart

36.5.2 Clock Synchronization

In general input mode and interrupt input mode, external input signals on the port pins of all the channels are synchronized with the clock signal.

36.6 Interrupt Display Timing Charts

Figure 36.3 shows the interrupt display timing and figure 36.4 shows the note on the timing. In both figures, the positive logic and edge-sensitive input are assumed.

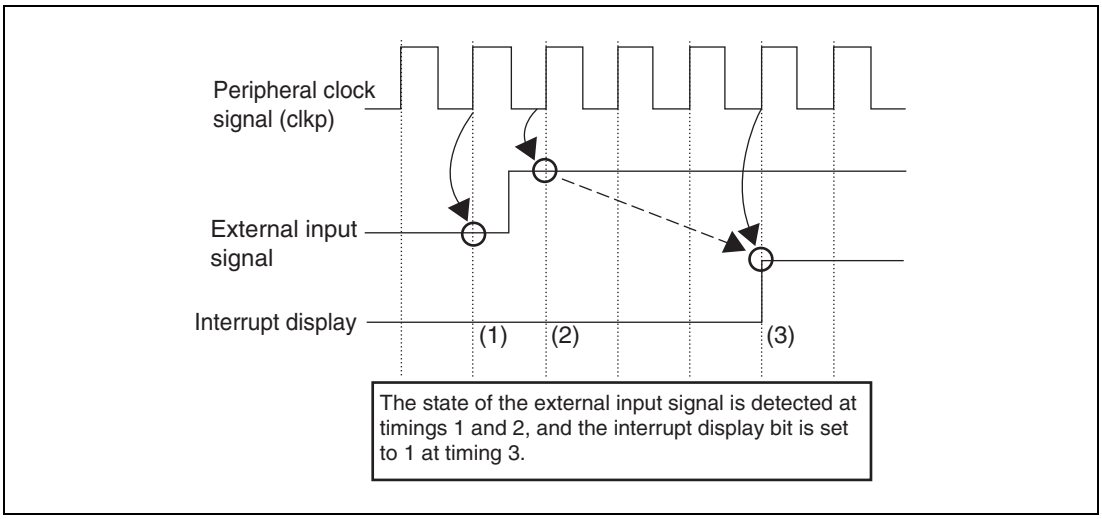


Figure 36.3 Interrupt Display Timing

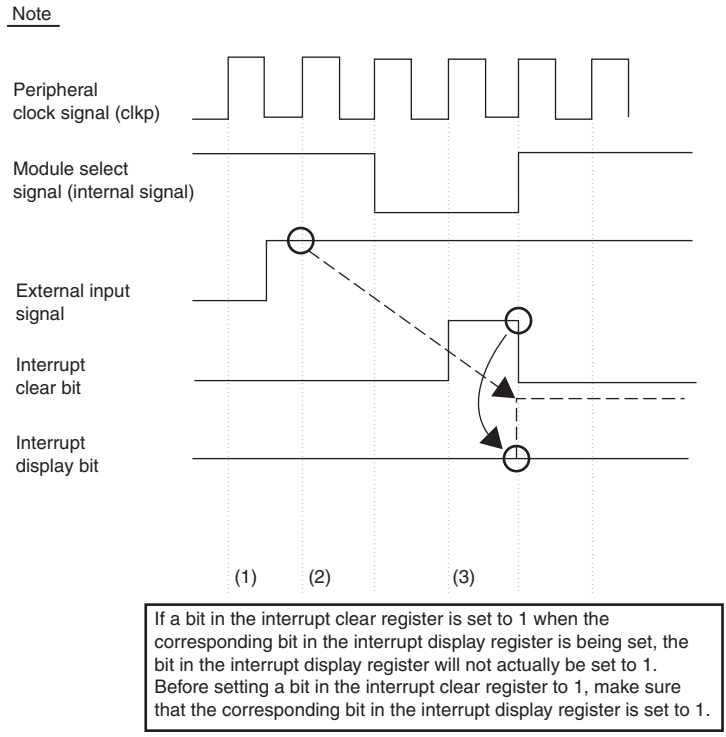


Figure 36.4 Note on Interrupt Display Timing

36.7 Using GPIO

The following sections describe how to use the GPIO. If the GPIO is not used according to the procedures shown here, operations are not guaranteed.

36.7.1 Setting Edge-Sensitive Interrupt Input Mode

For setting edge-sensitive interrupt input mode, refer to the procedure shown in figure 36.5.

Note that an unexpected interrupt might be generated in the module if setting 1, 2, or 3 in the flowchart is changed. When changing the setting, 4 and 5 should be done.

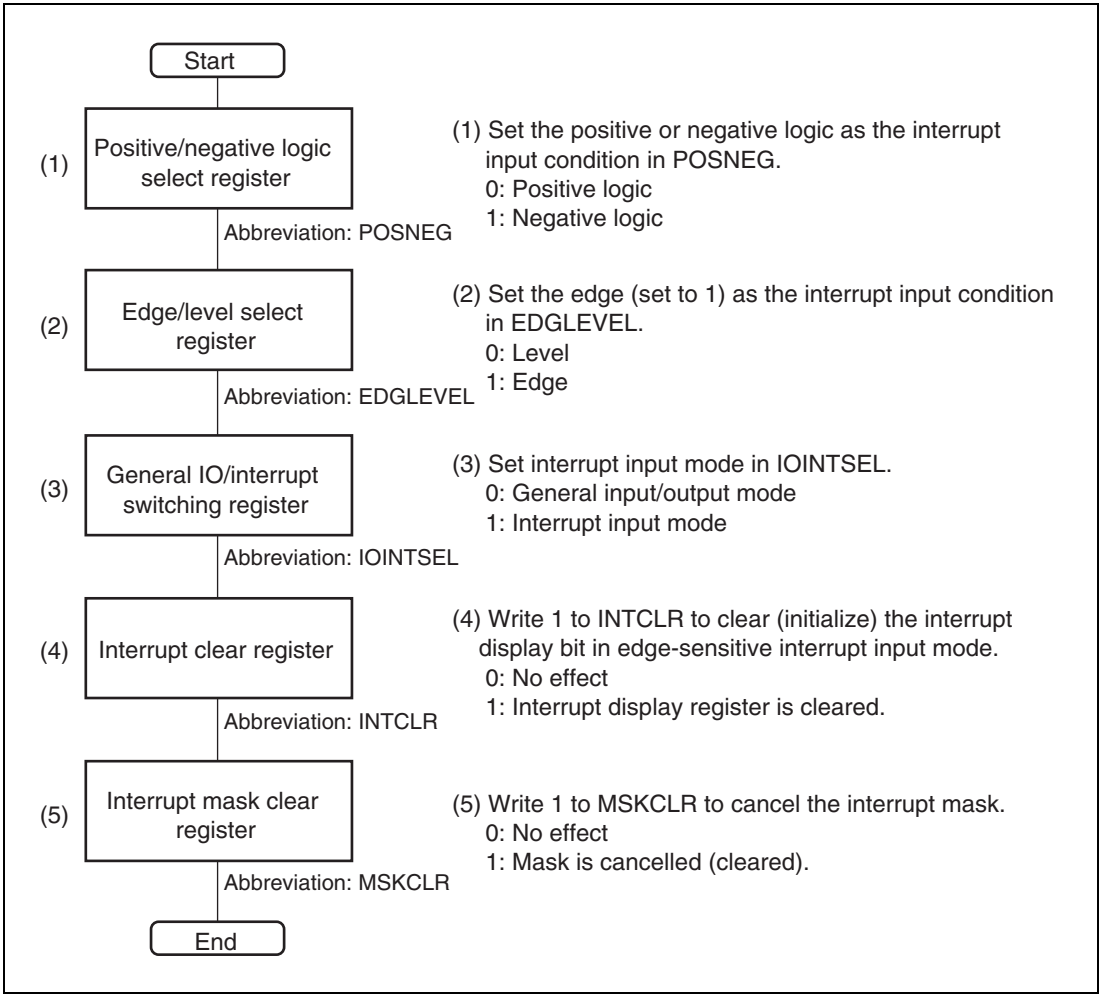


Figure 36.5 Flowchart of Setting the GPIO to Edge-Sensitive Interrupt Input Mode

36.7.2 Setting Level-Sensitive Interrupt Input Mode

For setting level-sensitive interrupt input mode, refer to the procedure shown in figure 36.6.

Note that when an external level-sensitive interrupt input signal is stopped, the corresponding interrupt is canceled automatically. In level-sensitive interrupt input mode, the interrupt clear register is invalid.

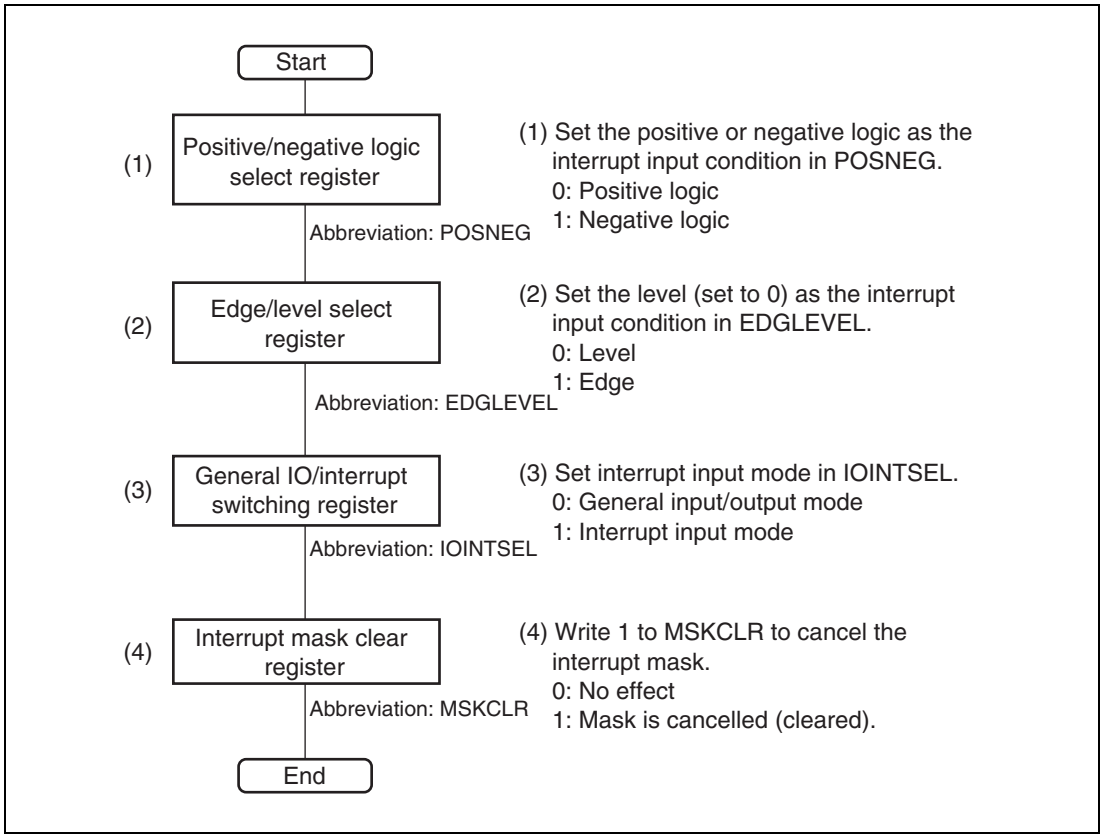


Figure 36.6 Flowchart of Setting the GPIO to Level-Sensitive Interrupt Input Mode

36.7.3 Setting General Output Mode

For setting general output mode, refer to the procedure shown in figure 36.7.

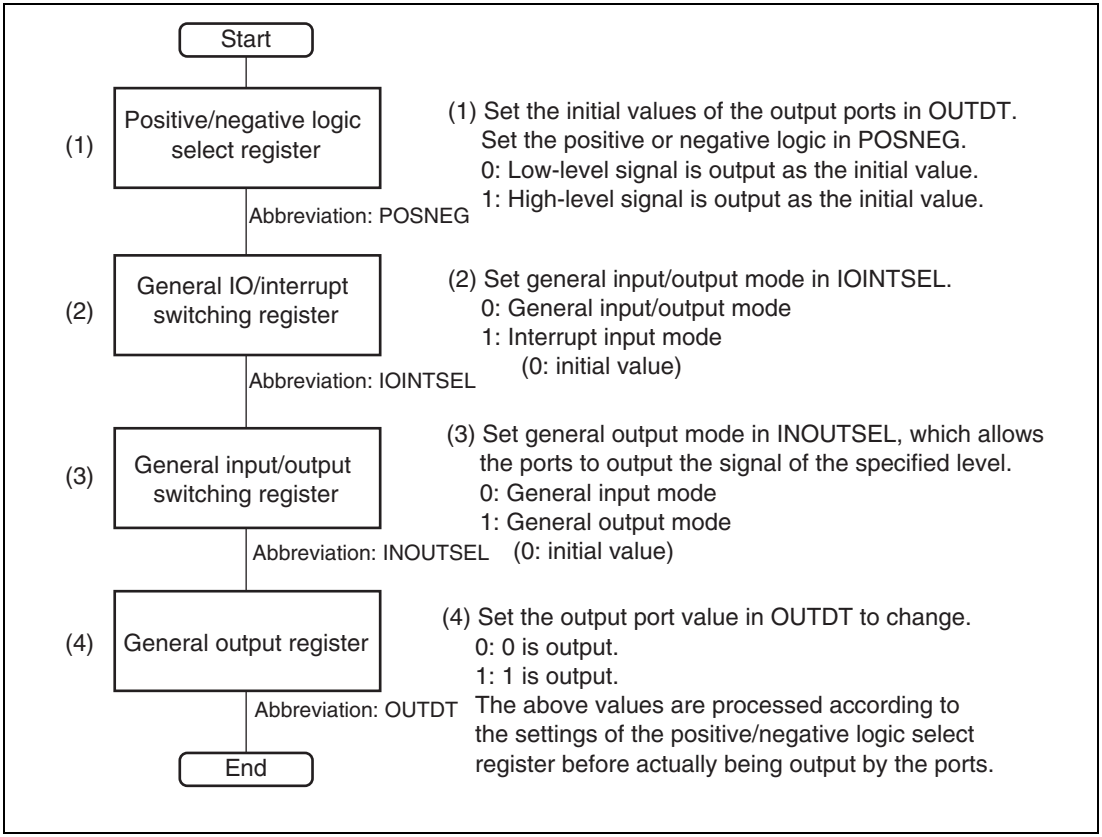


Figure 36.7 Flowchart of Setting the GPIO to General Output Mode

36.7.4 Setting General Input Mode

For setting general input mode, refer to the procedure shown in figure 36.8.

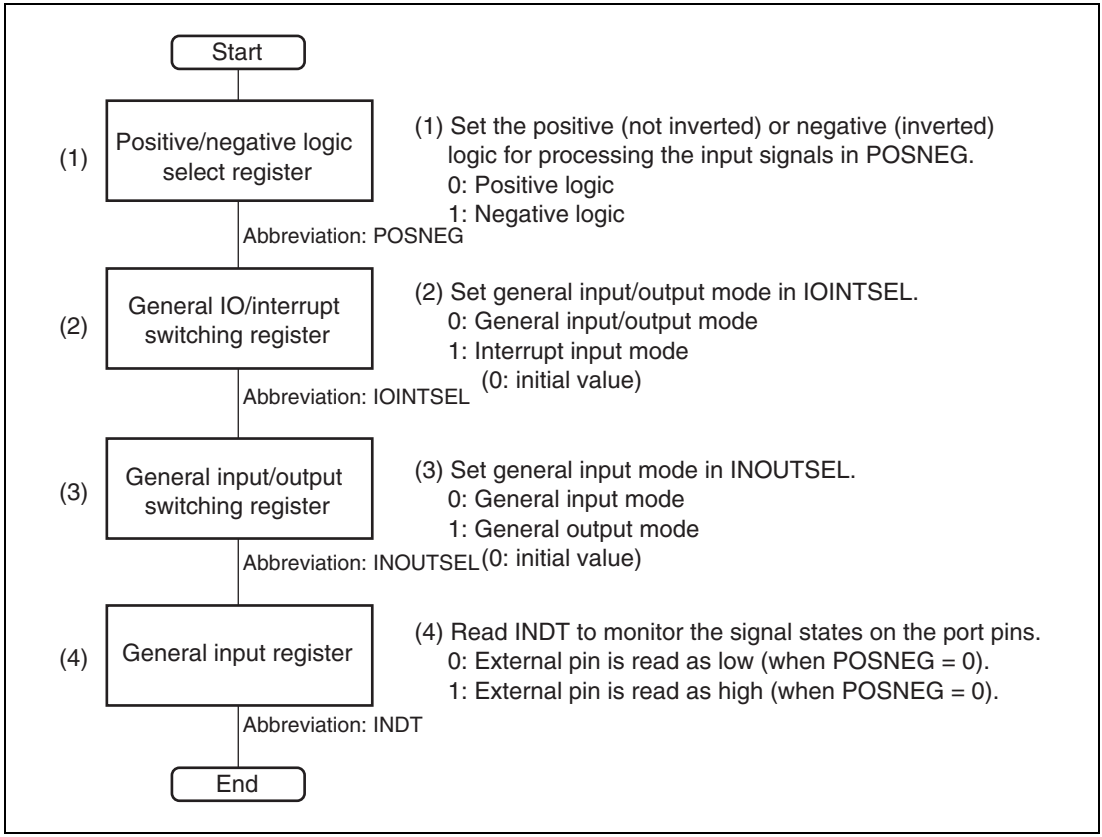


Figure 36.8 Flowchart of Setting the GPIO to General Input Mode

Section 37 Pin Function Controller (PFC)

37.1 Overview

The pin function controller (PFC) is a module that consists of registers for selecting the function of the multiplexed pins and controlling the pull-up resistor on each LSI pin.

37.1.1 Features

- Register access through the HPB bus interface
- Setting multiplexed pin functions for LSI pins
  - Function of this LSI pin selectable by setting the registers in the PFC module

    (The function of the LSI pin can be selected by the GPIO/peripheral function select registers 0 to 5 (GPSR0 to GPSR5) and the peripheral function select registers 0 to 11 (IPSR0 to IPSR11) in the PFC module. For details, see sections 37.2.2, GPIO/Peripheral Function Select Register 0 (GPSR0) through 37.2.19, Peripheral Function Select Register 11 (IPSR11).)
- Module select function

  Selection is handled by the module select register (MOD_SEL) and the module select register 2 (MOD_SEL2). For details, see section 37.2.20, Module Select Register (MOD_SEL) and section 37.2.21, Module Select Register 2 (MOD_SEL2).
- Pull-up control for each LSI pin
  - On/off of the pull-up resistor on each pin of this LSI can be controlled by setting the registers in the PFC module.

    (The pull-up resistor on each LSI pin can be turned on or off individually by setting the LSI pin pull-up control registers 0 to 5 (PUPCTL0 to PUPCTL5) in the PFC module. For details, see sections 37.2.22, LSI Pin Pull-Up Control Register 0 (PUPCTL0) through 37.2.27, LSI Pin Pull-Up Control Register 5 (PUPCTL5).)

37.1.2 Register Configuration

All the registers in the PFC are mapped into the HPB bus space. Table 37.1 shows the configuration of the registers provided in the PFC and table 37.2 shows the register state in each operating mode. Details on each register in the PFC are given in section 37.2, Register Descriptions.

Table 37.1 Configuration of Registers in PFC

Address	Register Name	Abbr.	Power-On Reset	Manual Reset	Access Size	Store Block
H'FFFC0000	LSI Multiplexed Pin Setting Mask Register	PMMR	H'0000 0000	Retained	32	PFC
H'FFFC 0004	GPIO/peripheral function select register 0	GPSR0	H'F9FF FFFF	Retained	32	PFC
H'FFFC 0008	GPIO/peripheral function select register 1	GPSR1	H'73A0 7FF1	Retained	32	PFC
H'FFFC 000C	GPIO/peripheral function select register 2	GPSR2	H'0000 0000	Retained	32	PFC
H'FFFC 0010	GPIO/peripheral function select register 3	GPSR3	H'0000 0000 (in normal operation) H'007F FFFF (in HIF boot mode)	Retained	32	PFC
H'FFFC 0014	GPIO/peripheral function select register 4	GPSR4	H'1400 0000 (in normal operation) H'1600 0000 (in HIF boot mode)	Retained	32	PFC
H'FFFC 0018	GPIO/peripheral function select register 5	GPSR5	H'0000 0000	Retained	32	PFC
H'FFFC 001C	Peripheral function select register 0	IPSR0	H'0000 0000	Retained	32	PFC
H'FFFC 0020	Peripheral function select register 1	IPSR1	H'0000 0000	Retained	32	PFC

Address	Register Name	Abbr.	Power-On Reset	Manual Reset	Access Size	Store Block
H'FFFC 0024	Peripheral function select register 2	IPSR2	H'0000 0000	Retained	32	PFC
H'FFFC 0028	Peripheral function select register 3	IPSR3	H'0000 0000	Retained	32	PFC
H'FFFC 002C	Peripheral function select register 4	IPSR4	H'0000 0000	Retained	32	PFC
H'FFFC 0030	Peripheral function select register 5	IPSR5	H'0000 0000	Retained	32	PFC
H'FFFC 0034	Peripheral function select register 6	IPSR6	H'0000 0000 (in normal operation) H'0097 FFED (in HIF boot mode)	Retained	32	PFC
H'FFFC 0038	Peripheral function select register 7	IPSR7	H'0000 0000 (in normal operation) H'5C92 4924 (in HIF boot mode)	Retained	32	PFC
H'FFFC 003C	Peripheral function select register 8	IPSR8	H'0000 0000 (in normal operation) H'0000 000A (in HIF boot mode)	Retained	32	PFC
H'FFFC 0040	Peripheral function select register 9	IPSR9	H'0000 0000	Retained	32	PFC
H'FFFC 0044	Peripheral function select register 10	IPSR10	H'0000 0000	Retained	32	PFC
H'FFFC 0048	Peripheral function select register 11	IPSR11	H'0000 0000 (in normal operation) H'0000 0004 (in HIF boot mode)	Retained	32	PFC
H'FFFC 004C	Module select register	MOD_SEL	H'0000 0000	Retained	32	PFC
H'FFFC 0050	Module select register 2	MOD_SEL2	H'0000 0000	Retained	32	PFC
H'FFFC0100	LSI pin pull-up control register 0	PUPCTL0	H'0000 0000	Retained	32	PFC
H'FFFC0104	LSI pin pull-up control register 1	PUPCTL1	H'DE29 C000	Retained	32	PFC
H'FFFC0108	LSI pin pull-up control register 2	PUPCTL2	H'7FFF FDEF	Retained	32	PFC

Address	Register Name	Abbr.	Power-On Reset	Manual Reset	Access Size	Store Block
H'FFFC010C	LSI pin pull-up control register 3	PUPCTL3	H'7F00 0000	Retained	32	PFC
H'FFFC0110	LSI pin pull-up control register 4	PUPCTL4	H'0ACF FFFE	Retained	32	PFC
H'FFFC0114	LSI pin pull-up control register 5	PUPCTL5	H'0000 01F1	Retained	32	PFC
H'FFFC 0118	LSI pin drive function switching register	DRV_SEL	H'0000 0000	Retained	32	PFC

Note: Do not write to any addresses other than listed above. Operations cannot be guaranteed if writing is attempted. Values read from addresses other than those listed above are undefined.

Table 37.2 Register State in Each Operating Mode

Abbr.	Power-On Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
PMMR	Initialized	Retained	Retained	Retained	Retained	Initialized
GPSR0	Initialized	Retained	Retained	Retained	Retained	Initialized
GPSR1	Initialized	Retained	Retained	Retained	Retained	Initialized
GPSR2	Initialized	Retained	Retained	Retained	Retained	Initialized
GPSR3	Initialized	Retained	Retained	Retained	Retained	Initialized
GPSR4	Initialized	Retained	Retained	Retained	Retained	Initialized
GPSR5	Initialized	Retained	Retained	Retained	Retained	Initialized
IPSR0	Initialized	Retained	Retained	Retained	Retained	Initialized
IPSR1	Initialized	Retained	Retained	Retained	Retained	Initialized
IPSR2	Initialized	Retained	Retained	Retained	Retained	Initialized
IPSR3	Initialized	Retained	Retained	Retained	Retained	Initialized
IPSR4	Initialized	Retained	Retained	Retained	Retained	Initialized
IPSR5	Initialized	Retained	Retained	Retained	Retained	Initialized
IPSR6	Initialized	Retained	Retained	Retained	Retained	Initialized
IPSR7	Initialized	Retained	Retained	Retained	Retained	Initialized
IPSR8	Initialized	Retained	Retained	Retained	Retained	Initialized
IPSR9	Initialized	Retained	Retained	Retained	Retained	Initialized
IPSR10	Initialized	Retained	Retained	Retained	Retained	Initialized
IPSR11	Initialized	Retained	Retained	Retained	Retained	Initialized

Abbr.	Power-On Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
MOD_SEL	Initialized	Retained	Retained	Retained	Retained	Initialized
MOD_SEL2	Initialized	Retained	Retained	Retained	Retained	Initialized
PUPCTL0	Initialized	Retained	Retained	Retained	Retained	Initialized
PUPCTL1	Initialized	Retained	Retained	Retained	Retained	Initialized
PUPCTL2	Initialized	Retained	Retained	Retained	Retained	Initialized
PUPCTL3	Initialized	Retained	Retained	Retained	Retained	Initialized
PUPCTL4	Initialized	Retained	Retained	Retained	Retained	Initialized
PUPCTL5	Initialized	Retained	Retained	Retained	Retained	Initialized
DRV_SEL	Initialized	Retained	Retained	Retained	Retained	Initialized

37.2 Register Descriptions

Legend for symbols used in register descriptions:

Initial value:	The value in the register after reset
—:	Undefined value
R/W:	The bit is readable and writable. The written value can be read.
R:	The bit is readable. The write value should always be 0.
R/WC0:	The bit is readable and writable. Writing 0 to the bit initializes the bit. Writing 1 to the bit is ignored.
R/WC1:	The bit is readable and writable. Writing 1 to the bit initializes the bit. Writing 0 to the bit is ignored.
W:	The bit is writable. Reading the bit is prohibited. When the bit is reserved, the write value should always be 0.
—/W:	The bit is writable. The read value is undefined.

All the bits are active high unless otherwise specified, and deactivated on reset.

Data access is always performed in longword units.

The write value to a reserved bit should always be 0.

37.2.1 LSI Multiplexed Pin Setting Mask Register (PMMR)

Function: PMMR enables/disables writing to the multiplexed pin setting registers.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MPM [31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MPM [15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	MPM	H'0000 0000	R/W	Multiplexed Pin Setting Mask Writing a value to any register from among the GPIO/peripheral function select registers, the peripheral function select registers, and the module select registers is enabled by writing the inverse of the value to this register.

Note: This register must be set before setting each of the GPIO/peripheral function select registers 0 to 5, the peripheral function select registers 0 to 11, the module select register, and the module select register 2.

37.2.2 GPIO/Peripheral Function Select Register 0 (GPSR0)

Function: GPSR0 selects the functions of multiplexed LSI pins.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GP0 [31]	GP0 [30]	GP0 [29]	GP0 [28]	GP0 [27]	GP0 [26]	GP0 [25]	GP0 [24]	GP0 [23]	GP0 [22]	GP0 [21]	GP0 [20]	GP0 [19]	GP0 [18]	GP0 [17]	GP0 [16]
Initial value:	1	1	1	1	1	0	0	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GP0 [15]	GP0 [14]	GP0 [13]	GP0 [12]	GP0 [11]	GP0 [10]	GP0 [9]	GP0 [8]	GP0 [7]	GP0 [6]	GP0 [5]	GP0 [4]	GP0 [3]	GP0 [2]	GP0 [1]	GP0 [0]
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GP0[31:0]	H'F9FF FFFF	R/W	The functions of the LSI pins are selected according to the table below.

Note: To enable this register to be set, appropriately set the LSI multiplexed pin setting mask register (PMMR) immediately before setting this register.

Bit Name	GPIO (Set Value = H'0)	Peripheral Function (Set Value = H'1)
GP0[0]	GP-0-0	Peripheral function selected by IP1[9:8]
GP0[1]	GP-0-1	Peripheral function selected by IP1[11:10]
GP0[2]	GP-0-2	Peripheral function selected by IP1[13:12]
GP0[3]	GP-0-3	Peripheral function selected by IP1[15:14]
GP0[4]	GP-0-4	Peripheral function selected by IP0[7:6]
GP0[5]	GP-0-5	Peripheral function selected by IP0[9:8]
GP0[6]	GP-0-6	Peripheral function selected by IP0[11:10]
GP0[7]	GP-0-7	Peripheral function selected by IP0[13:12]
GP0[8]	GP-0-8	Peripheral function selected by IP0[15:14]
GP0[9]	GP-0-9	Peripheral function selected by IP0[17:16]
GP0[10]	GP-0-10	Peripheral function selected by IP0[19:18]
GP0[11]	GP-0-11	Peripheral function selected by IP0[21:20]
GP0[12]	GP-0-12	Peripheral function selected by IP0[23:22]
GP0[13]	GP-0-13	Peripheral function selected by IP0[25:24]
GP0[14]	GP-0-14	Peripheral function selected by IP0[27:26]
GP0[15]	GP-0-15	Peripheral function selected by IP0[29:28]
GP0[16]	GP-0-16	Peripheral function selected by IP0[31:30]
GP0[17]	GP-0-17	Peripheral function selected by IP1[1:0]
GP0[18]	GP-0-18	Peripheral function selected by IP1[3:2]
GP0[19]	GP-0-19	Peripheral function selected by IP1[5:4]
GP0[20]	GP-0-20	Peripheral function selected by IP1[7:6]
GP0[21]	GP-0-21	Peripheral function selected by IP11[28]
GP0[22]	GP-0-22	Peripheral function selected by IP0[1:0]
GP0[23]	GP-0-23	Peripheral function selected by IP0[3:2]
GP0[24]	GP-0-24	Peripheral function selected by IP0[5:4]
GP0[25]	GP-0-25	Peripheral function selected by IP1[17:16]
GP0[26]	GP-0-26	Peripheral function selected by IP1[19:18]
GP0[27]	GP-0-27	Peripheral function selected by IP1[22:20]
GP0[28]	GP-0-28	Peripheral function selected by IP1[25:23]
GP0[29]	GP-0-29	Peripheral function selected by IP1[28:26]
GP0[30]	GP-0-30	Peripheral function selected by IP1[31:29]
GP0[31]	GP-0-31	Peripheral function selected by IP2[2:0]

37.2.3 GPIO/Peripheral Function Select Register 1 (GPSR1)

Function: GPSR1 selects the functions of multiplexed LSI pins.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GP1 [31]	GP1 [30]	GP1 [29]	GP1 [28]	GP1 [27]	GP1 [26]	GP1 [25]	GP1 [24]	GP1 [23]	GP1 [22]	GP1 [21]	GP1 [20]	GP1 [19]	GP1 [18]	GP1 [17]	GP1 [16]
Initial value:	0	1	1	1	0	0	1	1	1	0	1	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GP1 [15]	GP1 [14]	GP1 [13]	GP1 [12]	GP1 [11]	GP1 [10]	GP1 [9]	GP1 [8]	GP1 [7]	GP1 [6]	GP1 [5]	GP1 [4]	GP1 [3]	GP1 [2]	GP1 [1]	GP1 [0]
Initial value:	0	1	1	1	1	1	1	1	1	1	1	1	0	0	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GP1[31:0]	H'73A0 7FF1	R/W	The functions of the LSI pins are selected according to the table below.

Note: To enable this register to be set, appropriately set the LSI multiplexed pin setting mask register (PMMR) immediately before setting this register.

Bit Name	GPIO (Set Value = H'0)	Peripheral Function (Set Value = H'1)
GP1[0]	GP-1-0	Peripheral function selected by IP3[20]
GP1[1]	GP-1-1	Peripheral function selected by IP3[29:27]
GP1[2]	GP-1-2	Peripheral function selected by IP11[20:19]
GP1[3]	GP-1-3	Peripheral function selected by IP11[22:21]
GP1[4]	GP-1-4	Peripheral function selected by IP2[16:14]
GP1[5]	GP-1-5	Peripheral function selected by IP2[19:17]
GP1[6]	GP-1-6	Peripheral function selected by IP2[22:20]
GP1[7]	GP-1-7	Peripheral function selected by IP2[24:23]
GP1[8]	GP-1-8	Peripheral function selected by IP2[27:25]
GP1[9]	GP-1-9	Peripheral function selected by IP2[30:28]
GP1[10]	GP-1-10	Peripheral function selected by IP3[1:0]
GP1[11]	GP-1-11	CLKOUT
GP1[12]	GP-1-12	$\overline{BS}$
GP1[13]	GP-1-13	$\overline{CS0}$

Bit Name	GPIO (Set Value = H'0)	Peripheral Function (Set Value = H'1)
GP1[14]	GP-1-14	Peripheral function selected by IP3[2]
GP1[15]	GP-1-15	$\overline{\text{EX\_CS0}}$
GP1[16]	GP-1-16	Peripheral function selected by IP3[5:3]
GP1[17]	GP-1-17	Peripheral function selected by IP3[8:6]
GP1[18]	GP-1-18	Peripheral function selected by IP3[11:9]
GP1[19]	GP-1-19	Peripheral function selected by IP3[14:12]
GP1[20]	GP-1-20	Peripheral function selected by IP3[17:15]
GP1[21]	GP-1-21	$\overline{\text{RD}}$
GP1[22]	GP-1-22	Peripheral function selected by IP3[19:18]
GP1[23]	GP-1-23	$\overline{\text{WE0}}$
GP1[24]	GP-1-24	$\overline{\text{WE1}}$
GP1[25]	GP-1-25	Peripheral function selected by IP2[4:3]
GP1[26]	GP-1-26	Peripheral function selected by IP3[23:21]
GP1[27]	GP-1-27	Peripheral function selected by IP3[26:24]
GP1[28]	GP-1-28	Peripheral function selected by IP2[7:5]
GP1[29]	GP-1-29	Peripheral function selected by IP2[10:8]
GP1[30]	GP-1-30	Peripheral function selected by IP2[13:11]
GP1[31]	GP-1-31	Peripheral function selected by IP11[25:23]

37.2.4 GPIO/Peripheral Function Select Register 2 (GPSR2)

Function: GPSR2 selects the functions of multiplexed LSI pins.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GP2 [31]	GP2 [30]	GP2 [29]	GP2 [28]	GP2 [27]	GP2 [26]	GP2 [25]	GP2 [24]	GP2 [23]	GP2 [22]	GP2 [21]	GP2 [20]	GP2 [19]	GP2 [18]	GP2 [17]	GP2 [16]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GP2 [15]	GP2 [14]	GP2 [13]	GP2 [12]	GP2 [11]	GP2 [10]	GP2 [9]	GP2 [8]	GP2 [7]	GP2 [6]	GP2 [5]	GP2 [4]	GP2 [3]	GP2 [2]	GP2 [1]	GP2 [0]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GP2[31:0]	H'0000 0000	R/W	The functions of the LSI pins are selected according to the table below.

Note: To enable this register to be set, appropriately set the LSI multiplexed pin setting mask register (PMMR) immediately before setting this register.

Bit Name	GPIO (Set Value = H'0)	Peripheral Function (Set Value = H'1)
GP2[0]	GP-2-0	Peripheral function selected by IP11[6:4]
GP2[1]	GP-2-1	Peripheral function selected by IP11[9:7]
GP2[2]	GP-2-2	Peripheral function selected by IP11[11:10]
GP2[3]	GP-2-3	Peripheral function selected by IP4[2:0]
GP2[4]	GP-2-4	Peripheral function selected by IP8[29:28]
GP2[5]	GP-2-5	Peripheral function selected by IP11[27:26]
GP2[6]	GP-2-6	Peripheral function selected by IP8[22:20]
GP2[7]	GP-2-7	Peripheral function selected by IP8[25:23]
GP2[8]	GP-2-8	Peripheral function selected by IP11[12]
GP2[9]	GP-2-9	Peripheral function selected by IP8[27:26]
GP2[10]	GP-2-10	Peripheral function selected by IP4[5:3]
GP2[11]	GP-2-11	Peripheral function selected by IP4[8:6]
GP2[12]	GP-2-12	Peripheral function selected by IP4[11:9]
GP2[13]	GP-2-13	Peripheral function selected by IP4[14:12]

Bit Name	GPIO (Set Value = H'0)	Peripheral Function (Set Value = H'1)
GP2[14]	GP-2-14	Peripheral function selected by IP4[17:15]
GP2[15]	GP-2-15	Peripheral function selected by IP4[19:18]
GP2[16]	GP-2-16	Peripheral function selected by IP4[21:20]
GP2[17]	GP-2-17	Peripheral function selected by IP4[23:22]
GP2[18]	GP-2-18	Peripheral function selected by IP4[25:24]
GP2[19]	GP-2-19	Peripheral function selected by IP4[27:26]
GP2[20]	GP-2-20	Peripheral function selected by IP4[29:28]
GP2[21]	GP-2-21	Peripheral function selected by IP4[31:30]
GP2[22]	GP-2-22	Peripheral function selected by IP5[2:0]
GP2[23]	GP-2-23	Peripheral function selected by IP5[5:3]
GP2[24]	GP-2-24	Peripheral function selected by IP5[8:6]
GP2[25]	GP-2-25	Peripheral function selected by IP5[11:9]
GP2[26]	GP-2-26	Peripheral function selected by IP5[14:12]
GP2[27]	GP-2-27	Peripheral function selected by IP5[17:15]
GP2[28]	GP-2-28	Peripheral function selected by IP5[20:18]
GP2[29]	GP-2-29	Peripheral function selected by IP5[22:21]
GP2[30]	GP-2-30	Peripheral function selected by IP5[24:23]
GP2[31]	GP-2-31	Peripheral function selected by IP5[26:25]

37.2.5 GPIO/Peripheral Function Select Register 3 (GPSR3)

Function: GPSR3 selects the functions of multiplexed LSI pins.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GP3 [31]	GP3 [30]	GP3 [29]	GP3 [28]	GP3 [27]	GP3 [26]	GP3 [25]	GP3 [24]	GP3 [23]	GP3 [22]	GP3 [21]	GP3 [20]	GP3 [19]	GP3 [18]	GP3 [17]	GP3 [16]
Initial value:	0	0	0	0	0	0	0	0	0	*	*	*	*	*	*	*
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GP3 [15]	GP3 [14]	GP3 [13]	GP3 [12]	GP3 [11]	GP3 [10]	GP3 [9]	GP3 [8]	GP3 [7]	GP3 [6]	GP3 [5]	GP3 [4]	GP3 [3]	GP3 [2]	GP3 [1]	GP3 [0]
Initial value:	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: * The initial value differs according to the operating mode : 0 in normal operation and 1 in HIF boot mode.

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GP3[31:0]	H'0000 0000 (in normal operation) H'007F FFFF (in HIF boot mode)	R/W	The functions of the LSI pins are selected according to the table below.

Note: To enable this register to be set, appropriately set the LSI multiplexed pin setting mask register (PMMR) immediately before setting this register.

Bit Name	GPIO (Set Value = H'0)	Peripheral Function (Set Value = H'1)
GP3[0]	GP-3-0	Peripheral function selected by IP6[2:0]
GP3[1]	GP-3-1	Peripheral function selected by IP6[5:3]
GP3[2]	GP-3-2	Peripheral function selected by IP6[7:6]
GP3[3]	GP-3-3	Peripheral function selected by IP6[9:8]
GP3[4]	GP-3-4	Peripheral function selected by IP6[11:10]
GP3[5]	GP-3-5	Peripheral function selected by IP6[13:12]
GP3[6]	GP-3-6	Peripheral function selected by IP6[15:14]
GP3[7]	GP-3-7	Peripheral function selected by IP6[17:16]
GP3[8]	GP-3-8	Peripheral function selected by IP6[20:18]
GP3[9]	GP-3-9	Peripheral function selected by IP6[23:21]
GP3[10]	GP-3-10	Peripheral function selected by IP7[2:0]
GP3[11]	GP-3-11	Peripheral function selected by IP7[5:3]
GP3[12]	GP-3-12	Peripheral function selected by IP7[8:6]

Bit Name	GPIO (Set Value = H'0)	Peripheral Function (Set Value = H'1)
GP3[13]	GP-3-13	Peripheral function selected by IP7[11:9]
GP3[14]	GP-3-14	Peripheral function selected by IP7[14:12]
GP3[15]	GP-3-15	Peripheral function selected by IP7[17:15]
GP3[16]	GP-3-16	Peripheral function selected by IP7[20:18]
GP3[17]	GP-3-17	Peripheral function selected by IP7[23:21]
GP3[18]	GP-3-18	Peripheral function selected by IP7[26:24]
GP3[19]	GP-3-19	Peripheral function selected by IP7[28:27]
GP3[20]	GP-3-20	Peripheral function selected by IP7[30:29]
GP3[21]	GP-3-21	Peripheral function selected by IP8[1:0]
GP3[22]	GP-3-22	Peripheral function selected by IP8[3:2]
GP3[23]	GP-3-23	Peripheral function selected by IP8[5:4]
GP3[24]	GP-3-24	Peripheral function selected by IP8[7:6]
GP3[25]	GP-3-25	Peripheral function selected by IP8[9:8]
GP3[26]	GP-3-26	Peripheral function selected by IP8[11:10]
GP3[27]	GP-3-27	Peripheral function selected by IP8[13:12]
GP3[28]	GP-3-28	Peripheral function selected by IP8[15:14]
GP3[29]	GP-3-29	Peripheral function selected by IP8[17:16]
GP3[30]	GP-3-30	Peripheral function selected by IP8[19:18]
GP3[31]	GP-3-31	Peripheral function selected by IP9[1:0]

37.2.6 GPIO/Peripheral Function Select Register 4 (GPSR4)

Function: GPSR4 selects the functions of multiplexed LSI pins.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GP4 [31]	GP4 [30]	GP4 [29]	GP4 [28]	GP4 [27]	GP4 [26]	GP4 [25]	GP4 [24]	GP4 [23]	GP4 [22]	GP4 [21]	GP4 [20]	GP4 [19]	GP4 [18]	GP4 [17]	GP4 [16]
Initial value:	0	0	0	1	0	1	*	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GP4 [15]	GP4 [14]	GP4 [13]	GP4 [12]	GP4 [11]	GP4 [10]	GP4 [9]	GP4 [8]	GP4 [7]	GP4 [6]	GP4 [5]	GP4 [4]	GP4 [3]	GP4 [2]	GP4 [1]	GP4 [0]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: * The initial value differs according to the operating mode : 0 in normal operation and 1 in HIF boot mode.

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GP4[31:0]	H'1400 0000 (in normal operation) H'1600 0000 (in HIF boot mode)	R/W	The functions of the LSI pins are selected according to the table below.

Note: To enable this register to be set, appropriately set the LSI multiplexed pin setting mask register (PMMR) immediately before setting this register.

Bit Name	GPIO (Set Value = H'0)	Peripheral Function (Set Value = H'1)
GP4[0]	GP-4-0	Peripheral function selected by IP9[19:18]
GP4[1]	GP-4-1	Peripheral function selected by IP9[21:20]
GP4[2]	GP-4-2	Peripheral function selected by IP9[23:22]
GP4[3]	GP-4-3	Peripheral function selected by IP9[25:24]
GP4[4]	GP-4-4	Peripheral function selected by IP9[11:10]
GP4[5]	GP-4-5	Peripheral function selected by IP9[13:12]
GP4[6]	GP-4-6	Peripheral function selected by IP9[15:14]
GP4[7]	GP-4-7	Peripheral function selected by IP9[17:16]
GP4[8]	GP-4-8	Peripheral function selected by IP9[3:2]
GP4[9]	GP-4-9	Peripheral function selected by IP9[5:4]
GP4[10]	GP-4-10	Peripheral function selected by IP9[7:6]
GP4[11]	GP-4-11	Peripheral function selected by IP9[9:8]
GP4[12]	GP-4-12	Peripheral function selected by IP9[27:26]

Bit Name	GPIO (Set Value = H'0)	Peripheral Function (Set Value = H'1)
GP4[13]	GP-4-13	Peripheral function selected by IP9[29:28]
GP4[14]	GP-4-14	Peripheral function selected by IP10[2:0]
GP4[15]	GP-4-15	Peripheral function selected by IP10[5:3]
GP4[16]	GP-4-16	Peripheral function selected by IP10[8:6]
GP4[17]	GP-4-17	Peripheral function selected by IP10[11:9]
GP4[18]	GP-4-18	Peripheral function selected by IP10[14:12]
GP4[19]	GP-4-19	Peripheral function selected by IP10[15]
GP4[20]	GP-4-20	Peripheral function selected by IP10[18:16]
GP4[21]	GP-4-21	Peripheral function selected by IP10[21:19]
GP4[22]	GP-4-22	Peripheral function selected by IP11[0]
GP4[23]	GP-4-23	Peripheral function selected by IP11[1]
GP4[24]	GP-4-24	SCL0
GP4[25]	GP-4-25	Peripheral function selected by IP11[2]
GP4[26]	GP-4-26	PENC0
GP4[27]	GP-4-27	Peripheral function selected by IP11[15:13]
GP4[28]	GP-4-28	USB_OVC0
GP4[29]	GP-4-29	Peripheral function selected by IP11[18:16]
GP4[30]	GP-4-30	Peripheral function selected by IP10[22]
GP4[31]	GP-4-31	Peripheral function selected by IP10[24:23]

37.2.7 GPIO/Peripheral Function Select Register 5 (GPSR5)

Function: GPSR5 selects the functions of multiplexed LSI pins.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	GP5 [11]	GP5 [10]	—	—	—	—	—	—	GP5 [3]	GP5 [2]	GP5 [1]	GP5 [0]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	These bits are always read as 0. The write value should always be 0.
11, 10	GP5[11:10]	H'0	R/W	The functions of the LSI pins are selected according to the table below.
9 to 4	—	All 0	R	These bits are always read as 0. The write value should always be 0.
3 to 0	GP5[3:0]	H'0	R/W	The functions of the LSI pins are selected according to the table below.

Note: To enable this register to be set, appropriately set the LSI multiplexed pin setting mask register (PMMR) immediately before setting this register.

Bit Name	GPIO (Set Value = H'0)	Peripheral Function (Set Value = H'1)
GP5[0]	GP-5-0	Peripheral function selected by IP10[25]
GP5[1]	GP-5-1	Peripheral function selected by IP11[3]
GP5[2]	GP-5-2	IRQ2_B
GP5[3]	GP-5-3	IRQ3_B
GP5[10]	GP-5-10	Peripheral function selected by IP10[27:26]
GP5[11]	GP-5-11	Peripheral function selected by IP10[29:28]

Note: Although pins 4 to 9 of GPIO channel 5 are multiplexed with ADC or AN2 to 7 pin functions, digital input for a dedicated cell is used with the ADC, so this cannot be mixed with the analogue input for the ADC. Accordingly, using the GPIO peripheral function register to switch pin functions is not necessary.

37.2.8 Peripheral Function Select Register 0 (IPSR0)

Function: IPSR0 selects the functions of multiplexed LSI pins.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IP0 [31]	IP0 [30]	IP0 [29]	IP0 [28]	IP0 [27]	IP0 [26]	IP0 [25]	IP0 [24]	IP0 [23]	IP0 [22]	IP0 [21]	IP0 [20]	IP0 [19]	IP0 [18]	IP0 [17]	IP0 [16]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IP0 [15]	IP0 [14]	IP0 [13]	IP0 [12]	IP0 [11]	IP0 [10]	IP0 [9]	IP0 [8]	IP0 [7]	IP0 [6]	IP0 [5]	IP0 [4]	IP0 [3]	IP0 [2]	IP0 [1]	IP0 [0]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	IP0[31:0]	H'0000 0000	R/W	The functions of the LSI pins are selected according to the table below.

Note: To enable this register to be set, appropriately set the LSI multiplexed pin setting mask register (PMMR) immediately before setting this register.

Bit Name	Function 1 (Set Value = H'0)	Function 2 (Set Value = H'1)	Function 3 (Set Value = H'2)	Function 4 (Set Value = H'3)
IP0[1:0]	A0	ST0_CLKIN	LCD_DATA0_A	TCLKA_C
IP0[3:2]	A1	ST0_REQ	LCD_DATA1_A	TCLKB_C
IP0[5:4]	A2	ST0_SYC	LCD_DATA2_A	TCLKC_C
IP0[7:6]	A3	ST0_VLD	LCD_DATA3_A	TCLKD_C
IP0[9:8]	A4	ST0_D[0]	LCD_DATA4_A	TIOC0A_C
IP0[11:10]	A5	ST0_D[1]	LCD_DATA5_A	TIOC0B_C
IP0[13:12]	A6	ST0_D[2]	LCD_DATA6_A	TIOC0C_C
IP0[15:14]	A7	ST0_D[3]	LCD_DATA7_A	TIOC0D_C
IP0[17:16]	A8	ST0_D[4]	LCD_DATA8_A	TIOC1A_C
IP0[19:18]	A9	ST0_D[5]	LCD_DATA9_A	TIOC1B_C
IP0[21:20]	A10	ST0_D[6]	LCD_DATA10_A	TIOC2A_C
IP0[23:22]	A11	ST0_D[7]	LCD_DATA11_A	TIOC2B_C
IP0[25:24]	A12	LCD_DATA12_A	TIOC3A_C	—
IP0[27:26]	A13	LCD_DATA13_A	TIOC3B_C	—
IP0[29:28]	A14	LCD_DATA14_A	TIOC3C_C	—
IP0[31:30]	A15	ST0_VCO_CLKIN	LCD_DATA15_A	TIOC3D_C

[Legend] —: Setting invalid or impossible

37.2.9 Peripheral Function Select Register 1 (IPSR1)

Function: IPSR1 selects the functions of multiplexed LSI pins.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IP1 [31]	IP1 [30]	IP1 [29]	IP1 [28]	IP1 [27]	IP1 [26]	IP1 [25]	IP1 [24]	IP1 [23]	IP1 [22]	IP1 [21]	IP1 [20]	IP1 [19]	IP1 [18]	IP1 [17]	IP1 [16]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IP1 [15]	IP1 [14]	IP1 [13]	IP1 [12]	IP1 [11]	IP1 [10]	IP1 [9]	IP1 [8]	IP1 [7]	IP1 [6]	IP1 [5]	IP1 [4]	IP1 [3]	IP1 [2]	IP1 [1]	IP1 [0]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	IP1[31:0]	H'0000 0000	R/W	The functions of the LSI pins are selected according to the table below.

Note: To enable this register to be set, appropriately set the LSI multiplexed pin setting mask register (PMMR) immediately before setting this register.

Bit Name	Function 1 (Set Value = H'0)	Function 2 (Set Value = H'1)	Function 3 (Set Value = H'2)	Function 4 (Set Value = H'3)	Function 5 (Set Value = H'4)
IP1[1:0]	A16	ST0_PWM	LCD_DON_A	TIOC4A_C	—
IP1[3:2]	A17	ST1_VCO_CLKIN	LCD_CL1_A	TIOC4B_C	—
IP1[5:4]	A18	ST1_PWM	LCD_CL2_A	TIOC4C_C	—
IP1[7:6]	A19	ST1_CLKIN	LCD_CLK_A	TIOC4D_C	—
IP1[9:8]	A20	ST1_REQ	LCD_FLM_A	—	—
IP1[11:10]	A21	ST1_SYC	LCD_VCPWC_A	—	—
IP1[13:12]	A22	ST1_VLD	LCD_VEPWC_A	—	—
IP1[15:14]	A23	ST1_D[0]	LCD_M_DISP_A	—	—
IP1[17:16]	A24	RX2_D	ST1_D[1]	—	—
IP1[19:18]	A25	TX2_D	ST1_D[2]	—	—
IP1[22:20]	D0	SD0_DAT0_A	MMC_D0_A	ST1_D[3]	NAF0_A
IP1[25:23]	D1	SD0_DAT1_A	MMC_D1_A	ST1_D[4]	NAF1_A
IP1[28:26]	D2	SD0_DAT2_A	MMC_D2_A	ST1_D[5]	NAF2_A
IP1[31:29]	D3	SD0_DAT3_A	MMC_D3_A	ST1_D[6]	NAF3_A

[Legend] —: Setting invalid or impossible

37.2.10 Peripheral Function Select Register 2 (IPSR2)

Function: IPSR2 selects the functions of multiplexed LSI pins.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	IP2 [30]	IP2 [29]	IP2 [28]	IP2 [27]	IP2 [26]	IP2 [25]	IP2 [24]	IP2 [23]	IP2 [22]	IP2 [21]	IP2 [20]	IP2 [19]	IP2 [18]	IP2 [17]	IP2 [16]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IP2 [15]	IP2 [14]	IP2 [13]	IP2 [12]	IP2 [11]	IP2 [10]	IP2 [9]	IP2 [8]	IP2 [7]	IP2 [6]	IP2 [5]	IP2 [4]	IP2 [3]	IP2 [2]	IP2 [1]	IP2 [0]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	This bit is always read as 0. The write value should always be 0.
30 to 0	IP2[30:0]	H'0000 0000	R/W	The functions of the LSI pins are selected according to the table below.

Note: To enable this register to be set, appropriately set the LSI multiplexed pin setting mask register (PMMR) immediately before setting this register.

Bit Name	Function 1 (Set Value = H'0)	Function 2 (Set Value = H'1)	Function 3 (Set Value = H'2)	Function 4 (Set Value = H'3)	Function 5 (Set Value = H'4)	Function 6 (Set Value = H'5)
IP2[2:0]	D4	SD0_CD_A	MMC_D4_A	ST1_D[7]	NAF4_A	—
IP2[4:3]	D5	SD0_WP_A	MMC_D5_A	NAF5_A	—	—
IP2[7:5]	D6	RSPI_RSPCK_A	MMC_D6_A	QSPCLK_A	NAF6_A	—
IP2[10:8]	D7	RSPI_SSL_A	MMC_D7_A	QSSL_A	NAF7_A	—
IP2[13:11]	D8	SD0_CLK_A	MMC_CLK_A	QIO2_A	$\overline{\text{FCE\_A}}$	ET0_GTX_CLK_B
IP2[16:14]	D9	SD0_CMD_A	MMC_CMD_A	QIO3_A	FCLE_A	ET0_ETXD1_B
IP2[19:17]	D10	RSPI_MOSI_A	—	QMO / QIO0_A	FALE_A	ET0_ETXD2_B
IP2[22:20]	D11	RSPI_MISO_A	—	QMI / QIO1_A	$\overline{\text{FRE\_A}}$	ET0_ETXD3_B
IP2[24:23]	D12	—	$\overline{\text{FWE\_A}}$	ET0_ETXD5_B	—	—
IP2[27:25]	D13	RX2_B	—	FRB_A	ET0_ETXD6_B	—
IP2[30:28]	D14	TX2_B	—	—	ET0_TX_CLK_B	—

[Legend] —: Setting invalid or impossible

37.2.11 Peripheral Function Select Register 3 (IPSR3)

Function: IPSR3 selects the functions of multiplexed LSI pins.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	IP3 [29]	IP3 [28]	IP3 [27]	IP3 [26]	IP3 [25]	IP3 [24]	IP3 [23]	IP3 [22]	IP3 [21]	IP3 [20]	IP3 [19]	IP3 [18]	IP3 [17]	IP3 [16]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IP3 [15]	IP3 [14]	IP3 [13]	IP3 [12]	IP3 [11]	IP3 [10]	IP3 [9]	IP3 [8]	IP3 [7]	IP3 [6]	IP3 [5]	IP3 [4]	IP3 [3]	IP3 [2]	IP3 [1]	IP3 [0]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	These bits are always read as 0. The write value should always be 0.
29 to 0	IP3[29:0]	H'0000 0000	R/W	The functions of the LSI pins are selected according to the table below.

Note: To enable this register to be set, appropriately set the LSI multiplexed pin setting mask register (PMMR) immediately before setting this register.

Bit Name	Function 1 (Set Value = H'0)	Function 2 (Set Value = H'1)	Function 3 (Set Value = H'2)	Function 4 (Set Value = H'3)	Function 5 (Set Value = H'4)	Function 6 (Set Value = H'5)
IP3[1:0]	D15	SCK2_B	—	—	—	—
IP3[2]	$\overline{\text{CS1}}$/A26	QIO3_B	—	—	—	—
IP3[5:3]	$\overline{\text{EX\_CS1}}$	RX3_B	$\overline{\text{ATACS0}}$	QIO2_B	ET0_ETXD0	—
IP3[8:6]	$\overline{\text{EX\_CS2}}$	TX3_B	$\overline{\text{ATACS1}}$	QSPCLK_B	ET0_GTX_CLK_A	—
IP3[11:9]	$\overline{\text{EX\_CS3}}$	SD1_CD_A	$\overline{\text{ATARD}}$	QMO / QIO0_B	ET0_ETXD1_A	—
IP3[14:12]	$\overline{\text{EX\_CS4}}$	SD1_WP_A	$\overline{\text{ATAWR}}$	QMI / QIO1_B	ET0_ETXD2_A	—
IP3[17:15]	$\overline{\text{EX\_CS5}}$	SD1_CMD_A	$\overline{\text{ATADIR}}$	QSSL_B	ET0_ETXD3_A	—
IP3[19:18]	RD/$\overline{\text{WR}}$	TCLK0	CAN_CLK_B	ET0_ETXD4	—	—
IP3[20]	EX_WAIT0	TCLK1_B	—	—	—	—
IP3[23:21]	EX_WAIT1	SD1_DAT0_A	DREQ2	CAN1_TX_C	ET0_LINK_C	ET0_ETXD5_A
IP3[26:24]	EX_WAIT2	SD1_DAT1_A	DACK2	CAN1_RX_C	ET0_MAGIC_C	ET0_ETXD6_A
IP3[29:27]	DRACK0	SD1_DAT2_A	$\overline{\text{ATAG}}$	TCLK1_A	ET0_ETXD7	—

[Legend] —: Setting invalid or impossible

37.2.12 Peripheral Function Select Register 4 (IPSR4)

Function: IPSR4 selects the functions of multiplexed LSI pins.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IP4 [31]	IP4 [30]	IP4 [29]	IP4 [28]	IP4 [27]	IP4 [26]	IP4 [25]	IP4 [24]	IP4 [23]	IP4 [22]	IP4 [21]	IP4 [20]	IP4 [19]	IP4 [18]	IP4 [17]	IP4 [16]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IP4 [15]	IP4 [14]	IP4 [13]	IP4 [12]	IP4 [11]	IP4 [10]	IP4 [9]	IP4 [8]	IP4 [7]	IP4 [6]	IP4 [5]	IP4 [4]	IP4 [3]	IP4 [2]	IP4 [1]	IP4 [0]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	IP4[31:0]	H'0000 0000	R/W	The functions of the LSI pins are selected according to the table below.

Note: To enable this register to be set, appropriately set the LSI multiplexed pin setting mask register (PMMR) immediately before setting this register.

Bit Name	Function 1 (Set Value = H'0)	Function 2 (Set Value = H'1)	Function 3 (Set Value = H'2)	Function 4 (Set Value = H'3)	Function 5 (Set Value = H'4)	Function 6 (Set Value = H'5)
IP4[2:0]	$\overline{\text{HCTS0\_A}}$	$\overline{\text{CTS1\_A}}$	VI0_FIELD	RMII0_RXD1_A	ET0_ERXD7	—
IP4[5:3]	$\overline{\text{HRTS0\_A}}$	$\overline{\text{RTS1\_A}}$	$\overline{\text{VI0\_HSYNC}}$	RMII0_TXD_EN_A	ET0_RX_DV	—
IP4[8:6]	HSCK0_A	SCK1_A	$\overline{\text{VI0\_VSYNC}}$	RMII0_RX_ER_A	ET0_RX_ER	—
IP4[11:9]	HRX0_A	RX1_A	VI0_DATA0/VI0_B0	RMII0_CRS_DV_A	ET0_CRS	—
IP4[14:12]	HTX0_A	TX1_A	VI0_DATA1/VI0_B1	RMII0_MDC_A	ET0_COL	—
IP4[17:15]	—	$\overline{\text{CTS0\_B}}$	VI0_DATA2/VI0_B2	RMII0_MDIO_A	ET0_MDC	—
IP4[19:18]	—	$\overline{\text{RTS0\_B}}$	VI0_DATA3/VI0_B3	ET0_MDIO_A	—	—
IP4[21:20]	—	SCK1_B	VI0_DATA4/VI0_B4	ET0_LINK_A	—	—
IP4[23:22]	—	RX1_B	VI0_DATA5/VI0_B5	ET0_MAGIC_A	—	—
IP4[25:24]	—	TX1_B	VI0_DATA6/VI0_G0	ET0_PHY_INT_A	—	—
IP4[27:26]	—	$\overline{\text{CTS1\_B}}$	VI0_DATA7/VI0_G1	—	—	—
IP4[29:28]	—	$\overline{\text{RTS1\_B}}$	VI0_G2	—	—	—
IP4[31:30]	—	SCK2_A	VI0_G3	—	—	—

[Legend] —: Setting invalid or impossible

37.2.13 Peripheral Function Select Register 5 (IPSR5)

Function: IPSR5 selects the functions of multiplexed LSI pins.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	IP5 [26]	IP5 [25]	IP5 [24]	IP5 [23]	IP5 [22]	IP5 [21]	IP5 [20]	IP5 [19]	IP5 [18]	IP5 [17]	IP5 [16]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IP5 [15]	IP5 [14]	IP5 [13]	IP5 [12]	IP5 [11]	IP5 [10]	IP5 [9]	IP5 [8]	IP5 [7]	IP5 [6]	IP5 [5]	IP5 [4]	IP5 [3]	IP5 [2]	IP5 [1]	IP5 [0]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	These bits are always read as 0. The write value should always be 0.
26 to 0	IP5[26:0]	H'000 0000	R/W	The functions of the LSI pins are selected according to the table below.

Note: To enable this register to be set, appropriately set the LSI multiplexed pin setting mask register (PMMR) immediately before setting this register.

Bit Name	Function 1 (Set Value = H'0)	Function 2 (Set Value = H'1)	Function 3 (Set Value = H'2)	Function 4 (Set Value = H'3)	Function 5 (Set Value = H'4)
IP5[2:0]	SD2_CLK_A	RX2_A	VI0_G4	—	ET0_RX_CLK_B
IP5[5:3]	SD2_CMD_A	TX2_A	VI0_G5	—	ET0_ERXD2_B
IP5[8:6]	SD2_DAT0_A	RX3_A	VI0_R0	—	ET0_ERXD3_B
IP5[11:9]	SD2_DAT1_A	TX3_A	VI0_R1	—	ET0_MDIO_B
IP5[14:12]	SD2_DAT2_A	RX4_A	VI0_R2	—	ET0_LINK_B
IP5[17:15]	SD2_DAT3_A	TX4_A	VI0_R3	—	ET0_MAGIC_B
IP5[20:18]	SD2_CD_A	RX5_A	VI0_R4	—	ET0_PHY_INT_B
IP5[22:21]	SD2_WP_A	TX5_A	VI0_R5	—	—
IP5[24:23]	REF125CK	$\overline{\text{ADTRG}}$	RX5_C	—	—
IP5[26:25]	REF50CK	$\overline{\text{CTS1\_E}}$	$\overline{\text{HCTS0\_D}}$	—	—

[Legend] —: Setting invalid or impossible

37.2.14 Peripheral Function Select Register 6 (IPSR6)

Function: IPSR6 selects the functions of multiplexed LSI pins.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	IP6 [23]	IP6 [22]	IP6 [21]	IP6 [20]	IP6 [19]	IP6 [18]	IP6 [17]	IP6 [16]
Initial value:	0	0	0	0	0	0	0	0	*	0	0	*	0	*	*	*
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IP6 [15]	IP6 [14]	IP6 [13]	IP6 [12]	IP6 [11]	IP6 [10]	IP6 [9]	IP6 [8]	IP6 [7]	IP6 [6]	IP6 [5]	IP6 [4]	IP6 [3]	IP6 [2]	IP6 [1]	IP6 [0]
Initial value:	*	*	*	*	*	*	*	*	*	*	*	0	*	*	0	*
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: * The initial value differs according to the operating mode : 0 in normal operation and 1 in HIF boot mode.

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	These bits are always read as 0. The write value should always be 0.
23 to 0	IP6[23:0]	H'00 0000 (in normal operation) H'97 FFED (in HIF boot mode)	R/W	The functions of the LSI pins are selected according to the table below.

Note: To enable this register to be set, appropriately set the LSI multiplexed pin setting mask register (PMMR) immediately before setting this register.

Bit Name	Function 1 (Set Value = H'0)	Function 2 (Set Value = H'1)	Function 3 (Set Value = H'2)	Function 4 (Set Value = H'3)	Function 5 (Set Value = H'4)	Function 6 (Set Value = H'5)
IP6[2:0]	DU0_DR0	SCIF_CLK_B	HRX0_D	IETX_A	TCLKA_A	HIFD00
IP6[5:3]	DU0_DR1	SCK0_B	HTX0_D	IERX_A	TCLKB_A	HIFD01
IP6[7:6]	DU0_DR2	RX0_B	TCLKC_A	HIFD02	—	—
IP6[9:8]	DU0_DR3	TX0_B	TCLKD_A	HIFD03	—	—
IP6[11:10]	DU0_DR4	$\overline{\text{CTS0\_C}}$	TIOC0A_A	HIFD04	—	—
IP6[13:12]	DU0_DR5	$\overline{\text{RTS0\_C}}$	TIOC0B_A	HIFD05	—	—
IP6[15:14]	DU0_DR6	SCK1_C	TIOC0C_A	HIFD06	—	—
IP6[17:16]	DU0_DR7	RX1_C	TIOC0D_A	HIFD07	—	—
IP6[20:18]	DU0_DG0	TX1_C	HSCK0_D	IECLK_A	TIOC1A_A	HIFD08
IP6[23:21]	DU0_DG1	$\overline{\text{CTS1\_C}}$	$\overline{\text{HRTS0\_D}}$	TIOC1B_A	HIFD09	—

[Legend] —: Setting invalid or impossible

37.2.15 Peripheral Function Select Register 7 (IPSR7)

Function: IPSR7 selects the functions of multiplexed LSI pins.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	IP7 [30]	IP7 [29]	IP7 [28]	IP7 [27]	IP7 [26]	IP7 [25]	IP7 [24]	IP7 [23]	IP7 [22]	IP7 [21]	IP7 [20]	IP7 [19]	IP7 [18]	IP7 [17]	IP7 [16]
Initial value:	0	*	0	*	*	*	0	0	*	0	0	*	0	0	*	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IP7 [15]	IP7 [14]	IP7 [13]	IP7 [12]	IP7 [11]	IP7 [10]	IP7 [9]	IP7 [8]	IP7 [7]	IP7 [6]	IP7 [5]	IP7 [4]	IP7 [3]	IP7 [2]	IP7 [1]	IP7 [0]
Initial value:	0	*	0	0	*	0	0	*	0	0	*	0	0	*	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: * The initial value differs according to the operating mode : 0 in normal operation and 1 in HIF boot mode.

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	This bit is always read as 0. The write value should always be 0.
30 to 0	IP7[30:0]	H'0000 0000 (in normal operation) H'5C92 4924 (in HIF boot mode)	R/W	The functions of the LSI pins are selected according to the table below.

Note: To enable this register to be set, appropriately set the LSI multiplexed pin setting mask register (PMMR) immediately before setting this register.

Bit Name	Function 1 (Set Value = H'0)	Function 2 (Set Value = H'1)	Function 3 (Set Value = H'2)	Function 4 (Set Value = H'3)	Function 5 (Set Value = H'4)	Function 6 (Set Value = H'5)
IP7[2:0]	DU0_DG2	$\overline{\text{RTS1\_C}}$	RMII0_MDC_B	TIOC2A_A	HIFD10	—
IP7[5:3]	DU0_DG3	SCK2_C	RMII0_MDIO_B	TIOC2B_A	HIFD11	—
IP7[8:6]	DU0_DG4	RX2_C	RMII0_CRS_DV_B	TIOC3A_A	HIFD12	—
IP7[11:9]	DU0_DG5	TX2_C	RMII0_RX_ER_B	TIOC3B_A	HIFD13	—
IP7[14:12]	DU0_DG6	RX3_C	RMII0_RXD0_B	TIOC3C_A	HIFD14	—
IP7[17:15]	DU0_DG7	TX3_C	RMII0_RXD1_B	TIOC3D_A	HIFD15	—
IP7[20:18]	DU0_DB0	RX4_C	RMII0_TXD_EN_B	TIOC4A_A	$\overline{\text{HIFCS}}$	—
IP7[23:21]	DU0_DB1	TX4_C	RMII0_TXD0_B	TIOC4B_A	HIFRS	—
IP7[26:24]	DU0_DB2	RX5_B	RMII0_TXD1_B	TIOC4C_A	$\overline{\text{HIFWR}}$	—
IP7[28:27]	DU0_DB3	TX5_B	TIOC4D_A	$\overline{\text{HIFRD}}$	—	—
IP7[30:29]	DU0_DB4	SD2_CLK_B	$\overline{\text{HIFINT}}$	—	—	—

[Legend] — : Setting invalid or impossible

37.2.16 Peripheral Function Select Register 8 (IPSR8)

Function: IPSR8 selects the functions of multiplexed LSI pins.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	IP8 [29]	IP8 [28]	IP8 [27]	IP8 [26]	IP8 [25]	IP8 [24]	IP8 [23]	IP8 [22]	IP8 [21]	IP8 [20]	IP8 [19]	IP8 [18]	IP8 [17]	IP8 [16]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IP8 [15]	IP8 [14]	IP8 [13]	IP8 [12]	IP8 [11]	IP8 [10]	IP8 [9]	IP8 [8]	IP8 [7]	IP8 [6]	IP8 [5]	IP8 [4]	IP8 [3]	IP8 [2]	IP8 [1]	IP8 [0]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	*	0	*	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: * The initial value differs according to the operating mode : 0 in normal operation and 1 in HIF boot mode.

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	These bits are always read as 0. The write value should always be 0.
29 to 0	IP8[29:0]	H'0000 0000 (in normal operation) H'0000 000A (in HIF boot mode)	R/W	The functions of the LSI pins are selected according to the table below.

Note: To enable this register to be set, appropriately set the LSI multiplexed pin setting mask register (PMMR) immediately before setting this register.

Bit Name	Function 1 (Set Value = H'0)	Function 2 (Set Value = H'1)	Function 3 (Set Value = H'2)	Function 4 (Set Value = H'3)	Function 5 (Set Value = H'4)
IP8[1:0]	DU0_DB5	SD2_CMD_B	HIFDREQ	—	—
IP8[3:2]	DU0_DB6	SD2_DAT0_B	HIFRDY	—	—
IP8[5:4]	DU0_DB7	SD2_DAT1_B	SSI_SCK0_B	HIFEBL_B	—
IP8[7:6]	DU0_DOTCLKIN	SD2_DAT2_B	$\overline{\text{HSPI\_CS\_C}}$	SSI_WS0_B	—
IP8[9:8]	DU0_DOTCLKOUT	SD2_DAT3_B	HSPI_CLK_C	SSI_SDATA0_B	—
IP8[11:10]	DU0_EXHSYNC/ DU0_HSYNC	SD2_CD_B	HSPI_TX_C	SSI_SCK1_B	—
IP8[13:12]	DU0_EXVSYNC/ DU0_VSYNC	SD2_WP_B	HSPI_RX_C	SSI_WS1_B	—
IP8[15:14]	DU0_EXODDF/ DU0_ODDF	CAN0_RX_B	HSCK0_B	SSI_SDATA1_B	—

Bit Name	Function 1 (Set Value = H'0)	Function 2 (Set Value = H'1)	Function 3 (Set Value = H'2)	Function 4 (Set Value = H'3)	Function 5 (Set Value = H'4)
IP8[17:16]	DU0_DISP	CAN0_TX_B	HRX0_B	AUDIO_CLKA_B	—
IP8[19:18]	DU0_CDE	HTX0_B	AUDIO_CLKB_B	LCD_VCPWC_B	—
IP8[22:20]	IRQ0_A	—	HSPI_TX_B	RX3_E	ET0_ERXD0
IP8[25:23]	IRQ1_A	—	HSPI_RX_B	TX3_E	ET0_ERXD1
IP8[27:26]	IRQ2_A	$\overline{\text{CTS0\_A}}$	$\overline{\text{HCTS0\_B}}$	ET0_ERXD2_A	—
IP8[29:28]	IRQ3_A	$\overline{\text{RTS0\_A}}$	$\overline{\text{HRTS0\_B}}$	ET0_ERXD3_A	—

[Legend] — : Setting invalid or impossible

37.2.17 Peripheral Function Select Register 9 (IPSR9)

Function: IPSR9 selects the functions of multiplexed LSI pins.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	IP9 [29]	IP9 [28]	IP9 [27]	IP9 [26]	IP9 [25]	IP9 [24]	IP9 [23]	IP9 [22]	IP9 [21]	IP9 [20]	IP9 [19]	IP9 [18]	IP9 [17]	IP9 [16]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IP9 [15]	IP9 [14]	IP9 [13]	IP9 [12]	IP9 [11]	IP9 [10]	IP9 [9]	IP9 [8]	IP9 [7]	IP9 [6]	IP9 [5]	IP9 [4]	IP9 [3]	IP9 [2]	IP9 [1]	IP9 [0]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	These bits are always read as 0. The write value should always be 0.
29 to 0	IP9[29:0]	H'0000 0000	R/W	The functions of the LSI pins are selected according to the table below.

Note: To enable this register to be set, appropriately set the LSI multiplexed pin setting mask register (PMMR) immediately before setting this register.

Bit Name	Function 1 (Set Value = H'0)	Function 2 (Set Value = H'1)	Function 3 (Set Value = H'2)	Function 4 (Set Value = H'3)
IP9[1:0]	VI1_CLK_A	—	NAF0_B	LCD_DATA0_B
IP9[3:2]	VI1_0_A	—	NAF1_B	LCD_DATA1_B
IP9[5:4]	VI1_1_A	—	NAF2_B	LCD_DATA2_B
IP9[7:6]	VI1_2_A	—	NAF3_B	LCD_DATA3_B
IP9[9:8]	VI1_3_A	—	NAF4_B	LCD_DATA4_B
IP9[11:10]	VI1_4_A	—	NAF5_B	LCD_DATA5_B
IP9[13:12]	VI1_5_A	—	NAF6_B	LCD_DATA6_B
IP9[15:14]	VI1_6_A	—	NAF7_B	LCD_DATA7_B
IP9[17:16]	VI1_7_A	$\overline{\text{FCE\_B}}$	LCD_DATA8_B	—
IP9[19:18]	SSI_SCK0_A	TIOC1A_B	LCD_DATA9_B	—
IP9[21:20]	SSI_WS0_A	TIOC1B_B	LCD_DATA10_B	—
IP9[23:22]	SSI_SDATA0_A	VI1_0_B	TIOC2A_B	LCD_DATA11_B
IP9[25:24]	SSI_SCK1_A	VI1_1_ B	TIOC2B_B	LCD_DATA12_B
IP9[27:26]	SSI_WS1_A	VI1_2_ B	LCD_DATA13_B	—
IP9[29:28]	SSI_SDATA1_A	VI1_3_ B	LCD_DATA14_B	—

[Legend] —: Setting invalid or impossible

37.2.18 Peripheral Function Select Register 10 (IPSR10)

Function: IPSR10 selects the functions of multiplexed LSI pins.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	IP10 [29]	IP10 [28]	IP10 [27]	IP10 [26]	IP10 [25]	IP10 [24]	IP10 [23]	IP10 [22]	IP10 [21]	IP10 [20]	IP10 [19]	IP10 [18]	IP10 [17]	IP10 [16]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IP10 [15]	IP10 [14]	IP10 [13]	IP10 [12]	IP10 [11]	IP10 [10]	IP10 [9]	IP10 [8]	IP10 [7]	IP10 [6]	IP10 [5]	IP10 [4]	IP10 [3]	IP10 [2]	IP10 [1]	IP10 [0]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	These bits are always read as 0. The write value should always be 0.
29 to 0	IP10[29:0]	H'0000 0000	R/W	The functions of the LSI pins are selected according to the table below.

Note: To enable this register to be set, appropriately set the LSI multiplexed pin setting mask register (PMMR) immediately before setting this register.

Bit Name	Function 1 (Set Value = H'0)	Function 2 (Set Value = H'1)	Function 3 (Set Value = H'2)	Function 4 (Set Value = H'3)	Function 5 (Set Value = H'4)	Function 6 (Set Value = H'5)
IP10[2:0]	SSI_SCK23	VI1_4_B	RX1_D	FCLE_B	LCD_DATA15_B	—
IP10[5:3]	SSI_WS23	VI1_5_B	TX1_D	HSCK0_C	FALE_B	LCD_DON_B
IP10[8:6]	SSI_SDATA2	VI1_6_B	—	HRX0_C	$\overline{\text{FRE\_B}}$	LCD_CL1_B
IP10[11:9]	SSI_SDATA3	VI1_7_B	—	HTX0_C	$\overline{\text{FWE\_B}}$	LCD_CL2_B
IP10[14:12]	AUDIO_CLKA_A	VI1_CLK_B	SCK1_D	IECLK_B	LCD_FLM_B	—
IP10[15]	AUDIO_CLKB_A	LCD_CLK_B	—	—	—	—
IP10[18:16]	AUDIO_CLKC	SCK1_E	—	$\overline{\text{HCTS0\_C}}$	FRB_B	LCD_VEPWC_B
IP10[21:19]	AUDIO_CLKOUT	TX1_E	—	$\overline{\text{HRTS0\_C}}$	—	LCD_M_DISP_B
IP10[22]	CAN_CLK_A	RX4_D	—	—	—	—
IP10[24:23]	CAN0_TX_A	TX4_D	MLB_CLK	—	—	—
IP10[25]	CAN1_RX_A	IRQ1_B	—	—	—	—
IP10[27:26]	CAN0_RX_A	IRQ0_B	MLB_SIG	—	—	—
IP10[29:28]	CAN1_TX_A	TX5_C	MLB_DAT	—	—	—

[Legend] —: Setting invalid or impossible

37.2.19 Peripheral Function Select Register 11 (IPSR11)

Function: IPSR11 selects the functions of multiplexed LSI pins.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	IP11 [28]	IP11 [27]	IP11 [26]	IP11 [25]	IP11 [24]	IP11 [23]	IP11 [22]	IP11 [21]	IP11 [20]	IP11 [19]	IP11 [18]	IP11 [17]	IP11 [16]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IP11 [15]	IP11 [14]	IP11 [13]	IP11 [12]	IP11 [11]	IP11 [10]	IP11 [9]	IP11 [8]	IP11 [7]	IP11 [6]	IP11 [5]	IP11 [4]	IP11 [3]	IP11 [2]	IP11 [1]	IP11 [0]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	*	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: * The initial value differs according to the operating mode : 0 in normal operation and 1 in HIF boot mode.

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	These bits are always read as 0. The write value should always be 0.
28 to 0	IP11[28:0]	H'0000 0000 (in normal operation) H'0000 0004 (in HIF boot mode)	R/W	The functions of the LSI pins are selected according to the table below.

Note: To enable this register to be set, appropriately set the LSI multiplexed pin setting mask register (PMMR) immediately before setting this register.

Bit Name	Function 1 (Set Value = H'0)	Function 2 (Set Value = H'1)	Function 3 (Set Value = H'2)	Function 4 (Set Value = H'3)	Function 5 (Set Value = H'4)
IP11[0]	SCL1	SCIF_CLK_C	—	—	—
IP11[1]	SDA1	RX1_E	—	—	—
IP11[2]	SDA0	HIFEBL_A	—	—	—
IP11[3]	SDSELF	$\overline{\text{RTS1\_E}}$	—	—	—
IP11[6:4]	SCIF_CLK_A	HSPI_CLK_A	VI0_CLK	RMII0_TXD0_A	ET0_ERXD4
IP11[9:7]	SCK0_A	$\overline{\text{HSPI\_CS\_A}}$	VI0_CLKENB	RMII0_TXD1_A	ET0_ERXD5
IP11[11:10]	RX0_A	HSPI_RX_A	RMII0_RXD0_A	ET0_ERXD6	—
IP11[12]	TX0_A	HSPI_TX_A	—	—	—
IP11[15:13]	PENC1	TX3_D	CAN1_TX_B	TX5_D	IETX_B
IP11[18:16]	USB_OVC1	RX3_D	CAN1_RX_B	RX5_D	IERX_B

Bit Name	Function 1 (Set Value = H'0)	Function 2 (Set Value = H'1)	Function 3 (Set Value = H'2)	Function 4 (Set Value = H'3)	Function 5 (Set Value = H'4)
IP11[20:19]	DREQ0	SD1_CLK_A	ET0_TX_EN	—	—
IP11[22:21]	DACK0	SD1_DAT3_A	ET0_TX_ER	—	—
IP11[25:23]	DREQ1	HSPI_CLK_B	RX4_B	ET0_PHY_INT_C	ET0_TX_CLK_A
IP11[27:26]	DACK1	$\overline{\text{HSPI\_CS\_B}}$	TX4_B	ET0_RX_CLK_A	—
IP11[28]	$\overline{\text{PRESETOUT}}$	ST_CLKOUT	—	—	—

[Legend] — : Setting invalid or impossible

Table 37.3 shows the correspondence between the function signals and the bit settings in the GPIO/peripheral function select registers and peripheral function select registers.

Table 37.3 Correspondence between Function Signals and Register Bit Settings

Selection between GPIO and Peripheral Function

GPIO (GP Set Value = 0)	Peripheral Module (GP Set Value = 1)						GPIO/ Peripheral Function Select Bit	Peripheral Function Select Bit
	Function Selected by IP Bits							
	Function 1 (IP Set Value = 0)	Function 2 (IP Set Value = 1)	Function 3 (IP Set Value = 2)	Function 4 (IP Set Value = 3)	Function 5 (IP Set Value = 4)	Function 6 (IP Set Value = 5)		
GP-0-0	A20	ST1_REQ	LCD_FLM_A	—	—	—	GP0[0]	IP1[9:8]
GP-0-1	A21	ST1_SYC	LCD_VCPWC_A	—	—	—	GP0[1]	IP1[11:10]
GP-0-2	A22	ST1_VLD	LCD_VEPWC_A	—	—	—	GP0[2]	IP1[13:12]
GP-0-3	A23	ST1_D[0]	LCD_M_DISP_A	—	—	—	GP0[3]	IP1[15:14]
GP-0-4	A3	ST0_VLD	LCD_DATA3_A	TCLKD_C	—	—	GP0[4]	IP0[7:6]
GP-0-5	A4	ST0_D[0]	LCD_DATA4_A	TIOC0A_C	—	—	GP0[5]	IP0[9:8]
GP-0-6	A5	ST0_D[1]	LCD_DATA5_A	TIOC0B_C	—	—	GP0[6]	IP0[11:10]
GP-0-7	A6	ST0_D[2]	LCD_DATA6_A	TIOC0C_C	—	—	GP0[7]	IP0[13:12]
GP-0-8	A7	ST0_D[3]	LCD_DATA7_A	TIOC0D_C	—	—	GP0[8]	IP0[15:14]
GP-0-9	A8	ST0_D[4]	LCD_DATA8_A	TIOC1A_C	—	—	GP0[9]	IP0[17:16]
GP-0-10	A9	ST0_D[5]	LCD_DATA9_A	TIOC1B_C	—	—	GP0[10]	IP0[19:18]
GP-0-11	A10	ST0_D[6]	LCD_DATA10_A	TIOC2A_C	—	—	GP0[11]	IP0[21:20]
GP-0-12	A11	ST0_D[7]	LCD_DATA11_A	TIOC2B_C	—	—	GP0[12]	IP0[23:22]
GP-0-13	A12	LCD_DATA12_A	TIOC3A_C	—	—	—	GP0[13]	IP0[25:24]

Selection between GPIO and Peripheral Function

GPIO (GP Set Value = 0)	Peripheral Module (GP Set Value = 1)						GPIO/	
	Function Selected by IP Bits							
	Function 1 (IP Set Value = 0)	Function 2 (IP Set Value = 1)	Function 3 (IP Set Value = 2)	Function 4 (IP Set Value = 3)	Function 5 (IP Set Value = 4)	Function 6 (IP Set Value = 5)	Peripheral Function Select Bit	Peripheral Function Select Bit
GP-0-14	A13	LCD_DATA13_A	TIOC3B_C	—	—	—	GP0[14]	IP0[27:26]
GP-0-15	A14	LCD_DATA14_A	TIOC3C_C	—	—	—	GP0[15]	IP0[29:28]
GP-0-16	A15	ST0_VCO_CLKIN	LCD_DATA15_A	TIOC3D_C	—	—	GP0[16]	IP0[31:30]
GP-0-17	A16	ST0_PWM	LCD_DON_A	TIOC4A_C	—	—	GP0[17]	IP1[1:0]
GP-0-18	A17	ST1_VCO_CLKIN	LCD_CL1_A	TIOC4B_C	—	—	GP0[18]	IP1[3:2]
GP-0-19	A18	ST1_PWM	LCD_CL2_A	TIOC4C_C	—	—	GP0[19]	IP1[5:4]
GP-0-20	A19	ST1_CLKIN	LCD_CLK_A	TIOC4D_C	—	—	GP0[20]	IP1[7:6]
GP-0-21	$\overline{\text{PRESETOUT}}$	ST_CLKOUT	—	—	—	—	GP0[21]	IP11[28]
GP-0-22	A0	ST0_CLKIN	LCD_DATA0_A	TCLKA_C	—	—	GP0[22]	IP0[1:0]
GP-0-23	A1	ST0_REQ	LCD_DATA1_A	TCLKB_C	—	—	GP0[23]	IP0[3:2]
GP-0-24	A2	ST0_SYC	LCD_DATA2_A	TCLKC_C	—	—	GP0[24]	IP0[5:4]
GP-0-25	A24	RX2_D	ST1_D[1]	—	—	—	GP0[25]	IP1[17:16]
GP-0-26	A25	TX2_D	ST1_D[2]	—	—	—	GP0[26]	IP1[19:18]
GP-0-27	D0	SD0_DAT0_A	MMC_D0_A	ST1_D[3]	NAF0_A	—	GP0[27]	IP1[22:20]
GP-0-28	D1	SD0_DAT1_A	MMC_D1_A	ST1_D[4]	NAF1_A	—	GP0[28]	IP1[25:23]
GP-0-29	D2	SD0_DAT2_A	MMC_D2_A	ST1_D[5]	NAF2_A	—	GP0[29]	IP1[28:26]
GP-0-30	D3	SD0_DAT3_A	MMC_D3_A	ST1_D[6]	NAF3_A	—	GP0[30]	IP1[31:29]
GP-0-31	D4	SD0_CD_A	MMC_D4_A	ST1_D[7]	NAF4_A	—	GP0[31]	IP2[2:0]
GP-1-0	EX_WAIT0	TCLK1_B	—	—	—	—	GP1[0]	IP3[20]
GP-1-1	DRACK0	SD1_DAT2_A	$\overline{\text{ATAG}}$	TCLK1_A	ET0_ETXD7	—	GP1[1]	IP3[29:27]
GP-1-2	DREQ0	SD1_CLK_A	ET0_TX_EN	—	—	—	GP1[2]	IP11[20:19]
GP-1-3	DACK0	SD1_DAT3_A	ET0_TX_ER	—	—	—	GP1[3]	IP11[22:21]

Selection between GPIO and Peripheral Function

GPIO (GP Set Value = 0)	Peripheral Module (GP Set Value = 1)						GPIO/ Peripheral Function Select Bit	Peripheral Function Select Bit
	Function Selected by IP Bits							
	Function 1 (IP Set Value = 0)	Function 2 (IP Set Value = 1)	Function 3 (IP Set Value = 2)	Function 4 (IP Set Value = 3)	Function 5 (IP Set Value = 4)	Function 6 (IP Set Value = 5)		
GP-1-4	D9	SD0_CMD_A	MMC_CMD_A	QIO3_A	FCLE_A	ET0_ETXD1_B	GP1[4]	IP2[16:14]
GP-1-5	D10	RSPI_MOSI_A	—	QMO / QIO0_A	FALE_A	ET0_ETXD2_B	GP1[5]	IP2[19:17]
GP-1-6	D11	RSPI_MISO_A	—	QMI / QIO1_A	$\overline{\text{FRE\_A}}$	ET0_ETXD3_B	GP1[6]	IP2[22:20]
GP-1-7	D12	—	$\overline{\text{FWE\_A}}$	ET0_ETXD5_B	—	—	GP1[7]	IP2[24:23]
GP-1-8	D13	RX2_B	—	FRB_A	ET0_ETXD6_B	—	GP1[8]	IP2[27:25]
GP-1-9	D14	TX2_B	—	—	ET0_TX_CLK_B	—	GP1[9]	IP2[30:28]
GP-1-10	D15	SCK2_B	—	—	—	—	GP1[10]	IP3[1:0]
GP-1-11	CLKOUT	—	—	—	—	—	GP1[11]	—
GP-1-12	$\overline{\text{BS}}$	—	—	—	—	—	GP1[12]	—
GP-1-13	$\overline{\text{CS0}}$	—	—	—	—	—	GP1[13]	—
GP-1-14	$\overline{\text{CS1}}$/A26	QIO3_B	—	—	—	—	GP1[14]	IP3[2]
GP-1-15	$\overline{\text{EX\_CS0}}$	—	—	—	—	—	GP1[15]	—
GP-1-16	$\overline{\text{EX\_CS1}}$	RX3_B	$\overline{\text{ATACS0}}$	QIO2_B	ET0_ETXD0	—	GP1[16]	IP3[5:3]
GP-1-17	$\overline{\text{EX\_CS2}}$	TX3_B	$\overline{\text{ATACS1}}$	QSPCLK_B	ET0_GTX_CLK_A	—	GP1[17]	IP3[8:6]
GP-1-18	$\overline{\text{EX\_CS3}}$	SD1_CD_A	$\overline{\text{ATARD}}$	QMO / QIO0_B	ET0_ETXD1_A	—	GP1[18]	IP3[11:9]
GP-1-19	$\overline{\text{EX\_CS4}}$	SD1_WP_A	$\overline{\text{ATAWR}}$	QMI / QIO1_B	ET0_ETXD2_A	—	GP1[19]	IP3[14:12]
GP-1-20	$\overline{\text{EX\_CS5}}$	SD1_CMD_A	$\overline{\text{ATADIR}}$	QSSL_B	ET0_ETXD3_A	—	GP1[20]	IP3[17:15]
GP-1-21	$\overline{\text{RD}}$	—	—	—	—	—	GP1[21]	—
GP-1-22	RD/$\overline{\text{WR}}$	TCLK0	CAN_CLK_B	ET0_ETXD4	—	—	GP1[22]	IP3[19:18]
GP-1-23	$\overline{\text{WE0}}$	—	—	—	—	—	GP1[23]	—

Selection between GPIO and Peripheral Function

GPIO (GP Set Value = 0)	Peripheral Module (GP Set Value = 1)						GPIO/	
	Function Selected by IP Bits							
	Function 1 (IP Set Value = 0)	Function 2 (IP Set Value = 1)	Function 3 (IP Set Value = 2)	Function 4 (IP Set Value = 3)	Function 5 (IP Set Value = 4)	Function 6 (IP Set Value = 5)	Peripheral Function Select Bit	Peripheral Function Select Bit
GP-1-24	$\overline{\text{WE1}}$	—	—	—	—	—	GP1[24]	—
GP-1-25	D5	SD0_WP_A	MMC_D5_A	NAF5_A	—	—	GP1[25]	IP2[4:3]
GP-1-26	EX_WAIT1	SD1_DAT0_A	DREQ2	CAN1_TX_C	ET0_LINK_C	ET0_ETXD5_A	GP1[26]	IP3[23:21]
GP-1-27	EX_WAIT2	SD1_DAT1_A	DACK2	CAN1_RX_C	ET0_MAGIC_C	ET0_ETXD6_A	GP1[27]	IP3[26:24]
GP-1-28	D6	RSPI_RSPCK_A	MMC_D6_A	QSPCLK_A	NAF6_A	—	GP1[28]	IP2[7:5]
GP-1-29	D7	RSPI_SSL_A	MMC_D7_A	QSSL_A	NAF7_A	—	GP1[29]	IP2[10:8]
GP-1-30	D8	SD0_CLK_A	MMC_CLK_A	QIO2_A	$\overline{\text{FCE\_A}}$	ET0_GTX_CLK_B	GP1[30]	IP2[13:11]
GP-1-31	DREQ1	HSPI_CLK_B	RX4_B	ET0_PHY_INT_C	ET0_TX_CLK_A	—	GP1[31]	IP11[25:23]
GP-2-0	SCIF_CLK_A	HSPI_CLK_A	VI0_CLK	RMII0_TXD0_A	ET0_ERXD4	—	GP2[0]	IP11[6:4]
GP-2-1	SCK0_A	$\overline{\text{HSPI\_CS\_A}}$	VI0_CLKENB	RMII0_TXD1_A	ET0_ERXD5	—	GP2[1]	IP11[9:7]
GP-2-2	RX0_A	HSPI_RX_A	RMII0_RXD0_A	ET0_ERXD6	—	—	GP2[2]	IP11[11:10]
GP-2-3	$\overline{\text{HCTS0\_A}}$	$\overline{\text{CTS1\_A}}$	VI0_FIELD	RMII0_RXD1_A	ET0_ERXD7	—	GP2[3]	IP4[2:0]
GP-2-4	IRQ3_A	$\overline{\text{RTS0\_A}}$	$\overline{\text{HRTS0\_B}}$	ET0_ERXD3_A	—	—	GP2[4]	IP8[29:28]
GP-2-5	DACK1	$\overline{\text{HSPI\_CS\_B}}$	TX4_B	ET0_RX_CLK_A	—	—	GP2[5]	IP11[27:26]
GP-2-6	IRQ0_A	—	HSPI_TX_B	RX3_E	ET0_ERXD0	—	GP2[6]	IP8[22:20]
GP-2-7	IRQ1_A	—	HSPI_RX_B	TX3_E	ET0_ERXD1	—	GP2[7]	IP8[25:23]
GP-2-8	TX0_A	HSPI_TX_A	—	—	—	—	GP2[8]	IP11[12]
GP-2-9	IRQ2_A	$\overline{\text{CTS0\_A}}$	$\overline{\text{HCTS0\_B}}$	ET0_ERXD2_A	—	—	GP2[9]	IP8[27:26]
GP-2-10	$\overline{\text{HRTS0\_A}}$	$\overline{\text{RTS1\_A}}$	$\overline{\text{VI0\_HSYNC}}$	RMII0_TXD_EN_A	ET0_RX_DV	—	GP2[10]	IP4[5:3]

Selection between GPIO and Peripheral Function

GPIO (GP Set Value = 0)	Peripheral Module (GP Set Value = 1)						GPIO/	
	Function Selected by IP Bits							
	Function 1 (IP Set Value = 0)	Function 2 (IP Set Value = 1)	Function 3 (IP Set Value = 2)	Function 4 (IP Set Value = 3)	Function 5 (IP Set Value = 4)	Function 6 (IP Set Value = 5)	Peripheral Function Select Bit	Peripheral Function Select Bit
GP-2-11	HSCK0_A	SCK1_A	$\overline{\text{VI0\_VSYNC}}$	RMII0_RX_ER_A	ET0_RX_ER	—	GP2[11]	IP4[8:6]
GP-2-12	HRX0_A	RX1_A	VI0_DATA0/ VI0_B0	RMII0_ CRS_DV_A	ET0_CRS	—	GP2[12]	IP4[11:9]
GP-2-13	HTX0_A	TX1_A	VI0_DATA1/ VI0_B1	RMII0_ MDC_A	ET0_COL	—	GP2[13]	IP4[14:12]
GP-2-14	—	$\overline{\text{CTS0\_B}}$	VI0_DATA2/ VI0_B2	RMII0_ MDIO_A	ET0_MDC	—	GP2[14]	IP4[17:15]
GP-2-15	—	$\overline{\text{RTS0\_B}}$	VI0_DATA3/ VI0_B3	ET0_MDIO _A	—	—	GP2[15]	IP4[19:18]
GP-2-16	—	SCK1_B	VI0_DATA4/ VI0_B4	ET0_LINK_ A	—	—	GP2[16]	IP4[21:20]
GP-2-17	—	RX1_B	VI0_DATA5/ VI0_B5	ET0_MAGI C_A	—	—	GP2[17]	IP4[23:22]
GP-2-18	—	TX1_B	VI0_DATA6/ VI0_G0	ET0_PHY_ INT_A	—	—	GP2[18]	IP4[25:24]
GP-2-19	—	$\overline{\text{CTS1\_B}}$	VI0_DATA7/ VI0_G1	—	—	—	GP2[19]	IP4[27:26]
GP-2-20	—	$\overline{\text{RTS1\_B}}$	VI0_G2	—	—	—	GP2[20]	IP4[29:28]
GP-2-21	—	SCK2_A	VI0_G3	—	—	—	GP2[21]	IP4[31:30]
GP-2-22	SD2_CLK _A	RX2_A	VI0_G4	—	ET0_RX_ CLK_B	—	GP2[22]	IP5[2:0]
GP-2-23	SD2_CMD _A	TX2_A	VI0_G5	—	ET0_ERXD2 _B	—	GP2[23]	IP5[5:3]
GP-2-24	SD2_DAT0 _A	RX3_A	VI0_R0	—	ET0_ERXD3 _B	—	GP2[24]	IP5[8:6]
GP-2-25	SD2_DAT1 _A	TX3_A	VI0_R1	—	ET0_MDIO_ B	—	GP2[25]	IP5[11:9]
GP-2-26	SD2_DAT2 _A	RX4_A	VI0_R2	—	ET0_LINK_B	—	GP2[26]	IP5[14:12]
GP-2-27	SD2_DAT3 _A	TX4_A	VI0_R3	—	ET0_MAGIC _B	—	GP2[27]	IP5[17:15]

Selection between GPIO and Peripheral Function

GPIO (GP Set Value = 0)	Peripheral Module (GP Set Value = 1) Function Selected by IP Bits						GPIO/ Peripheral Function Select Bit	Peripheral Function Select Bit
	Function 1 (IP Set Value = 0)	Function 2 (IP Set Value = 1)	Function 3 (IP Set Value = 2)	Function 4 (IP Set Value = 3)	Function 5 (IP Set Value = 4)	Function 6 (IP Set Value = 5)		
GP-2-28	SD2_CD_A	RX5_A	VI0_R4	—	ET0_PHY_INT_B	—	GP2[28]	IP5[20:18]
GP-2-29	SD2_WP_A	TX5_A	VI0_R5	—	—	—	GP2[29]	IP5[22:21]
GP-2-30	REF125CK	$\overline{\text{ADTRG}}$	RX5_C	—	—	—	GP2[30]	IP5[24:23]
GP-2-31	REF50CK	$\overline{\text{CTS1\_E}}$	$\overline{\text{HCTS0\_D}}$	—	—	—	GP2[31]	IP5[26:25]
GP-3-0	DU0_DR0	SCIF_CLK_B	HRX0_D	IETX_A	TCLKA_A	HIFD00	GP3[0]	IP6[2:0]
GP-3-1	DU0_DR1	SCK0_B	HTX0_D	IERX_A	TCLKB_A	HIFD01	GP3[1]	IP6[5:3]
GP-3-2	DU0_DR2	RX0_B	TCLKC_A	HIFD02	—	—	GP3[2]	IP6[7:6]
GP-3-3	DU0_DR3	TX0_B	TCLKD_A	HIFD03	—	—	GP3[3]	IP6[9:8]
GP-3-4	DU0_DR4	$\overline{\text{CTS0\_C}}$	TIOC0A_A	HIFD04	—	—	GP3[4]	IP6[11:10]
GP-3-5	DU0_DR5	$\overline{\text{RTS0\_C}}$	TIOC0B_A	HIFD05	—	—	GP3[5]	IP6[13:12]
GP-3-6	DU0_DR6	SCK1_C	TIOC0C_A	HIFD06	—	—	GP3[6]	IP6[15:14]
GP-3-7	DU0_DR7	RX1_C	TIOC0D_A	HIFD07	—	—	GP3[7]	IP6[17:16]
GP-3-8	DU0_DG0	TX1_C	HSCK0_D	IECLK_A	TIOC1A_A	HIFD08	GP3[8]	IP6[20:18]
GP-3-9	DU0_DG1	$\overline{\text{CTS1\_C}}$	$\overline{\text{HRTS0\_D}}$	TIOC1B_A	HIFD09	—	GP3[9]	IP6[23:21]
GP-3-10	DU0_DG2	$\overline{\text{RTS1\_C}}$	RMII0_MDC_B	TIOC2A_A	HIFD10	—	GP3[10]	IP7[2:0]
GP-3-11	DU0_DG3	SCK2_C	RMII0_MDIO_B	TIOC2B_A	HIFD11	—	GP3[11]	IP7[5:3]
GP-3-12	DU0_DG4	RX2_C	RMII0_CRS_DV_B	TIOC3A_A	HIFD12	—	GP3[12]	IP7[8:6]
GP-3-13	DU0_DG5	TX2_C	RMII0_RX_ER_B	TIOC3B_A	HIFD13	—	GP3[13]	IP7[11:9]
GP-3-14	DU0_DG6	RX3_C	RMII0_RXD0_B	TIOC3C_A	HIFD14	—	GP3[14]	IP7[14:12]
GP-3-15	DU0_DG7	TX3_C	RMII0_RXD1_B	TIOC3D_A	HIFD15	—	GP3[15]	IP7[17:15]
GP-3-16	DU0_DB0	RX4_C	RMII0_TXD_EN_B	TIOC4A_A	$\overline{\text{HIFCS}}$	—	GP3[16]	IP7[20:18]
GP-3-17	DU0_DB1	TX4_C	RMII0_TXD0_B	TIOC4B_A	HIFRS	—	GP3[17]	IP7[23:21]

Selection between GPIO and Peripheral Function

GPIO (GP Set Value = 0)	Peripheral Module (GP Set Value = 1) Function Selected by IP Bits						GPIO/ Peripheral Function Select Bit	Peripheral Function Select Bit
	Function 1 (IP Set Value = 0)	Function 2 (IP Set Value = 1)	Function 3 (IP Set Value = 2)	Function 4 (IP Set Value = 3)	Function 5 (IP Set Value = 4)	Function 6 (IP Set Value = 5)		
GP-3-18	DU0_DB2	RX5_B	RMII0_TXD1_B	TIOC4C_A	$\overline{\text{HIFWR}}$	—	GP3[18]	IP7[26:24]
GP-3-19	DU0_DB3	TX5_B	TIOC4D_A	$\overline{\text{HIFRD}}$	—	—	GP3[19]	IP7[28:27]
GP-3-20	DU0_DB4	SD2_CLK_B	$\overline{\text{HIFINT}}$	—	—	—	GP3[20]	IP7[30:29]
GP-3-21	DU0_DB5	SD2_CMD_B	HIFDREQ	—	—	—	GP3[21]	IP8[1:0]
GP-3-22	DU0_DB6	SD2_DAT0_B	HIFRDY	—	—	—	GP3[22]	IP8[3:2]
GP-3-23	DU0_DB7	SD2_DAT1_B	SSI_SCK0_B	HIFEBL_B	—	—	GP3[23]	IP8[5:4]
GP-3-24	DU0_DOTCLKIN	SD2_DAT2_B	$\overline{\text{HSPI\_CS\_C}}$	SSI_WS0_B	—	—	GP3[24]	IP8[7:6]
GP-3-25	DU0_DOTCLKOUT	SD2_DAT3_B	HSPI_CLK_C	SSI_SDATA0_B	—	—	GP3[25]	IP8[9:8]
GP-3-26	DU0_EXHSYNC/DU0_HSYNC	SD2_CD_B	HSPI_TX_C	SSI_SCK1_B	—	—	GP3[26]	IP8[11:10]
GP-3-27	DU0_EXVSYNC/DU0_VSYNC	SD2_WP_B	HSPI_RX_C	SSI_WS1_B	—	—	GP3[27]	IP8[13:12]
GP-3-28	DU0_EXODDF/DU0_ODDF	CAN0_RX_B	HSCK0_B	SSI_SDATA1_B	—	—	GP3[28]	IP8[15:14]
GP-3-29	DU0_DISP	CAN0_TX_B	HRX0_B	AUDIO_CLKA_B	—	—	GP3[29]	IP8[17:16]
GP-3-30	DU0_CDE	HTX0_B	AUDIO_CLKB_B	LCD_VCPWC_B	—	—	GP3[30]	IP8[19:18]
GP-3-31	VI1_CLK_A	—	NAF0_B	LCD_DATA0_B	—	—	GP3[31]	IP9[1:0]
GP-4-0	SSI_SCK0_A	TIOC1A_B	LCD_DATA9_B	—	—	—	GP4[0]	IP9[19:18]
GP-4-1	SSI_WS0_A	TIOC1B_B	LCD_DATA10_B	—	—	—	GP4[1]	IP9[21:20]
GP-4-2	SSI_SDATA0_A	VI1_0_B	TIOC2A_B	LCD_DATA11_B	—	—	GP4[2]	IP9[23:22]

Selection between GPIO and Peripheral Function

GPIO (GP Set Value = 0)	Peripheral Module (GP Set Value = 1) Function Selected by IP Bits						GPIO/ Peripheral Function Select Bit	Peripheral Function Select Bit
	Function 1 (IP Set Value = 0)	Function 2 (IP Set Value = 1)	Function 3 (IP Set Value = 2)	Function 4 (IP Set Value = 3)	Function 5 (IP Set Value = 4)	Function 6 (IP Set Value = 5)		
GP-4-3	SSI_SCK1_A	VI1_1_B	TIOC2B_B	LCD_DATA12_B	—	—	GP4[3]	IP9[25:24]
GP-4-4	VI1_4_A	—	NAF5_B	LCD_DATA5_B	—	—	GP4[4]	IP9[11:10]
GP-4-5	VI1_5_A	—	NAF6_B	LCD_DATA6_B	—	—	GP4[5]	IP9[13:12]
GP-4-6	VI1_6_A	—	NAF7_B	LCD_DATA7_B	—	—	GP4[6]	IP9[15:14]
GP-4-7	VI1_7_A	$\overline{\text{FCE\_B}}$	LCD_DATA8_B	—	—	—	GP4[7]	IP9[17:16]
GP-4-8	VI1_0_A	—	NAF1_B	LCD_DATA1_B	—	—	GP4[8]	IP9[3:2]
GP-4-9	VI1_1_A	—	NAF2_B	LCD_DATA2_B	—	—	GP4[9]	IP9[5:4]
GP-4-10	VI1_2_A	—	NAF3_B	LCD_DATA3_B	—	—	GP4[10]	IP9[7:6]
GP-4-11	VI1_3_A	—	NAF4_B	LCD_DATA4_B	—	—	GP4[11]	IP9[9:8]
GP-4-12	SSI_WS1_A	VI1_2_B	LCD_DATA13_B	—	—	—	GP4[12]	IP9[27:26]
GP-4-13	SSI_SDATA1_A	VI1_3_B	LCD_DATA14_B	—	—	—	GP4[13]	IP9[29:28]
GP-4-14	SSI_SCK23	VI1_4_B	RX1_D	FCLE_B	LCD_DATA15_B	—	GP4[14]	IP10[2:0]
GP-4-15	SSI_WS23	VI1_5_B	TX1_D	HSCK0_C	FALE_B	LCD_DON_B	GP4[15]	IP10[5:3]
GP-4-16	SSI_SDATA2	VI1_6_B	—	HRX0_C	$\overline{\text{FRE\_B}}$	LCD_CL1_B	GP4[16]	IP10[8:6]
GP-4-17	SSI_SDATA3	VI1_7_B	—	HTX0_C	$\overline{\text{FWE\_B}}$	LCD_CL2_B	GP4[17]	IP10[11:9]
GP-4-18	AUDIO_CLKA_A	VI1_CLK_B	SCK1_D	IECLK_B	LCD_FLM_B	—	GP4[18]	IP10[14:12]

Selection between GPIO and Peripheral Function

GPIO (GP Set Value = 0)	Peripheral Module (GP Set Value = 1)						GPIO/ Peripheral Function Select Bit	Peripheral Function Select Bit
	Function Selected by IP Bits							
	Function 1 (IP Set Value = 0)	Function 2 (IP Set Value = 1)	Function 3 (IP Set Value = 2)	Function 4 (IP Set Value = 3)	Function 5 (IP Set Value = 4)	Function 6 (IP Set Value = 5)		
GP-4-19	AUDIO_CLKB_A	LCD_CLK_B	—	—	—	—	GP4[19]	IP10[15]
GP-4-20	AUDIO_CLKC	SCK1_E	—	$\overline{\text{HCTS0\_C}}$	FRB_B	LCD_VEPWC_B	GP4[20]	IP10[18:16]
GP-4-21	AUDIO_CLKOUT	TX1_E	—	$\overline{\text{HRTS0\_C}}$	—	LCD_M_DISP_B	GP4[21]	IP10[21:19]
GP-4-22	SCL1	SCIF_CLK_C	—	—	—	—	GP4[22]	IP11[0]
GP-4-23	SDA1	RX1_E	—	—	—	—	GP4[23]	IP11[1]
GP-4-24	SCL0	—	—	—	—	—	GP4[24]	—
GP-4-25	SDA0	HIFEBL_A	—	—	—	—	GP4[25]	IP11[2]
GP-4-26	PENC0	—	—	—	—	—	GP4[26]	—
GP-4-27	PENC1	TX3_D	CAN1_TX_B	TX5_D	IETX_B	—	GP4[27]	IP11[15:13]
GP-4-28	USB_OVC0	—	—	—	—	—	GP4[28]	—
GP-4-29	USB_OVC1	RX3_D	CAN1_RX_B	RX5_D	IERX_B	—	GP4[29]	IP11[18:16]
GP-4-30	CAN_CLK_A	RX4_D	—	—	—	—	GP4[30]	IP10[22]
GP-4-31	CAN0_TX_A	TX4_D	MLB_CLK	—	—	—	GP4[31]	IP10[24:23]
GP-5-0	CAN1_RX_A	IRQ1_B	—	—	—	—	GP5[0]	IP10[25]
GP-5-1	SDSELF	$\overline{\text{RTS1\_E}}$	—	—	—	—	GP5[1]	IP11[3]
GP-5-2	IRQ2_B	—	—	—	—	—	GP5[2]	—
GP-5-3	IRQ3_B	—	—	—	—	—	GP5[3]	—
GP-5-10	CAN0_RX_A	IRQ0_B	MLB_SIG	—	—	—	GP5[10]	IP10[27:26]
GP-5-11	CAN1_TX_A	TX5_C	MLB_DAT	—	—	—	GP5[11]	IP10[29:28]

[Legend] —: Setting invalid

37.2.20 Module Select Register (MOD_SEL)

Function: MOD_SEL selects the group for LSI pins with multiplexed pin functions.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	iebus_sel	rqspi_sel	vin_sel	hif_sel	rspi_sel	lcd_sel	get_et0_ctl_sel[1]	get_et0_ctl_sel[0]	get_et0_sel	get_rmii_sel	tmu_sel	hspi0_sel[1]	hspi0_sel[0]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	hscif_sel[1]	hscif_sel[0]	rcan_clk_sel	rcan1_sel[1]	rcan1_sel[0]	rcan0_sel	sdhi2_sel	sdhi1_sel	sdhi0_sel	ssi1_sel	ssi0_sel	audiob_sel	audioa_sel	flctl_sel	mmc_sel	intc_sel
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Initial Value	R/W	Description
31 to 29	All 0	R	These bits are always read as 0. The write value should always be 0.
28 to 0	H'0000 0000	R/W	These bits select multiplexed pin functions as indicated in the table below.

Note: To enable this register to be set, appropriately set the LSI multiplexed pin setting mask register (PMMR) immediately before setting this register.

Each input or input/output signal of the INTC, FLCTL, RCAN (channels 0 and 1, CLK), SDHI (channel 2), HSCIF (channel 0), SSS (channels 0 and 1, AUDIO_CLKA/ AUDIO_CLKB), HSPI, TMU, GETHER (RMII/ET0), LCDC, HIF, VIN (channel 1), RQSPI, and IEBus is assigned to two or more groups of pins. Select one from among these groups when using these signals. Do not use the module pins in the non-selected group; if a module pin in the non-selected group is used, correct operation is not guaranteed.

For some modules, however, although the output signals are assigned to two or more groups of pins, there is no bit for selecting the group because two or more groups of pins are output. Use the corresponding peripheral function select register to select one from among these pins for each output signal.

Bit Name	Function 1 (Set Value = H'0)	Function 2 (Set Value = H'1)	Function 3 (Set Value = H'2)	Function 4 (Set Value = H'3)
intc_sel	Selects the group A side for INTC	Selects the group B side for INTC	—	—
mmc_sel	Selects the group A side for MMC	—	—	—
flctl_sel	Selects the group A side for FLCTL	Selects the group B side for FLCTL	—	—
audioa_sel	Selects the group A side for AUDIO_CLKA	Selects the group B side for AUDIO_CLKA	—	—
audiob_sel	Selects the group A side for AUDIO_CLKB	Selects the group B side for AUDIO_CLKB	—	—
ssi0_sel	Selects the group A side for SSI0	Selects the group B side for SSI0	—	—
ssi1_sel	Selects the group A side for SSI1	Selects the group B side for SSI1	—	—
sdhi0_sel	Selects the group A side for SDHI channel 0	—	—	—
sdhi1_sel	Selects the group A side for SDHI channel 1	—	—	—
sdhi2_sel	Selects the group A side for SDHI channel 2	Selects the group B side for SDHI channel 2	—	—
rcan0_sel	Selects the group A side for RCAN0	Selects the group B side for RCAN0	—	—
rcan1_sel[1:0]	Selects the group A side for RCAN1	Selects the group B side for RCAN1	Selects the group C side for RCAN1	—
rcan_clk_sel	Selects the group A side for RCAN CLK	Selects the group B side for RCAN CLK	—	—
hscif_sel[1:0]	Selects the group A side for HSCIF	Selects the group B side for HSCIF	Selects the group C side for HSCIF	Selects the group D side for HSCIF
hspi0_sel[1:0]	Selects the group A side for HSPI	Selects the group B side for HSPI	Selects the group C side for HSPI	—
tmu_sel	Selects the group A side for TMU	Selects the group B side for TMU	—	—
get_rmii_sel	Selects the group A side for GETHER (RMII)	Selects the group B side for GETHER (RMII)	—	—
get_et0_sel	Selects the group A side for GETHER (ET0)*1	Selects the group B side for GETHER (ET0)*1	—	—
get_et0_ctl_sel[1:0]	Selects the group A side for GETHER (ET0)*2	Selects the group B side for GETHER (ET0)*2	Selects the group C side for GETHER (ET0)*2	—
lcd_sel	Selects the group A side for LCDC	Selects the group B side for LCDC	—	—
rspi_sel	Selects the group A side for RSPI	—	—	—
hif_sel	Selects the group A side for HIF	Selects the group B side for HIF	—	—

Bit Name	Function 1 (Set Value = H'0)	Function 2 (Set Value = H'1)	Function 3 (Set Value = H'2)	Function 4 (Set Value = H'3)
vin_sel	Selects the group A side for VIN1	Selects the group B side for VIN1	—	—
rqspi_sel	Selects the group A side for RQSPI	Selects the group B side for RQSPI	—	—
iebus_sel	Selects the group A side for IEBus	Selects the group B side for IEBus	—	—

[Legend] —: Setting invalid or impossible

Notes: 1. Pins ET0_TX_CLK, ET0_RX_CLK, ET0_ERXD2, ET0_ERXD3, and ET0_MDIO
2. Pins ET0_LINK and ET0_PHY_INT

37.2.21 Module Select Register 2 (MOD_SEL2)

Function: MOD_SEL2 selects the group for multiple LSI pins with multiplexed pin functions.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	mtuclk_sel	mtu4_sel	mtu3_sel	mtu2_sel[1]	mtu2_sel[0]	mtu1_sel[1]	mtu1_sel[0]	mtu0_sel
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	scif5_sel[1]	scif5_sel[0]	scif4_sel[1]	scif4_sel[0]	scif3_sel[2]	scif3_sel[1]	scif3_sel[0]	scif2_sel[1]	scif2_sel[0]	scif1_sel[2]	scif1_sel[1]	scif1_sel[0]	scif0_sel[1]	scif0_sel[0]	scif_clk_sel[1]	scif_clk_sel[0]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Initial Value	R/W	Description
31 to 24	All 0	R	These bits are always read as 0. The write value should always be 0.
23 to 0	H'00 0000	R/W	These bits select multiplexed pin functions as indicated in the table below.

Note: To enable this register to be set, appropriately set the LSI multiplexed pin setting mask register (PMMR) immediately before setting this register.

Each input or input/output signal of the SCIF (CLK, channels 1 to 5), MTU2 (channels 0 to 4, CLK) is assigned to two or more groups of pins. Select one from among these groups when using these signals. Do not use the module pins in the non-selected group; if a module pin in the non-selected group is used, correct operation is not guaranteed.

For some modules, however, although the output signals are assigned to two or more groups of pins, there is no bit for selecting the group because two or more groups of pins are output. Use the corresponding peripheral function select register to select one from among these pins for each output signal.

Bit Name	Function 1 (Set Value = H'0)	Function 2 (Set Value = H'1)	Function 3 (Set Value = H'2)	Function 4 (Set Value = H'3)	Function 5 (Set Value = H'4)
scif_clk_sel[1:0]	Selects the group A side for SCIF_CLK	Selects the group B side for SCIF_CLK	Selects the group C side for SCIF_CLK	—	—
scif0_sel[1:0]	Selects the group A side for SCIF0	Selects the group B side for SCIF0	Selects the group C side for SCIF0	—	—
scif1_sel[2:0]	Selects the group A side for SCIF1	Selects the group B side for SCIF1	Selects the group C side for SCIF1	Selects the group D side for SCIF1	Selects the group E side for SCIF1
scif2_sel[1:0]	Selects the group A side for SCIF2	Selects the group B side for SCIF2	Selects the group C side for SCIF2	Selects the group D side for SCIF2	—
scif3_sel[2:0]	Selects the group A side for SCIF3	Selects the group B side for SCIF3	Selects the group C side for SCIF3	Selects the group D side for SCIF3	Selects the group E side for SCIF3
scif4_sel[1:0]	Selects the group A side for SCIF4	Selects the group B side for SCIF4	Selects the group C side for SCIF4	Selects the group D side for SCIF4	—
scif5_sel[1:0]	Selects the group A side for SCIF5	Selects the group B side for SCIF5	Selects the group C side for SCIF5	Selects the group D side for SCIF5	—
mtu0_sel	Selects the group A side for MTU2 channel 0	Selects the group C side for MTU2 channel 0	—	—	—
mtu1_sel[1:0]	Selects the group A side for MTU2 channel 1	Selects the group B side for MTU2 channel 1	Selects the group C side for MTU2 channel 1	—	—
mtu2_sel[1:0]	Selects the group A side for MTU2 channel 2	Selects the group B side for MTU2 channel 2	Selects the group C side for MTU2 channel 2	—	—
mtu3_sel	Selects the group A side for MTU2 channel 3	Selects the group C side for MTU2 channel 3	—	—	—
mtu4_sel	Selects the group A side for MTU2 channel 4	Selects the group C side for MTU2 channel 4	—	—	—
mtuclk_sel	Selects the group A side for MTU2 CLK	Selects the group C side for MTU2 CLK	—	—	—

[Legend] —: Setting invalid or impossible

Table 37.4 List of Module Pins Multiplexed with Several LSI Pins

Target Pin	Group A	Group B	Group C	Group D	Group E
IRQ0	Selects IRQ0_A pin	Selects IRQ0 which is multiplexed with CAN0_RX_A pin	—	—	—
IRQ1	Selects IRQ1_A pin	Selects IRQ1 which is multiplexed with CAN1_RX_A pin	—	—	—
IRQ2	Selects IRQ2_A pin	Selects IRQ2 which is multiplexed with AN0 pin	—	—	—
IRQ3	Selects IRQ3_A pin	Selects IRQ3 which is multiplexed with AN1 pin	—	—	—
MMC_D0	Selects MMC_D0 which is multiplexed with D0 pin	—	—	—	—
MMC_D1	Selects MMC_D1 which is multiplexed with D1 pin	—	—	—	—
MMC_D2	Selects MMC_D2 which is multiplexed with D2 pin	—	—	—	—
MMC_D3	Selects MMC_D3 which is multiplexed with D3 pin	—	—	—	—
MMC_D4	Selects MMC_D4 which is multiplexed with D4 pin	—	—	—	—
MMC_D5	Selects MMC_D5 which is multiplexed with D5 pin	—	—	—	—
MMC_D6	Selects MMC_D6 which is multiplexed with D6 pin	—	—	—	—
MMC_D7	Selects MMC_D7 which is multiplexed with D7 pin	—	—	—	—
MMC_CMD	Selects MMC_CMD which is multiplexed with D9 pin	—	—	—	—

Target Pin	Group A	Group B	Group C	Group D	Group E
NAF0	Selects NAF0 which is multiplexed with D0 pin	Selects NAF0 which is multiplexed with VI1_CLK_A pin	—	—	—
NAF1	Selects NAF1 which is multiplexed with D1 pin	Selects NAF1 which is multiplexed with VI1_0_A pin	—	—	—
NAF2	Selects NAF2 which is multiplexed with D2 pin	Selects NAF2 which is multiplexed with VI1_1_A pin	—	—	—
NAF3	Selects NAF3 which is multiplexed with D3 pin	Selects NAF3 which is multiplexed with VI1_2_A pin	—	—	—
NAF4	Selects NAF4 which is multiplexed with D4 pin	Selects NAF4 which is multiplexed with VI1_3_A pin	—	—	—
NAF5	Selects NAF5 which is multiplexed with D5 pin	Selects NAF5 which is multiplexed with VI1_4_A pin	—	—	—
NAF6	Selects NAF6 which is multiplexed with D6 pin	Selects NAF6 which is multiplexed with VI1_5_A pin	—	—	—
NAF7	Selects NAF7 which is multiplexed with D7 pin	Selects NAF7 which is multiplexed with VI1_6_A pin	—	—	—
FRB	Selects FRB which is multiplexed with D13 pin	Selects FRB which is multiplexed with AUDIO_CLKC pin	—	—	—
AUDIO_CLKA	Selects AUDIO_CLKA_A pin	Selects AUDIO_CLKA which is multiplexed with DU0_DISP pin	—	—	—
AUDIO_CLKB	Selects AUDIO_CLKB_A pin	Selects AUDIO_CLKB which is multiplexed with DU0_CDE pin	—	—	—
SSI_SCK0	Selects SSI_SCK0_A pin	Selects SSI_SCK0 which is multiplexed with DU0_DB7 pin	—	—	—
SSI_WS0	Selects SSI_WS0_A pin	Selects SSI_ WS0 which is multiplexed with DU0_DOTCLKIN pin	—	—	—

Target Pin	Group A	Group B	Group C	Group D	Group E
SSI_SDATA0	Selects SSI_SDATA0_A pin	Selects SSI_ SDATA0 which is multiplexed with DU0_DOTCLKOUT pin	—	—	—
SSI_SCK1	Selects SSI_SCK1_A pin	Selects SSI_SCK1 which is multiplexed with DU0_EXHSYNC/ DU0_HSYNC pin	—	—	—
SSI_WS1	Selects SSI_WS1_A pin	Selects SSI_ WS1 which is multiplexed with DU0_EXVSYNC/ DU0_VSYNC pin	—	—	—
SSI_SDATA1	Selects SSI_SDATA1_A pin	Selects SSI_SDATA1 which is multiplexed with DU0_EXODDF/ DU0_ODDF pin	—	—	—
SD0_CMD	Selects SD0_CMD which is multiplexed with D9 pin	—	—	—	—
SD0_DAT0	Selects SD0_DAT0 which is multiplexed with D0 pin	—	—	—	—
SD0_DAT1	Selects SD0_DAT1 which is multiplexed with D1 pin	—	—	—	—
SD0_DAT2	Selects SD0_DAT2 which is multiplexed with D2 pin	—	—	—	—
SD0_DAT3	Selects SD0_DAT 3 which is multiplexed with D3 pin	—	—	—	—
SD0_CD	Selects SD0_CD which is multiplexed with D4 pin	—	—	—	—
SD0_WP	Selects SD0_WP which is multiplexed with D5 pin	—	—	—	—
SD1_CMD	Selects SD1_CMD which is multiplexed with $\overline{\text{EX\_CS5}}$ pin	—	—	—	—

Target Pin	Group A	Group B	Group C	Group D	Group E
SD1_DAT0	Selects SD1_DAT0 which is multiplexed with EX_WAIT1 pin	—	—	—	—
SD1_DAT1	Selects SD1_DAT1 which is multiplexed with EX_WAIT2 pin	—	—	—	—
SD1_DAT2	Selects SD1_DAT2 which is multiplexed with DRACK0 pin	—	—	—	—
SD1_DAT3	Selects SD1_DAT 3 which is multiplexed with DACK0 pin	—	—	—	—
SD1_CD	Selects SD1_CD which is multiplexed with $\overline{\text{EX\_CS3}}$ pin	—	—	—	—
SD1_WP	Selects SD1_WP which is multiplexed with $\overline{\text{EX\_CS4}}$ pin	—	—	—	—
SD2_CMD	Selects SD2_CMD_A pin	Selects SD2_CMD which is multiplexed with DU0_DB5 pin	—	—	—
SD2_DAT0	Selects SD2_DAT0_A pin	Selects SD2_DAT0 which is multiplexed with DU0_DB6 pin	—	—	—
SD2_DAT1	Selects SD2_DAT1_A pin	Selects SD2_DAT1 which is multiplexed with DU0_DB7 pin	—	—	—
SD2_DAT2	Selects SD2_DAT2_A pin	Selects SD2_DAT2 which is multiplexed with DU0_DOTCLKIN pin	—	—	—
SD2_DAT3	Selects SD2_DAT3_A pin	Selects SD2_DAT3 which is multiplexed with DU0_DOTCLKOUT pin	—	—	—
SD2_CD	Selects SD2_CD_A pin	Selects SD2_CD which is multiplexed with DU0_EXHSYNC/DU0_HSYNC pin	—	—	—

Target Pin	Group A	Group B	Group C	Group D	Group E
SD2_WP	Selects SD2_WP_A pin	Selects SD2_WP which is multiplexed with DU0_EXVSYNC/DU0_VSYNC pin	—	—	—
CAN0_RX	Selects CAN0_RX_A pin	Selects CAN0_RX which is multiplexed with DU0_EXODDF/DU0_ODDF pin	—	—	—
CAN1_RX	Selects CAN1_RX_A pin	Selects CAN1_RX which is multiplexed with USB_OVC1 pin	Selects CAN1_RX which is multiplexed with EX_WAIT2 pin	—	—
CAN_CLK	Selects CAN_CLK_A pin	Selects CAN_CLK which is multiplexed with RD/$\overline{WR}$ pin	—	—	—
$\overline{HCTS0}$	Selects $\overline{HCTS0}$_A pin	Selects $\overline{HCTS0}$ which is multiplexed with IRQ2_A pin	Selects $\overline{HCTS0}$ which is multiplexed with SSI_WS23 pin	Selects $\overline{HCTS0}$ which is multiplexed with REF50CK pin	—
$\overline{HRTS0}$	Selects $\overline{HRTS0}$_A pin	Selects $\overline{HRTS0}$ which is multiplexed with IRQ3_A pin	Selects $\overline{HRTS0}$ which is multiplexed with SSI_SDATA2 pin	Selects $\overline{HRTS0}$ which is multiplexed with DU0_DG1 pin	—
HSCK0	Selects HSCK0_A pin	Selects HSCK0 which is multiplexed with DU0_EXODDF/DU0_ODDF pin	Selects HSCK0 which is multiplexed with AUDIO_CLKC pin	Selects HSCK0 which is multiplexed with DU0_DG0 pin	—
HRX0	Selects HRX0_A pin	Selects HRX0 which is multiplexed with DU0_DISP pin	Selects HRX0 which is multiplexed with AUDIO_CLKOUT pin	Selects HRX0 which is multiplexed with DU0_DR0 pin	—
HSPI_CLK	Selects HSPI_CLK which is multiplexed with SCIF_CLK_A pin	Selects HSPI_CLK which is multiplexed with DREQ1 pin	Selects HSPI_CLK which is multiplexed with DU0_DOTCLKOUT pin	—	—

Target Pin	Group A	Group B	Group C	Group D	Group E
$\overline{\text{HSPI\_CS}}$	Selects $\overline{\text{HSPI\_CS}}$ which is multiplexed with SCK0_A pin	Selects $\overline{\text{HSPI\_CS}}$ which is multiplexed with DACK1 pin	Selects $\overline{\text{HSPI\_CS}}$ which is multiplexed with DU0_DOTCLKIN pin	—	—
HSPI_RX	Selects HSPI_RX which is multiplexed with RX0_A pin	Selects HSPI_RX which is multiplexed with IRQ1_A pin	Selects HSPI_RX which is multiplexed with DU0_EXVSYNC/DU0_VSYNC pin	—	—
TCLK1	Selects TCLK1 which is multiplexed with DRACK0 pin	Selects TCLK1 which is multiplexed with EX_WAIT0 pin	—	—	—
RMII0_RXD0	Selects RMII0_RXD0 which is multiplexed with RX0_A pin	Selects RMII0_RXD0 which is multiplexed with DU0_DG6 pin	—	—	—
RMII0_RXD1	Selects RMII0_RXD1 which is multiplexed with $\overline{\text{HCTS0\_A}}$ pin	Selects RMII0_RXD1 which is multiplexed with DU0_DG7 pin	—	—	—
RMII0_MDIO	Selects RMII0_MDIO which is multiplexed with $\overline{\text{HRTS0\_A}}$ pin	Selects RMII0_MDIO which is multiplexed with DU0_DG3 pin	—	—	—
RMII0_RX_ER	Selects RMII0_RX_ER which is multiplexed with HSCK0_A pin	Selects RMII0_RX_ER which is multiplexed with DU0_DG5 pin	—	—	—
RMII0_CRS_DV	Selects RMII0_CRS_DV which is multiplexed with HRX0_A pin	Selects RMII0_CRS_DV which is multiplexed with DU0_DG4 pin	—	—	—
ET0_TX_CLK	Selects ET0_TX_CLK which is multiplexed with DREQ1 pin	Selects ET0_TX_CLK which is multiplexed with D14 pin	—	—	—
ET0_RX_CLK	Selects ET0_RX_CLK which is multiplexed with DACK1 pin	Selects ET0_RX_CLK which is multiplexed with SD2_CLK_A pin	—	—	—

Target Pin	Group A	Group B	Group C	Group D	Group E
ET0_ERXD2	Selects ET0_ERXD2 which is multiplexed with IRQ2_A pin	Selects ET0_ERXD2 which is multiplexed with SD2_CLK_A pin	—	—	—
ET0_ERXD3	Selects ET0_ERXD3 which is multiplexed with IRQ3_A pin	Selects ET0_ERXD3 which is multiplexed with SD2_DAT0_A pin	—	—	—
ET0_MDIO	Selects ET0_MDIO which is multiplexed with $\overline{\text{RTS0\_B}}$ pin	Selects ET0_MDIO which is multiplexed with SD2_DAT1_A pin	—	—	—
ET0_LINK	Selects ET0_LINK which is multiplexed with SCK1_B pin	Selects ET0_LINK which is multiplexed with SD2_DAT2_A pin	Selects ET0_LINK which is multiplexed with EX_WAIT1 pin	—	—
ET0_PHY_INT	Selects ET0_PHY_INT which is multiplexed with TX1_B pin	Selects ET0_PHY_INT which is multiplexed with SD2_CD_A pin	Selects ET0_PHY_INT which is multiplexed with DREQ1 pin	—	—
LCD_CLK	Selects LCD_CLK which is multiplexed with A19 pin	Selects LCD_CLK which is multiplexed with AUDIO_CLKB_A pin	—	—	—
RSPI_RSPCK	Selects RSPI_RSPCK which is multiplexed with D6 pin	—	—	—	—
RSPI_SSL	Selects RSPI_SSL which is multiplexed with D7 pin	—	—	—	—
RSPI_MOSI	Selects RSPI_MOSI which is multiplexed with D10 pin	—	—	—	—
RSPI_MISO	Selects RSPI_MISO which is multiplexed with D11 pin	—	—	—	—
HIFEBL	Selects HIFEBL which is multiplexed with SDA0 pin	Selects HIFEBL which is multiplexed with DU0_DB7 pin	—	—	—

Target Pin	Group A	Group B	Group C	Group D	Group E
VI1_CLK	Selects VI1_CLK_A pin	Selects VI1_CLK which is multiplexed with AUDIO_CLKA_A pin	—	—	—
VI1_0	Selects VI1_0_A pin	Selects VI1_0 which is multiplexed with SSI_SDATA0_A pin	—	—	—
VI1_1	Selects VI1_1_A pin	Selects VI1_1 which is multiplexed with SSI_SCK1_A pin	—	—	—
VI1_2	Selects VI1_2_A pin	Selects VI1_2 which is multiplexed with SSI_WS1_A pin	—	—	—
VI1_3	Selects VI1_3_A pin	Selects VI1_3 which is multiplexed with SSI_SDATA1_A pin	—	—	—
VI1_4	Selects VI1_4_A pin	Selects VI1_4 which is multiplexed with SSI_SCK23 pin	—	—	—
VI1_5	Selects VI1_5_A pin	Selects VI1_5 which is multiplexed with SSI_WS23 pin	—	—	—
VI1_6	Selects VI1_6_A pin	Selects VI1_6 which is multiplexed with SSI_SDATA2 pin	—	—	—
VI1_7	Selects VI1_7_A pin	Selects VI1_7 which is multiplexed with SSI_SDATA3 pin	—	—	—
QSPCLK	Selects QSPCLK which is multiplexed with D6 pin	Selects QSPCLK which is multiplexed with $\overline{\text{EX\_CS2}}$ pin	—	—	—
QSSL	Selects QSSL which is multiplexed with D7 pin	Selects QSSL which is multiplexed with $\overline{\text{EX\_CS5}}$ pin	—	—	—
QIO2	Selects QIO2 which is multiplexed with D8 pin	Selects QIO2 which is multiplexed with $\overline{\text{EX\_CS1}}$ pin	—	—	—
QIO3	Selects QIO3 which is multiplexed with D9 pin	Selects QIO3 which is multiplexed with $\overline{\text{CS1}}$/A26 pin	—	—	—

Target Pin	Group A	Group B	Group C	Group D	Group E
QMO / QIO0	Selects QMO / QIO0 which is multiplexed with D10 pin	Selects QMO / QIO0 which is multiplexed with $\overline{\text{EX\_CS3}}$ pin	—	—	—
QMI / QIO1	Selects QMI / QIO1 which is multiplexed with D11 pin	Selects QMI / QIO1 which is multiplexed with $\overline{\text{EX\_CS4}}$ pin	—	—	—
IERX	Selects IERX which is multiplexed with DU0_DR1 pin	Selects IERX which is multiplexed with USB_OVC1 pin	—	—	—
IECLK	Selects IECLK which is multiplexed with DU0_DG0 pin	Selects IECLK which is multiplexed with AUDIO_CLKA_A pin	—	—	—
SCIF_CLK	Selects SCIF_CLK_A pin	Selects SCIF_CLK pin which is multiplexed with DU0_DR0 pin	Selects SCIF_CLK which is multiplexed with SCL1 pin	—	—
SCK0	Selects SCK0_A pin	Selects SCK0 which is multiplexed with DU0_DR1 pin	—	—	—
RX0	Selects RX0_A pin	Selects RX0 which is multiplexed with DU0_DR2 pin	—	—	—
$\overline{\text{CTS0}}$	Selects $\overline{\text{CTS0}}$ which is multiplexed with IRQ2_A pin	Selects $\overline{\text{CTS0\_B}}$ pin	Selects $\overline{\text{CTS0}}$ which is multiplexed with DU0_DR4 pin	—	—
$\overline{\text{RTS0}}$	Selects $\overline{\text{RTS0}}$ which is multiplexed with IRQ3_A pin	Selects $\overline{\text{RTS0\_B}}$ pin	Selects $\overline{\text{RTS0}}$ which is multiplexed with DU0_DR5 pin	—	—
SCK1	Selects SCK1 which is multiplexed with HSCK0_A pin	Selects SCK1_B pin	Selects SCK1 which is multiplexed with DU0_DR6 pin	Selects SCK1 which is multiplexed with AUDIO_CLKA_A pin	Selects SCK1 which is multiplexed with AUDIO_CLKC pin
RX1	Selects RX1 which is multiplexed with HRX0_A pin	Selects RX1_B pin	Selects RX1 which is multiplexed with DU0_DR7 pin	Selects RX1 which is multiplexed with SSI_SCK23 pin	Selects RX1 which is multiplexed with SDA1 pin

Target Pin	Group A	Group B	Group C	Group D	Group E
$\overline{\text{CTS1}}$	Selects $\overline{\text{CTS1}}$ which is multiplexed with $\overline{\text{HCTS0\_A}}$ pin	Selects $\overline{\text{CTS1\_B}}$ pin	Selects $\overline{\text{CTS1}}$ which is multiplexed with DU0_DG1 pin	—	Selects $\overline{\text{CTS1}}$ which is multiplexed with REF50CK pin
$\overline{\text{RTS1}}$	Selects $\overline{\text{RTS1}}$ which is multiplexed with $\overline{\text{HRTS0\_A}}$ pin	Selects $\overline{\text{RTS1\_B}}$ pin	Selects $\overline{\text{RTS1}}$ which is multiplexed with DU0_DG2 pin	—	Selects $\overline{\text{RTS1}}$ which is multiplexed with SDSELF pin
SCK2	Selects SCK2_A pin	Selects SCK2 which is multiplexed with D15 pin	Selects SCK2 which is multiplexed with DU0_DG3 pin	—	—
RX2	Selects RX2 which is multiplexed with SD2_CLK_A pin	Selects RX2 which is multiplexed with D13 pin	Selects RX2 which is multiplexed with DU0_DG4 pin	Selects RX2 which is multiplexed with A24 pin	—
RX3	Selects RX3 which is multiplexed with SD2_DAT0_A pin	Selects RX3 which is multiplexed with $\overline{\text{EX\_CS1}}$ pin	Selects RX3 which is multiplexed with DU0_DG6 pin	Selects RX3 which is multiplexed with USB_OVC1 pin	Selects RX3 which is multiplexed with IRQ0_A pin
RX4	Selects RX4 which is multiplexed with SD2_DAT2_A pin	Selects RX4 which is multiplexed with DREQ1 pin	Selects RX4 which is multiplexed with DU0_DB0 pin	Selects RX4 which is multiplexed with CAN_CLK_A pin	—
RX5	Selects RX5 which is multiplexed with SD2_CD_A pin	Selects RX5 which is multiplexed with DU0_DB2 pin	Selects RX5 which is multiplexed with REF125CK pin	Selects RX5 which is multiplexed with USB_OVC1 pin	—
TCLKA	Selects TCLKA which is multiplexed with DU0_DR0 pin	Selects TCLKA which is multiplexed with A0 pin	—	—	—
TCLKB	Selects TCLKB which is multiplexed with DU0_DR1 pin	Selects TCLKB which is multiplexed with A1 pin	—	—	—
TCLKC	Selects TCLKC which is multiplexed with DU0_DR2 pin	Selects TCLKC which is multiplexed with A2 pin	—	—	—
TCLKD	Selects TCLKD which is multiplexed with DU0_DR3 pin	Selects TCLKD which is multiplexed with A3 pin	—	—	—
TIOC0A	Selects TIOC0A which is multiplexed with DU0_DR4 pin	Selects TIOC0A which is multiplexed with A4 pin	—	—	—

Target Pin	Group A	Group B	Group C	Group D	Group E
TIOC0B	Selects TIOC0B which is multiplexed with DU0_DR5 pin	Selects TIOC0B which is multiplexed with A5 pin	—	—	—
TIOC0C	Selects TIOC0C which is multiplexed with DU0_DR6 pin	Selects TIOC0C which is multiplexed with A6 pin	—	—	—
TIOC0D	Selects TIOC0D which is multiplexed with DU0_DR7 pin	Selects TIOC0D which is multiplexed with A7 pin	—	—	—
TIOC1A	Selects TIOC1A which is multiplexed with DU0_DG0 pin	Selects TIOC1A which is multiplexed with SSI_SCK0_A pin	Selects TIOC1A which is multiplexed with A8 pin	—	—
TIOC1B	Selects TIOC1B which is multiplexed with DU0_DG1 pin	Selects TIOC1B which is multiplexed with SSI_WS0_A pin	Selects TIOC1B which is multiplexed with A9 pin	—	—
TIOC2A	Selects TIOC2A which is multiplexed with DU0_DG2 pin	Selects TIOC2A which is multiplexed with SSI_SDATA0_A pin	Selects TIOC2A which is multiplexed with A10 pin	—	—
TIOC2B	Selects TIOC2B which is multiplexed with DU0_DG3 pin	Selects TIOC2B which is multiplexed with SSI_SCK1_A pin	Selects TIOC2B which is multiplexed with A11 pin	—	—
TIOC3A	Selects TIOC3A which is multiplexed with DU0_DG4 pin	Selects TIOC3A which is multiplexed with A12 pin	—	—	—
TIOC3B	Selects TIOC3B which is multiplexed with DU0_DG5 pin	Selects TIOC3B which is multiplexed with A13 pin	—	—	—
TIOC3C	Selects TIOC3C which is multiplexed with DU0_DG6 pin	Selects TIOC3C which is multiplexed with A14 pin	—	—	—
TIOC3D	Selects TIOC3D which is multiplexed with DU0_DG7 pin	Selects TIOC3D which is multiplexed with A15 pin	—	—	—
TIOC4A	Selects TIOC4A which is multiplexed with DU0_DB0 pin	Selects TIOC4A which is multiplexed with A16 pin	—	—	—

Target Pin	Group A	Group B	Group C	Group D	Group E
TIOC4B	Selects TIOC4B which is multiplexed with DU0_DB1 pin	Selects TIOC4B which is multiplexed with A17 pin	—	—	—
TIOC4C	Selects TIOC4C which is multiplexed with DU0_DB2 pin	Selects TIOC4C which is multiplexed with A18 pin	—	—	—
TIOC4D	Selects TIOC4D which is multiplexed with DU0_DB3 pin	Selects TIOC4D which is multiplexed with A19 pin	—	—	—

[Legend] —: Setting invalid

37.2.22 LSI Pin Pull-Up Control Register 0 (PUPCTL0)

Function: PUPCTL0 performs on/off control of the pull-up resistors.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	D5	D4	D3	D2	D1	D0	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	—	H'0000 0000	R/W	Performs individual on/off control of the pull-up resistor provided in each signal pin of the LSI. 0: Pull-up function is disabled. 1: Pull-up function is enabled.

37.2.23 LSI Pin Pull-Up Control Register 1 (PUPCTL1)

Function: PUPCTL1 performs on/off control of the pull-up resistors.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DACK1	$\overline{\text{DREQ1}}$	DACK0	$\overline{\text{DREQ0}}$	DRACK0	EX_WAIT2	EX_WAIT1	EX_WAIT0	$\overline{\text{WE1}}$	$\overline{\text{WE0}}$	RD/$\overline{\text{WR}}$	RD	$\overline{\text{EX_CS5}}$	$\overline{\text{EX_CS4}}$	$\overline{\text{EX_CS3}}$	$\overline{\text{EX_CS2}}$	$\overline{\text{EX_CS1}}$	$\overline{\text{EX_CS0}}$	$\overline{\text{CS1}}$/A26	$\overline{\text{CS0}}$	$\overline{\text{BS}}$	CLKOUT	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6
Initial value:	1	1	0	1	1	1	1	0	0	0	1	0	1	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	—	H'DE29 C000	R/W	Performs individual on/off control of the pull-up resistor provided in each signal pin of the LSI. 0: Pull-up function is disabled. 1: Pull-up function is enabled.

37.2.24 LSI Pin Pull-Up Control Register 2 (PUPCTL2)

Function: PUPCTL2 performs on/off control of the pull-up resistors.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	SD2_WP_A	SD2_CD_A	SD2_DAT3_A	SD2_DAT2_A	SD2_DAT1_A	SD2_DAT0_A	SD2_CMD_A	SD2_CLK_A	SCK2_A	$\overline{\text{RTS1\_B}}$	$\overline{\text{CTS1\_B}}$	TX1_B	RX1_B	SCK1_B	$\overline{\text{RTS0\_B}}$	$\overline{\text{CTS0\_B}}$	HTX0_A	HRX0_A	HSCK0_A	$\overline{\text{HRTS0\_A}}$	$\overline{\text{HCTS0\_A}}$	TX0_A	RX0_A	SCK0_A	SCIF_CLK_A	ASEBRK_N_ACK	TDO	TDI	TMS	TCK	TRST_N
Initial value:	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	0	1	1	1	1
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	This bit is always read as 0. The write value should always be 0.
30 to 0	—	H'7FFF FDEF	R/W	Performs individual on/off control of the pull-up resistor provided in each signal pin of the LSI. 0: Pull-up function is disabled. 1: Pull-up function is enabled.

37.2.25 LSI Pin Pull-Up Control Register 3 (PUPCTL3)

Function: PUPCTL3 performs on/off control of the pull-up resistors.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	DU0_CDE	DU0_DISP	DU0_EXODDF_DU0_ODDF	DU0_EXVSYNC_DU0_VSYNC	DU0_EXHSYNC_DU0_HSYNC	DU0_DOTCLKOUT	DU0_DOTCLKIN	DU0_DB7	DU0_DB6	DU0_DB5	DU0_DB4	DU0_DB3	DU0_DB2	DU0_DB1	DU0_DB0	DU0_DG7	DU0_DG6	DU0_DG5	DU0_DG4	DU0_DG3	DU0_DG2	DU0_DG1	DU0_DG0	DU0_DR7	DU0_DR6	DU0_DR5	DU0_DR4	DU0_DR3	DU0_DR2	DU0_DR1	DU0_DR0
Initial value:	0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	This bit is always read as 0. The write value should always be 0.
30 to 0	—	H'7F00 0000	R/W	Performs individual on/off control of the pull-up resistor provided in each signal pin of the LSI. 0: Pull-up function is disabled. 1: Pull-up function is enabled.

37.2.26 LSI Pin Pull-Up Control Register 4 (PUPCTL4)

Function: PUPCTL4 performs on/off control of the pull-up resistors.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	USB_OVC1	USB_OVC0	PENC1	PENC0	SDSELF	AUDIO_CLKOUT	AUDIO_CLKC	AUDIO_CLKB_A	AUDIO_CLKA_A	SSI_SDATA3	SSI_SDATA2	SSI_WS23	SSI_SCK23	SSI_SDATA1_A	SSI_WS1_A	SSI_SCK1_A	SSI_SDATA0_A	SSI_WS0_A	SSI_SCK0_A	VI1_7_A	VI1_6_A	VI1_5_A	VI1_4_A	VI1_3_A	VI1_2_A	VI1_1_A	VI1_0_A	VI1_CLK_A
Initial value:	0	0	0	0	1	0	1	0	1	1	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	These bits are always read as 0. The write value should always be 0.
27 to 0	—	H'ACF FFFE	R/W	Performs individual on/off control of the pull-up resistor provided in each signal pin of the LSI. 0: Pull-up function is disabled. 1: Pull-up function is enabled.

37.2.27 LSI Pin Pull-Up Control Register 5 (PUPCTL5)

Function: PUPCTL5 performs on/off control of the pull-up resistors.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PRESETOUT	REF50CK	REF125CK	IRQ3_A	IRQ2_A	IRQ1_A	IRQ0_A	CAN1_RX_A	CAN1_TX_A	CAN0_RX_A	CAN0_TX_A	CAN_CLK_A
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	These bits are always read as 0. The write value should always be 0.
11 to 0	—	H'1F1	R/W	Performs individual on/off control of the pull-up resistor provided in each signal pin of the LSI. 0: Pull-up function is disabled. 1: Pull-up function is enabled.

37.2.28 LSI Pin Drive Function Switching Register (DRV_SEL)

Function: DRV_SEL selects the LSI pin driving ability.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	sel_drv_stif	sel_drv_gr.b	sel_drv_gr.a	sel_drv_gr.com
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Initial Value	R/W	Description
31 to 4	All 0	R	These bits are always read as 0. The write value should always be 0.
3	0	R/W	Selects the driving ability of the $\overline{\text{PRESETOUT}}$ pin 0: 6 mA 1: 8 mA
2	0	R/W	Selects the driving ability of the D8, D9, D10, D11, D12, and D13 pins 0: 6 mA 1: 8 mA
1	0	R/W	Selects the driving ability of the $\overline{\text{EX\_CS2}}$, $\overline{\text{EX\_CS3}}$, $\overline{\text{EX\_CS4}}$, $\overline{\text{EX\_CS5}}$, EX_WAIT1, and EX_WAIT2 pins 0: 6 mA 1: 8 mA
0	0	R/W	Selects the driving ability of the $\overline{\text{EX\_CS1}}$, RD/$\overline{\text{WR}}$, DRACK0, DREQ0, and DACK0 pins 0: 6 mA 1: 8 mA

37.3 Operation

37.3.1 Function Setting for Multiplexed Pins

Setting the LSI multiplexed pin setting mask register (PMMR) is necessary before setting each of the GPIO/peripheral function select registers 0 to 5 (GPSR0 to GPSR5), the peripheral function select registers 0 to 11 (IPSR0 to IPSR11), the module select register (MOD_SEL), and the module select register 2 (MOD_SEL2). Specifically, the inverse of the value to be set in the select register must be written to the LSI multiplexed pin setting mask register. Otherwise, the GPIO/peripheral function select registers 0 to 5 (GPSR0 to GPSR5), the peripheral function select registers 0 to 11 (IPSR0 to IPSR11), the module select register (MOD_SEL), and the module select register 2 (MOD_SEL2) cannot be set.

In addition, before setting input and output pins for INTC, MMC, FLCTL, RCAN (channels 0 and 1, CLK), SDHI (channels 0 to 2), HSCIF (channel 0), SSS (channels 0 and 1, AUDIO_CLKA/ AUDIO_CLKB), HSPI, TMU, GETHER (RMII/ET0), LCDC, RSPI, SCIF (SCIF_CLK, channels 1 to 5), HIF, MTU2 (channels 0 to 4, CLK), VIN (channel 1), RQSPI, and IEBus, by the function of the multiplexed pins, set the module select register or the module select register 2.

Furthermore, after a pin function has been switched to GPIO and is then switched back to a peripheral module function or is switched from one peripheral function to another, waveforms output by the peripheral function may be incomplete and the change may occur along the way.

(1) Procedure for changing pin function from GPIO to peripheral function

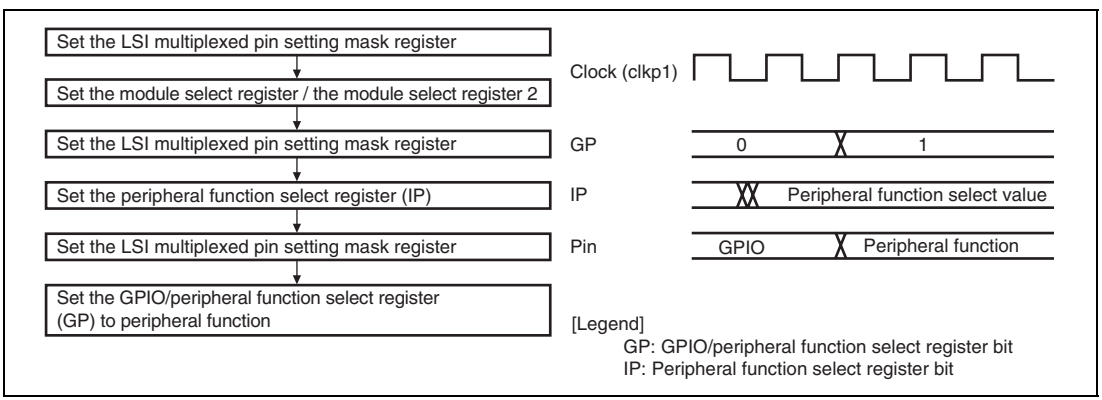


Figure 37.1 Procedure for Changing Pin Function from GPIO to Peripheral Function

(2) Procedure for changing pin function from peripheral function to GPIO

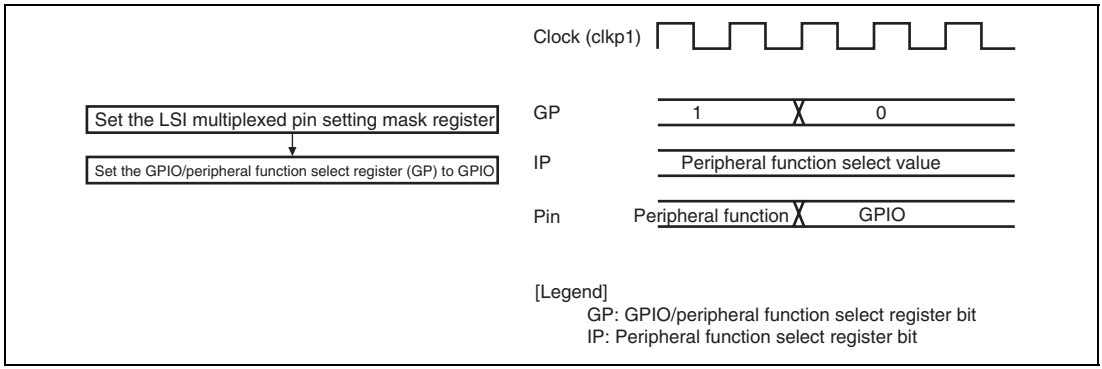


Figure 37.2 Procedure for Changing Pin Function from Peripheral function to GPIO

(3) Procedure 1 for changing pin function from one peripheral function to another peripheral function

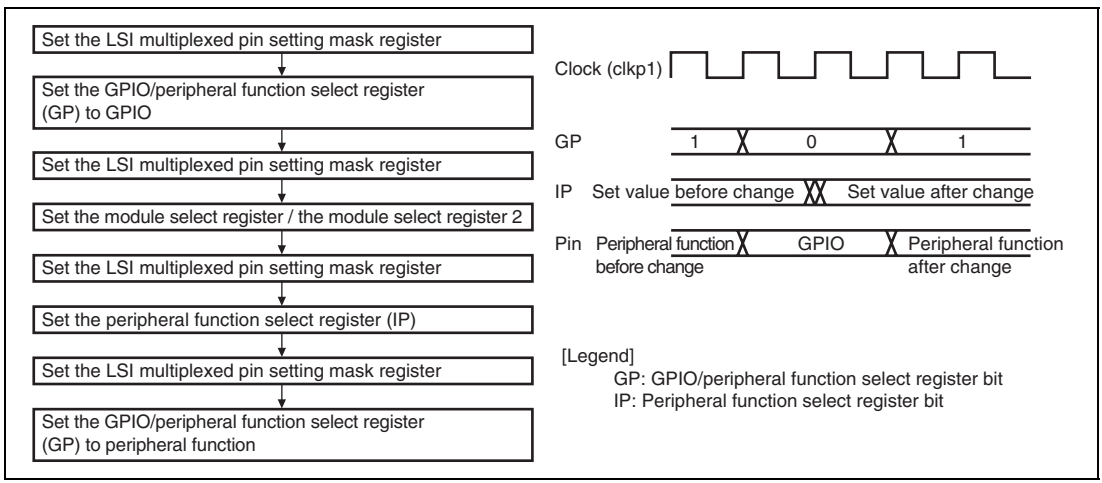


Figure 37.3 Procedure for Changing Pin Function from One Peripheral Function to Another Peripheral Function (with GPIO Setting)

(4) Procedure 2 for changing pin function from one peripheral function to another peripheral function

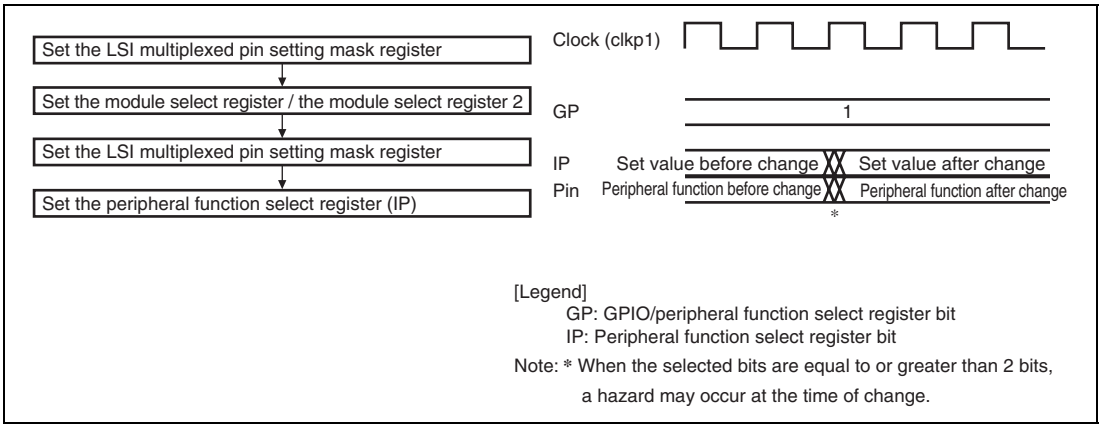


Figure 37.4 Procedure for Changing Pin Function from One Peripheral Function to Another Peripheral Function (without GPIO Setting)

37.3.2 Setting Pull-Up Resistors

The on/off of the pull-up resistors is controlled by the LSI pin pull-up control registers 0 to 5 (PUPCTL0 to PUPCTL5).

37.4 Notes

The PENC1 pin is pulled up in its initial state.

When the PENC1 function is in use, pulling up of the PENC1 pin is enabled until it is disabled by the setting of bit 25 in the PUPCTL4 register. Accordingly, if pulling up is disabled, obtain the same state by including an appropriate external pull-down resistor.

Section 38 User Break Controller (UBC)

The user break controller (UBC) provides versatile functions to facilitate program debugging. These functions help to ease creation of a self-monitor/debugger, which allows easy program debugging using this LSI alone, without using the in-circuit emulator. Various break conditions can be set in the UBC: instruction fetch or read/write access of an operand, operand size, data contents, address value, and program stop timing for instruction fetch.

38.1 Features

1. The following break conditions can be set.

   Break channels: Two (channels 0 and 1)

   User break conditions can be set independently for channels 0 and 1, and can also be set as a single sequential condition for the two channels, that is, a sequential break. (Sequential break involves two cases such that the channel 0 break condition is satisfied in a certain bus cycle and then the channel 1 break condition is satisfied in a different bus cycle, and vice versa.)

- Address

  When 40 bits containing ASID and 32-bit address are compared with the specified value, all the ASID bits can be compared or masked.

  32-bit address can be masked bit by bit, allowing the user to mask the address in desired page sizes such as lower 12 bits (4-Kbyte page) and lower 10 bits (1-Kbyte page).

- Data

  32 bits can be masked only for channel 1.

- Bus cycle

  The program can break either for instruction fetch (PC break) or operand access.

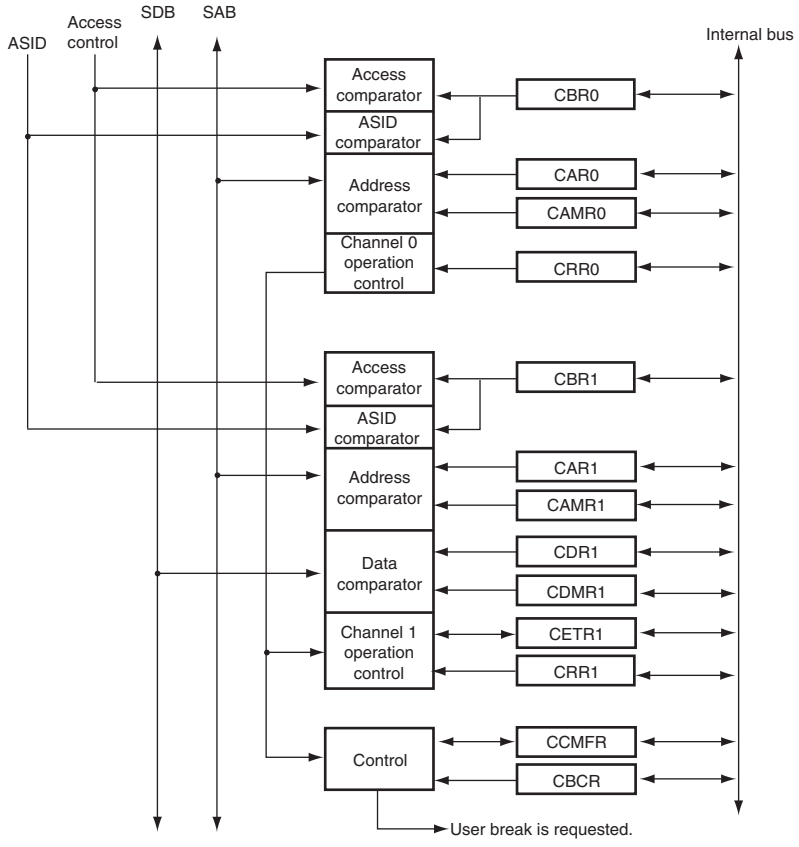
- Read or write access
- Operand sizes

  Byte, word, longword, and quadword are supported.

2. The user-designated exception handling routine for the user break condition can be executed.
3. Pre-instruction-execution or post-instruction-execution can be selected as the PC break timing.
4. A maximum of $2^{12} - 1$ repetition counts can be specified as the break condition (available only for channel 1).

38.1.1 Block Diagram

Figure 38.1 shows the UBC block diagram.



[Legend]

CBR0:	Match condition setting register 0
CRR0:	Match operation setting register 0
CAR0:	Match address setting register 0
CAMR0:	Match address mask setting register 0
CBR1:	Match condition setting register 1
CRR1:	Match operation setting register 1
CAR1:	Match address setting register 1
CAMR1:	Match address mask setting register 1
CDR1:	Match data setting register 1
CDMR1:	Match data mask setting register 1
CETR1:	Execution count break register 1
CCMFR:	Channel match flag register
CBCR:	Break control register
SAB:	Operand address bus
SDB:	Operand data bus

Figure 38.1 Block Diagram of UBC

38.2 Register Configuration

Tables 38.1 and 38.2 show the register configuration and the register status in each processing state, respectively.

Do not write to addresses other than those listed below, otherwise normal operation cannot be guaranteed.

Table 38.1 Register Configuration

Name	Abbreviation	R/W	Area P4 Address*	Area 7 Address*	Access Size
Match condition setting register 0	CBR0	R/W	H'FF20 0000	H'1F20 0000	32
Match operation setting register 0	CRR0	R/W	H'FF20 0004	H'1F20 0004	32
Match address setting register 0	CAR0	R/W	H'FF20 0008	H'1F20 0008	32
Match address mask setting register 0	CAMR0	R/W	H'FF20 000C	H'1F20 000C	32
Match condition setting register 1	CBR1	R/W	H'FF20 0020	H'1F20 0020	32
Match operation setting register 1	CRR1	R/W	H'FF20 0024	H'1F20 0024	32
Match address setting register 1	CAR1	R/W	H'FF20 0028	H'1F20 0028	32
Match address mask setting register 1	CAMR1	R/W	H'FF20 002C	H'1F20 002C	32
Match data setting register 1	CDR1	R/W	H'FF20 0030	H'1F20 0030	32
Match data mask setting register 1	CDMR1	R/W	H'FF20 0034	H'1F20 0034	32
Execution count break register 1	CETR1	R/W	H'FF20 0038	H'1F20 0038	32
Channel match flag register	CCMFR	R/W	H'FF20 0600	H'1F20 0600	32
Break control register	CBCR	R/W	H'FF20 0620	H'1F20 0620	32

Note: * P4 addresses are used when area P4 in the virtual address space is used, and area 7 addresses are used when accessing the register through area 7 in the physical address space using the TLB.

Table 38.2 Register Status in Each Processing State

Register Name	Abbreviation	Power-on Reset	Manual Reset	Sleep	Software Standby	Deep Standby
Match condition setting register 0	CBR0	H'2000 0000	Retained	Retained	Retained	H'2000 0000
Match operation setting register 0	CRR0	H'0000 2000	Retained	Retained	Retained	H'0000 2000
Match address setting register 0	CAR0	Undefined	Retained	Retained	Retained	Undefined
Match address mask setting register 0	CAMR0	Undefined	Retained	Retained	Retained	Undefined
Match condition setting register 1	CBR1	H'2000 0000	Retained	Retained	Retained	H'2000 0000
Match operation setting register 1	CRR1	H'0000 2000	Retained	Retained	Retained	H'0000 2000
Match address setting register 1	CAR1	Undefined	Retained	Retained	Retained	Undefined
Match address mask setting register 1	CAMR1	Undefined	Retained	Retained	Retained	Undefined
Match data setting register 1	CDR1	Undefined	Retained	Retained	Retained	Undefined
Match data mask setting register 1	CDMR1	Undefined	Retained	Retained	Retained	Undefined
Execution count break register 1	CETR1	Undefined	Retained	Retained	Retained	Undefined
Channel match flag register	CCMFR	H'0000 0000	Retained	Retained	Retained	H'0000 0000
Break control register	CBCR	H'0000 0000	Retained	Retained	Retained	H'0000 0000

The access size must be the same as the control register size. If the size is different, the register is not written to if attempted, and reading the register returns the undefined value. A desired break may not occur between the time when the instruction for rewriting the control register is executed and the time when the written value is actually reflected on the register. In order to confirm the exact timing when the control register is updated, read the data, which has been written most recently. The subsequent instructions are valid for the most recently written register value.

38.2.1 Match Condition Setting Registers 0 and 1 (CBR0 and CBR1)

CBR0 and CBR1 are readable/writable 32-bit registers which specify the break conditions for channels 0 and 1, respectively. The following break conditions can be set in the CBR0 and CBR1: (1) whether or not to include the match flag in the conditions, (2) whether or not to include the ASID, and the ASID value when included, (3) whether or not to include the data value, (4) operand size, (5) whether or not to include the execution count, (6) bus type, (7) instruction fetch cycle or operand access cycle, and (8) read or write access cycle.

- CBR0

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MFE	AIE	MFI						AIV							
Initial value :	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	SZ			—	—	—	—	CD		ID		—	RW		CE
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	MFE	0	R/W	Match Flag Enable Specifies whether or not to include the match flag value specified by the MFI bit of this register in the match conditions. When the specified match flag value is 1, the condition is determined to be satisfied. 0: The match flag is not included in the match conditions; thus, not checked. 1: The match flag is included in the match conditions.
30	AIE	0	R/W	ASID Enable Specifies whether or not to include the ASID specified by the AIV bit of this register in the match conditions. 0: The ASID is not included in the match conditions; thus, not checked. 1: The ASID is included in the match conditions.

Bit	Bit Name	Initial Value	R/W	Description
29 to 24	MFI	100000	R/W	Match Flag Specify Specifies the match flag to be included in the match conditions. 000000: The MF0 bit of the CCMFR register 000001: The MF1 bit of the CCMFR register Others: Reserved (setting prohibited) Note: The initial value is the reserved value, but when 1 is written into CBR1[0], MFI must be set to 000000 or 000001. And note that channel 0 is not hit when the MFE bit of this register is 1 and MFI bits are 000000 in the condition of CCMFR.MF0 = 0.
23 to 16	AIV	H'00	R/W	ASID Specify Specifies the ASID value to be included in the match conditions.
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14 to 12	SZ	000	R/W	Operand Size Select Specifies the operand size to be included in the match conditions. This bit is valid only when the operand access cycle is specified as a match condition. 000: The operand size is not included in the match conditions; thus, not checked (any operand size specifies the match condition).*1 001: Byte access 010: Word access 011: Longword access 100: Quadword access*3 Others: Reserved (setting prohibited)
11 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
7, 6	CD	00	R/W	Bus Select Specifies the bus to be included in the match conditions. This bit is valid only when the operand access cycle is specified as a match condition. 00: Operand bus for operand access Others: Reserved (setting prohibited)
5, 4	ID	00	R/W	Instruction Fetch/Operand Access Select Specifies the instruction fetch cycle or operand access cycle as the match condition. 00: Instruction fetch cycle or operand access cycle 01: Instruction fetch cycle 10: Operand access cycle 11: Instruction fetch cycle or operand access cycle
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2, 1	RW	00	R/W	Bus Command Select Specifies the read/write cycle as the match condition. This bit is valid only when the operand access cycle is specified as a match condition. 00: Read cycle or write cycle 01: Read cycle 10: Write cycle 11: Read cycle or write cycle
0	CE	0	R/W	Channel Enable Validates/invalidates the channel. If this bit is 0, all the other bits in this register are invalid. 0: Invalidates the channel. 1: Validates the channel.

- CBR1

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MFE	AIE	MFI						AIV							
Initial value :	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DBE	SZ			ETBE	—	—	—	CD		ID		—	RW		CE
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	MFE	0	R/W	Match Flag Enable Specifies whether or not to include the match flag value specified by the MFI bit of this register in the match conditions. When the specified match flag value is 1, the condition is determined to be satisfied. 0: The match flag is not included in the match conditions; thus, not checked. 1: The match flag is included in the match conditions.
30	AIE	0	R/W	ASID Enable Specifies whether or not to include the ASID specified by the AIV bit of this register in the match conditions. 0: The ASID is not included in the match conditions; thus, not checked. 1: The ASID is included in the match conditions.
29 to 24	MFI	100000	R/W	Match Flag Specify Specifies the match flag to be included in the match conditions. 000000: The MF0 bit of the CCMFR register 000001: The MF1 bit of the CCMFR register Others: Reserved (setting prohibited) Note: The initial value is the reserved value, but when 1 is written into CBR1[0], MFI must be set to 000000 or 000001. And note that channel 1 is not hit when the MFE bit of this register is 1 and MFI bits are 000001 in the condition of CCMFR.MF1 = 0.

Bit	Bit Name	Initial Value	R/W	Description
23 to 16	AIV	H'00	R/W	ASID Specify Specifies the ASID value to be included in the match conditions.
15	DBE	0	R/W	Data Value Enable*2 Specifies whether or not to include the data value in the match condition. This bit is valid only when the operand access cycle is specified as a match condition. 0: The data value is not included in the match conditions; thus, not checked. 1: The data value is included in the match conditions.
14 to 12	SZ	000	R/W	Operand Size Select Specifies the operand size to be included in the match conditions. This bit is valid only when the operand access cycle is specified as a match condition. 000: The operand size is not included in the match condition; thus, not checked (any operand size specifies the match condition).*1 001: Byte access 010: Word access 011: Longword access 100: Quadword access*3 Others: Reserved (setting prohibited)
11	ETBE	0	R/W	Execution Count Value Enable Specifies whether or not to include the execution count value in the match conditions. If this bit is 1 and the match condition satisfaction count matches the value specified by the CETR1 register, the operation specified by the CRR1 register is performed. 0: The execution count value is not included in the match conditions; thus, not checked. 1: The execution count value is included in the match conditions.
10 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
7, 6	CD	00	R/W	Bus Select Specifies the bus to be included in the match conditions. This bit is valid only when the operand access cycle is specified as a match condition. 00: Operand bus for operand access Others: Reserved (setting prohibited)
5, 4	ID	00	R/W	Instruction Fetch/Operand Access Select Specifies the instruction fetch cycle or operand access cycle as the match condition. 00: Instruction fetch cycle or operand access cycle 01: Instruction fetch cycle 10: Operand access cycle 11: Instruction fetch cycle or operand access cycle
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2, 1	RW	00	R/W	Bus Command Select Specifies the read/write cycle as the match condition. This bit is valid only when the operand access cycle is specified as a match condition. 00: Read cycle or write cycle 01: Read cycle 10: Write cycle 11: Read cycle or write cycle
0	CE	0	R/W	Channel Enable Validates/invalidates the channel. If this bit is 0, all the other bits in this register are invalid. 0: Invalidates the channel. 1: Validates the channel.

Notes: 1. If the data value is included in the match conditions, be sure to specify the operand size.

2. The OCBI instruction is handled as longword write access without the data value, and the PREF, OCBP, and OCBWB instructions are handled as longword read access without the data value. Therefore, do not include the data value in the match conditions for these instructions.

3. If the quadword access is specified and the data value is included in the match conditions, the upper and lower 32 bits of 64-bit data are each compared with the contents of both the match data setting register and the match data mask setting register.

38.2.2 Match Operation Setting Registers 0 and 1 (CRR0 and CRR1)

CRR0 and CRR1 are readable/writable 32-bit registers which specify the operation to be executed when channels 0 and 1 satisfy the match condition, respectively. The following operations can be set in the CRR0 and CRR1 registers: (1) breaking at a desired timing for the instruction fetch cycle and (2) requesting a break.

- CRR0

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PCB	BIE
Initial value :	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
12 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1	PCB	0	R/W	PC Break Select Specifies either before or after instruction execution as the break timing for the instruction fetch cycle. This bit is invalid for breaks other than ones for the instruction fetch cycle. 0: Sets the PC break before instruction execution. 1: Sets the PC break after instruction execution.
0	BIE	0	R/W	Break Enable Specifies whether or not to request a break when the match condition is satisfied for the channel. 0: Does not request a break. 1: Requests a break.

- CRR1

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PCB	BIE
Initial value :	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

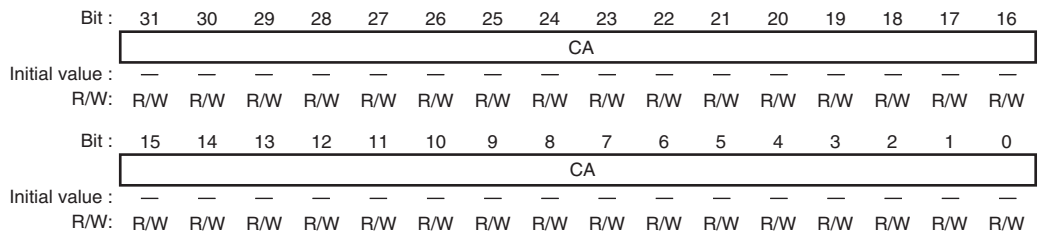
Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
12 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1	PCB	0	R/W	PC Break Select Specifies either before or after instruction execution as the break timing for the instruction fetch cycle. This bit is invalid for breaks other than ones for the instruction fetch cycle. 0: Sets the PC break before instruction execution. 1: Sets the PC break after instruction execution.
0	BIE	0	R/W	Break Enable Specifies whether or not to request a break when the match condition is satisfied for the channel. 0: Does not request a break. 1: Requests a break.

38.2.3 Match Address Setting Registers 0 and 1 (CAR0 and CAR1)

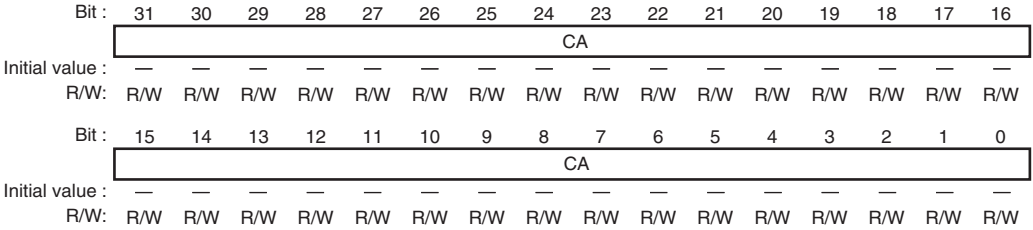
CAR0 and CAR1 are readable/writable 32-bit registers specifying the virtual address to be included in the break conditions for channels 0 and 1, respectively.

- CAR0



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CA	Undefined	R/W	Compare Address Specifies the address to be included in the break conditions. When the operand bus has been specified using the CBR0 register, specify the SAB address in CA[31:0].

- CAR1

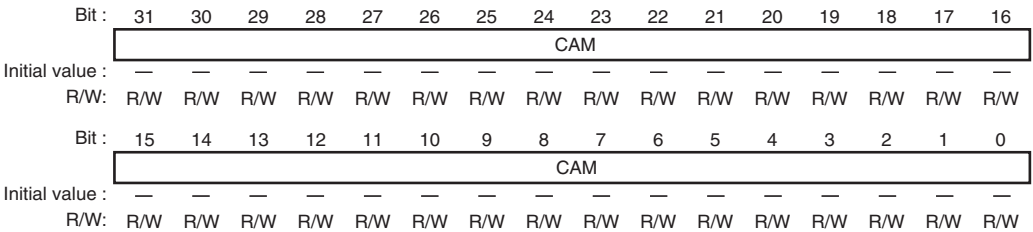


Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CA	Undefined	R/W	Compare Address Specifies the address to be included in the break conditions. When the operand bus has been specified using the CBR1 register, specify the SAB address in CA[31:0].

38.2.4 Match Address Mask Setting Registers 0 and 1 (CAMR0 and CAMR1)

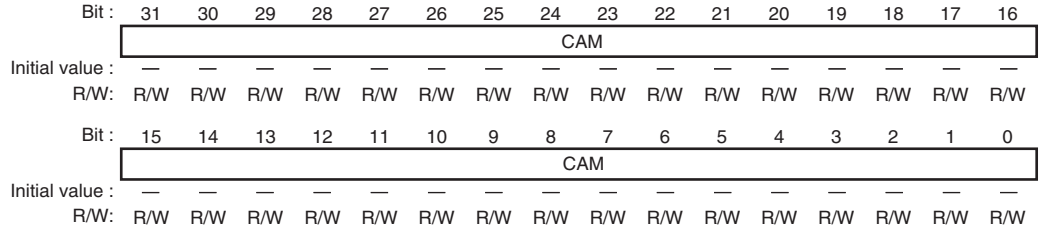
CMAR0 and CMAR1 are readable/writable 32-bit registers which specify the bits to be masked among the address bits specified by using the match address setting register of the corresponding channel. (Set the bits to be masked to 1.)

- CAMR0



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CAM	Undefined	R/W	Compare Address Mask Specifies the bits to be masked among the address bits which are specified using the CAR0 register. (Set the bits to be masked to 1.) 0: Address bits CA[n] are included in the break condition. 1: Address bits CA[n] are masked and not included in the break condition. [n] = any values from 31 to 0

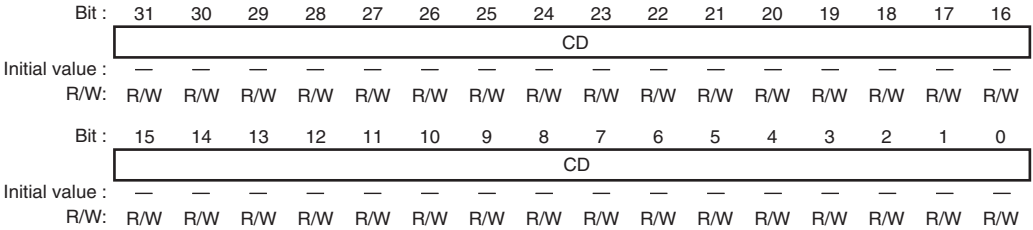
- CAMR1



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CAM	Undefined	R/W	Compare Address Mask Specifies the bits to be masked among the address bits which are specified using the CAR1 register. (Set the bits to be masked to 1.) 0: Address bits CA[n] are included in the break condition. 1: Address bits CA[n] are masked and not included in the break condition. [n] = any values from 31 to 0

38.2.5 Match Data Setting Register 1 (CDR1)

CDR1 is a readable/writable 32-bit register which specifies the data value to be included in the break conditions for channel 1.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CD	Undefined	R/W	Compare Data Value Specifies the data value to be included in the break conditions. When the operand bus has been specified using the CBR1 register, specify the SDB data value in CD[31:0].

Table 38.3 Settings for Match Data Setting Register

Bus and Size Selected Using CBR1	CD[31:24]	CD[23:16]	CD[15:8]	CD[7:0]
Operand bus (byte)	Don't care	Don't care	Don't care	SDB7 to SDB0
Operand bus (word)	Don't care	Don't care	SDB15 to SDB8	SDB7 to SDB0
Operand bus (longword)	SDB31 to SDB24	SDB23 to SDB16	SDB15 to SDB8	SDB7 to SDB0

Notes: 1. If the data value is included in the match conditions, be sure to specify the operand size.
2. The OCBI instruction is handled as longword write access without the data value, and the PREF, OCBP, and OCBWB instructions are handled as longword read access without the data value. Therefore, do not include the data value in the match conditions for these instructions.
3. If the quadword access is specified and the data value is included in the match conditions, the upper and lower 32 bits of 64-bit data are each compared with the contents of both the match data setting register and match data mask setting register.

38.2.6 Match Data Mask Setting Register 1 (CDMR1)

CDMR1 is a readable/writable 32-bit register which specifies the bits to be masked among the data value bits specified using the match data setting register. (Set the bits to be masked to 1.)

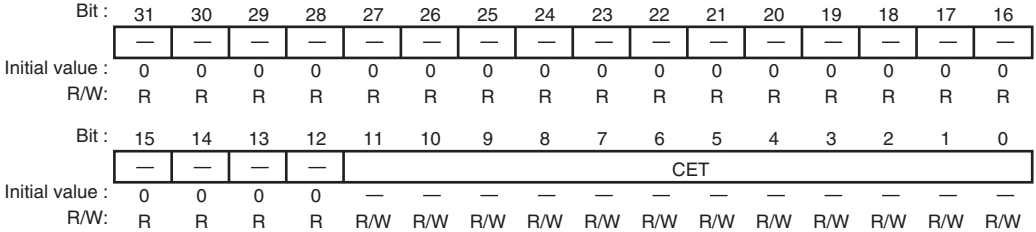
Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CDM															
Initial value :	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CDM															
Initial value :	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CDM	Undefined	R/W	Compare Data Value Mask Specifies the bits to be masked among the data value bits specified using the CDR1 register. (Set the bits to be masked to 1.) 0: Data value bits CD[n] are included in the break condition. 1: Data value bits CD[n] are masked and not included in the break condition. [n] = any values from 31 to 0

38.2.7 Execution Count Break Register 1 (CETR1)

CETR1 is a readable/writable 32-bit register which specifies the number of the channel hits before a break occurs. A maximum value of $2^{12} - 1$ can be specified. When the execution count value is included in the match conditions by using the match condition setting register, the value of this register is decremented by one every time the channel is hit. When the channel is hit after the register value reaches H'001, a break occurs.



Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	CET	Undefined	R/W	Execution Count Specifies the execution count to be included in the break conditions.

38.2.8 Channel Match Flag Register (CCMFR)

CCMFR is a readable/writable 32-bit register which indicates whether or not the match conditions have been satisfied for each channel. When a channel match condition has been satisfied, the corresponding flag bit is set to 1. To clear the flags, write the data containing value 0 for the bits to be cleared and value 1 for the other bits to this register. (The logical AND between the value which has been written and the current register value is actually written to the register.) Sequential operation using multiple channels is available by using these match flags.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MF1	MF0
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	MF1	0	R/W	Channel 1 Condition Match Flag This flag is set to 1 when the channel 1 match condition has been satisfied. To clear the flag, write 0 to this bit. 0: Channel 1 match condition has not been satisfied. 1: Channel 1 match condition has been satisfied.
0	MF0	0	R/W	Channel 0 Condition Match Flag This flag is set to 1 when the channel 0 match condition has been satisfied. To clear the flag, write 0 to this bit. 0: Channel 0 match condition has not been satisfied. 1: Channel 0 match condition has been satisfied.

38.2.9 Break Control Register (CBCR)

CBCR is a readable/writable 32-bit register which specifies whether or not to use the user break debugging support function. For details on the user break debugging support function, refer to section 38.3.7, User Break Debugging Support Function.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UBDE
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	UBDE	0	R/W	User Break Debugging Support Function Enable Specifies whether or not to use the user break debugging support function. 0: Does not use the user break debugging support function. 1: Uses the user break debugging support function.

38.3 Operation Description

38.3.1 Definition of Words Related to Accesses

"Instruction fetch" refers to an access in which an instruction is fetched. For example, fetching the instruction located at the branch destination after executing a branch instruction is an instruction access. "Operand access" refers to any memory access accompanying execution of an instruction. For example, accessing an address (PC + disp × 2 + 4) in the instruction MOV.W@(disp,PC),Rn is an operand access. "Data" is used in contrast to "address".

All types of operand access are classified into read or write access. Special care must be taken in using the following instructions.

- PREF, OCBP, and OCBWB: Instructions for a read access
- MOVCA.L and OCBI: Instructions for a write access
- TAS.B: Instruction for a single read access or a single write access

The operand access accompanying the PREF, OCBP, OCBWB, and OCBI instructions is access without the data value; therefore, do not include the data value in the match conditions for these instructions.

The operand size should be defined for all types of operand access. Available operand sizes are byte, word, longword, and quadword. For operand access accompanying the PREF, OCBP, OCBWB, MOVCA.L, and OCBI instructions, the operand size is defined as longword.

38.3.2 User Break Operation Sequence

The following describes the sequence from when the break condition is set until the user break exception handling is initiated.

1. Specify the operand size, bus, instruction fetch/operand access, and read/write as the match conditions using the match condition setting register (CBR0 or CBR1). Specify the break address using the match address setting register (CAR0 or CAR1), and specify the address mask condition using the match address mask setting register (CAMR0 or CAMR1). To include the ASID in the match conditions, set the AIE bit in the match condition setting register and specify the ASID value by the AIV bit in the same register. To include the data value in the match conditions, set the DBE bit in the match condition setting register; specify the break data using the match data setting register (CDR1); and specify the data mask condition using the match data mask setting register (CDMR1). To include the execution count in the match conditions, set the ETBE bit of the match condition setting register; and specify the execution count using the execution count break register (CETR1). To use the sequential break, set the MFE bit of the match condition setting register; and specify the number of the first channel using the MFI bit.
2. Specify whether or not to request a break when the match condition is satisfied and the break timing when the match condition is satisfied as a result of fetching the instruction using the match operation setting register (CRR0 or CRR1). After having set all the bits in the match condition setting register except the CE bit and the other necessary registers, set the CE bit and read the match condition setting register again. This ensures that the set values in the control registers are valid for the subsequent instructions immediately after reading the register. Setting the CE bit of the match condition setting register in the initial state after reset via the control registers may cause an undesired break.
3. When the match condition has been satisfied, the corresponding condition match flag (MF1 or MF0) in the channel match flag register (CCMFR) is set. A break is also requested to the CPU according to the set values in the match operation setting register (CRR0 or CRR1). The CPU operates differently according to the BL bit value of the SR register: when the BL bit is 0, the CPU accepts the break request and executes the specified exception handling; and when the BL bit is 1, the CPU does not execute the exception handling.
4. The match flags (MF1 and MF0) can be used to confirm whether or not the corresponding match condition has been satisfied. Although the flag is set when the condition is satisfied, it is not cleared automatically; therefore, write 0 to the flag bit by issuing a memory store instruction to the channel match flag register (CCMFR) in order to use the flag again.
5. Breaks may occur virtually at the same time for channels 0 and 1. In this case, only one break request is sent to the CPU; however, the two condition match flags corresponding to these breaks may be set.

6. While the BL bit in the SR register is 1, no break requests are accepted. However, whether or not the condition has been satisfied is determined. When the condition is determined to be satisfied, the corresponding condition match flag is set.
7. If the sequential break conditions are set, the condition match flag is set every time the match conditions are satisfied for each channel. When the conditions have been satisfied for the first channel in the sequence but not for the second channel in the sequence, clear the condition match flag for the first channel in the sequence in order to release the first channel in the sequence from the match state.

38.3.3 Instruction Fetch Cycle Break

1. If the instruction fetch cycle is set in the match condition setting register (CBR0 or CBR1), the instruction fetch cycle is handled as a match condition. To request a break upon satisfying the match condition, set the BIE bit in the match operation setting register (CRR0 or CRR1) of the corresponding channel. Either before or after executing the instruction can be selected as the break timing according to the PCB bit value. If the instruction fetch cycle is specified as a match condition, be sure to clear the LSB to 0 in the match address setting register (CAR0 or CAR1); otherwise, no break occurs.
2. If pre-instruction-execution break is specified for the instruction fetch cycle, the break is requested when the instruction is fetched and determined to be executed. Therefore, this function cannot be used for the instructions which are fetched through overrun (i.e., the instructions fetched during branching or making transition to the interrupt routine but not executed). For priorities of pre-instruction-execution break and the other exceptions, refer to section 2.5, Exception Handling. If pre-instruction-execution break is specified for the delayed slot of the delayed branch instruction, the break is requested before the delayed branch instruction is executed. However, do not specify pre-instruction-execution break for the delayed slot of the RTE instruction.
3. If post-instruction-execution break is specified for the instruction fetch cycle, the break is requested after the instruction, which satisfied the match condition has been executed and before the next instruction is executed. Similar to pre-instruction-execution break, this function cannot be used for the instructions, which are fetched through overrun. For priorities of post-instruction-execution break and the other exceptions, refer to section 2.5, Exception Handling. If post-instruction-execution break is specified for the delayed branch instruction and its delayed slot, the break does not occur until the first instruction at the branch destination.
4. If the instruction fetch cycle is specified as the channel 1 match condition, the DBE bit of match condition setting register CBR1 becomes invalid, the settings of match data setting register CDR1 and match data mask setting register CDMR1 are ignored. Therefore, the data value cannot be specified for the instruction fetch cycle break.

38.3.4 Operand Access Cycle Break

1. Table 38.4 shows the relation between the operand sizes specified using the match condition setting register (CBR0 or CBR1) and the address bits to be compared for the operand access cycle break.

Table 38.4 Relation between Operand Sizes and Address Bits to be Compared

Selected Operand Size	Address Bits to be Compared
Quadword	Address bits A31 to A3
Longword	Address bits A31 to A2
Word	Address bits A31 to A1
Byte	Address bits A31 to A0
Operand size is not included in the match conditions	Address bits A31 to A3 for quadword access Address bits A31 to A2 for longword access Address bits A31 to A1 for word access Address bits A31 to A0 for byte access

The above table means that if address H'00001003 is set in the match address setting register (CAR0 or CAR1), for example, the match condition is satisfied for the following access cycles (assuming that all the other conditions are satisfied):

— Longword access to address H'00001000

— Word access to address H'00001002

— Byte access to address H'00001003

2. When the data value is included in the channel 1 match conditions:

If the data value is included in the match conditions, be sure to select the quadword, longword, word, or byte as the operand size using the operand size select bit (SZ) of the match condition setting register (CBR1), and also set the match data setting register (CDR1) and the match data mask setting register (CDMR1). With these settings, the match condition is satisfied when both of the address and data conditions are satisfied. The data value and mask control for byte access, word access, and longword access should be set in bits 7 to 0, 15 to 0, and 31 to 0 in the bits CDR1 and CDMR1, respectively. For quadword access, 64-bit data is divided into the upper and lower 32-bit data units, and each unit is independently compared with the specified condition. When either the upper or lower 32-bit data unit satisfies the match condition, the match condition for the 64-bit data is determined to be satisfied.

3. The operand access accompanying the PREF, OCBP, OCBWB, and OCBI instructions are access without the data value; therefore, if the data value is included in the match conditions for these instructions, the match conditions will never be satisfied.
4. If the operand bus is selected, a break occurs after executing the instruction which has satisfied the conditions and immediately before executing the next instruction. However, if the data value is included in the match conditions, a break may occur after executing several instructions after the instruction which has satisfied the conditions; therefore, it is impossible to identify the instruction causing the break. If such a break has occurred for the delayed branch instruction or its delayed slot, the break does not occur until the first instruction at the branch destination.

   However, do not specify the operand break for the delayed slot of the RTE instruction. And if the data value is included in the match conditions, it is not allowed to set the break for the preceding the RTE instruction by one to six instructions.

38.3.5 Sequential Break

1. Sequential break conditions can be specified by setting the MFE and MFI bits in the match condition setting registers (CBR0 and CBR1). (Sequential break involves two cases such that channel 0 break condition is satisfied then channel 1 break condition is satisfied, and vice versa.) To use the sequential break function, clear the MFE bit of the match condition setting register and the BIE bit of the match operation setting register of the first channel in the sequence, and set the MFE bit and specify the number of the second channel in the sequence using the MFI bit in the match condition setting register of the second channel in the sequence. If the sequential break condition is set, the condition match flag is set every time the match condition is satisfied for each channel. When the condition has been satisfied for the first channel in the sequence but not for the second channel in the sequence, clear the condition match flag for the first channel in the sequence in order to release the first channel in the sequence from the match state.
2. For channel 1, the execution count break condition can also be included in the sequential break conditions.
3. If the match conditions for the first and second channels in the sequence are satisfied within a significantly short time, sequential operation may not be guaranteed in some cases, as shown below.

- When the match condition is satisfied at the instruction fetch cycle for both the first and second channels in the sequence:

Instruction B is 0 instruction after instruction A	Equivalent to setting the same addresses; do not use this setting.
Instruction B is one instruction after instruction A	Sequential operation is not guaranteed.
Instruction B is two or more instructions after instruction A	Sequential operation is guaranteed.

- When the match condition is satisfied at the instruction fetch cycle for the first channel in the sequence whereas the match condition is satisfied at the operand access cycle for the second channel in the sequence:

Instruction B is 0 or one instruction after instruction A	Sequential operation is not guaranteed.
Instruction B is two or more instructions after instruction A	Sequential operation is guaranteed.

- When the match condition is satisfied at the operand access cycle for the first channel in the sequence whereas the match condition is satisfied at the instruction fetch cycle for the second channel in the sequence:

Instruction B is 0 to five instructions after instruction A	Sequential operation is not guaranteed.
Instruction B is six or more instructions after instruction A	Sequential operation is guaranteed.

- When the match condition is satisfied at the operand access cycle for both the first and second channels in the sequence:

Instruction B is 0 to five instructions after instruction A	Sequential operation is not guaranteed.
Instruction B is six or more instructions after instruction A	Sequential operation is guaranteed.

38.3.6 Program Counter Value to be Saved

When a break has occurred, the address of the instruction to be executed when the program restarts is saved in the SPC then the exception handling state is initiated. A unique instruction causing a break can be identified unless the data value is included in the match conditions.

1. When the instruction fetch cycle (before instruction execution) is specified as the match condition:

   The address of the instruction, which has satisfied the match conditions is saved in the SPC. The instruction, which has satisfied the match conditions is not executed, but a break occurs instead. However, if the match conditions are satisfied for the delayed slot instruction, the address of the delayed branch instruction is saved in the SPC.
2. When the instruction fetch cycle (after instruction execution) is specified as the match condition:

   The address of the instruction immediately after the instruction, which has satisfied the match conditions is saved in the SPC. The instruction, which has satisfied the match conditions is executed, then a break occurs before the next instruction. If the match conditions are satisfied for the delayed branch instruction or its delayed slot, these instructions are executed and the address of the branch destination is saved in the SPC.
3. When the operand access (address only) is specified as the match condition:

   The address of the instruction immediately after the instruction, which has satisfied the break conditions is saved in the SPC. The instruction, which has satisfied the match conditions are executed, then a break occurs before the next instruction. However, if the conditions are satisfied for the delayed slot, the address of the branch destination is saved in the SPC.
4. When the operand access (address and data) is specified as the match condition:

   If the data value is added to the match conditions, the instruction, which has satisfied the match conditions is executed. A user break occurs before executing an instruction that is one through six instructions after the instruction, which has satisfied the match conditions. The address of the instruction is saved in the SPC; thus, it is impossible to identify exactly where a break will occur. If the conditions are satisfied for the delayed slot instruction, the address of the branch destination is saved in the SPC. If a branch instruction follows the instruction, which has satisfied the match conditions, a break may occur after the delayed instruction and delayed slot are executed. In this case, the address of the branch destination is also saved in the SPC.

38.3.7 User Break Debugging Support Function

By using the user break debugging support function, the branch destination address can be modified when the CPU accepts the user break request. Specifically, setting the UBDE bit of break control register CBCR to 1 allows branching to the address indicated by DBR instead of branching to the address indicated by the [VBR + offset]. Figure 38.2 shows the flowchart of the user break debugging support function.

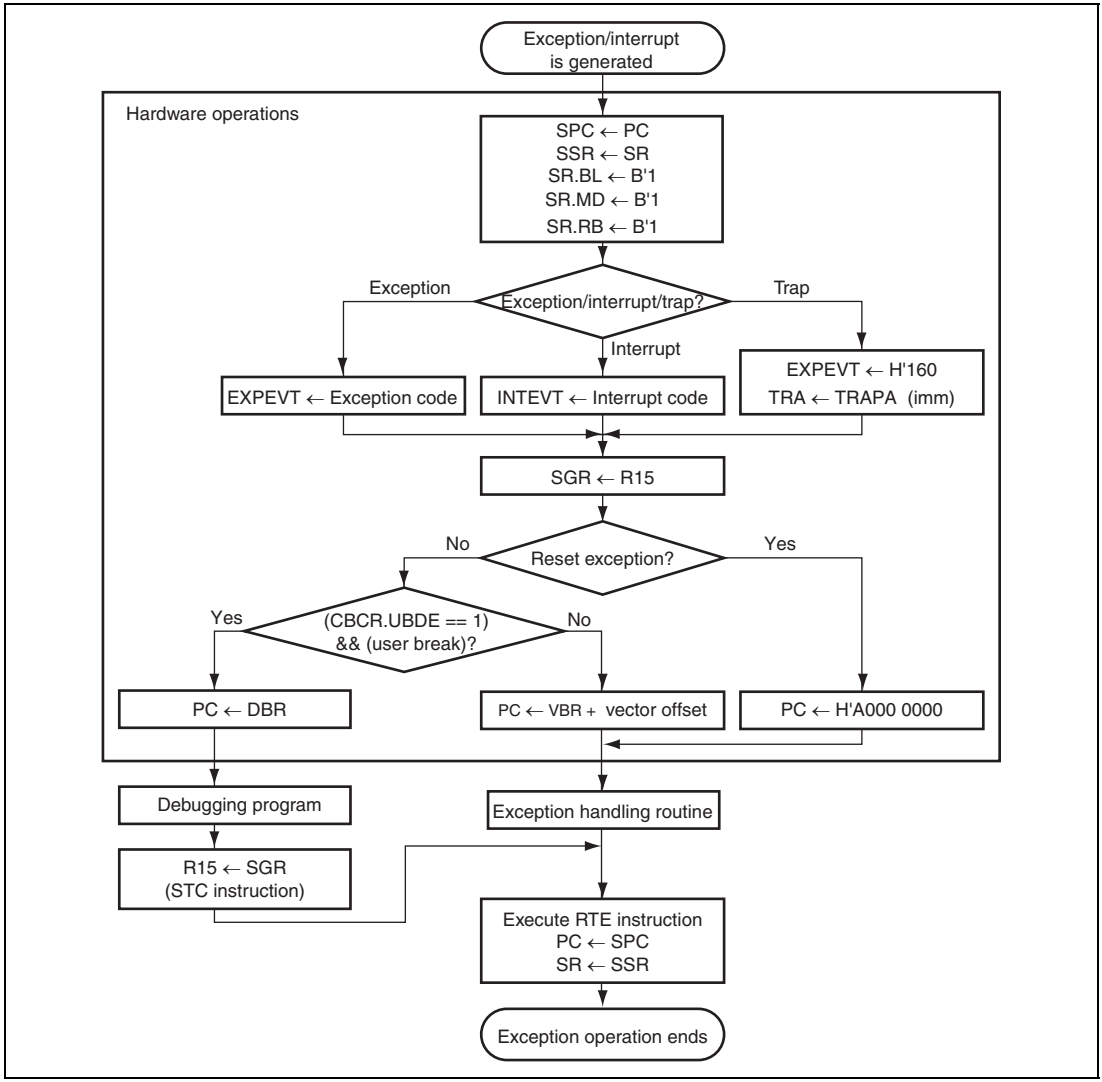


Figure 38.2 Flowchart of User Break Debugging Support Function

38.3.8 User Break Examples

(1) Match Conditions are Specified for an Instruction Fetch Cycle

- Example 1-1

  Register settings: CBR0 = H'00000013 / CRR0 = H'00002003 / CAR0 = H'00000404 / CAMR0 = H'00000000 / CBR1 = H'00000013 / CRR1 = H'00002001 / CAR1 = H'00008010 / CAMR1 = H'00000006 / CDR1 = H'00000000 / CDMR1 = H'00000000 / CETR1 = H'00000000 / CBCR = H'00000000

  Specified conditions: Independent for channels 0 and 1

  — Channel 0

  Address: H'00000404 / Address mask: H'00000000

  Bus cycle: Instruction fetch (after executing the instruction)

  ASID is not included in the conditions.

  — Channel 1

  Address: H'00008010 / Address mask: H'00000006

  Data: H'00000000 / Data mask: H'00000000 / Execution count: H'00000000

  Bus cycle: Instruction fetch (before executing instruction)

  ASID, data values, and execution count are not included in the conditions.

  With the above settings, the user break occurs after executing the instruction at address H'00000404 or before executing the instruction at address H'00008010 to H'00008016.

- Example 1-2

  Register settings: CBR0 = H'40800013 / CRR0 = H'00002000 / CAR0 = H'00037226 / CAMR0 = H'00000000 / CBR1 = H'C0700013 / CRR1 = H'00002001 / CAR1 = H'0003722E / CAMR1 = H'00000000 / CDR1 = H'00000000 / CDMR1 = H'00000000 / CETR1 = H'00000000 / CBCR = H'00000000

  Specified conditions: Channel 0 → Channel 1 sequential mode

  — Channel 0

  Address: H'00037226 / Address mask: H'00000000 / ASID: H'80

  Bus cycle: Instruction fetch (before executing the instruction)

  — Channel 1

  Address: H'0003722E / Address mask: H'00000000 / ASID: H'70

  Data: H'00000000 / Data mask: H'00000000 / Execution count: H'00000000

  Bus cycle: Instruction fetch (before executing the instruction)

  Data values and execution count are not included in the conditions.

  With the above settings, the user break occurs after executing the instruction at address H'00037226 where ASID is H'80 and before executing the instruction at address H'0003722E where ASID is H'70.

- Example 1-3

  Register settings: CBR0 = H'00000013 / CRR0 = H'00002001 / CAR0 = H'00027128 / CAMR0 = H'00000000 / CBR1 = H'00000013 / CRR1 = H'00002001 / CAR1 = H'00031415 / CAMR1 = H'00000000 / CDR1 = H'00000000 / CDMR1 = H'00000000 / CETR1 = H'00000000 / CBCR = H'00000000

  Specified conditions: Independent for channels 0 and 1

  — Channel 0

    Address: H'00027128 / Address mask: H'00000000

    Bus cycle: Instruction fetch (before executing the instruction)

    ASID is not included in the conditions.

  — Channel 1

    Address: H'00031415 / Address mask: H'00000000

    Data: H'00000000 / Data mask: H'00000000 / Execution count: H'00000000

    Bus cycle: Instruction fetch (before executing the instruction)

    ASID, data values, and execution count are not included in the conditions.

  With the above settings, the user break occurs for channel 0 before executing the instruction at address H'00027128. No user break occurs for channel 1 since the instruction fetch is executed only at even addresses.

- Example 1-4

  Register settings: CBR0 = H'40800013 / CRR0 = H'00002000 / CAR0 = H'00037226 / CAMR0 = H'00000000 / CBR1 = H'C0700013 / CRR1 = H'00002001 / CAR1 = H'0003722E / CAMR1 = H'00000000 / CDR1 = H'00000000 / CDMR1 = H'00000000 / CETR1 = H'00000000 / CBCR = H'00000000

  Specified conditions: Channel 0 → Channel 1 sequential mode

  — Channel 0

    Address: H'00037226 / Address mask: H'00000000 / ASID: H'80

    Bus cycle: Instruction fetch (before executing the instruction)

  — Channel 1

    Address: H'0003722E / Address mask: H'00000000 / ASID: H'70

    Data: H'00000000 / Data mask: H'00000000 / Execution count: H'00000000

    Bus cycle: Instruction fetch (before executing the instruction)

    Data values and execution count are not included in the conditions.

  With the above settings, the user break occurs after executing the instruction at address H'00037226 where ASID is H'80 and before executing the instruction at address H'0003722E where ASID is H'70.

- Example 1-5

  Register settings: CBR0 = H'00000013 / CRR0 = H'00002001 / CAR0 = H'00000500 / CAMR0 = H'00000000 / CBR1 = H'00000813 / CRR1 = H'00002001 / CAR1 = H'00001000 / CAMR1 = H'00000000 / CDR1 = H'00000000 / CDMR1 = H'00000000 / CETR1 = H'00000005 / CBCR = H'00000000

  Specified conditions: Independent for channels 0 and 1

  — Channel 0

  Address: H'00000500 / Address mask: H'00000000

  Bus cycle: Instruction fetch (before executing the instruction)

  ASID is not included in the conditions.

  — Channel 1

  Address: H'00001000 / Address mask: H'00000000

  Data: H'00000000 / Data mask: H'00000000 / Execution count: H'00000005

  Bus cycle: Instruction fetch (before executing the instruction)

  Execution count: 5

  ASID and data values are not included in the conditions.

  With the above settings, the user break occurs for channel 0 before executing the instruction at address H'00000500. The user break occurs for channel 1 after executing the instruction at address H'00001000 four times; before executing the instruction five times.

- Example 1-6

  Register settings: CBR0 = H'40800013 / CRR0 = H'00002003 / CAR0 = H'00008404 / CAMR0 = H'00000FFF / CBR1 = H'40700013 / CRR1 = H'00002001 / CAR1 = H'00008010 / CAMR1 = H'00000006 / CDR1 = H'00000000 / CDMR1 = H'00000000 / CETR1 = H'00000000 / CBCR = H'00000000

  Specified conditions: Independent for channels 0 and 1

  — Channel 0

  Address: H'00008404 / Address mask: H'00000FFF / ASID: H'80

  Bus cycle: Instruction fetch (after executing the instruction)

  — Channel 1

  Address: H'00008010 / Address mask: H'00000006 / ASID: H'70

  Data: H'00000000 / Data mask: H'00000000 / Execution count: H'00000000

  Bus cycle: Instruction fetch (before executing the instruction)

  Data values and execution count are not included in the conditions.

  With the above settings, the user break occurs after executing the instruction at address H'00008000 to H'00008FFE where ASID is H'80 or before executing the instruction at address H'00008010 to H'00008016 where ASID is H'70.

(2) Match Conditions are Specified for an Operand Access Cycle

- Example 2-1

  Register settings: CBR0 = H'40800023 / CRR0 = H'00002001 / CAR0 = H'00123456 / CAMR0 = H'00000000 / CBR1 = H'4070A025 / CRR1 = H'00002001 / CAR1 = H'000ABCDE / CAMR1 = H'000000FF / CDR1 = H'0000A512 / CDMR1 = H'00000000 / CETR1 = H'00000000 / CBCR = H'00000000

  Specified conditions: Independent for channels 0 and 1

  — Channel 0

  Address: H'00123456 / Address mask: H'00000000 / ASID: H'80

  Bus cycle: Operand bus, operand access, and read (operand size is not included in the conditions.)

  — Channel 1

  Address: H'000ABCDE / Address mask: H'000000FF / ASID: H'70

  Data: H'0000A512 / Data mask: H'00000000 / Execution count: H'00000000

  Bus cycle: Operand bus, operand access, write, and word size

  Execution count is not included in the conditions.

  With these settings, the user break occurs for channel 0 for the following accesses: longword read access to address H'000123454, word read access to address H'000123456, byte read access to address H'000123456 where ASID is H'80. The user break occurs for channel 1 when word H'A512 is written to address H'000ABC00 to H'000ABCFE where ASID is H'70.

38.4 Usage Notes

1. A desired break may not occur between the time when the instruction for rewriting the UBC register is executed and the time when the written value is actually reflected on the register.

   After the UBC register is updated, execute one of the following three methods.

   (1) Read the updated UBC register, and execute a branch using the RTE instruction.
   (It is not necessary that a branch using the RTE instruction is next to a reading UBC register.)

   (2) Execute the ICBI instruction for any address (including non-cacheable area).
   (It is not necessary that the ICBI instruction is next to a reading UBC register.)

   (3) Set 0 (initial value) to IRMCR.R1 before updating the UBC register and update with following sequence.

   a. Write the UBC register.

   b. Read the UBC register which is updated at a

   c. Write the value which is read at b to the UBC register.

Note: When two or more UBC registers are updated, executing these methods at each updating the UBC registers is not necessary. At only last updating the UBC register, execute one of these methods.

2. The PCB bit of the CRR0 and CRR1 registers is valid only when the instruction fetch is specified as the match condition.
3. If the sequential break conditions are set, the sequential break conditions are satisfied when the conditions for the first and second channels in the sequence are satisfied in this order. Therefore, if the conditions are set so that the conditions for channels 0 and 1 should be satisfied simultaneously for the same bus cycle, the sequential break conditions will not be satisfied, causing no break.
4. For the SLEEP instruction, do not allow the post-instruction-execution break where the instruction fetch cycle is the match condition. For the instructions preceding the SLEEP instruction by one to five instructions, do not allow the break where the operand access is the match condition.
5. If the user break and other exceptions occur for the same instruction, they are determined according to the specified priority. For the priority, refer to section 2.5, Exception Handling. If the exception having the higher priority occurs, the user break does not occur.

   — The pre-instruction-execution break is accepted prior to any other exception.

— If the post-instruction-execution break and data access break have occurred simultaneously with the re-execution type exception (including the pre-instruction-execution break) having a higher priority, only the re-execution type exception is accepted, and no condition match flags are set. When the exception handling has finished thus clearing the exception source, and when the same instruction has been executed again, the break occurs setting the corresponding flag.

— If the post-instruction-execution break or operand access break has occurred simultaneously with the completion-type exception (TRAPA) having a higher priority, then no user break occurs; however, the condition match flag is set.

6. When conditions have been satisfied simultaneously and independently for channels 0 and 1, resulting in identical SPC values for both of the breaks, the user break occurs only once. However, the condition match flags are set for both channels. For example,

   Instruction at address 110 (post-instruction-execution break for instruction fetch for channel 0) → SPC = 112, CCMFR.MF0 = 1

   Instruction at address 112 (pre-instruction-execution break for instruction fetch for channel 1) → SPC = 112, CCMFR.MF1 = 1

7. It is not allowed to set the pre-instruction-execution break or the operand break in the delayed slot instruction of the RTE instruction. And if the data value is included in the match conditions of the operand break, do not set the break for the preceding the RTE instruction by one to six instructions.

8. If the re-execution type exception and the post-instruction-execution break are in conflict for the instruction requiring two or more execution states, then the re-execution type exception occurs. Here, the CCMFR.MF0 (or CCMFR.MF1) bit may or may not be set to 1 when the break conditions have been satisfied.

Section 39 User Debugging Interface (H-UDI)

The H-UDI is a serial input/output interface which conforms to JTAG (IEEE 1149.1) with the subset. The H-UDI is used to connect emulators.

39.1 Features

The H-UDI is a serial input/output interface which conforms to JTAG (IEEE 1149.1: IEEE Standard Test Access Port and Boundary-Scan Architecture) with the subset and Renesas extension functions. The H-UDI is used to connect emulators. Do not use the JTAG functions of this interface when using an emulator. For the method of connecting the emulator, see emulator manuals.

The H-UDI has six pins, the TCK, TMS, TDI, TDO, $\overline{\text{TRST}}$, and $\overline{\text{ASEBRK}}$/BRKACK pins. The pin functions except $\overline{\text{ASEBRK}}$/BRKACK, and serial transfer protocols conform to JTAG with the subset. Furthermore, the H-UDI has ten signal pins (AUDSYNC, AUDCK, and AUDATA7 to AUDATA0) for an emulator and one signal pin (MPMD) for specifying chip mode.

In the H-UDI in this LSI, the boundary-scan test access port (TAP) controller is separated from the TAP controller for other H-UDI function control. When the $\overline{\text{TRST}}$ is asserted (including when the power is turned on), the boundary-scan TAP controller is selected. Therefore, the switching command should be input to use the H-UDI functions. The boundary-scan TAP controller cannot be accessed through the CPU.

Figure 39.1 shows a block diagram of the H-UDI.

The H-UDI circuit has TAP controllers and four registers (SDBPR, SDBSR, SDIR, and SDINT). SDBPR supports the JTAG bypass mode, SDBSR supports the JTAG boundary scan mode, SDIR is used for commands, and SDINT is used for H-UDI interrupts. SDIR can be directly accessed through the TDI and TDO pins.

Without reset pins of the chip, the TAP controller, control registers, and boundary-scan TAP controller are reset when the $\overline{\text{TRST}}$ pin is set to low or when five or more TCK cycles are elapsed after TMS is set to 1. The other circuits are reset in a normal reset period, and initialized.

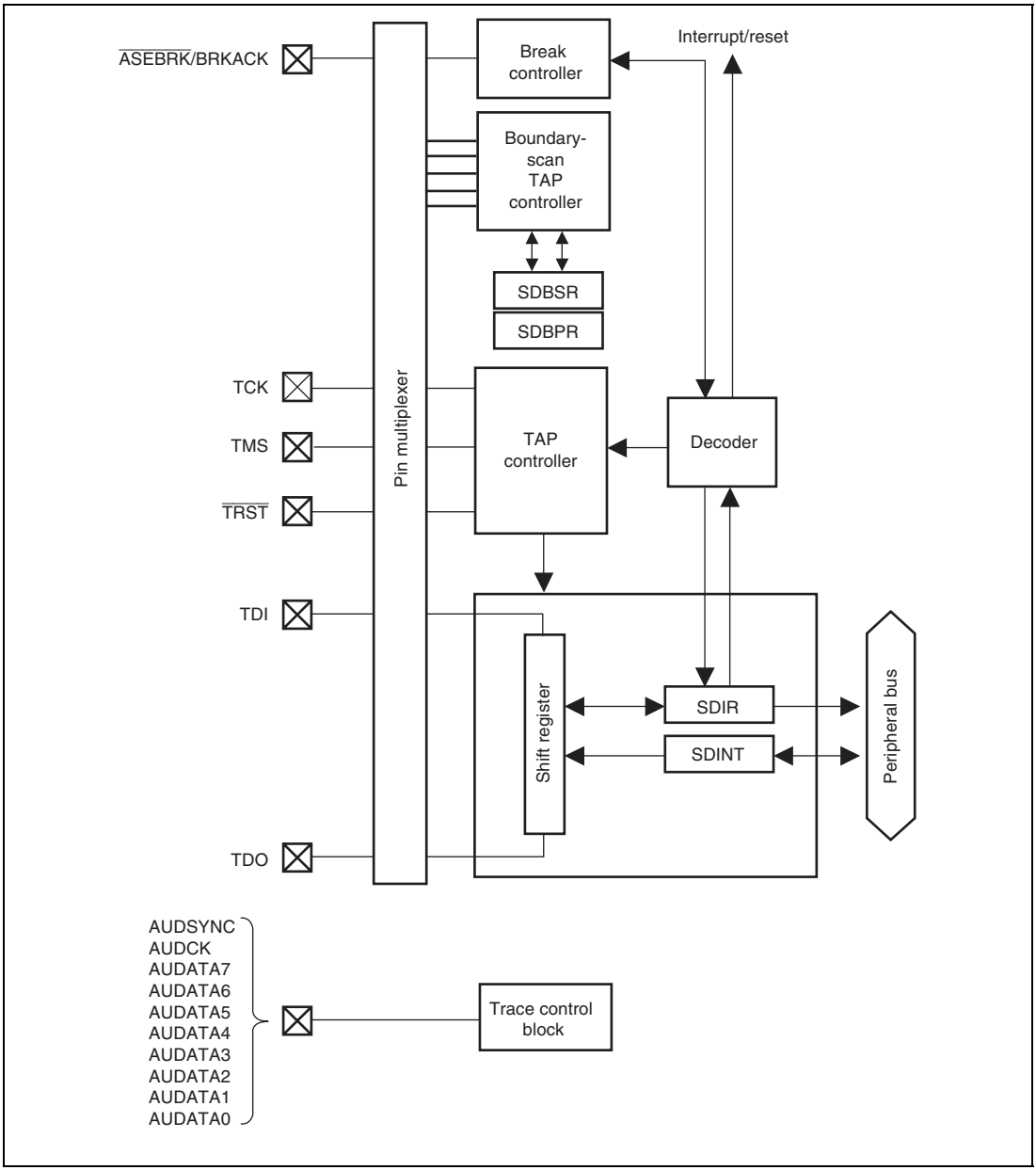


Figure 39.1 H-UDI Block Diagram

39.2 Pin Configuration

Table 39.1 shows the pin configuration for the H-UDI.

Table 39.1 H-UDI Pin Configuration

Pin Name	Function	I/O	Description	When Not in Use
TCK*[1]	Clock	Input	Functions as the serial clock input pin stipulated in the JTAG standard. Data input to the H-UDI circuit via the TDI (data input) pin or data output via the TDO (data output) pin is performed in synchronization with this signal.	Open
TMS*[1]	Mode	Input	Changing this signal in synchronization with the TCK signal determines the significance of data input via the TDI pin. Its protocol conforms to the subset of the JTAG standard (IEEE standard 1149.1).	Open
$\overline{\text{TRST}}$*[1]*[2]*[3]	Reset	Input	This signal is received asynchronously with a TCK signal. Asserting this signal resets the JTAG interface circuit. When power is supplied, the $\overline{\text{TRST}}$ pin should be asserted for a given period regardless of whether or not the JTAG function is used, which differs from the IEEE standard. If you are not using an emulator, fix this pin to the low level.	Tied to ground or connected to the $\overline{\text{PRESET}}$ pin
TDI*[1]	Data input	Input	Data is sent to the H-UDI circuit by changing this signal in synchronization with the TCK signal.	Open
TDO	Data output	Output	Data is read from the H-UDI circuit by reading from this signal in synchronization with the TCK signal.	Open
$\overline{\text{ASEBRK}}$/ BRKACK*[1]	Pin for an emulator	I/O	Pin for an emulator	Open
AUDSYNC, AUDCK, AUDATA7 to AUDATA0	Pins for an emulator	Output	Pins for an emulator	Open
MPMD	Chip mode specification	Input	Specifies whether this LSI operates in emulation support mode (MPMD = 0) or in independent chip mode (MPMD = 1).	Fixed at 3.3 V

Notes: 1. These pins are pulled up internally in this LSI. When designing a board that can use an emulator or when using interrupts and resets through the H-UDI, external pull-up resistors may be attached to these pins without problem.

2. When designing a board that can use an emulator or when using interrupts and resets through the H-UDI, the $\overline{\text{TRST}}$ pin circuit should be designed so that it can be controlled independently and can be set low during the period the $\overline{\text{PRESET}}$ pin is asserted low at power-on.

3. This pin should be either tied to ground or connected to the $\overline{\text{PRESET}}$ pin (or another pin which operates in the same manner as the $\overline{\text{PRESET}}$ pin). Note, however, that the following problem occurs when the $\overline{\text{TRST}}$ pin is tied to ground. Since the $\overline{\text{TRST}}$ pin is pulled up within this LSI, a weak current will flow if the pin is tied to ground externally. The value of the current is determined by the pull-up resistor of the port pin. Although this current does not affect the operation of this LSI, it does consume unnecessary power.

The TCK frequency or the CPG of this LSI should be set so that the TCK frequency is lower than the peripheral clock frequency of this LSI. For details on setting the CPG, refer to section 8, Clock Pulse Generator (CPG).

39.3 Register Description

The H-UDI has the following registers.

Do not write to any addresses other than listed below. Operations cannot be guaranteed if writing is attempted.

Table 39.2 Register Configuration (1)

		CPU Side			
Register Name	**Abbreviation**	**R/W**	**Address**	**Access Size**	**Sync Clock**
Instruction register	SDIR	R	H'FC11 0000	16	clkp1
Interrupt source register	SDINT	R/W	H'FC11 0018	16	clkp1
Boundary scan register	SDBSR	—	—	—	—
Bypass register	SDBPR	—	—	—	—

Table 39.3 Register Configuration (2)

		H-UDI Side		
Register Name	**Abbreviation**	**R/W**	**Access Size**	**Sync Clock**
Instruction register	SDIR	R/W*[1]	32	clkp1
Interrupt source register	SDINT	R/W*[2]	32	clkp1
Boundary scan register	SDBSR	R/W	—	—
Bypass register	SDBPR	R/W	1	—

Notes: 1. When reading from the H-UDI, the value is always H'FFFF FFFD.
2. Only 1 can be written to the LSB by the H-UDI interrupt command.

Table 39.4 Register States in Each Operating Mode

Register Name	**Abbreviation**	**Power-on Reset**	**Manual Reset**	**Sleep**	**Software Standby**	**Module Standby**	**Deep Standby**
Instruction register	SDIR	H'0EFF	Retained	Retained	Retained	Retained	Retained*
Interrupt source register	SDINT	H'0000	Retained	Retained	Retained	Retained	Retained*

Note: * This is the case when the MPMD pin is at the low level. When it is at the high level, the registers enter the same states as in a power-on reset.

Legend for symbols used in register descriptions:

Initial value:	The value in the register after reset
— :	Undefined value
R/W:	The bit is readable and writable. The written value can be read.
R/WC0:	The bit is readable and writable. Writing 0 to the bit initializes the bit. Writing 1 to the bit is ignored.
R:	Read-only.

39.3.1 Instruction Register (SDIR)

SDIR is a 16-bit register that is read-only for the CPU. Values (commands) are set in this register from the serial input (TDI). SDIR is initialized when the $\overline{\text{TRST}}$ pin is set to the low level or when the TAP controller is in the Test-Logic-Reset state. When writing to this register from the H-UDI, it can be written to regardless of the CPU mode. When reading from this register from the H-UDI, a fixed value (H'FFFF FFFD) will be read. Note that operation is not guaranteed if a reserved command is set into this register.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TI								—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	1	1	1	0	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	TI	00001110	R	Test instruction bits 7 to 0 0110xxxx: H-UDI reset negate 0111xxxx: H-UDI reset assert 101xxxxx: H-UDI interrupt 00001110: Initial state Other than above: Setting prohibited
7 to 0	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.

39.3.2 Interrupt Source Register (SDINT)

SDINT is a 16-bit register that can be read from or written to by the CPU. Specifying an H-UDI interrupt command in SDIR with the H-UDI pin (Update-IR) sets the INTREQ bit to 1. While an H-UDI interrupt command is set in SDIR, SDINT is connected between H-UDI pins TDI and TDO, and can be read as a 32-bit register. In this case, the upper 16 bits will be 0 and the lower 16 bits represent the SDINT value.

Only 0 can be written to the INTREQ bit by the CPU. Interrupt requests will continue to be generated while this bit is set to 1. Therefore, this bit must be cleared by the interrupt handler, and this bit must be read again to confirm that it has been cleared to 0. This register is initialized when the $\overline{\text{TRST}}$ pin is set to the low level or when the TAP controller is in the Test-Logic-Reset state.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	INTREQ
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/WC0

Bit	Bit Name	Initial Value	R/W	Description
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	INTREQ	0	R/W	Interrupt Request Indicates whether or not there is an interrupt request due to an H-UDI interrupt command. The CPU can clear the interrupt request by writing 0 to this bit. If 1 is written to this bit, its immediately prior value will be retained.

39.3.3 Bypass Register (SDBPR)

SDBPR is a 1-bit register used for supporting the JTAG bypass mode. When the BYPASS command is set in the boundary-scan TAP controller, SDBPR is connected between the TDI and TDO pins. SDBPR cannot be accessed by the CPU. This register is not initialized by a power-on reset or when the $\overline{\text{TRST}}$ pin is set to the low level. This register is initialized to 0 only when the TAP controller is in the Capture-DR state.

39.3.4 Boundary Scan Register (SDBSR)

SDBSR is a register that supports the JTAG boundary scan mode. SDBSR is a shift register that is located on the pad to control the input/output pins. By using the SAMPLE/PRELOAD and EXTEST commands, this register can perform the boundary scan test that supports the JTAG standard (IEEE 1149.1) with the subset. This register cannot be accessed through the CPU, regardless of the chip's mode of operation.

Table 39.5 Register Configuration of Boundary Scan Register

Bit	Pin Name	Type
	From TDI	
641	$\overline{\text{ASEBRK}}$/ACK	OUTPUT
640	$\overline{\text{ASEBRK}}$/ACK	CONTROL
639	$\overline{\text{ASEBRK}}$/ACK	INPUT
638	GP/CLKOUT	OUTPUT
637	GP/CLKOUT	CONTROL
636	GP/CLKOUT	OBSERVE_ONLY
635	GP/D0/SD0_DAT0_A/MMC_D0_A/ ST1_D[3]/NAF0_A	OUTPUT
634	GP/D0/SD0_DAT0_A/MMC_D0_A/ ST1_D[3]/NAF0_A	CONTROL
633	GP/D0/SD0_DAT0_A/MMC_D0_A/ ST1_D[3]/NAF0_A	INPUT
632	GP/D1/SD0_DAT1_A/MMC_D1_A/ ST1_D[4]/NAF1_A	OUTPUT
631	GP/D1/SD0_DAT1_A/MMC_D1_A/ ST1_D[4]/NAF1_A	CONTROL
630	GP/D1/SD0_DAT1_A/MMC_D1_A/ ST1_D[4]/NAF1_A	INPUT
629	GP/D2/SD0_DAT2_A/MMC_D2_A/ ST1_D[5]/NAF2_A	OUTPUT
628	GP/D2/SD0_DAT2_A/MMC_D2_A/ ST1_D[5]/NAF2_A	CONTROL
627	GP/D2/SD0_DAT2_A/MMC_D2_A/ ST1_D[5]/NAF2_A	INPUT
626	GP/D3/SD0_DAT3_A/MMC_D3_A/ ST1_D[6]/NAF3_A	OUTPUT
625	GP/D3/SD0_DAT3_A/MMC_D3_A/ ST1_D[6]/NAF3_A	CONTROL
624	GP/D3/SD0_DAT3_A/MMC_D3_A/ ST1_D[6]/NAF3_A	INPUT
623	GP/D4/SD0_CD_A/MMC_D4_A/ ST1_D[7]/NAF4_A	OUTPUT
622	GP/D4/SD0_CD_A/MMC_D4_A/ ST1_D[7]/NAF4_A	CONTROL
621	GP/D4/SD0_CD_A/MMC_D4_A/ ST1_D[7]/NAF4_A	INPUT
620	GP/D5/SD0_WP_A/MMC_D5_A/NAF5_A	OUTPUT
619	GP/D5/SD0_WP_A/MMC_D5_A/NAF5_A	CONTROL
618	GP/D5/SD0_WP_A/MMC_D5_A/NAF5_A	INPUT
617	GP/D6/RSPI_RSPCK_A/MMC_D6_A/ QSPCLK_A/NAF6_A	OUTPUT
616	GP/D6/RSPI_RSPCK_A/MMC_D6_A/ QSPCLK_A/NAF6_A	CONTROL
615	GP/D6/RSPI_RSPCK_A/MMC_D6_A/ QSPCLK_A/NAF6_A	INPUT
614	GP/D7/RSPI_SSL_A/MMC_D7_A/ QSSL_A/NAF7_A	OUTPUT
613	GP/D7/RSPI_SSL_A/MMC_D7_A/ QSSL_A/NAF7_A	CONTROL
612	GP/D7/RSPI_SSL_A/MMC_D7_A/ QSSL_A/NAF7_A	INPUT
611	GP/D8/SD0_CLK_A/MMC_CLK_A/ QIO2_A/$\overline{\text{FCE\_A}}$/ET0_GTX_CLK_B	OUTPUT

Bit	Pin Name	Type
610	GP/D8/SD0_CLK_A/MMC_CLK_A/ QIO2_A/$\overline{\text{FCE\_A}}$/ET0_GTX_CLK_B	CONTROL
609	GP/D8/SD0_CLK_A/MMC_CLK_A/ QIO2_A/$\overline{\text{FCE\_A}}$/ET0_GTX_CLK_B	INPUT
608	GP/D9/SD0_CMD_A/MMC_CMD_A/ QIO3_A/FCLE_A/ET0_ETXD1_B	OUTPUT
607	GP/D9/SD0_CMD_A/MMC_CMD_A/ QIO3_A/FCLE_A/ET0_ETXD1_B	CONTROL
606	GP/D9/SD0_CMD_A/MMC_CMD_A/ QIO3_A/FCLE_A/ET0_ETXD1_B	INPUT
605	GP/D10/RSPI_MOSI_A/TXCLK0/QMO/ QIO0_A/FALE_A/ET0_ETXD2_B	OUTPUT
604	GP/D10/RSPI_MOSI_A/TXCLK0/QMO/ QIO0_A/FALE_A/ET0_ETXD2_B	CONTROL
603	GP/D10/RSPI_MOSI_A/TXCLK0/QMO/ QIO0_A/FALE_A/ET0_ETXD2_B	INPUT
602	GP/D11/RSPI_MISO_A/SAMPOINT0/ QMI/QIO1_A/$\overline{\text{FRE\_A}}$/ET0_ETXD3_B	OUTPUT
601	GP/D11/RSPI_MISO_A/SAMPOINT0/ QMI/QIO1_A/$\overline{\text{FRE\_A}}$/ET0_ETXD3_B	CONTROL
600	GP/D11/RSPI_MISO_A/SAMPOINT0/ QMI/QIO1_A/$\overline{\text{FRE\_A}}$/ET0_ETXD3_B	INPUT
599	GP/D12/EN_NEXTQU0/$\overline{\text{FWE\_A}}$/ ET0_ETXD5_B	OUTPUT
598	GP/D12/EN_NEXTQU0/$\overline{\text{FWE\_A}}$/ ET0_ETXD5_B	CONTROL
597	GP/D12/EN_NEXTQU0/$\overline{\text{FWE\_A}}$/ ET0_ETXD5_B	INPUT
596	GP/D13/RX2_B/TXCLK1/FRB_A/ ET0_ETXD6_B	OUTPUT
595	GP/D13/RX2_B/TXCLK1/FRB_A/ ET0_ETXD6_B	CONTROL
594	GP/D13/RX2_B/TXCLK1/FRB_A/ ET0_ETXD6_B	INPUT
593	GP/D14/TX2_B/SAMPOINT1/ ET0_TX_CLK_B	OUTPUT
592	GP/D14/TX2_B/SAMPOINT1/ ET0_TX_CLK_B	CONTROL
591	GP/D14/TX2_B/SAMPOINT1/ ET0_TX_CLK_B	INPUT
590	GP/D15/SCK2_B/EN_NEXTQU1	OUTPUT
589	GP/D15/SCK2_B/EN_NEXTQU1	CONTROL
588	GP/D15/SCK2_B/EN_NEXTQU1	INPUT
587	GP/A0/ST0_CLKIN/LCD_DATA0_A/ TCLKA_C	OUTPUT
586	GP/A0/ST0_CLKIN/LCD_DATA0_A/ TCLKA_C	CONTROL
585	GP/A0/ST0_CLKIN/LCD_DATA0_A/ TCLKA_C	INPUT
584	GP/A1/ST0_REQ/LCD_DATA1_A/ TCLKB_C	OUTPUT
583	GP/A1/ST0_REQ/LCD_DATA1_A/ TCLKB_C	CONTROL
582	GP/A1/ST0_REQ/LCD_DATA1_A/ TCLKB_C	INPUT
581	GP/A2/ST0_SYC/LCD_DATA2_A/ TCLKC_C	OUTPUT
580	GP/A2/ST0_SYC/LCD_DATA2_A/ TCLKC_C	CONTROL
579	GP/A2/ST0_SYC/LCD_DATA2_A/ TCLKC_C	INPUT
578	GP/A3/ST0_VLD/LCD_DATA3_A/ TCLKD_C	OUTPUT
577	GP/A3/ST0_VLD/LCD_DATA3_A/ TCLKD_C	CONTROL
576	GP/A3/ST0_VLD/LCD_DATA3_A/ TCLKD_C	INPUT
575	GP/A4/ST0_D[0]/LCD_DATA4_A/ TIOC0A_C	OUTPUT
574	GP/A4/ST0_D[0]/LCD_DATA4_A/ TIOC0A_C	CONTROL
573	GP/A4/ST0_D[0]/LCD_DATA4_A/ TIOC0A_C	INPUT
572	GP/A5/ST0_D[1]/LCD_DATA5_A/ TIOC0B_C	OUTPUT

Bit	Pin Name	Type
571	GP/A5/ST0_D[1]/LCD_DATA5_A/ TIOC0B_C	CONTROL
570	GP/A5/ST0_D[1]/LCD_DATA5_A/ TIOC0B_C	INPUT
569	GP/A6/ST0_D[2]/LCD_DATA6_A/ TIOC0C_C	OUTPUT
568	GP/A6/ST0_D[2]/LCD_DATA6_A/ TIOC0C_C	CONTROL
567	GP/A6/ST0_D[2]/LCD_DATA6_A/ TIOC0C_C	INPUT
566	GP/A7/ST0_D[3]/LCD_DATA7_A/ TIOC0D_C	OUTPUT
565	GP/A7/ST0_D[3]/LCD_DATA7_A/ TIOC0D_C	CONTROL
564	GP/A7/ST0_D[3]/LCD_DATA7_A/ TIOC0D_C	INPUT
563	GP/A8/ST0_D[4]/LCD_DATA8_A/ TIOC1A_C	OUTPUT
562	GP/A8/ST0_D[4]/LCD_DATA8_A/ TIOC1A_C	CONTROL
561	GP/A8/ST0_D[4]/LCD_DATA8_A/ TIOC1A_C	INPUT
560	GP/A9/ST0_D[5]/LCD_DATA9_A/ TIOC1B_C	OUTPUT
559	GP/A9/ST0_D[5]/LCD_DATA9_A/ TIOC1B_C	CONTROL
558	GP/A9/ST0_D[5]/LCD_DATA9_A/ TIOC1B_C	INPUT
557	GP/A10/ST0_D[6]/LCD_DATA10_A/ TIOC2A_C	OUTPUT
556	GP/A10/ST0_D[6]/LCD_DATA10_A/ TIOC2A_C	CONTROL
555	GP/A10/ST0_D[6]/LCD_DATA10_A/ TIOC2A_C	INPUT
554	GP/A11/ST0_D[7]/LCD_DATA11_A/ TIOC2B_C	OUTPUT
553	GP/A11/ST0_D[7]/LCD_DATA11_A/ TIOC2B_C	CONTROL
552	GP/A11/ST0_D[7]/LCD_DATA11_A/ TIOC2B_C	INPUT
551	GP/A12/LCD_DATA12_A/TIOC3A_C	OUTPUT
550	GP/A12/LCD_DATA12_A/TIOC3A_C	CONTROL
549	GP/A12/LCD_DATA12_A/TIOC3A_C	INPUT
548	GP/A13/LCD_DATA13_A/TIOC3B_C	OUTPUT
547	GP/A13/LCD_DATA13_A/TIOC3B_C	CONTROL
546	GP/A13/LCD_DATA13_A/TIOC3B_C	INPUT
545	GP/A14/LCD_DATA14_A/TIOC3C_C	OUTPUT
544	GP/A14/LCD_DATA14_A/TIOC3C_C	CONTROL
543	GP/A14/LCD_DATA14_A/TIOC3C_C	INPUT
542	GP/$\overline{BS}$	OUTPUT
541	GP/$\overline{BS}$	CONTROL
540	GP/$\overline{BS}$	INPUT
539	GP/A15/ST0_VCO_CLKIN/ LCD_DATA15_A/TIOC3D_C	OUTPUT
538	GP/A15/ST0_VCO_CLKIN/ LCD_DATA15_A/TIOC3D_C	CONTROL
537	GP/A15/ST0_VCO_CLKIN/ LCD_DATA15_A/TIOC3D_C	INPUT
536	GP/A16/ST0_PWM/LCD_DON_A/ TIOC4A_C	OUTPUT
535	GP/A16/ST0_PWM/LCD_DON_A/ TIOC4A_C	CONTROL
534	GP/A16/ST0_PWM/LCD_DON_A/ TIOC4A_C	INPUT
533	GP/A17/ST1_VCO_CLKIN/ LCD_CL1_A/TIOC4B_C	OUTPUT
532	GP/A17/ST1_VCO_CLKIN/ LCD_CL1_A/TIOC4B_C	CONTROL
531	GP/A17/ST1_VCO_CLKIN/ LCD_CL1_A/TIOC4B_C	INPUT
530	GP/$\overline{CS0}$	OUTPUT
529	GP/$\overline{CS0}$	CONTROL
528	GP/$\overline{CS0}$	OBSERVE_ONLY

Bit	Pin Name	Type
527	GP/$\overline{CS1}$/A26/QIO3_B	OUTPUT
526	GP/$\overline{CS1}$/A26/QIO3_B	CONTROL
525	GP/$\overline{CS1}$/A26/QIO3_B	OBSERVE_ONLY
524	GP/A18/ST1_PWM/LCD_CL2_A/ TIOC4C_C	OUTPUT
523	GP/A18/ST1_PWM/LCD_CL2_A/ TIOC4C_C	CONTROL
522	GP/A18/ST1_PWM/LCD_CL2_A/ TIOC4C_C	INPUT
521	GP/A19/ST1_CLKIN/LCD_CLK_A/ TIOC4D_C	OUTPUT
520	GP/A19/ST1_CLKIN/LCD_CLK_A/ TIOC4D_C	CONTROL
519	GP/A19/ST1_CLKIN/LCD_CLK_A/ TIOC4D_C	INPUT
518	GP/A20/ST1_REQ/LCD_FLM_A	OUTPUT
517	GP/A20/ST1_REQ/LCD_FLM_A	CONTROL
516	GP/A20/ST1_REQ/LCD_FLM_A	INPUT
515	GP/A21/ST1_SYC/LCD_VCPWC_A	OUTPUT
514	GP/A21/ST1_SYC/LCD_VCPWC_A	CONTROL
513	GP/A21/ST1_SYC/LCD_VCPWC_A	INPUT
512	GP/A22/ST1_VLD/LCD_VEPWC_A	OUTPUT
511	GP/A22/ST1_VLD/LCD_VEPWC_A	CONTROL
510	GP/A22/ST1_VLD/LCD_VEPWC_A	INPUT
509	GP/A23/ST1_D[0]/LCD_M_DISP_A	OUTPUT
508	GP/A23/ST1_D[0]/LCD_M_DISP_A	CONTROL
507	GP/A23/ST1_D[0]/LCD_M_DISP_A	INPUT
506	GP/A24/RX2_D/ST1_D[1]	OUTPUT
505	GP/A24/RX2_D/ST1_D[1]	CONTROL
504	GP/A24/RX2_D/ST1_D[1]	INPUT
503	GP/A25/TX2_D/ST1_D[2]	OUTPUT
502	GP/A25/TX2_D/ST1_D[2]	CONTROL
501	GP/A25/TX2_D/ST1_D[2]	INPUT
500	GP/TX0_A/HSPI_TX_A	OUTPUT
499	GP/TX0_A/HSPI_TX_A	CONTROL
498	GP/TX0_A/HSPI_TX_A	INPUT
497	GP/$\overline{RD}$	OUTPUT
496	GP/$\overline{RD}$	CONTROL
495	GP/$\overline{RD}$	INPUT
494	GP/$\overline{WE0}$	OUTPUT
493	GP/$\overline{WE0}$	CONTROL
492	GP/$\overline{WE0}$	INPUT
491	GP/$\overline{EX\_CS2}$/TX3_B/$\overline{ATACS1}$/ QSPCLK_B/ET0_GTX_CLK_A	OUTPUT
490	GP/$\overline{EX\_CS2}$/TX3_B/$\overline{ATACS1}$/ QSPCLK_B/ET0_GTX_CLK_A	CONTROL
489	GP/$\overline{EX\_CS2}$/TX3_B/$\overline{ATACS1}$/ QSPCLK_B/ET0_GTX_CLK_A	INPUT
488	GP/RD/$\overline{WR}$/TCLK0/CAN_CLK_B/ ET0_ETXD4	OUTPUT
487	GP/RD/$\overline{WR}$/TCLK0/CAN_CLK_B/ ET0_ETXD4	CONTROL
486	GP/RD/$\overline{WR}$/TCLK0/CAN_CLK_B/ ET0_ETXD4	INPUT
485	GP/$\overline{EX\_CS3}$/SD1_CD_A/$\overline{ATARD}$/QMO/ QIO0_B/ET0_ETXD1_A	OUTPUT
484	GP/$\overline{EX\_CS3}$/SD1_CD_A/$\overline{ATARD}$/QMO/ QIO0_B/ET0_ETXD1_A	CONTROL
483	GP/$\overline{EX\_CS3}$/SD1_CD_A/$\overline{ATARD}$/QMO/ QIO0_B/ET0_ETXD1_A	INPUT
482	GP/$\overline{WE1}$	OUTPUT
481	GP/$\overline{WE1}$	CONTROL
480	GP/$\overline{WE1}$	INPUT
479	GP/$\overline{EX\_CS5}$/SD1_CMD_A/$\overline{ATADIR}$/ QSSL_B/ET0_ETXD3_A	OUTPUT
478	GP/$\overline{EX\_CS5}$/SD1_CMD_A/$\overline{ATADIR}$/ QSSL_B/ET0_ETXD3_A	CONTROL
477	GP/$\overline{EX\_CS5}$/SD1_CMD_A/$\overline{ATADIR}$/ QSSL_B/ET0_ETXD3_A	INPUT
476	GP/EX_WAIT0/TCLK1_B	OUTPUT

Bit	Pin Name	Type
475	GP/EX_WAIT0/TCLK1_B	CONTROL
474	GP/EX_WAIT0/TCLK1_B	INPUT
473	GP/$\overline{\text{EX\_CS0}}$	OUTPUT
472	GP/$\overline{\text{EX\_CS0}}$	CONTROL
471	GP/$\overline{\text{EX\_CS0}}$	INPUT
470	GP/$\overline{\text{EX\_CS1}}$/RX3_B/$\overline{\text{ATACS0}}$/QIO2_B/ ET0_ETXD0	OUTPUT
469	GP/$\overline{\text{EX\_CS1}}$/RX3_B/$\overline{\text{ATACS0}}$/QIO2_B/ ET0_ETXD0	CONTROL
468	GP/$\overline{\text{EX\_CS1}}$/RX3_B/$\overline{\text{ATACS0}}$/QIO2_B/ ET0_ETXD0	INPUT
467	GP/EX_WAIT2/SD1_DAT1_A/DACK2/ CAN1_RX_C/ET0_MAGIC_C/ ET0_ETXD6_A	OUTPUT
466	GP/EX_WAIT2/SD1_DAT1_A/DACK2/ CAN1_RX_C/ET0_MAGIC_C/ ET0_ETXD6_A	CONTROL
465	GP/EX_WAIT2/SD1_DAT1_A/DACK2/ CAN1_RX_C/ET0_MAGIC_C/ ET0_ETXD6_A	INPUT
464	GP/DACK0/SD1_DAT3_A/ET0_TX_ER	OUTPUT
463	GP/DACK0/SD1_DAT3_A/ET0_TX_ER	CONTROL
462	GP/DACK0/SD1_DAT3_A/ET0_TX_ER	INPUT
461	GP/DREQ0/SD1_CLK_A/ET0_TX_EN	OUTPUT
460	GP/DREQ0/SD1_CLK_A/ET0_TX_EN	CONTROL
459	GP/DREQ0/SD1_CLK_A/ET0_TX_EN	INPUT
458	GP/EX_WAIT1/SD1_DAT0_A/DREQ2/ CAN1_TX_C/ET0_LINK_C/ ET0_ETXD5_A	OUTPUT
457	GP/EX_WAIT1/SD1_DAT0_A/DREQ2/ CAN1_TX_C/ET0_LINK_C/ ET0_ETXD5_A	CONTROL
456	GP/EX_WAIT1/SD1_DAT0_A/DREQ2/ CAN1_TX_C/ET0_LINK_C/ ET0_ETXD5_A	INPUT
455	GP/REF125CK/$\overline{\text{ADTRG}}$/RX5_C	OUTPUT
454	GP/REF125CK/$\overline{\text{ADTRG}}$/RX5_C	CONTROL
453	GP/REF125CK/$\overline{\text{ADTRG}}$/RX5_C	INPUT
452	GP/REF50CK/$\overline{\text{CTS1\_E}}$/$\overline{\text{HCTS0\_D}}$	OUTPUT
451	GP/REF50CK/$\overline{\text{CTS1\_E}}$/$\overline{\text{HCTS0\_D}}$	CONTROL
450	GP/REF50CK/$\overline{\text{CTS1\_E}}$/$\overline{\text{HCTS0\_D}}$	INPUT
449	GP/IRQ0_A/$\overline{\text{PCMOE}}$/HSPI_TX_B/ RX3_E/ET0_ERXD0	OUTPUT
448	GP/IRQ0_A/$\overline{\text{PCMOE}}$/HSPI_TX_B/ RX3_E/ET0_ERXD0	CONTROL
447	GP/IRQ0_A/$\overline{\text{PCMOE}}$/HSPI_TX_B/ RX3_E/ET0_ERXD0	INPUT
446	GP/DRACK0/SD1_DAT2_A/$\overline{\text{ATAG}}$/ TCLK1_A/ET0_ETXD7	OUTPUT
445	GP/DRACK0/SD1_DAT2_A/$\overline{\text{ATAG}}$/ TCLK1_A/ET0_ETXD7	CONTROL
444	GP/DRACK0/SD1_DAT2_A/$\overline{\text{ATAG}}$/ TCLK1_A/ET0_ETXD7	INPUT
443	GP/DREQ1/HSPI_CLK_B/RX4_B/ ET0_PHY_INT_C/ET0_TX_CLK_A	OUTPUT
442	GP/DREQ1/HSPI_CLK_B/RX4_B/ ET0_PHY_INT_C/ET0_TX_CLK_A	CONTROL
441	GP/DREQ1/HSPI_CLK_B/RX4_B/ ET0_PHY_INT_C/ET0_TX_CLK_A	INPUT
440	GP/$\overline{\text{HCTS0\_A}}$/$\overline{\text{CTS1\_A}}$/VI0_FIELD/ RMII0_RXD1_A/ET0_ERXD7	OUTPUT
439	GP/$\overline{\text{HCTS0\_A}}$/$\overline{\text{CTS1\_A}}$/VI0_FIELD/ RMII0_RXD1_A/ET0_ERXD7	CONTROL
438	GP/$\overline{\text{HCTS0\_A}}$/$\overline{\text{CTS1\_A}}$/VI0_FIELD/ RMII0_RXD1_A/ET0_ERXD7	INPUT
437	GP/IRQ1_A/$\overline{\text{PCMWE}}$/HSPI_RX_B/ TX3_E/ET0_ERXD1	OUTPUT
436	GP/IRQ1_A/$\overline{\text{PCMWE}}$/HSPI_RX_B/ TX3_E/ET0_ERXD1	CONTROL
435	GP/IRQ1_A/$\overline{\text{PCMWE}}$/HSPI_RX_B/ TX3_E/ET0_ERXD1	INPUT
434	GP/IRQ2_A/$\overline{\text{CTS0\_A}}$/$\overline{\text{HCTS0\_B}}$/ ET0_ERXD2_A	OUTPUT

Bit	Pin Name	Type	Bit	Pin Name	Type
433	GP/IRQ2_A/CTS0_A/HCTS0_B/ ET0_ERXD2_A	CONTROL	414	GP/CTS0_B/VI0_DATA2/VI0_B2/ RMII0_MDIO_A/ET0_MDC	INPUT
432	GP/IRQ2_A/CTS0_A/HCTS0_B/ ET0_ERXD2_A	INPUT	413	GP/HRTS0_A/RTS1_A/VI0_HSYNC/ RMII0_TXD_EN_A/ET0_RX_DV	OUTPUT
431	GP/DACK1/HSPI_CS_B/TX4_B/ ET0_RX_CLK_A	OUTPUT	412	GP/HRTS0_A/RTS1_A/VI0_HSYNC/ RMII0_TXD_EN_A/ET0_RX_DV	CONTROL
430	GP/DACK1/HSPI_CS_B/TX4_B/ ET0_RX_CLK_A	CONTROL	411	GP/HRTS0_A/RTS1_A/VI0_HSYNC/ RMII0_TXD_EN_A/ET0_RX_DV	INPUT
429	GP/DACK1/HSPI_CS_B/TX4_B/ ET0_RX_CLK_A	INPUT	410	GP/HSCK0_A/SCK1_A/VI0_VSYNC/ RMII0_RX_ER_A/ET0_RX_ER	OUTPUT
428	GP/RX0_A/HSPI_RX_A/ RMII0_RXD0_A/ET0_ERXD6	OUTPUT	409	GP/HSCK0_A/SCK1_A/VI0_VSYNC/ RMII0_RX_ER_A/ET0_RX_ER	CONTROL
427	GP/RX0_A/HSPI_RX_A/ RMII0_RXD0_A/ET0_ERXD6	CONTROL	408	GP/HSCK0_A/SCK1_A/VI0_VSYNC/ RMII0_RX_ER_A/ET0_RX_ER	INPUT
426	GP/RX0_A/HSPI_RX_A/ RMII0_RXD0_A/ET0_ERXD6	INPUT	407	GP/EX_CS4/SD1_WP_A/ATAWR/QMI/ QIO1_B/ET0_ETXD2_A	OUTPUT
425	GP/SCK0_A/HSPI_CS_A/VI0_CLKENB/ RMII0_TXD1_A/ET0_ERXD5	OUTPUT	406	GP/EX_CS4/SD1_WP_A/ATAWR/QMI/ QIO1_B/ET0_ETXD2_A	CONTROL
424	GP/SCK0_A/HSPI_CS_A/VI0_CLKENB/ RMII0_TXD1_A/ET0_ERXD5	CONTROL	405	GP/EX_CS4/SD1_WP_A/ATAWR/QMI/ QIO1_B/ET0_ETXD2_A	INPUT
423	GP/SCK0_A/HSPI_CS_A/VI0_CLKENB/ RMII0_TXD1_A/ET0_ERXD5	INPUT	404	GP/RX1_B/VI0_DATA5/VI0_B5/ ET0_MAGIC_A	OUTPUT
422	GP/SCIF_CLK_A/HSPI_CLK_A/ VI0_CLK/RMII0_TXD0_A/ET0_ERXD4	OUTPUT	403	GP/RX1_B/VI0_DATA5/VI0_B5/ ET0_MAGIC_A	CONTROL
421	GP/SCIF_CLK_A/HSPI_CLK_A/ VI0_CLK/RMII0_TXD0_A/ET0_ERXD4	CONTROL	402	GP/RX1_B/VI0_DATA5/VI0_B5/ ET0_MAGIC_A	INPUT
420	GP/SCIF_CLK_A/HSPI_CLK_A/ VI0_CLK/RMII0_TXD0_A/ET0_ERXD4	INPUT	401	GP/TX1_B/VI0_DATA6/VI0_G0/ ET0_PHY_INT_A	OUTPUT
419	GP/IRQ3_A/RTS0_A/HRTS0_B/ ET0_ERXD3_A	OUTPUT	400	GP/TX1_B/VI0_DATA6/VI0_G0/ ET0_PHY_INT_A	CONTROL
418	GP/IRQ3_A/RTS0_A/HRTS0_B/ ET0_ERXD3_A	CONTROL	399	GP/TX1_B/VI0_DATA6/VI0_G0/ ET0_PHY_INT_A	INPUT
417	GP/IRQ3_A/RTS0_A/HRTS0_B/ ET0_ERXD3_A	INPUT	398	GP/RTS0_B/VI0_DATA3/VI0_B3/ ET0_MDIO_A	OUTPUT
416	GP/CTS0_B/VI0_DATA2/VI0_B2/ RMII0_MDIO_A/ET0_MDC	OUTPUT	397	GP/RTS0_B/VI0_DATA3/VI0_B3/ ET0_MDIO_A	CONTROL
415	GP/CTS0_B/VI0_DATA2/VI0_B2/ RMII0_MDIO_A/ET0_MDC	CONTROL	396	GP/RTS0_B/VI0_DATA3/VI0_B3/ ET0_MDIO_A	INPUT

Bit	Pin Name	Type
395	GP/HTX0_A/TX1_A/VI0_DATA1/ VI0_B1/RMII0_MDC_A/ET0_COL	OUTPUT
394	GP/HTX0_A/TX1_A/VI0_DATA1/ VI0_B1/RMII0_MDC_A/ET0_COL	CONTROL
393	GP/HTX0_A/TX1_A/VI0_DATA1/ VI0_B1/RMII0_MDC_A/ET0_COL	INPUT
392	GP/SCK1_B/VI0_DATA4/VI0_B4/ ET0_LINK_A	OUTPUT
391	GP/SCK1_B/VI0_DATA4/VI0_B4/ ET0_LINK_A	CONTROL
390	GP/SCK1_B/VI0_DATA4/VI0_B4/ ET0_LINK_A	INPUT
389	GP/HRX0_A/RX1_A/VI0_DATA0/ VI0_B0/RMII0_CRS_DV_A/ET0_CRS	OUTPUT
388	GP/HRX0_A/RX1_A/VI0_DATA0/ VI0_B0/RMII0_CRS_DV_A/ET0_CRS	CONTROL
387	GP/HRX0_A/RX1_A/VI0_DATA0/ VI0_B0/RMII0_CRS_DV_A/ET0_CRS	INPUT
386	GP/CAN1_TX_A/TX5_C/MLB_DAT	OUTPUT
385	GP/CAN1_TX_A/TX5_C/MLB_DAT	CONTROL
384	GP/CAN1_TX_A/TX5_C/MLB_DAT	INPUT
383	GP/CAN_CLK_A/RX4_D	OUTPUT
382	GP/CAN_CLK_A/RX4_D	CONTROL
381	GP/CAN_CLK_A/RX4_D	INPUT
380	NMI	INPUT
379	GP/CAN1_RX_A/IRQ1_B	OUTPUT
378	GP/CAN1_RX_A/IRQ1_B	CONTROL
377	GP/CAN1_RX_A/IRQ1_B	INPUT
376	GP/CAN0_RX_A/IRQ0_B/MLB_SIG	OUTPUT
375	GP/CAN0_RX_A/IRQ0_B/MLB_SIG	CONTROL
374	GP/CAN0_RX_A/IRQ0_B/MLB_SIG	INPUT
373	GP/CAN0_TX_A/TX4_D/MLB_CLK	OUTPUT
372	GP/CAN0_TX_A/TX4_D/MLB_CLK	CONTROL
371	GP/CAN0_TX_A/TX4_D/MLB_CLK	INPUT
370	AN0/IRQ2_B	INPUT
369	AN1/IRQ3_B	INPUT
368	AN2	INPUT
367	AN3	INPUT
366	AN4	INPUT
365	AN5	INPUT
364	AN6	INPUT
363	AN7	INPUT
362	SDA1/RX1_E	OUTPUT
361	SDA1/RX1_E	INPUT
360	SCL1/SCIF_CLK_C	OUTPUT
359	SCL1/SCIF_CLK_C	INPUT
358	SDA0/HIFEBL_A	OUTPUT
357	SDA0/HIFEBL_A	INPUT
356	SCL0	OUTPUT
355	SCL0	INPUT
354	GP/PENC1/TX3_D/CAN1_TX_B/ TX5_D/IETX_B	OUTPUT
353	GP/PENC1/TX3_D/CAN1_TX_B/ TX5_D/IETX_B	CONTROL
352	GP/PENC1/TX3_D/CAN1_TX_B/ TX5_D/IETX_B	INPUT
351	GP/USB_OVC1/RX3_D/CAN1_RX_B/ RX5_D/IERX_B	OUTPUT
350	GP/USB_OVC1/RX3_D/CAN1_RX_B/ RX5_D/IERX_B	CONTROL
349	GP/USB_OVC1/RX3_D/CAN1_RX_B/ RX5_D/IERX_B	INPUT
348	GP/PENC0	OUTPUT
347	GP/PENC0	CONTROL
346	GP/PENC0	INPUT
345	GP/USB_OVC0	OUTPUT
344	GP/USB_OVC0	CONTROL
343	GP/USB_OVC0	INPUT

Bit	Pin Name	Type	Bit	Pin Name	Type
342	GP/DU0_DR0/SCIF_CLK_B/ HRX0_D/IETX_A/TCLKA_A/HIFD00	OUTPUT	323	GP/DU0_DR6/SCK1_C/TIOC0C_A/ HIFD06	CONTROL
341	GP/DU0_DR0/SCIF_CLK_B/ HRX0_D/IETX_A/TCLKA_A/HIFD00	CONTROL	322	GP/DU0_DR6/SCK1_C/TIOC0C_A/ HIFD06	INPUT
340	GP/DU0_DR0/SCIF_CLK_B/ HRX0_D/IETX_A/TCLKA_A/HIFD00	INPUT	321	GP/DU0_DR7/RX1_C/TIOC0D_A/ HIFD07	OUTPUT
339	GP/DU0_DR1/SCK0_B/HTX0_D/ IERX_A/TCLKB_A/HIFD01	OUTPUT	320	GP/DU0_DR7/RX1_C/TIOC0D_A/ HIFD07	CONTROL
338	GP/DU0_DR1/SCK0_B/HTX0_D/ IERX_A/TCLKB_A/HIFD01	CONTROL	319	GP/DU0_DR7/RX1_C/TIOC0D_A/ HIFD07	INPUT
337	GP/DU0_DR1/SCK0_B/HTX0_D/ IERX_A/TCLKB_A/HIFD01	INPUT	318	GP/DU0_DG0/TX1_C/HSCK0_D/ IECLK_A/TIOC1A_A/HIFD08	OUTPUT
336	GP/DU0_DR2/RX0_B/TCLKC_A/ HIFD02	OUTPUT	317	GP/DU0_DG0/TX1_C/HSCK0_D/ IECLK_A/TIOC1A_A/HIFD08	CONTROL
335	GP/DU0_DR2/RX0_B/TCLKC_A/ HIFD02	CONTROL	316	GP/DU0_DG0/TX1_C/HSCK0_D/ IECLK_A/TIOC1A_A/HIFD08	INPUT
334	GP/DU0_DR2/RX0_B/TCLKC_A/ HIFD02	INPUT	315	GP/DU0_DG1/$\overline{\text{CTS1\_C}}$/$\overline{\text{HRTS0\_D}}$/ TIOC1B_A/HIFD09	OUTPUT
333	GP/DU0_DR3/TX0_B/TCLKD_A/ HIFD03	OUTPUT	314	GP/DU0_DG1/$\overline{\text{CTS1\_C}}$/$\overline{\text{HRTS0\_D}}$/ TIOC1B_A/HIFD09	CONTROL
332	GP/DU0_DR3/TX0_B/TCLKD_A/ HIFD03	CONTROL	313	GP/DU0_DG1/$\overline{\text{CTS1\_C}}$/$\overline{\text{HRTS0\_D}}$/ TIOC1B_A/HIFD09	INPUT
331	GP/DU0_DR3/TX0_B/TCLKD_A/ HIFD03	INPUT	312	GP/DU0_DG2/$\overline{\text{RTS1\_C}}$/RMII0_MDC_B/ TIOC2A_A/HIFD10	OUTPUT
330	GP/DU0_DR4/$\overline{\text{CTS0\_C}}$/TIOC0A_A/ HIFD04	OUTPUT	311	GP/DU0_DG2/$\overline{\text{RTS1\_C}}$/RMII0_MDC_B/ TIOC2A_A/HIFD10	CONTROL
329	GP/DU0_DR4/$\overline{\text{CTS0\_C}}$/TIOC0A_A/ HIFD04	CONTROL	310	GP/DU0_DG2/$\overline{\text{RTS1\_C}}$/RMII0_MDC_B/ TIOC2A_A/HIFD10	INPUT
328	GP/DU0_DR4/$\overline{\text{CTS0\_C}}$/TIOC0A_A/ HIFD04	INPUT	309	GP/DU0_DG3/SCK2_C/ RMII0_MDIO_B/TIOC2B_A/HIFD11	OUTPUT
327	GP/DU0_DR5/$\overline{\text{RTS0\_C}}$/TIOC0B_A/ HIFD05	OUTPUT	308	GP/DU0_DG3/SCK2_C/ RMII0_MDIO_B/TIOC2B_A/HIFD11	CONTROL
326	GP/DU0_DR5/$\overline{\text{RTS0\_C}}$/TIOC0B_A/ HIFD05	CONTROL	307	GP/DU0_DG3/SCK2_C/ RMII0_MDIO_B/TIOC2B_A/HIFD11	INPUT
325	GP/DU0_DR5/$\overline{\text{RTS0\_C}}$/TIOC0B_A/ HIFD05	INPUT	306	GP/DU0_DG4/RX2_C/ RMII0_CRS_DV_B/TIOC3A_A/HIFD12	OUTPUT
324	GP/DU0_DR6/SCK1_C/TIOC0C_A/ HIFD06	OUTPUT	305	GP/DU0_DG4/RX2_C/ RMII0_CRS_DV_B/TIOC3A_A/HIFD12	CONTROL

Bit	Pin Name	Type
304	GP/DU0_DG4/RX2_C/ RMII0_CRS_DV_B/TIOC3A_A/HIFD12	INPUT
303	GP/DU0_DG5/TX2_C/ RMII0_RX_ER_B/TIOC3B_A/HIFD13	OUTPUT
302	GP/DU0_DG5/TX2_C/ RMII0_RX_ER_B/TIOC3B_A/HIFD13	CONTROL
301	GP/DU0_DG5/TX2_C/ RMII0_RX_ER_B/TIOC3B_A/HIFD13	INPUT
300	GP/DU0_DG6/RX3_C/ RMII0_RXD0_B/TIOC3C_A/HIFD14	OUTPUT
299	GP/DU0_DG6/RX3_C/ RMII0_RXD0_B/TIOC3C_A/HIFD14	CONTROL
298	GP/DU0_DG6/RX3_C/ RMII0_RXD0_B/TIOC3C_A/HIFD14	INPUT
297	GP/DU0_DG7/TX3_C/ RMII0_RXD1_B/TIOC3D_A/HIFD15	OUTPUT
296	GP/DU0_DG7/TX3_C/ RMII0_RXD1_B/TIOC3D_A/HIFD15	CONTROL
295	GP/DU0_DG7/TX3_C/RMII0_RXD1_B/ TIOC3D_A/HIFD15	INPUT
294	GP/DU0_DB0/RX4_C/ RMII0_TXD_EN_B/TIOC4A_A/$\overline{\text{HIFCS}}$	OUTPUT
293	GP/DU0_DB0/RX4_C/ RMII0_TXD_EN_B/TIOC4A_A/$\overline{\text{HIFCS}}$	CONTROL
292	GP/DU0_DB0/RX4_C/ RMII0_TXD_EN_B/TIOC4A_A/$\overline{\text{HIFCS}}$	INPUT
291	GP/DU0_DB1/TX4_C/RMII0_TXD0_B/ TIOC4B_A/HIFRS	OUTPUT
290	GP/DU0_DB1/TX4_C/RMII0_TXD0_B/ TIOC4B_A/HIFRS	CONTROL
289	GP/DU0_DB1/TX4_C/RMII0_TXD0_B/ TIOC4B_A/HIFRS	INPUT
288	GP/DU0_DB2/RX5_B/RMII0_TXD1_B/ TIOC4C_A/$\overline{\text{HIFWR}}$	OUTPUT
287	GP/DU0_DB2/RX5_B/RMII0_TXD1_B/ TIOC4C_A/$\overline{\text{HIFWR}}$	CONTROL
286	GP/DU0_DB2/RX5_B/RMII0_TXD1_B/ TIOC4C_A/$\overline{\text{HIFWR}}$	INPUT
285	GP/DU0_DB3/TX5_B/TIOC4D_A/ $\overline{\text{HIFRD}}$	OUTPUT
284	GP/DU0_DB3/TX5_B/TIOC4D_A/ $\overline{\text{HIFRD}}$	CONTROL
283	GP/DU0_DB3/TX5_B/TIOC4D_A/ $\overline{\text{HIFRD}}$	INPUT
282	GP/DU0_DB7/SD2_DAT1_B/ SSI_SCK0_B/HIFEBL_B	OUTPUT
281	GP/DU0_DB7/SD2_DAT1_B/ SSI_SCK0_B/HIFEBL_B	CONTROL
280	GP/DU0_DB7/SD2_DAT1_B/ SSI_SCK0_B/HIFEBL_B	INPUT
279	GP/DU0_DB6/SD2_DAT0_B/HIFRDY	OUTPUT
278	GP/DU0_DB6/SD2_DAT0_B/HIFRDY	CONTROL
277	GP/DU0_DB6/SD2_DAT0_B/HIFRDY	INPUT
276	GP/DU0_DB5/SD2_CMD_B/HIFDREQ	OUTPUT
275	GP/DU0_DB5/SD2_CMD_B/HIFDREQ	CONTROL
274	GP/DU0_DB5/SD2_CMD_B/HIFDREQ	INPUT
273	GP/DU0_DB4/SD2_CLK_B/$\overline{\text{HIFINT}}$	OUTPUT
272	GP/DU0_DB4/SD2_CLK_B/$\overline{\text{HIFINT}}$	CONTROL
271	GP/DU0_DB4/SD2_CLK_B/$\overline{\text{HIFINT}}$	INPUT
270	GP/DU0_DISP/CAN0_TX_B/HRX0_B/ AUDIO_CLKA_B	OUTPUT
269	GP/DU0_DISP/CAN0_TX_B/HRX0_B/ AUDIO_CLKA_B	CONTROL
268	GP/DU0_DISP/CAN0_TX_B/HRX0_B/ AUDIO_CLKA_B	INPUT
267	GP/DU0_EXHSYNC/DU0_HSYNC/ SD2_CD_B/HSPI_TX_C/SSI_SCK1_B	OUTPUT
266	GP/DU0_EXHSYNC/DU0_HSYNC/ SD2_CD_B/HSPI_TX_C/SSI_SCK1_B	CONTROL
265	GP/DU0_EXHSYNC/DU0_HSYNC/ SD2_CD_B/HSPI_TX_C/SSI_SCK1_B	INPUT
264	GP/DU0_DOTCLKOUT/SD2_DAT3_B/ HSPI_CLK_C/SSI_SDATA0_B	OUTPUT

Bit	Pin Name	Type
263	GP/DU0_DOTCLKOUT/SD2_DAT3_B/ HSPI_CLK_C/SSI_SDATA0_B	CONTROL
262	GP/DU0_DOTCLKOUT/SD2_DAT3_B/ HSPI_CLK_C/SSI_SDATA0_B	INPUT
261	GP/DU0_DOTCLKIN/SD2_DAT2_B/ HSPI_CS_C/SSI_WS0_B	OUTPUT
260	GP/DU0_DOTCLKIN/SD2_DAT2_B/ HSPI_CS_C/SSI_WS0_B	CONTROL
259	GP/DU0_DOTCLKIN/SD2_DAT2_B/ HSPI_CS_C/SSI_WS0_B	INPUT
258	GP/RTS1_B/VI0_G2	OUTPUT
257	GP/RTS1_B/VI0_G2	CONTROL
256	GP/RTS1_B/VI0_G2	INPUT
255	GP/DU0_CDE/HTX0_B/ AUDIO_CLKB_B/LCD_VCPWC_B	OUTPUT
254	GP/DU0_CDE/HTX0_B/ AUDIO_CLKB_B/LCD_VCPWC_B	CONTROL
253	GP/DU0_CDE/HTX0_B/ AUDIO_CLKB_B/LCD_VCPWC_B	INPUT
252	GP/DU0_EXODDF/DU0_ODDF/ CAN0_RX_B/HSCK0_B/SSI_SDATA1_B	OUTPUT
251	GP/DU0_EXODDF/DU0_ODDF/ CAN0_RX_B/HSCK0_B/SSI_SDATA1_B	CONTROL
250	GP/DU0_EXODDF/DU0_ODDF/ CAN0_RX_B/HSCK0_B/SSI_SDATA1_B	INPUT
249	GP/DU0_EXVSYNC/DU0_VSYNC/ SD2_WP_B/HSPI_RX_C/SSI_WS1_B	OUTPUT
248	GP/DU0_EXVSYNC/DU0_VSYNC/ SD2_WP_B/HSPI_RX_C/SSI_WS1_B	CONTROL
247	GP/DU0_EXVSYNC/DU0_VSYNC/ SD2_WP_B/HSPI_RX_C/SSI_WS1_B	INPUT
246	GP/SD2_CLK_A/RX2_A/VI0_G4/ ET0_RX_CLK_B	OUTPUT
245	GP/SD2_CLK_A/RX2_A/VI0_G4/ ET0_RX_CLK_B	CONTROL
244	GP/SD2_CLK_A/RX2_A/VI0_G4/ ET0_RX_CLK_B	OBSERVE_ONLY
243	GP/SCK2_A/VI0_G3	OUTPUT
242	GP/SCK2_A/VI0_G3	CONTROL
241	GP/SCK2_A/VI0_G3	INPUT
240	GP/SD2_DAT0_A/RX3_A/VI0_R0/ ET0_ERXD3_B	OUTPUT
239	GP/SD2_DAT0_A/RX3_A/VI0_R0/ ET0_ERXD3_B	CONTROL
238	GP/SD2_DAT0_A/RX3_A/VI0_R0/ ET0_ERXD3_B	OBSERVE_ONLY
237	GP/CTS1_B/VI0_DATA7/VI0_G1	OUTPUT
236	GP/CTS1_B/VI0_DATA7/VI0_G1	CONTROL
235	GP/CTS1_B/VI0_DATA7/VI0_G1	INPUT
234	GP/SD2_DAT2_A/RX4_A/VI0_R2/ ET0_LINK_B	OUTPUT
233	GP/SD2_DAT2_A/RX4_A/VI0_R2/ ET0_LINK_B	CONTROL
232	GP/SD2_DAT2_A/RX4_A/VI0_R2/ ET0_LINK_B	OBSERVE_ONLY
231	GP/SD2_DAT1_A/TX3_A/VI0_R1/ ET0_MDIO_B	OUTPUT
230	GP/SD2_DAT1_A/TX3_A/VI0_R1/ ET0_MDIO_B	CONTROL
229	GP/SD2_DAT1_A/TX3_A/VI0_R1/ ET0_MDIO_B	OBSERVE_ONLY
228	GP/SD2_WP_A/TX5_A/VI0_R5	OUTPUT
227	GP/SD2_WP_A/TX5_A/VI0_R5	CONTROL
226	GP/SD2_WP_A/TX5_A/VI0_R5	OBSERVE_ONLY
225	GP/SD2_CMD_A/TX2_A/VI0_G5/ ET0_ERXD2_B	OUTPUT
224	GP/SD2_CMD_A/TX2_A/VI0_G5/ ET0_ERXD2_B	CONTROL
223	GP/SD2_CMD_A/TX2_A/VI0_G5/ ET0_ERXD2_B	OBSERVE_ONLY
222	GP/VI1_CLK_A/AUDCK/NAF0_B/ LCD_DATA0_B	OUTPUT

Bit	Pin Name	Type	Bit	Pin Name	Type
221	GP/VI1_CLK_A/AUDCK/NAF0_B/ LCD_DATA0_B	CONTROL	202	GP/VI1_2_A/AUDATA1/NAF3_B/ LCD_DATA3_B	INPUT
220	GP/VI1_CLK_A/AUDCK/NAF0_B/ LCD_DATA0_B	INPUT	201	GP/VI1_6_A/AUDATA5/NAF7_B/ LCD_DATA7_B	OUTPUT
219	GP/VI1_0_A/AUDSYNC/NAF1_B/ LCD_DATA1_B	OUTPUT	200	GP/VI1_6_A/AUDATA5/NAF7_B/ LCD_DATA7_B	CONTROL
218	GP/VI1_0_A/AUDSYNC/NAF1_B/ LCD_DATA1_B	CONTROL	199	GP/VI1_6_A/AUDATA5/NAF7_B/ LCD_DATA7_B	INPUT
217	GP/VI1_0_A/AUDSYNC/NAF1_B/ LCD_DATA1_B	INPUT	198	GP/VI1_5_A/AUDATA4/NAF6_B/ LCD_DATA6_B	OUTPUT
216	GP/SD2_CD_A/RX5_A/VI0_R4/ ET0_PHY_INT_B	OUTPUT	197	GP/VI1_5_A/AUDATA4/NAF6_B/ LCD_DATA6_B	CONTROL
215	GP/SD2_CD_A/RX5_A/VI0_R4/ ET0_PHY_INT_B	CONTROL	196	GP/VI1_5_A/AUDATA4/NAF6_B/ LCD_DATA6_B	INPUT
214	GP/SD2_CD_A/RX5_A/VI0_R4/ ET0_PHY_INT_B	OBSERVE_ONLY	195	GP/VI1_1_A/AUDATA0/NAF2_B/ LCD_DATA2_B	OUTPUT
213	GP/SD2_DAT3_A/TX4_A/VI0_R3/ ET0_MAGIC_B	OUTPUT	194	GP/VI1_1_A/AUDATA0/NAF2_B/ LCD_DATA2_B	CONTROL
212	GP/SD2_DAT3_A/TX4_A/VI0_R3/ ET0_MAGIC_B	CONTROL	193	GP/VI1_1_A/AUDATA0/NAF2_B/ LCD_DATA2_B	INPUT
211	GP/SD2_DAT3_A/TX4_A/VI0_R3/ ET0_MAGIC_B	OBSERVE_ONLY	192	GP/SSI_SDATA1_A/VI1_3_B/ LCD_DATA14_B	OUTPUT
210	GP/VI1_4_A/AUDATA3/NAF5_B/ LCD_DATA5_B	OUTPUT	191	GP/SSI_SDATA1_A/VI1_3_B/ LCD_DATA14_B	CONTROL
209	GP/VI1_4_A/AUDATA3/NAF5_B/ LCD_DATA5_B	CONTROL	190	GP/SSI_SDATA1_A/VI1_3_B/ LCD_DATA14_B	OBSERVE_ONLY
208	GP/VI1_4_A/AUDATA3/NAF5_B/ LCD_DATA5_B	INPUT	189	GP/VI1_7_A/AUDATA6/$\overline{\text{FCE\_B}}$/ LCD_DATA8_B	OUTPUT
207	GP/VI1_3_A/AUDATA2/NAF4_B/ LCD_DATA4_B	OUTPUT	188	GP/VI1_7_A/AUDATA6/$\overline{\text{FCE\_B}}$/ LCD_DATA8_B	CONTROL
206	GP/VI1_3_A/AUDATA2/NAF4_B/ LCD_DATA4_B	CONTROL	187	GP/VI1_7_A/AUDATA6/$\overline{\text{FCE\_B}}$/ LCD_DATA8_B	INPUT
205	GP/VI1_3_A/AUDATA2/NAF4_B/ LCD_DATA4_B	INPUT	186	GP/SSI_SCK0_A/AUDATA7/TIOC1A_B/ LCD_DATA9_B	OUTPUT
204	GP/VI1_2_A/AUDATA1/NAF3_B/ LCD_DATA3_B	OUTPUT	185	GP/SSI_SCK0_A/AUDATA7/TIOC1A_B/ LCD_DATA9_B	CONTROL
203	GP/VI1_2_A/AUDATA1/NAF3_B/ LCD_DATA3_B	CONTROL	184	GP/SSI_SCK0_A/AUDATA7/TIOC1A_B/ LCD_DATA9_B	OBSERVE_ONLY

Bit	Pin Name	Type
183	GP/SSI_SCK1_A/VI1_1_B/TIOC2B_B/ LCD_DATA12_B	OUTPUT
182	GP/SSI_SCK1_A/VI1_1_B/TIOC2B_B/ LCD_DATA12_B	CONTROL
181	GP/SSI_SCK1_A/VI1_1_B/TIOC2B_B/ LCD_DATA12_B	OBSERVE_ONLY
180	GP/SSI_WS0_A/TIOC1B_B/ LCD_DATA10_B	OUTPUT
179	GP/SSI_WS0_A/TIOC1B_B/ LCD_DATA10_B	CONTROL
178	GP/SSI_WS0_A/TIOC1B_B/ LCD_DATA10_B	OBSERVE_ONLY
177	GP/SSI_WS1_A/VI1_2_B/ LCD_DATA13_B	OUTPUT
176	GP/SSI_WS1_A/VI1_2_B/ LCD_DATA13_B	CONTROL
175	GP/SSI_WS1_A/VI1_2_B/ LCD_DATA13_B	OBSERVE_ONLY
174	GP/SSI_WS23/VI1_5_B/TX1_D/ HSCK0_C/FALE_B/LCD_DON_B	OUTPUT
173	GP/SSI_WS23/VI1_5_B/TX1_D/ HSCK0_C/FALE_B/LCD_DON_B	CONTROL
172	GP/SSI_WS23/VI1_5_B/TX1_D/ HSCK0_C/FALE_B/LCD_DON_B	OBSERVE_ONLY
171	GP/SSI_SDATA0_A/VI1_0_B/ TIOC2A_B/LCD_DATA11_B	OUTPUT
170	GP/SSI_SDATA0_A/VI1_0_B/ TIOC2A_B/LCD_DATA11_B	CONTROL
169	GP/SSI_SDATA0_A/VI1_0_B/ TIOC2A_B/LCD_DATA11_B	OBSERVE_ONLY
168	GP/SSI_SCK23/VI1_4_B/RX1_D/ FCLE_B/LCD_DATA15_B	OUTPUT
167	GP/SSI_SCK23/VI1_4_B/RX1_D/ FCLE_B/LCD_DATA15_B	CONTROL
166	GP/SSI_SCK23/VI1_4_B/RX1_D/ FCLE_B/LCD_DATA15_B	OBSERVE_ONLY
165	GP/SSI_SDATA2/VI1_6_B/HRX0_C/ $\overline{\text{FRE\_B}}$/LCD_CL1_B	OUTPUT
164	GP/SSI_SDATA2/VI1_6_B/HRX0_C/ $\overline{\text{FRE\_B}}$/LCD_CL1_B	CONTROL
163	GP/SSI_SDATA2/VI1_6_B/HRX0_C/ $\overline{\text{FRE\_B}}$/LCD_CL1_B	INPUT
162	GP/AUDIO_CLKA_A/VI1_CLK_B/ SCK1_D/IECLK_B/LCD_FLM_B	OUTPUT
161	GP/AUDIO_CLKA_A/VI1_CLK_B/ SCK1_D/IECLK_B/LCD_FLM_B	CONTROL
160	GP/AUDIO_CLKA_A/VI1_CLK_B/ SCK1_D/IECLK_B/LCD_FLM_B	INPUT
159	GP/AUDIO_CLKB_A/LCD_CLK_B	OUTPUT
158	GP/AUDIO_CLKB_A/LCD_CLK_B	CONTROL
157	GP/AUDIO_CLKB_A/LCD_CLK_B	INPUT
156	GP/AUDIO_CLKC/SCK1_E/$\overline{\text{HCTS0\_C}}$/ FRB_B/LCD_VEPWC_B	OUTPUT
155	GP/AUDIO_CLKC/SCK1_E/$\overline{\text{HCTS0\_C}}$/ FRB_B/LCD_VEPWC_B	CONTROL
154	GP/AUDIO_CLKC/SCK1_E/$\overline{\text{HCTS0\_C}}$/ FRB_B/LCD_VEPWC_B	INPUT
153	GP/SSI_SDATA3/VI1_7_B/HTX0_C/ $\overline{\text{FWE\_B}}$/LCD_CL2_B	OUTPUT
152	GP/SSI_SDATA3/VI1_7_B/HTX0_C/ $\overline{\text{FWE\_B}}$/LCD_CL2_B	CONTROL
151	GP/SSI_SDATA3/VI1_7_B/HTX0_C/ $\overline{\text{FWE\_B}}$/LCD_CL2_B	INPUT
150	GP/AUDIO_CLKOUT/TX1_E/$\overline{\text{HRTS0\_C}}$/ /LCD_M_DISP_B	OUTPUT
149	GP/AUDIO_CLKOUT/TX1_E/$\overline{\text{HRTS0\_C}}$/ /LCD_M_DISP_B	CONTROL
148	GP/AUDIO_CLKOUT/TX1_E/$\overline{\text{HRTS0\_C}}$/ /LCD_M_DISP_B	INPUT
147	GP/SDSELF/$\overline{\text{RTS1\_E}}$	OUTPUT
146	GP/SDSELF/$\overline{\text{RTS1\_E}}$	CONTROL
145	GP/SDSELF/$\overline{\text{RTS1\_E}}$	INPUT
144	*	INTERNAL
143	*	INTERNAL

Bit	Pin Name	Type	Bit	Pin Name	Type
142	*	INTERNAL	110	*	INTERNAL
141	*	INTERNAL	109	*	INTERNAL
140	*	INTERNAL	108	*	INTERNAL
139	*	INTERNAL	107	*	INTERNAL
138	*	INTERNAL	106	*	INTERNAL
137	*	INTERNAL	105	*	INTERNAL
136	*	INTERNAL	104	*	INTERNAL
135	*	INTERNAL	103	*	INTERNAL
134	*	INTERNAL	102	*	INTERNAL
133	*	INTERNAL	101	*	INTERNAL
132	*	INTERNAL	100	*	INTERNAL
131	*	INTERNAL	99	*	INTERNAL
130	*	INTERNAL	98	*	INTERNAL
129	*	INTERNAL	97	*	INTERNAL
128	*	INTERNAL	96	*	INTERNAL
127	*	INTERNAL	95	*	INTERNAL
126	*	INTERNAL	94	*	INTERNAL
125	*	INTERNAL	93	*	INTERNAL
124	*	INTERNAL	92	*	INTERNAL
123	*	INTERNAL	91	*	INTERNAL
122	*	INTERNAL	90	*	INTERNAL
121	*	INTERNAL	89	*	INTERNAL
120	*	INTERNAL	88	*	INTERNAL
119	*	INTERNAL	87	*	INTERNAL
118	*	INTERNAL	86	*	INTERNAL
117	*	INTERNAL	85	*	INTERNAL
116	*	INTERNAL	84	*	INTERNAL
115	*	INTERNAL	83	*	INTERNAL
114	*	INTERNAL	82	*	INTERNAL
113	*	INTERNAL	81	*	INTERNAL
112	*	INTERNAL	80	*	INTERNAL
111	*	INTERNAL	79	*	INTERNAL

Bit	Pin Name	Type	Bit	Pin Name	Type
78	*	INTERNAL	46	*	INTERNAL
77	*	INTERNAL	45	*	INTERNAL
76	*	INTERNAL	44	*	INTERNAL
75	*	INTERNAL	43	*	INTERNAL
74	*	INTERNAL	42	*	INTERNAL
73	*	INTERNAL	41	*	INTERNAL
72	*	INTERNAL	40	*	INTERNAL
71	*	INTERNAL	39	*	INTERNAL
70	*	INTERNAL	38	*	INTERNAL
69	*	INTERNAL	37	*	INTERNAL
68	*	INTERNAL	36	*	INTERNAL
67	*	INTERNAL	35	*	INTERNAL
66	*	INTERNAL	34	*	INTERNAL
65	*	INTERNAL	33	*	INTERNAL
64	*	INTERNAL	32	*	INTERNAL
63	*	INTERNAL	31	*	INTERNAL
62	*	INTERNAL	30	*	INTERNAL
61	*	INTERNAL	29	*	INTERNAL
60	*	INTERNAL	28	*	INTERNAL
59	*	INTERNAL	27	*	INTERNAL
58	*	INTERNAL	26	*	INTERNAL
57	*	INTERNAL	25	*	INTERNAL
56	*	INTERNAL	24	*	INTERNAL
55	*	INTERNAL	23	*	INTERNAL
54	*	INTERNAL	22	*	INTERNAL
53	*	INTERNAL	21	*	INTERNAL
52	*	INTERNAL	20	*	INTERNAL
51	*	INTERNAL	19	*	INTERNAL
50	*	INTERNAL	18	*	INTERNAL
49	*	INTERNAL	17	*	INTERNAL
48	*	INTERNAL	16	*	INTERNAL
47	*	INTERNAL	15	*	INTERNAL

Bit	Pin Name	Type
14	*	INTERNAL
13	*	INTERNAL
12	*	INTERNAL
11	*	INTERNAL
10	*	INTERNAL
9	*	INTERNAL
8	*	INTERNAL
7	*	INTERNAL
6	*	INTERNAL
5	*	INTERNAL
4	*	INTERNAL
3	$\overline{\text{PRESET}}$	OBSERVE_ONLY
2	GP/$\overline{\text{PRESETOUT}}$/ST_CLKOUT	OUTPUT
1	GP/$\overline{\text{PRESETOUT}}$/ST_CLKOUT	CONTROL
0	GP/$\overline{\text{PRESETOUT}}$/ST_CLKOUT	INPUT
	To TDO	

Note: Bits of type "control" are active low. When the control bit is at the low level, the corresponding pin becomes an output, and the level output on the pin corresponds to the value of the "output" bit.

39.4 Operation

39.4.1 Boundary-Scan TAP Controller (IDCODE, EXTEST, SAMPLE/PRELOAD, BYPASS)

The H-UDI of this LSI separates the TAP controller for boundary scan from the TAP controller that controls the H-UDI reset and H-UDI interrupt functions. The boundary-scan TAP controller is enabled and the boundary scan function stipulated by the JTAG standard can be used when the $\overline{\text{TRST}}$ pin is asserted, including when the power is turned on. Moreover, the H-UDI reset and H-UDI interrupt functions can be enabled by inputting the H-UDI switching command. Note, however, that this LSI has the following restrictions.

- Clock related signals (EXTAL and XTAL) are not covered by boundary scan.
- H-UDI related signals (TCK, TDI, TDO, TMS, $\overline{\text{TRST}}$, and MPMD) are not covered by boundary scan.
- DDR2/DDR3 interface related pins are not covered by boundary scan.
- USB interface related pins (USB_EXTAL, USB_XTAL, REFRIN, DP0, DP1, DM0, DM1, OVC0/VBUS0, and OVC1/VBUS1) are not covered by boundary scan.
- When boundary scan (IDCODE, EXTEST, SAMPLE/PRELOAD, BYPASS, or H-UDI switching command) is executed, the maximum TCK frequency is 2 MHz.
- The size for access from the H-UDI (external controller) to the boundary-scan TAP controller is 8 bits.

The supported commands of the boundary-scan TAP controller are shown in table 39.6.

Note: When using the boundary scan function, fix TEST1 to the low level, TEST2 to the low level, BSMODE to the high level, and MPMD to the high level. When this LSI operates in emulation support mode (MPMD = 0), the boundary scan function cannot be used. The sequence for switching from the boundary-scan TAP controller to the H-UDI is shown in figure 39.2.

Table 39.6 Supported Commands of Boundary-Scan TAP Controller

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Description
0	1	0	1	0	1	0	1	IDCODE
1	1	1	1	1	1	1	1	BYPASS
0	0	0	0	0	0	0	0	EXTEST
0	1	0	0	0	0	0	0	SAMPLE/PRELOAD
0	0	0	0	1	0	0	0	H-UDI switching command
Other than above								Setting prohibited

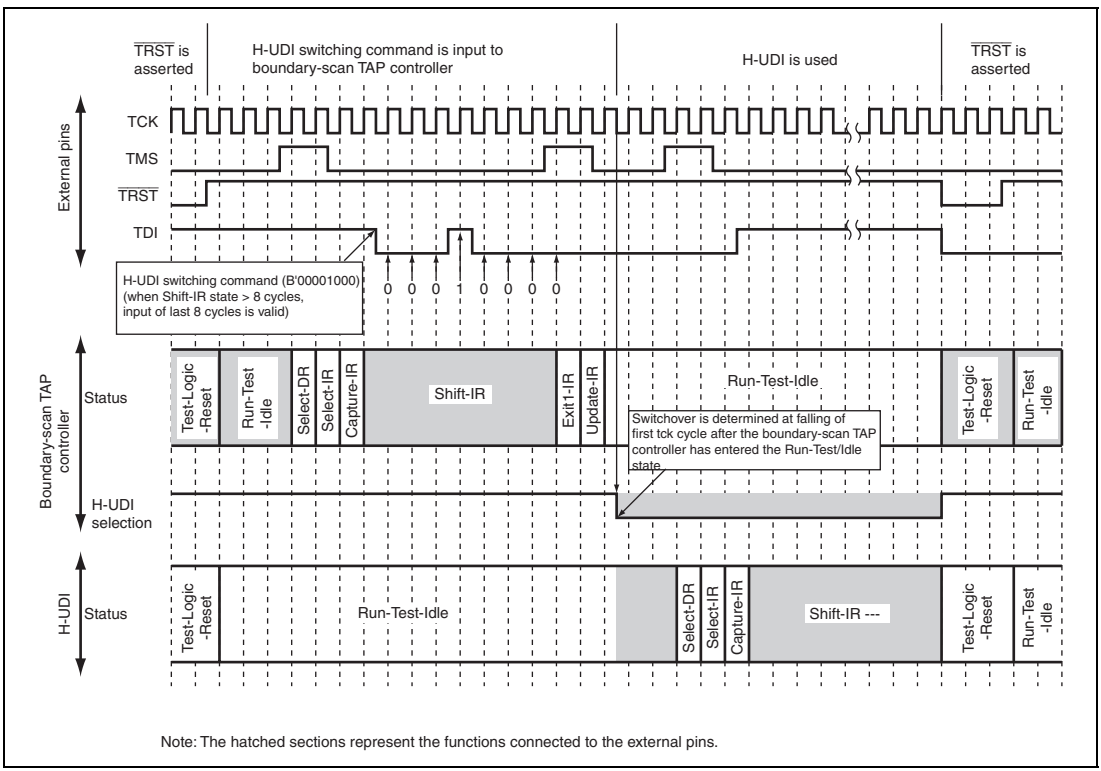


Figure 39.2 Sequence for Switching from Boundary-Scan TAP Controller to H-UDI

39.4.2 TAP Control

Figure 39.3 shows the internal states of the TAP controller. The state transitions conform to the subset of the JTAG standard.

- State transitions occur according to the TMS value at the rising edge of the TCK signal.
- The TDI value is sampled at the rising edge of the TCK signal and shifted at the falling edge of the TCK signal.
- The TDO value is changed at the falling edge of the TCK signal. The TDO signal is in a Hi-Z state for TAP controller states other than Shift-DR and Shift-IR.
- A transition to the Test-Logic-Reset state by clearing $\overline{\text{TRST}}$ to 0 is performed asynchronously with the TCK signal.

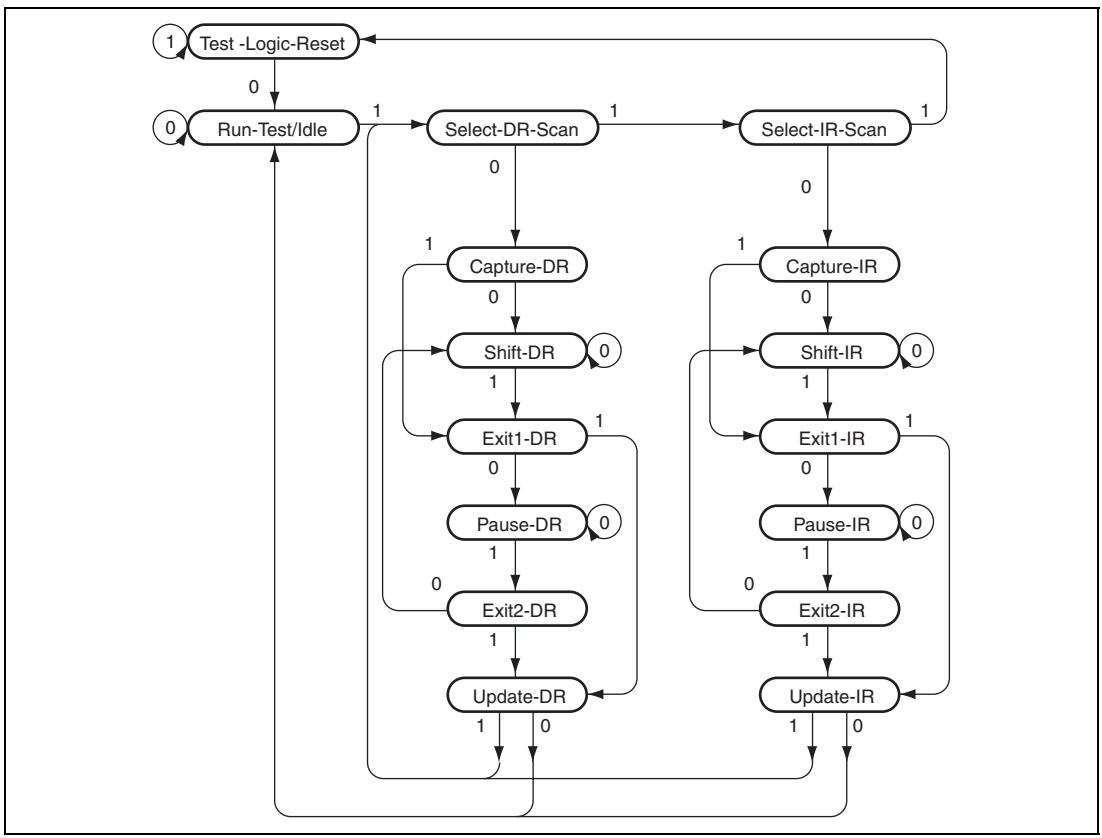


Figure 39.3 TAP Controller State Transitions

39.4.3 H-UDI Reset

A power-on reset is generated by the SDIR command. After the H-UDI reset assert command has been sent from the H-UDI pin, sending the H-UDI reset negate command will reset the CPU (see figure 39.4). The required time between the H-UDI reset assert and H-UDI reset negate commands is the same as the time for holding the reset pin low in order to reset this LSI by a power-on reset. When the H-UDI reset assert command is set, an LSI internal reset is asserted after four cycles of clkp1. When the H-UDI reset negate command is set, the LSI internal reset is negated after the reset hold time (38 cycles of clkp1 as the minimum and 73 cycles of clkp1 as the maximum).

Note: The RESET/WDT module is not initialized. Note, however, that the overflow counter in the RESET/WDT module will be initialized.

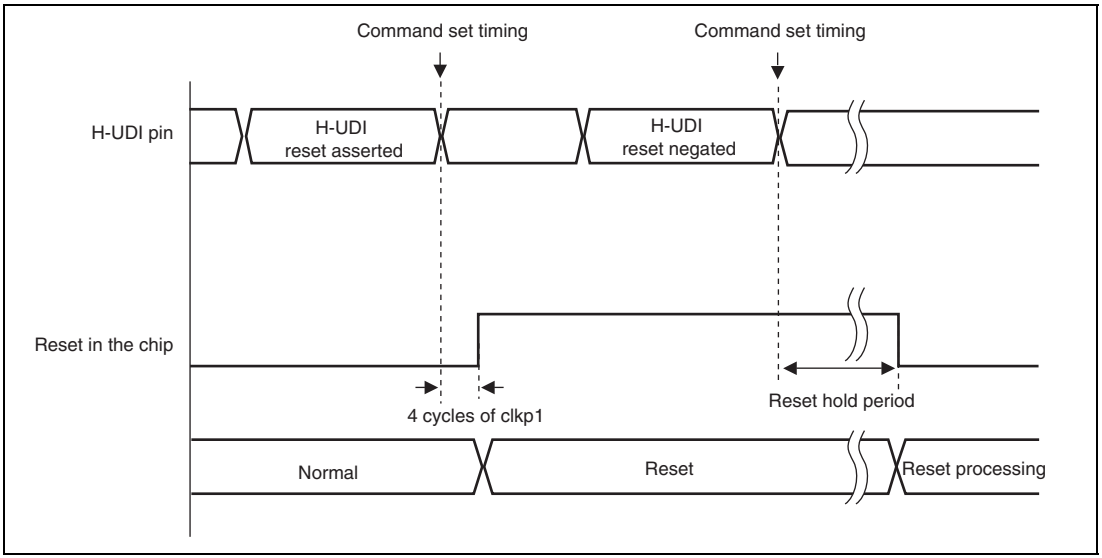


Figure 39.4 H-UDI Reset

39.4.4 H-UDI Interrupt

The H-UDI interrupt function generates an interrupt by setting the appropriate command in SDIR from the H-UDI.

The H-UDI interrupt is a normal exception or interrupt operation. The CPU branches to an address based on VBR and returns when the RTE instruction is executed. In this case, the exception code to be stored in the control register INTEVT is H'600. Bits 28 to 24 of the control register INT2PRI2 can be used to control the priority of the H-UDI interrupt. For details, refer to section 7, INTC/INTC2.

The H-UDI interrupt request signal is asserted by setting the INTREQ bit in SDINT to 1 after the appropriate command has been set (Update-IR). Since the interrupt request signal is not negated until the INTREQ bit is cleared to 0 by software, it is not possible to lose the interrupt request. While an H-UDI interrupt command is set in SDIR, SDINT is connected between the TDI and TDO pins. For the value read from the TDO pin, refer to section 39.3.2, Interrupt Source Register (SDINT).

39.5 Usage Notes

1. Once a command is set in SDIR, it will not be changed unless another command is written from the H-UDI or the H-UDI is initialized by the $\overline{\text{TRST}}$ pin being set to the low level or the TAP controller being set to the Test-Logic-Reset state.
2. If you are not using an emulator, fix the $\overline{\text{TRST}}$ pin to the low level.
3. In emulation support mode, an H-UDI interrupt or H-UDI reset is the trigger for release from sleep mode, software standby mode, and deep standby mode. Accordingly, these exception requests are accepted.
4. The H-UDI is used for emulator connection. Therefore the JTAG functions cannot be used when using an emulator.
5. In deep standby mode, do not issue an H-UDI reset after an H-UDI interrupt until the fetching of instructions starts.

Section 40 Quad-SPI

40.1 Features

This module has the following features.

- Capable of communications to the serial flash memory through single-/dual-/quad-SPI operation

  Single-SPI operation

  Use of MO (master out), MI (master in), QSSL (slave select), and QSPCLK (SPI clock) signals allow for communications to the serial flash memory through SPI operation (four-wire method).

  QMO output pin and QMI input pin

  QSSL and QSPCLK serve as output pins.

  Dual-SPI operation

  Use of QIO1, QIO0, QSSL, and QSPCLK signals allow for serial communications through SPI operation (four-wire method).

  Bidirectional QIO1 and QIO0 pins

  QSSL and QSPCLK serve as output pins.

  Quad-SPI operation

  Use of QIO3 to QIO0, QSSL, and QSPCLK signals allow for serial communications through SPI operation (six-wire method).

  Bidirectional QIO3 to QIO0 pins

  QSSL and QSPCLK serve as output pins
- Transfer data length

  Transfer data length is selectable from 8 bits to 128 Gbits

  Data is continuously transferred one through 4,294,967,296 times in 8-, 16-, or 32-bit units
- Bit rate

  QSPCLK can be divided by 2 to 4080

  QSPCLK can be generated by dividing CLKS1 by the on-chip baud rate generator.

- Buffer configuration
  8 bits × 32 buffers for transmission and 8 bits × 32 buffers for reception
- Shift registers
  32 bits each for transmission and reception
- QSSL control function
  Controllable delay from QSSL output assertion to QSPCLK operation (clock delay)
  Range: 0 and 1.5 to 8.5 QSPCLK cycles (set in QSPCLK-cycle units)
  Controllable delay from QSPCLK stoppage to QSSL output negation (QSSL negation delay)
  Range: 0 to 8 QSPCLK cycles (set in QSPCLK-cycle units)
  Controllable wait for next-access QSSL output assertion (next-access delay)
  Range: 0 to 8 QSPCLK cycles (set in QSPCLK-cycle units)
  Capable of holding QSSL output value from transfer end to next access
  Function for changing QSSL polarity
- Transfer control
  A transfer of up to four commands can be executed sequentially in looped execution.
  Single-SPI or dual-/quad-SPI write operation: A transfer can be started when data is written to the transmit buffer while the SPI function is enabled.
  Dual-/quad-SPI read operation: A transfer can be started when the SPI function is enabled while there is enough space for receiving the specified length of data in the receive buffer.
  QIO3 to QIO0 and QMO output values can be specified during QSSL negation
  QIO3 and QIO2 output values can be specified in single-/dual-SPI modes
- Interrupt sources
  Maskable interrupt sources:
  Receive buffer full interrupt
  Transmit buffer empty interrupt
- Others
  Provides loop back mode
  Provides a function for initializing this module

40.2 Input/Output Pins

Table 40.1 shows the pin configuration.

Table 40.1 Pin Configuration

Pin Name	Pin Name	I/O	Function
Clock pin	QSPCLK	Output	Clock output
Master transmit data/data 0 pin*2	QMO/QIO0	I/O	Master transmit data/data 0
Master input data/data 1 pin*2	QMI/QIO1	I/O	Master input data/data 1
Data 2 pin*1	QIO2	I/O	Data 2
Data 3 pin*2	QIO3	I/O	Data 3
Slave select pin	QSSL	Output	Slave selection

Notes: 1. In single-SPI mode, QMO and QMI are enabled; QIO0 and QIO1 in dual-/quad-SPI modes.
2. In single-/dual-SPI modes, fixed value according to register setting is output; QIO2 and QIO3 in quad-SPI mode.

40.3 Register Descriptions

Table 40.2 shows the register configuration.

The initial address is H'FFFC 3000.

Table 40.2 (1) Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Control register	SPCR	R/W	H'00	H'00	8, 16, 32
Slave select polarity register	SSLP	R/W	H'00	H'01	8, 16, 32
Pin control register	SPPCR	R/W	H'06	H'02	8, 16, 32
Status register	SPSR	R/(W)*	H'60	H'03	8, 16, 32
Data register	SPDR	R/W	Undefined	H'04	8, 16, 32
Sequence control register	SPSCR	R/W	H'00	H'08	8, 16, 32
Sequence status register	SPSSR	R	H'00	H'09	8, 16, 32
Bit rate register	SPBR	R/W	H'FF	H'0A	8, 16, 32
Data control register	SPDCR	R/W	H'00	H'0B	8, 16, 32
Clock delay register	SPCKD	R/W	H'00	H'0C	8, 16, 32
Slave select negation delay register	SSLND	R/W	H'00	H'0D	8, 16, 32
Next-access delay register	SPND	R/W	H'00	H'0E	8, 16, 32
Command register 0	SPCMD0	R/W	H'E001	H'10	16, 32
Command register 1	SPCMD1	R/W	H'E001	H'12	16, 32
Command register 2	SPCMD2	R/W	H'E001	H'14	16, 32
Command register 3	SPCMD3	R/W	H'E001	H'16	16, 32
Buffer control register	SPBFCR	R/W	H'00	H'18	8, 16, 32
Buffer data count setting register	SPBDCR	R	H'0000	H'1A	16, 32
Transfer data length multiplier setting register 0	SPBMUL0	R/W	H'00000001	H'1C	32
Transfer data length multiplier setting register 1	SPBMUL1	R/W	H'00000001	H'20	32
Transfer data length multiplier setting register 2	SPBMUL2	R/W	H'00000001	H'24	32
Transfer data length multiplier setting register 3	SPBMUL3	R/W	H'00000001	H'28	32

Note: * Only 0 can be written to clear the flag.

Table 40.2 (2) Register State in Each Operating Mode

Register Name	Power-on Reset	Manual Reset	Sleep	Software Standby	Module Standby	Deep Standby
All registers of Quad-SPI	Initialized	Initialized	Retained	Retained	Retained	Initialized

40.3.1 Control Register (SPCR)

SPCR sets the operating mode.

Bit:	7	6	5	4	3	2	1	0
	SPRIE	SPE	SPTIE	—	—	—	—	—
Initial Value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	SPRIE	0	R/W	Receive Interrupt Enable Enables or disables generation of receive interrupt requests when the number of receive data units in the receive buffer is equal to or greater than the specified receive buffer data triggering number and the receive buffer full flag (SPRFF) in the status register (SPSR) is set to 1. 0: Disables the generation of receive interrupt requests. 1: Enables the generation of receive interrupt requests.
6	SPE	0	R/W	SPI Function Enable Setting this bit to 1 enables the SPI module function. Setting this bit to 0 initializes a part of the module function. 0: Disables the module function 1: Enables the module function
5	SPTIE	0	R/W	Transmit Interrupt Enable Enables or disables generation of transmit interrupt requests when the number of transmit data units in the transmit buffer is equal to or less than the specified transmit buffer data triggering number and the transmit buffer empty flag (SPTEF) in SPSR is set to 1. 0: Disables the generation of transmit interrupt requests. 1: Enables the generation of transmit interrupt requests.

Bit	Bit Name	Initial Value	R/W	Description
4	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
3	—	0	R/W	Reserved The write value should always be 1. Otherwise, operation cannot be guaranteed.
2 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

40.3.2 Slave Select Polarity Register (SSLP)

SSLP sets the polarity of the QSSL signal. If the contents of SSLP are modified while the SPE bit in the control register (SPCR) is set to 1, the subsequent operation cannot be guaranteed.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SSLP
Initial Value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SSLP	0	R/W	QSSL Signal Polarity Setting Sets the polarity of the QSSL signal. 0: QSSL signal low-active 1: QSSL signal high-active

40.3.3 Pin Control Register (SPPCR)

SPPCR sets the modes of the pins. If the contents of SPPCR are modified while the SPE bit in SPCR is set to 1, the subsequent operation cannot be guaranteed.

Bit:	7	6	5	4	3	2	1	0
	—	—	MOIFE	MOIFV	—	IO3FV	IO2FV	SPLP
Initial Value:	0	0	0	0	0	1	1	0
R/W:	R	R	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	MOIFE	0	R/W	Data Output Idle Value Fixing Enable Fixes the pin output value in a QSSL negation period or the QSSL keeping period during a burst transfer. In single-SPI mode, this bit setting applies to QMO. In dual-SPI mode, this bit setting applies to QIO1 and QIO0. In quad-SPI mode, this bit setting applies to QIO3 to QIO0. 0: Output value equals final data from previous transfer 1: Output value equals the value set in the MOIFV bit Note: In dual-/quad-SPI modes, QIO1 and QIO0/QIO3 to QIO0 are driven to the Hi-Z state regardless of this bit setting (see section 40.4.2, Pin Control).
4	MOIFV	0	R/W	Data Output Idle Fixed Value If the data output idle value fixing enable bit (MOIFE) is 1, this module, according to data output idle fixed value (MOIFV) bit settings, determines the output value during the QSSL negation period. 0: Output pin idle fixed value equals 0 1: Output pin idle fixed value equals 1
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2	IO3FV	1	R/W	Single-/Dual-SPI Mode QIO3 Output Fixed Value Fixes the output direction of the QIO3 pin in single-/dual-SPI modes. This bit is valid only in single-/dual-SPI modes, and is not affected by the MOIFE or MOIFV bit values. 0: QIO3 output fixed value equals 0 1: QIO3 output fixed value equals 1
1	IO2FV	1	R/W	Single-/Dual-SPI Mode QIO2 Output Fixed Value Fixes the output direction of the QIO2 pin in single-/dual-SPI modes. This bit is valid only in single-/dual-SPI modes, and is not affected by the MOIFE or MOIFV bit values. 0: QIO2 output fixed value equals 0 1: QIO2 output fixed value equals 1
0	SPLP	0	R/W	Loopback Mode When the SPLP bit is set to 1, this module shuts off the path between the data I/O pin and the transmit/receive shift register, and connects the input path and the output path for the transmit/receive shift register. 0: Normal mode 1: Loopback mode Note: When the loopback mode is specified in dual-/quad-SPI modes, the SPI read/write access setting bit (SPRW) in command registers 0 to 3 (SPCMD0 to SPCMD3) should be set to 0 (write operation).

40.3.4 Status Register (SPSR)

SPSR indicates the operating status.

Bit:	7	6	5	4	3	2	1	0
	SPRFF	TEND	SPTEF	—	—	—	—	—
Initial Value:	0	1	1	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	SPRFF	0	R	Receive Buffer Full Flag Indicates that the number of receive data units in the receive buffer is equal to or greater than the receive buffer data triggering number specified in the buffer control register. 0: The number of receive data units in the receive buffer is less than the receive buffer data triggering number. 1: The number of receive data units in the receive buffer is equal to or greater than the receive buffer data triggering number. [Clearing conditions] • The receive buffer data is read until the number of data units in the receive buffer becomes less than the specified receive buffer data triggering number. • Receive buffer data reset is enabled. • Power-on reset [Setting condition] • The number of data units in the receive buffer is equal to or greater than the specified receive buffer data triggering number.

Bit	Bit Name	Initial Value	R/W	Description
6	TEND	1	R	Transmit End Flag This bit is set to 1 when transmission is completed, and this bit is 0 when transmission is not completed. [Clearing conditions] • When transmit data are moved from the transmit register to the transmit shift register. • When data reception is started in dual-/quad-SPI modes. [Setting condition] • When the number of data units in the transmit buffer is zero when a serial transfer is completed (except when the dummy transmission enable bit (TXDMY) is set to 1). • When there is not enough space for receiving the specified length of data in the receive buffer when a serial transfer is completed.

Bit	Bit Name	Initial Value	R/W	Description
5	SPTEF	1	R	Transmit Buffer Empty Flag Indicates that the number of transmit data units in the transmit buffer is equal to or less than the transmit buffer data triggering number specified in the buffer control register. 0: The number of transmit data units in the transmit buffer exceeds the specified transmit buffer data triggering number. 1: The number of transmit data units in the transmit buffer is equal to or less than the specified transmit buffer data triggering number. [Clearing condition] • When data is written to the transmit buffer until the number of transmit data units in the transmit buffer exceeds the specified transmit buffer data triggering number. [Setting conditions] • When the number of transmit data units in the transmit buffer is equal to or less than the specified transmit buffer data triggering number. • When transmit buffer data reset is enabled. • Power-on reset
4 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

40.3.5 Data Register (SPDR)

SPDR accesses transmit/receive data buffer.

The transmit buffer (SPTXB) and receive buffer (SPRXB) are independent and are mapped to SPDR.

When data is written to SPDR, the data will be written to the transmit buffer.

When data is read from SPDR, the data will be read from the receive buffer.

SPDR should be read or written to in byte, word, or longword units.

When SPDR is read or written to with the longword-, word-, or byte-access width, the receive or transmit data should be read from or written to the following bits.

Longword: Bits 31 to 0

Word: Bits 31 to 16

Byte: Bits 31 to 24

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SPD31	SPD30	SPD29	SPD28	SPD27	SPD26	SPD25	SPD24	SPD23	SPD22	SPD21	SPD20	SPD19	SPD18	SPD17	SPD16
Initial Value:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPD15	SPD14	SPD13	SPD12	SPD11	SPD10	SPD9	SPD8	SPD7	SPD6	SPD5	SPD4	SPD3	SPD2	SPD1	SPD0
Initial Value:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

40.3.6 Sequence Control Register (SPSCR)

SPSCR sets the sequence controlled method. If the contents of SPSCR are modified while the SPE bit in SPCR is 1, the subsequent operation cannot be guaranteed.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SPSC1	SPSC0
Initial Value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	SPSC1	0	R/W	Sequence Control Specification
0	SPSC0	0	R/W	These bits specify sequential operations. This module references SPCMD0 to SPCMD3 in the order according to these bit settings. 00: 0 → 0 → … 01: 0 → 1 → 0 → … 10: 0 → 1 → 2 → 0 → … 11: 0 → 1 → 2 → 3 → 0 → …

40.3.7 Sequence Status Register (SPSSR)

SPSSR indicates the sequence control status.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SPSS1	SPSS0
Initial Value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	SPSS1	0	R	Sequence Status
0	SPSS0	0	R	During sequence control, these bits indicate one of SPCMD0 to SPCMD3 that is currently referenced. 00: SPCMD0 01: SPCMD1 10: SPCMD2 11: SPCMD3

40.3.8 Bit Rate Register (SPBR)

SPBR sets the bit rate. If the contents of SPBR are modified while the SPE bit in SPCR is 1, the subsequent operation cannot be guaranteed.

Bit:	7	6	5	4	3	2	1	0
	SPBR7	SPBR6	SPBR5	SPBR4	SPBR3	SPBR2	SPBR1	SPBR0
Initial Value:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

The bit rate is determined by combinations of SPBR settings and the bit settings in the bit rate division setting bits (BRDV1 and BRDV0) in SPCMD0 to SPCMD3.

When SPBR is set to 0, the base bit rate is selected.

The equation for calculating the bit rate when SPBR is not 0 is given below. In the equation, n denotes an SPBR setting (1, …, 255), and N denotes bit settings in the bits BRDV1 and BRDV0 (0, 1, 2, 3).

$$\text{Bit rate} = \frac{f(P1\phi)}{2 \times n \times 2^{N}}$$

Table 40.3 shows examples of the relationship between bit rates and settings of SPBR and BRDV1 and BRDV0 bits.

Table 40.3 Relationship between Bit Rates and Settings of SPBR and BRDV1 and BRDV0 Bits

			Bit Rate
SPBR (n)	**BRDV[1:0] (N)**	**Division Ratio**	**CLKS1 = 100 MHz**
0	0	1	(Not supported)
1	0	2	50 Mbps
2	0	4	25.00 Mbps
3	0	6	16.66 Mbps
4	0	8	12.50 Mbps
5	0	10	10.00 Mbps
6	0	12	8.33 Mbps
6	1	24	4.16 Mbps
6	2	48	2.08 Mbps
6	3	96	1.04 Mbps
255	3	4080	24.51 kbps

Note: Set SPBR and BRDV that the division ratio of QSPCLK is 2 or more.

40.3.9 Data Control Register (SPDCR)

SPDCR enables or disables dummy data transmission.

Bit:	7	6	5	4	3	2	1	0
	TXDMY	—	—	—	—	—	—	—
Initial Value:	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	TXDMY	0	R/W	Dummy Data Transmission Enable Enables or disables dummy data transmission from the QMO pin in single-SPI mode and the transmit buffer is empty. Specifically, if this bit is set to 1 when the transmit buffer is empty, 0 is output from the QMO pin as dummy data. This bit setting can be changed while the transmit end flag (TEND) in SPSR is 1. Otherwise, operation cannot be guaranteed. 0: Disables dummy data transmission. 1: Enables dummy data transmission.
6 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

40.3.10 Clock Delay Register (SPCKD)

SPCKD sets a period (clock delay) from the beginning of QSSL signal assertion to QSPCLK oscillation when the QSPCLK delay setting enable bit (SCKDEN) in SPCMD0 to SPCMD3 is 1. If the contents of SPCKD are modified while the SPE bit in SPCR is 1, the subsequent operation cannot be guaranteed.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	SCKDL2	SCKDL1	SCKDL0
Initial Value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	SCKDL2	0	R/W	Clock Delay Setting
1	SCKDL1	0	R/W	These bits set a period (clock delay) from the beginning of QSSL signal assertion to QSPCLK oscillation when the SCKDEN bit in SPCMD0 to SPCMD3 is 1. 000: 1.5 QSPCLK cycles 001: 2.5 QSPCLK cycles 010: 3.5 QSPCLK cycles 011: 4.5 QSPCLK cycles 100: 5.5 QSPCLK cycles 101: 6.5 QSPCLK cycles 110: 7.5 QSPCLK cycles 111: 8.5 QSPCLK cycles
0	SCKDL0	0	R/W	

40.3.11 Slave Select Negation Delay Register (SSLND)

SSLND sets a period (QSSL negation delay) from the transmission of a final QSPCLK edge to the negation of the QSSL signal during a serial transfer when the QSSL negation delay setting enable bit (SLNDEN) in SPCMD0 to SPCMD3 is 1. If the contents of SSLND are modified while the SPE bit in SPCR is 1, the subsequent operation cannot be guaranteed.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	SLNDL2	SLNDL1	SLNDL0
Initial Value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	SLNDL2	0	R/W	QSSL Negation Delay Setting
1	SLNDL1	0	R/W	These bits set a period (QSSL negation delay) from the transmission of a final QSPCLK edge to the negation of the QSSL signal during a serial transfer when the SLNDEN bit in SPCMD0 to SPCMD3 is 1. 000: 1 QSPCLK cycle 001: 2 QSPCLK cycles 010: 3 QSPCLK cycles 011: 4 QSPCLK cycles 100: 5 QSPCLK cycles 101: 6 QSPCLK cycles 110: 7 QSPCLK cycles 111: 8 QSPCLK cycles If these bit settings are other than 2 QSPCLK or more cycles, the next-access delay setting (SPNDL) bit settings in the next-access delay register (SPND) should be other than 2 QSPCLK or more cycles.
0	SLNDL0	0	R/W	

40.3.12 Next-Access Delay Register (SPND)

SPND sets a period (next-access delay) from termination of a serial transfer to the beginning of the next serial transfer when the next-access delay enable bit (SPNDEN) in SPCMD0 to SPCMD3 is 1. If the contents of SPND are modified while the SPE bit in SPCR is 1, the subsequent operation cannot be guaranteed.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	SPNDL2	SPNDL1	SPNDL0
Initial Value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	SPNDL2	0	R/W	Next-Access Delay Setting
1	SPNDL1	0	R/W	These bits set a period (next-access delay) from termination of a serial transfer to the beginning of the next serial transfer when the SPNDEN bit in SPCMD0 to SPCMD3 is 1.
0	SPNDL0	0	R/W	000: 1 QSPCLK cycle 001: 2 QSPCLK cycles 010: 3 QSPCLK cycles 011: 4 QSPCLK cycles 100: 5 QSPCLK cycles 101: 6 QSPCLK cycles 110: 7 QSPCLK cycles 111: 8 QSPCLK cycles If these bit settings are other than 2 QSPCLK or more cycles, the QSSL negation delay setting (SLNDL) bit settings in the slave select negation delay register (SSLND) should be other than 2 QSPCLK or more cycles.

40.3.13 Command Register n (SPCMDn) (n = 0 to 3)

Each channel has four command registers (SPCMD0 to SPCMD3). SPCMD0 to SPCMD3 are used to set a transfer format. This module sequentially references SPCMD0 to SPCMD3 according to the settings in the sequence control register (SPSCR), and executes the serial transfer that is set in the referenced SPCMD.

If the contents of currently referred-to SPCMD are modified while the TEND bit in SPSR indicates that communication has not been completed, the subsequent operation cannot be guaranteed. The currently referred-to SPCMD can be checked by reading the sequence status register (SPSSR).

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0	SSLKP	SPIMOD1	SPIMOD0	SPRW	BRDV1	BRDV0	CPOL	CPHA
Initial Value:	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	SCKDEN	1	R/W	Clock Delay Setting Enable Sets a period (clock delay) from the beginning of QSSL signal assertion to QSPCLK oscillation. If this bit is 0, this module sets the clock delay to 0 QSPCLK cycle. If this bit is 1, this module starts QSPCLK oscillation in compliance with the clock delay register (SPCKD) settings. For the continuous access in which QSSL is kept asserted over the multiple commands, this bit can be set to 0 only when the pertinent command is the second or subsequent one. Otherwise, this bit should be set to 1. 0: A clock delay of 0 QSPCLK cycle 1: A clock delay equal to SPCKD settings.

Bit	Bit Name	Initial Value	R/W	Description
14	SLNDEN	1	R/W	QSSL Negation Delay Setting Enable Sets a period (QSSL negation delay) from QSPCLK oscillation stoppage to QSSL signal negation. If this bit is 0, this module sets the QSSL negation delay to 0 QSPCLK cycle. If this bit is 1, this module negates the QSSL signal in compliance with the slave select negation delay register (SSLND) settings. For the continuous access in which QSSL is kept asserted over the multiplier commands, this bit can be set to 0 only when the pertinent command is not the last one. Otherwise, this bit should be set to 1. 0: An QSSL negation delay of 0 QSPCLK cycle 1: An QSSL negation delay equal to SSLND settings.
13	SPNDEN	1	R/W	Next-Access Delay Enable Sets the period (next-access delay) from termination of a serial transfer to the beginning of the next serial transfer. If this bit is 0, this module sets the next-access delay to 0 QSPCLK cycle. If this bit is 1, this module starts next serial transfer in compliance with the next-access delay register (SPND) settings. For the continuous access in which QSSL is kept asserted over the multiple commands, this bit can be set to 0 only when the pertinent command is not the last one. Otherwise, this bit should be set to 1. 0: A next-access delay of 0 QSPCLK cycle. 1: A next-access delay equal to SPND settings.
12	LSBF	0	R/W	LSB First Sets the data format to MSB first or LSB first. 0: MSB first 1: LSB first

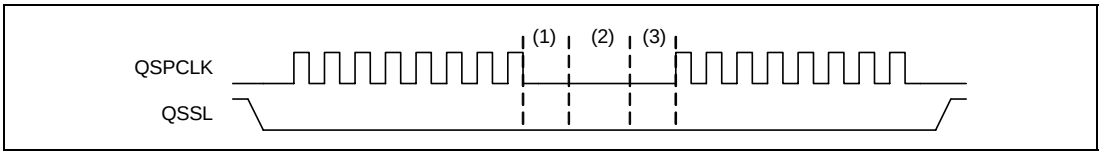
Bit	Bit Name	Initial Value	R/W	Description
11	SPB3	0	R/W	Transfer Data Length Setting
10	SPB2	0	R/W	These bits set the basic transfer data length for serial transfer. For LSB-first transfer, the transfer data is reversed within the data width specified with these bits. The actual amount of data to be transferred is determined by multiplying the value set with these bits by the value set with SPBMUL0 to SPBMUL3. 0000: 8 bits (1 byte) 0001: 16 bits (2 bytes) 0010: 32 bits (4 bytes) 0011 to 1111: Setting prohibited
9	SPB1	0	R/W	
8	SPB0	0	R/W	
7	SSLKP	0	R/W	QSSL Signal Level Keeping Specifies whether the QSSL signal level for the current command is to be kept or not from the end of the transfer for the current command to the beginning of the transfer for the next command. Setting this bit to 1 enables a transition to the next access while the QSSL signal is kept asserted. 0: Negates all QSSL signals upon completion of transfer. 1: Keeps the QSSL signal level from the end of the transfer to the beginning of the next access.
6	SPIMOD1	0	R/W	SPI Operating Mode
5	SPIMOD0	0	R/W	These bits select the operating mode from single-, dual-, or quad-SPI. 00: Single-SPI 01: Dual-SPI 10: Quad-SPI 11: Setting prohibited
4	SPRW	0	R/W	SPI Read/Write Access Sets an access direction in dual-/quad-SPI modes. This bit is invalid in single-SPI mode 0: Write operation (QIO1 and QIO0/QIO3 to QIO0: Output) 1: Read operation (QIO1 and QIO0/QIO3 to QIO0: Input)

Bit	Bit Name	Initial Value	R/W	Description
3	BRDV1	0	R/W	Bit Rate Frequency Division Setting
2	BRDV0	0	R/W	The settings of this field and of the bit rate register (SPBR) together determine the bit rate. The base bit rate depends on the setting of the SPBR. The setting of this field selects division of the base bit rate by one, two, four, or eight. Individual BRDV [1:0] values can be set in each of command registers 0 to 3. Therefore, serial transfers can be at different bit rates for each of the commands. 00: Base bit rate 01: Two division of the base 10: Four division of the base 11: Eight division of the base Note: Set SPBR and these bits that the division ratio of QSPCLK is 2 or more.
1	CPOL	0	R/W	QSPCLK Polarity Setting Sets an QSPCLK polarity. When data communication is performed between the Quad-SPI module and the other modules, the same QSPCLK polarity should be set for both modules. 0: Positive (QSPCLK = 0 when idle) 1: Negative (QSPCLK = 1 when idle)
0	CPHA	1	R/W	QSPCLK Phase Setting Sets an QSPCLK edge for latching and shifting data to be transferred. When data communication is performed between the Quad-SPI module and the other modules, the same QSPCLK edge should be set for both modules. 0: Data latch on odd edge, data shift on even edge 1: Data shift on odd edge, data latch on even edge Note: The first QSPCLK edge is treated as the first edge.

Reference: Some serial flash memory datasheets refer to QSPCLK specifications, which are determined by what this document refers to as CPOL and CPHA bits, as SPI modes 0 to 3. Assuming that SPI modes 0 to 3 are controlled by SPI mode bits [1:0], CPOL and CPHA in this document correspond to SPI mode bits 1 and 0, respectively.
In this module, the initial values of CPOL and CPHA are 0 and 1, respectively, selecting SPI mode 1 as the initial mode.

Notes: 1. When setting any or all of the clock delay period, QSSL negation delay period, and next-access delay period to 0, be sure to set SSLKP to 1 to select the continuous access in which QSSL is not negated. Otherwise, operation cannot be guaranteed. For the method of setting the various delay periods for the continuous access in which QSSL is not negated, see below.

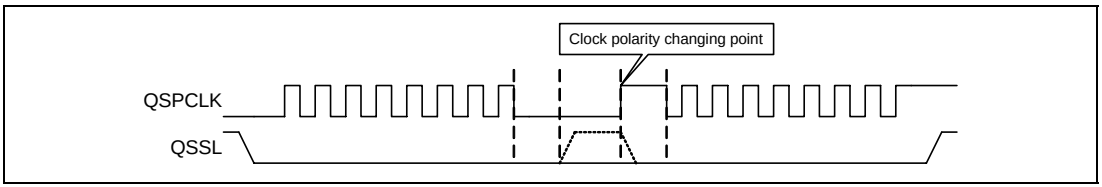
2. For the continuous access in which QSSL is not negated, QSPCLK clock stopping is followed by the QSSL negation delay period, next-access delay period, and next command clock delay period, in this order. When setting any of the QSSL negation delay setting enable bit (SLNDEN), next-access delay enable bit (SPNDEN), and clock delay setting enable bit (SCKDEN) to 0, be sure to set the bit corresponding to the later period prior to the bit corresponding to the earlier period.



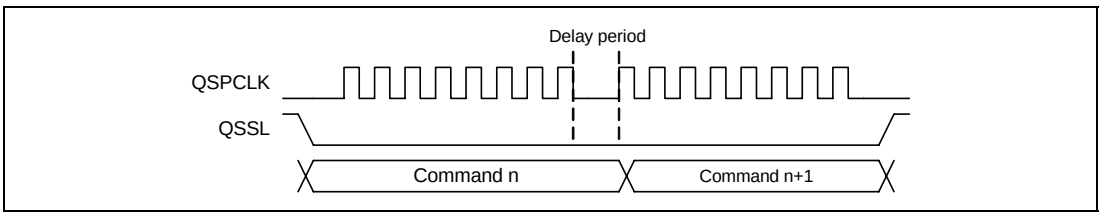
In the above figure, (1), (2), and (3) refer to the QSSL negation delay period, next-access delay period, and next command clock delay period, respectively. When setting any of these bits to 0, be sure to set (3) first. In other words, setting 1 after 0 as in {(1), (2), (3)} = {0, 0, 1}, {0, 1, 1}, {0, 1, 0} … is prohibited. Allowed setting is {(1), (2), (3)} = {1, 1, 1}, {1, 1, 0}, {1, 0, 0}, {0, 0, 0}. If set otherwise, operation cannot be guaranteed.

Notes: 1. When changing BRDV[1:0] or CPOL for each command for the continuous access in which the QSSL level is held, be sure to insert the QSSL negation delay period, next-access delay period, and clock delay period between commands. Otherwise, operation cannot be guaranteed.

2. A clock polarity changing point may be detected as a clock edge if CPOL is changed with the QSSL level held.



Note: When changing SPIMOD[1:0] or CPHA for each command for the continuous access in which the QSSL level is held, be sure to insert one cycle or more between commands. Otherwise, operation cannot be guaranteed.
(This also applies to write-to-read or read-to-write switching in dual-/quad-SPI modes.)
In the figure below, the data line is driven during the command n period if command n is for dual-/quad-SPI write access.



40.3.14 Buffer Control Register (SPBFCR)

SPBFCR resets the number of data units in the transmit buffer (SPTXB) or receive buffer (SPRXB) and sets the number of triggering data units.

Bit:	7	6	5	4	3	2	1	0
	TXRST	RXRST	TXTRG1	TXTRG0	—	RXTRG2	RXTRG1	RXTRG0
Initial Value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	TXRST	0	R/W	Transmit Buffer Data Reset Invalidates transmit data in the transmit buffer and resets the transmit buffer to an empty state. 0: Allows the transmit buffer normal operation. 1: Resets the transmit buffer.
6	RXRST	0	R/W	Receive Buffer Data Reset Invalidates receive data in the receive buffer and resets the receive buffer to an empty state. 0: Allows the receive buffer normal operation. 1: Resets the receive buffer.
5	TXTRG1	0	R/W	Transmit Buffer Data Triggering Number
4	TXTRG0	0	R/W	Specifies the timing at which the transmit buffer empty state is determined, that is when the SPTEF flag in the status register is set. When the number of bytes of data in the transmit buffer (SPTXB) is equal to or less than the specified triggering number, the SPTEF flag is set to 1. 00: 31 bytes (1 byte available) 01: 30 bytes (2 bytes available) 10: 28 bytes (4 bytes available) 11: 0 bytes (32 bytes available)
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2	RXTRG2	0	R/W	Receive Buffer Data Triggering Number
1	RXTRG1	0	R/W	Specifies the timing at which the receive buffer full state is determined, that is when the SPRFF flag in the status register is set. When the number of bytes of data in the receive buffer (SPRXB) is equal to or greater than the specified triggering number, the SPRFF flag is set to 1. 000: 1 byte (31 bytes available) 001: 2 bytes (30 bytes available) 010: 4 bytes (28 bytes available) 011: 5 bytes (27 bytes available) 100: 8 bytes (24 bytes available) 101: 16 bytes (16 bytes available) 110: 24 bytes (8 bytes available) 111: 32 bytes (0 byte available)
0	RXTRG0	0	R/W	

40.3.15 Buffer Data Count Setting Register (SPBDCR)

SPBDCR indicates the number of data units stored in the transmit buffer (SPTXB) and receive buffer (SPRXB). The upper eight bits indicate the number of transmit data units in the transmit buffer and the lower eight bits indicate the number of receive data units in the receive buffer.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	TXBC5	TXBC4	TXBC3	TXBC2	TXBC1	TXBC0	—	—	RXBC5	RXBC4	RXBC3	RXBC2	RXBC1	RXBC0
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13 to 8	TXBC[5:0]	000000	R	Transmit Data Byte Counter Indicates the number of transmit data bytes in the transmit data buffer (SPTXB). B'000000 indicates that SPTXB is empty. B'100000 indicates that SPTXB is full.
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5 to 0	RXBC[5:0]	000000	R	Receive Data Byte Counter Indicates the number of receive data bytes in the receive data buffer (SPRXB). B'000000 indicates that SPRXB is empty. B'100000 indicates that SPRXB is full.

40.3.16 Transfer Data Length Multiplier Setting Register n (SPBMULn) (n = 0, 1, 2, 3)

SPBMUL0 to SPBMUL3 set the number of times to transfer the specific length of data defined by the transfer data length setting bits (SPB[3:0]) in SPCMD0 to SPCMD3. SPBMUL0 to SPBMUL3 correspond to SPCMD0 to SPCMD3, respectively.

If a command register is referred to while the TEND bit in SPSR indicates that communication has not been completed and SPBMUL corresponding to the referred-to command register is modified, the subsequent operation is not guaranteed. The currently referred-to command register can be checked by reading the sequence status register (SPSSR).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SPBMUL [31:24]								SPBMUL [23:16]							
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPBMUL [15:8]								SPBMUL [7:0]							
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SPBMUL [31:0]	H'00000001	R/W	Transfer Data Length Multiplier Setting These bits set the multiplier for transfer data; that is, the number of times to transfer the specific length of data defined by SPB3 to SPB0 bits in SPCMD0 to SPCMD3. The actual amount of data to be transferred is determined by SPB[3:0] × SPBMUL[31:0]. Setting these bits to H'00000000 allows the defined size of data to be transferred 4,294,967,296 times.

40.4 Operation

In this section, the serial transfer period means a period from the beginning of driving valid data to the fetching of the final valid data, and the QSSL negation period means the idle period.

40.4.1 Overview of Operations

This module is capable of serial transfers in single-/dual-/quad-SPI modes. Table 40.4 gives the features of single-/dual-/quad-SPI modes.

Table 40.4 Features of Each SPI Mode

	Single-SPI	Dual-SPI	Quad-SPI
Number of data lines	One input line and one output line	Two IO lines	Four IO lines
Data line direction	Single-directional	Bidirectional	Bidirectional
Simultaneous transmission/reception	Supported	Not supported	Not supported

Table 40.5 gives the overview of operation.

Table 40.5 Overview of Operation

Items	Specification
QSPCLK signal	Output
QMO signal (single-SPI)	Output
QMI signal (single-SPI)	Input
QIO1 and QIO0 (dual-SPI)/ QIO3 to QIO0 (quad-SPI)	Input/output
QSSL signal	Output
Switching QSSL polarity	Supported
Transfer rate	Up to CLKS1
Clock source	On-chip baud rate generator
Clock polarity	Positive/negative
Clock phase	Latch at rising/output at falling Latch at falling/output at rising
Transfer bit order	MSB first/LSB first
Transfer data length	(8/16/32) × (1 to 4,294,967,296) bits
Burst transfer	Supported
QSPCLK delay control	Supported
QSSL negation delay control	Supported
Next-access delay control	Supported
Transfer start method	Writing data to the transmit buffer when SPE = 1 There is space in the receive buffer when SPE = 1*
Sequence control	Supported
Transmit buffer empty detection	Supported
Receive buffer full detection	Supported

Note: * During single-SPI operation and dual-/quad-SPI mode write operation, a transfer is started by setting SPE to 1 and writing data to the transmit buffer. During dual-/quad-SPI mode read operation, a transfer is started by setting SPE to 1 when there is space for the specified length of data in the receive buffer.

40.4.2 Pin Control

This module automatically switches the pin states according to the status after write/read transfer in single-/dual-/quad-SPI mode. The status of the data pins (QMO/QMI/QIO[3:0]) in the idle state depends on the MOIFE and MOIFV bit settings, the single-/dual-SPI mode QIO3 output fixed value bit (IO3FV) setting in, and the single-/dual-SPI mode QIO2 output fixed value bit (IO2FV) setting. Table 40.6 shows the pin states in single-SPI mode. Table 40.7 shows the pin states in dual-/quad-SPI modes.

Table 40.6 Pin States in Single-SPI Mode

Items	Single-SPI Mode
QSSL	Output
QSPCLK	Output
QMO	Output
QMI	Input
QMO in the idle state	MOIFE = 0: Final output value MOIFE = 1: MOIFV setting value
QMI in the idle state	—
QIO2	IO2FV setting value output or not used
QIO3	IO3FV setting value output or not used

Table 40.7 Pin States in Dual-/Quad-SPI Mode

Items	Dual-SPI Mode	Quad-SPI Mode
QSSL	Output	Output
QSPCLK	Output	Output
QIO0	I/O	I/O
QIO1	I/O	I/O
QIO2	IO2FV setting value output or not used	I/O
QIO3	IO3FV setting value output or not used	I/O
QIO0 in the idle state	After writing: MOIFE = 0: Final output value MOIFE = 1: MOIFV setting value After reading: Hi-Z	After writing: MOIFE = 0: Final output value MOIFE = 1: MOIFV setting value After reading: Hi-Z
QIO1 in the idle state	After writing: MOIFE = 0: Final output value MOIFE = 1: MOIFV setting value After reading: Hi-Z	After writing: MOIFE = 0: Final output value MOIFE = 1: MOIFV setting value After reading: Hi-Z
QIO2 in the idle state	IO2FV setting value output or not used	After writing: MOIFE = 0: Final output value MOIFE = 1: MOIFV setting value After reading: Hi-Z
QIO3 in the idle state	IO3FV setting value output or not used	After writing: MOIFE = 0: Final output value MOIFE = 1: MOIFV setting value After reading: Hi-Z

40.4.3 Transfer Format

The SPI has four clock settings determined by the QSPCLK polarity setting (CPOL) and QSPCLK phase setting (CPHA) bits in SPCMD0 to SPCMD3. Figure 40.1 shows the data latch/shift timing based on each setting in an 8-bit MSB first transfer. In figure 40.1, L indicates the latch timing and S indicates the shift timing. DATA corresponds to QMI/QMO in single-SPI mode; QIO1 and QIO0 in dual-SPI mode; or QIO3 to QIO0 in quad-SPI mode. t_{ckd} indicates the clock delay period when the SCKDEN bit in SPCMD0 to SPCMD3 is set to 1. Similarly, t_{slnd} indicates the QSSL negation delay period when the SLNDEN bit in SPCMD0 to SPCMD3 is set to 1, and t_{spnd} indicates the next-access delay period when the SPNDEN bit in SPCMD0 to SPCMD3 is set to 1.

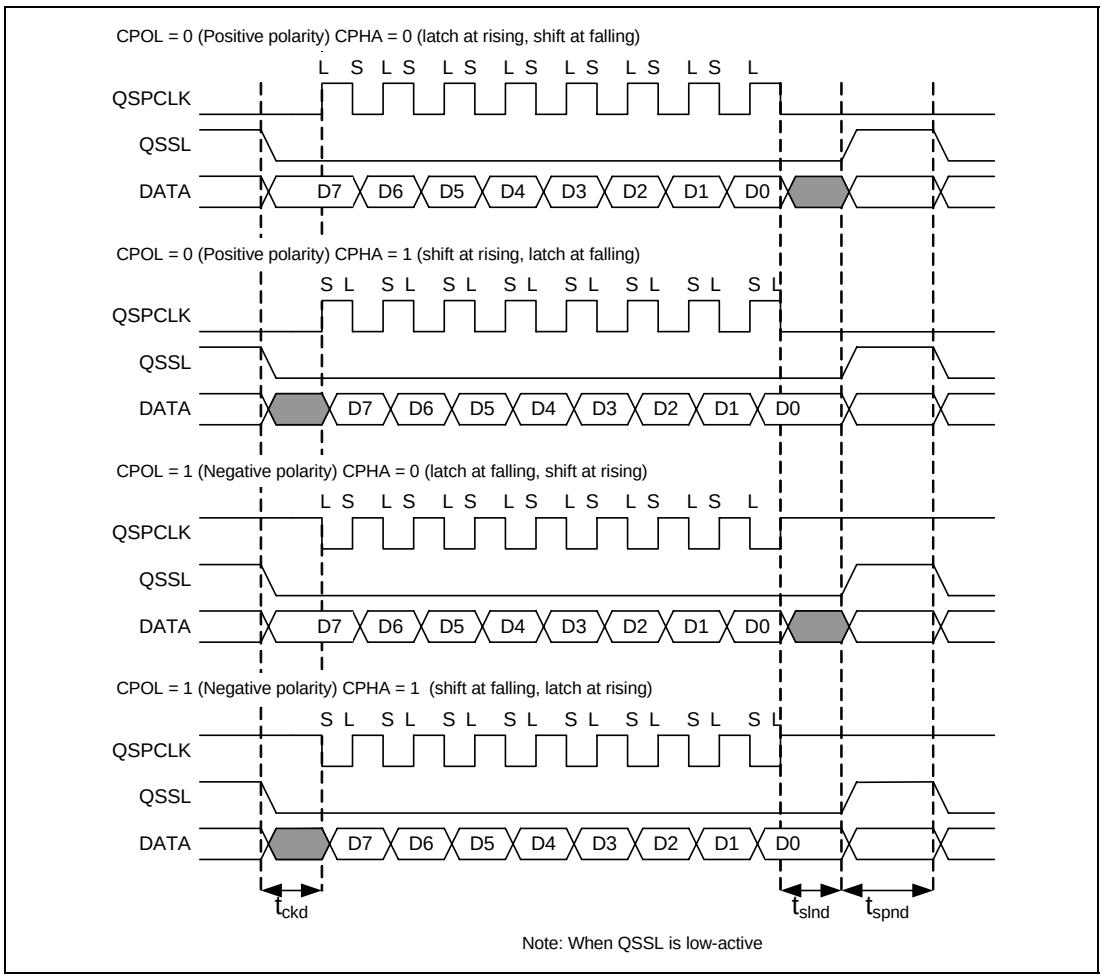


Figure 40.1 SPI Clock Setting and Transfer Timing

The following describes 8-bit MSB first transfer in single-/dual-/quad-SPI modes when CPOL = 0 and CPHA = 0.

(1) Single-SPI Mode

Figure 40.2 shows the transfer format in single-SPI mode. This mode provides transmission and reception simultaneously. Since one data line is used for serial communication both in transmission and reception, the communication speed is 1 bit per QSPCLK clock cycle. Transfer data is specified using SPCMD0 to SPCMD3. For details of transfer data, see section 40.4.4, Transfer Data.

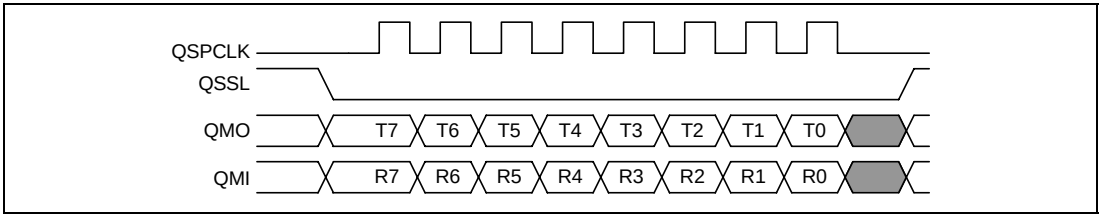


Figure 40.2 Transfer Format in Single-SPI Format

(2) Dual-SPI Mode

Figure 40.3 shows the transfer format in dual-SPI mode. This mode only provides operation of a single direction, that is, either transmission or reception. Transmission or reception can be set using the SPI read/write access setting bit (SPRW) in SPCMD0 to SPCMD3. Transmission is carried out by write operation and reception by read operation. The IO directions of QIO1 and QIO0 are switched accordingly. Since two data lines are used for serial communication both in transmission and reception, the communication speed is 2 bits per QSPCLK clock cycle. The start bit of the transfer data is output from QIO1. Transfer data is specified using SPCMD0 to SPCMD3. For details of transfer data, see section 40.4.4, Transfer Data.

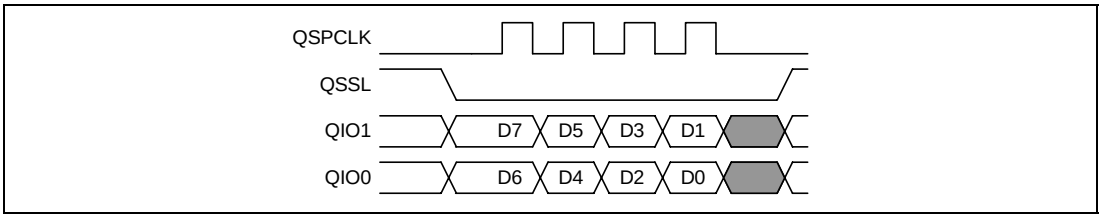


Figure 40.3 Transfer Format in Dual-SPI Format

(3) Quad-SPI Mode

Figure 40.4 shows the transfer format in quad-SPI mode. This mode provides operation of a single direction, that is, either transmission or reception. Transmission or reception can be set using the SPRW bit in SPCMD0 to SPCMD3. Transmission and reception are accomplished by writing and reading, respectively. The IO directions of QIO3 to QIO0 are switched accordingly. Since four data lines are used for serial communication both in transmission and reception, the communication speed is 4 bits per QSPCLK clock cycle. The start bit of the transfer data is output from QIO3. Transfer data is specified using SPCMD0 to SPCMD3. For details of transfer data, see section 40.4.4, Transfer Data.

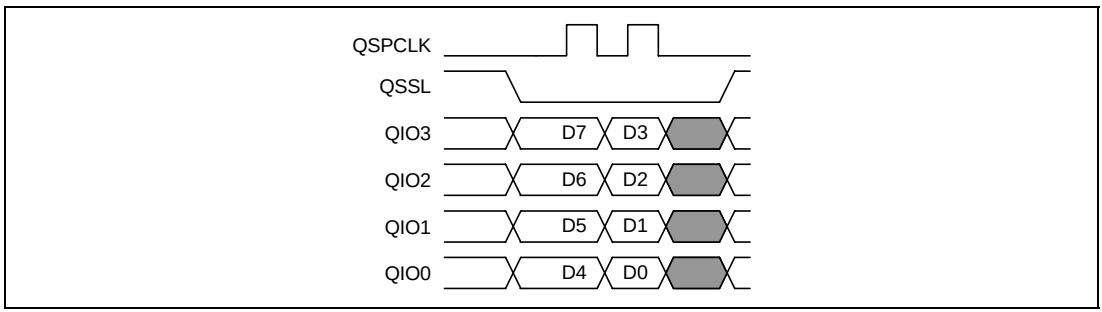


Figure 40.4 Transfer Format in Quad-SPI Format

40.4.4 Transfer Data

The data format is determined by the SPB3 to SPB0 and the LSB first (LSBF) bits in SPCMD0 to SPCMD3 and SPBMUL0 to SPBMUL3. In both MSB first and LSB first transfers, this module treats the specified size of data beginning at the MSB of the transmit shift register as transmit data, and the specified length of data beginning at the LSB of the receive shift register as receive data, regardless of whether the actual arrangement is MSB or LSB-first. The following sections describe MSB first and LSB first transfers in 32-bit, 16-bit, and 8-bit data units.

(1) MSB First Transfer (32-Bit Data)

Figure 40.5 shows the operation of the transmit buffer and transmit shift register, and the receive shift register and receive buffer when this module performs a 32-bit MSB-first data transfer.

For data transmission, the CPU or direct memory access controller writes 32-bit transmit data to the transmit buffer (SPTXB). If the transmit shift register is empty, this module copies the data with MSB-aligned in the transmit buffer to the transmit shift register, and fills the transmit shift register. When data transmission is started, this module outputs data beginning at the MSB (bit 31) of the transmit shift register, and when the QSPCLK clock cycle required for the serial transfer of 32 bits has passed, the transmit shift register becomes empty.

For data reception, data received from the data pin is stored in the receive shift register beginning at the LSB (bit 0). When the QSPCLK clock cycle required for the serial transfer of 32 bits has passed, the receive shift register becomes full. If the receive buffer (SPRXB) has a space for 32 bits or more, this module copies the 32-bit data beginning at the LSB from the receive shift register to the receive buffer, and empties the receive shift register. If the receive buffer does not have a space for 32 bits or more, data reception is not carried out. In order to start reception, the specified length of data should be read from the receive buffer to secure the space for 32 bits or more in the receive buffer.

In actual transfer, this operation is repeated for the number of times defined by SPBMUL0 to SPBMUL3.

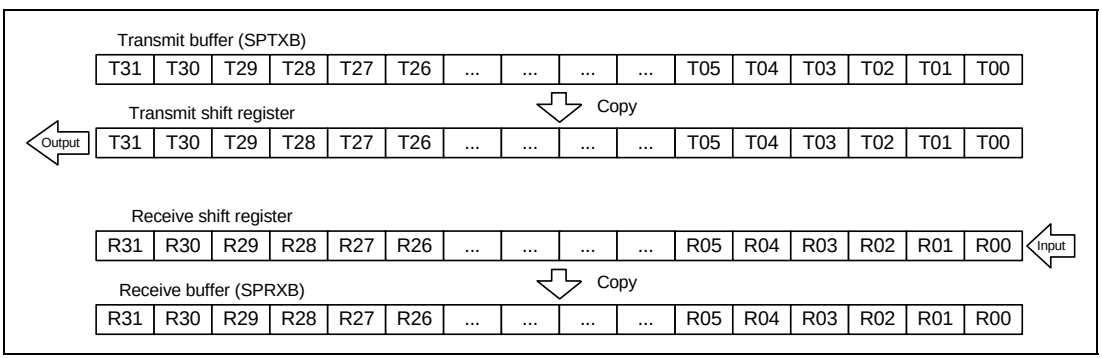


Figure 40.5 MSB First Transfer (32-Bit Data)

(2) MSB First Transfer (16-Bit Data)

Figure 40.6 shows the operation of the transmit buffer and transmit shift register, and the receive shift register and receive buffer when this module performs a 16-bit MSB-first data transfer.

For data transmission, the CPU or direct memory access controller writes 16-bit transmit data to the transmit buffer (SPTXB). If the transmit shift register is empty, this module copies the data with MSB-aligned in the transmit buffer to the transmit shift register, and fills the transmit shift register. When data transmission is started, this module outputs data beginning at the MSB (bit 31) of the transmit shift register, and when the QSPCLK clock cycle required for the serial transfer of 16 bits has passed, the transmit shift register becomes empty.

For data reception, data received from the data pin is stored in the receive shift register beginning at the LSB (bit 0). When the QSPCLK clock cycle required for the serial transfer of 16 bits has passed, the receive shift register becomes full. If the receive buffer (SPRXB) has a space for 16 bits or more, this module copies the 16-bit data beginning at the LSB from the receive shift register to the receive buffer, and empties the receive shift register. If the receive buffer does not have a space for 16 bits or more, data reception is not carried out. In order to start reception, the specified length of data should be read from the receive buffer to secure the space for 16 bits or more in the receive buffer.

In actual transfer, this operation is repeated for the number of times defined by SPBMUL0 to SPBMUL3.

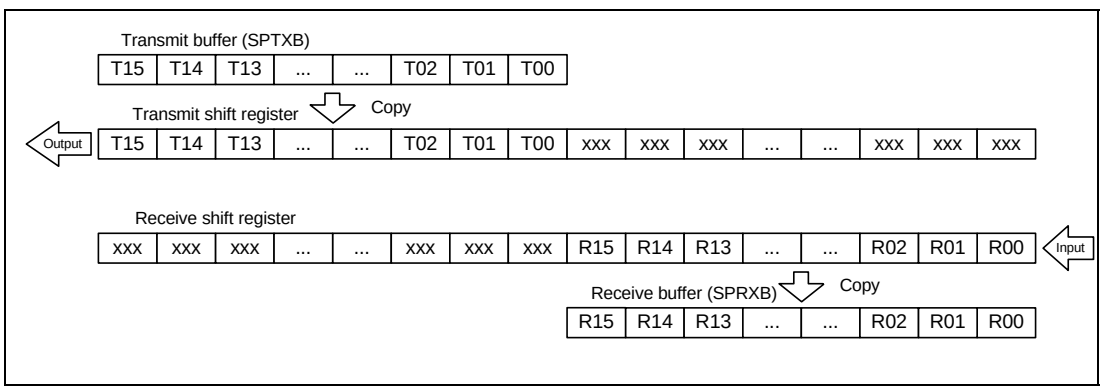


Figure 40.6 MSB First Transfer (16-Bit Data)

(3) MSB First Transfer (8-Bit Data)

Figure 40.7 shows the operation of the transmit buffer and transmit shift register, and the receive shift register and receive buffer when this module performs an 8-bit MSB-first data transfer.

For data transmission, the CPU or direct memory access controller writes 8-bit transmit data to the transmit buffer (SPTXB). If the transmit shift register is empty, this module copies the data with MSB-aligned in the transmit buffer to the transmit shift register, and fills the transmit shift register. When data transmission is started, this module outputs data beginning at the MSB (bit 31) of the transmit shift register, and when the QSPCLK clock cycle required for the serial transfer of 8 bits has passed, the transmit shift register becomes empty.

For data reception, data received from the data pin is stored in the receive shift register beginning at the LSB (bit 0). When the QSPCLK clock cycle required for the serial transfer of 8 bits has passed, the receive shift register becomes full. If the receive buffer (SPRXB) has a space for 8 bits or more, this module copies the 8-bit data beginning at the LSB from the receive shift register to the receive buffer, and empties the receive shift register. If the receive buffer does not have a space for 8 bits or more, data reception is not carried out. In order to start reception, data for the specified length of data should be read from the receive buffer to secure the space for 8 bits or more in the receive buffer.

In actual transfer, this operation is repeated for the number of times defined by SPBMUL0 to SPBMUL3.

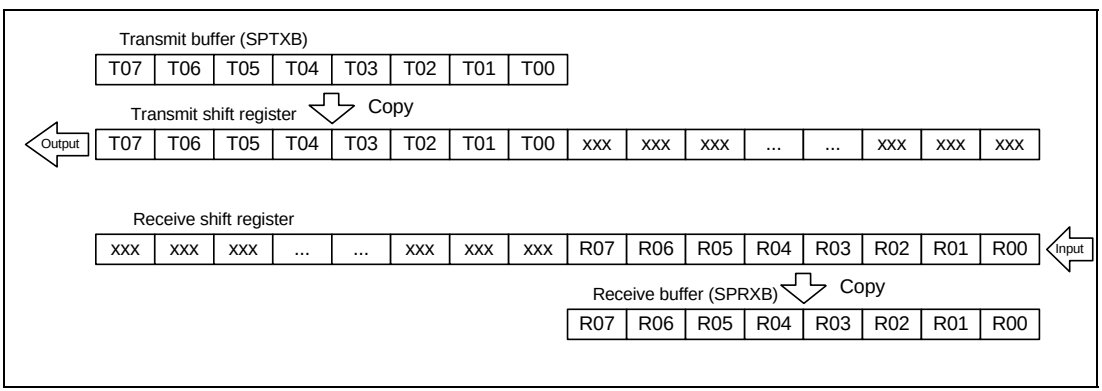


Figure 40.7 MSB First Transfer (8-Bit Data)

(4) LSB First Transfer (32-Bit Data)

Figure 40.8 shows the operation of the transmit buffer and transmit shift register, and the receive shift register and receive buffer when this module performs a 32-bit LSB-first data transfer.

For data transmission, the CPU or direct memory access controller writes 32-bit transmit data to the transmit buffer (SPTXB). If the transmit shift register is empty, this module reverses the bit order in the 32-bit transmit data, copies it with MSB-aligned to the transmit shift register, and fills the transmit shift register. When data transmission is started, this module outputs data beginning at the MSB (bit 31) of the transmit shift register, and when the QSPCLK clock cycle required for the serial transfer of 32 bits has passed, the transmit shift register becomes empty.

For data reception, data received from the data pin is stored in the receive shift register beginning at the LSB (bit 0). When the QSPCLK clock cycle required for the serial transfer of 32 bits has passed, the receive shift register becomes full. If the receive buffer (SPRXB) has a space for 32 bits or more, this module reverses the order of the bits of the 32-bit data, copies it beginning at the LSB from the receive shift register to the receive buffer, and empties the receive shift register. If the receive buffer does not have a space for 32 bits or more, data reception is not carried out. In order to start reception, the specified length of data should be read from the receive buffer to secure the space for 32 bits or more in the receive buffer.

In actual transfer, this operation is repeated for the number of times defined by SPBMUL0 to SPBMUL3.

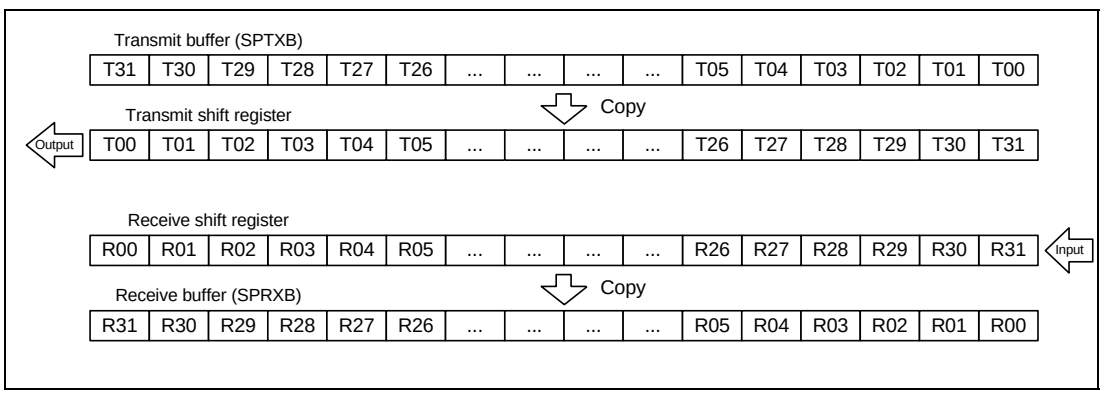


Figure 40.8 LSB First Transfer (32-Bit Data)

(5) LSB First Transfer (16-Bit Data)

Figure 40.9 shows the operation of the transmit buffer and transmit shift register, and the receive shift register and receive buffer when this module performs a 16-bit LSB-first data transfer.

For data transmission, the CPU or direct memory access controller writes 16-bit transmit data to the transmit buffer (SPTXB). If the transmit shift register is empty, this module reverses the bit order in the 16-bit transmit data, copies it with MSB-aligned to the transmit shift register, and fills the transmit shift register. When data transmission is started, this module outputs data beginning at the MSB (bit 31) of the transmit shift register, and when the QSPCLK clock cycle required for the serial transfer of 16 bits has passed, the transmit shift register becomes empty.

For data reception, data received from the data pin is stored in the receive shift register beginning at the LSB (bit 0). When the QSPCLK clock cycle required for the serial transfer of 16 bits has passed, the receive shift register becomes full. If the receive buffer (SPRXB) has a space for 16 bits or more, this module reverses the bit order in the 16-bit data, copies it beginning at the LSB from the receive shift register to the receive buffer, and empties the receive shift register. If the receive buffer does not have a space for 16 bits or more, data reception is not carried out. In order to start reception, the specified length of data should be read from the receive buffer to secure the space for 16 bits or more in the receive buffer.

In actual transfer, this operation is repeated for the number of times defined by SPBMUL0 to SPBMUL3.

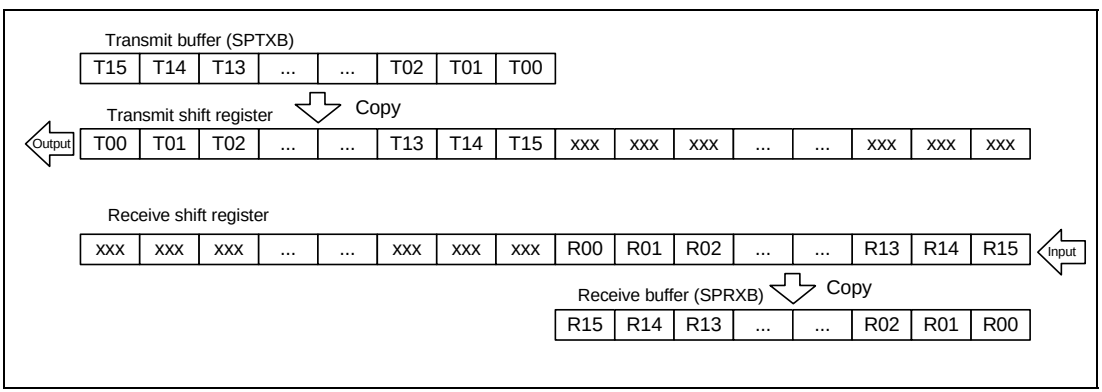


Figure 40.9 LSB First Transfer (16-Bit Data)

(6) LSB First Transfer (8-Bit Data)

Figure 40.10 shows the operation of the transmit buffer and transmit shift register, and the receive shift register and receive buffer when this module performs an 8-bit LSB-first data transfer.

For data transmission, the CPU or direct memory access controller writes 8-bit transmit data to the transmit buffer (SPTXB). If the transmit shift register is empty, this module reverses the bit order in the 8-bit transmit data, copies it with MSB-aligned to the transmit shift register, and fills the transmit shift register. When data transmission is started, this module outputs data beginning at the MSB (bit 31) of the transmit shift register, and when the QSPCLK clock cycle required for the serial transfer of 8 bits has passed, the transmit shift register becomes empty.

For data reception, data received from the data pin is stored in the received shift register beginning at the LSB (bit 0). When the QSPCLK clock cycle required for the serial transfer of 8 bits has passed, the receive shift register becomes full. If the receive buffer (SPRXB) has a space for 8 bits or more, this module reverses the bit order in the 8-bit data, copies it beginning at the LSB from the receive shift register to the receive buffer, and empties the receive shift register. If the receive buffer does not have a space for 8 bits or more, data reception is not carried out. In order to start reception, the specified length of data should be read from the receive buffer to secure the space for 8 bits or more in the receive buffer.

In actual transfer, this operation is repeated for the number of times defined by SPBMUL0 to SPBMUL3.

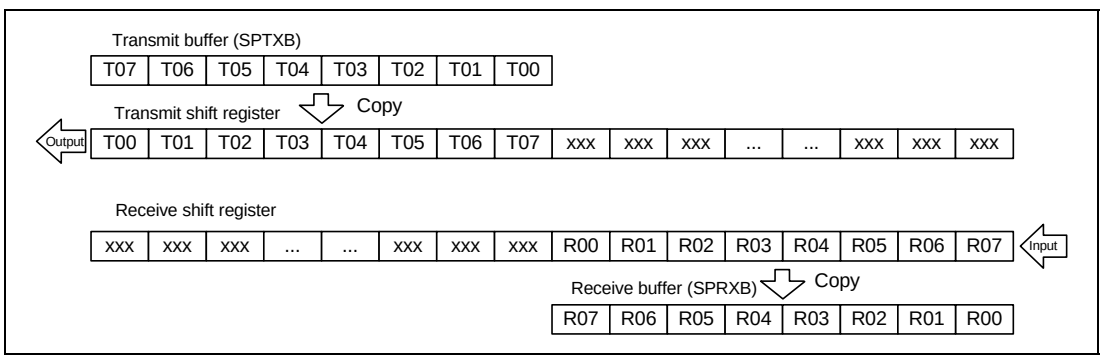


Figure 40.10 LSB First Transfer (8-Bit Data)

40.4.5 Non-Normal Transfer Operations

In the normal serial transfer, the data written from SPDR to the transmit buffer is serially transmitted, and the serially received data can be read from the receive buffer of SPDR. If access is made to SPDR, depending on the status of the transmit buffer/receive buffer, in some cases non-normal transfers can be executed.

Table 40.8 shows the relationship between non-normal transfer operations.

Table 40.8 Relationship between Non-Normal Transfer Operations

	Occurrence Condition	Operation
A	SPDR is written when the transmit buffer is full.	Missing write data.
B	SPDR is read when the receive buffer is empty.	The output data is undefined.

On operation A shown in table 40.8, whether SPDR can be written to or not can be checked using the transmit data byte counter bits (TXBC[5:0]) in the buffer data count setting register (SPBDCR).

Similarly, on operation B shown in table 40.8, whether the valid data is stored in the receive buffer or not can be checked by reading the receive data byte counter bits (RXBC[5:0]) in SPBDCR.

40.4.6 Initialization

If 0 is written to the SPE bit in SPCR, this module disables the module function, and initializes a part of the module function. When a power-on reset is generated, this module initializes all of the module function.

When the SPE bit in SPCR is cleared to 0, this module performs the following initialization:

- Suspending any serial transfer that is being executed
- Initializing the transmit shift register and the receive shift register
- Initializing the internal state machine
- Initializing the sequence
- Initializing the TEND bit in SPSR

Initialization by the clearing of the SPE bit to 0 does not initialize the control bits of this module and the transmit/receive buffer. For this reason, this module can be started in the same transfer mode as prior to the initialization if the SPE bit is re-set to 1. However, clearing the SPE bit to 0 initializes the transmit shift register and the receive shift register and allows the data that is being transferred to be discarded.

40.4.7 SPI Operation

The operating modes of this module are listed below.

- Single-SPI mode
- Dual-SPI mode/quad-SPI mode

The operation in each mode is described below.

(1) Single-SPI Mode

(a) Starting Serial Transfer

The serial transfer start conditions are: there is the specified length of data in the transmit buffer; and there is space for the specified length of data in the receive buffer.

(b) Terminating Serial Transfer

Irrespective of the clock setting, this module terminates the serial transfer after transmitting an QSPCLK edge corresponding to the final sampling timing. After the serial transfer is completed, receive data is copied from the receive shift register to the receive buffer. If there is not enough space for the specified length of data in the receive buffer after receive data is copied from the receive shift register to the receive buffer, another serial transfer will not be performed.

(c) Sequence Control

In single-SPI mode, according to the sequence length that is assigned to the sequence control register (SPSCR), this module makes up a sequence comprised of a part or all of SPCMD0 to SPCMD3 and SPBMUL0 to SPBMUL3. This module contains a pointer to the SPCMD that makes up the sequence. The value of this pointer can be checked by reading SPSSR.

When the SPE bit in SPCR is set to 1 and the function of this module is enabled, this module loads the pointer to the commands in SPCMD0, and incorporates the SPCMD0 and SPBMUL0 settings into the transfer format at the beginning of serial transfer. This module increments the pointer each time the next-access delay period for a data transfer that corresponds to the referenced SPCMD0 to SPCMD3 ends. Upon completion of the serial transfer that corresponds to the final command comprising the sequence, this module sets the pointer in SPCMD0, and in this manner the sequence is executed repeatedly.

The following items are set in command registers SPCMD0 to SPCMD3: basic transfer data length, MSB or LSB first, clock settings, some of the bit rate settings, SPI transfer mode and transfer direction (only in dual-/quad-SPI modes), whether QSSL level is held, a clock delay period, an QSSL negation delay period, and a next-access delay period. The total amount of data to be transferred is determined by multiplying the basic length of data to be transferred by the value set with SPBMUL0 to SPBMUL3.

Figure 40.11 shows an operation example when SPSCR is set to H'02, and the sequence is configured based on SPCMD0 to SPCMD2 settings. In figure 40.11, shaded areas of QMO/QMI indicate invalid data. Periods (1) to (3) in figure 40.11 indicate the followings.

(1) Clock delay period (SPCKD) setting value = B'000 (1 QSPCLK cycle)

(2) QSSL negation delay period (QSSLND) setting value = B'000 (1 QSPCLK cycle)

(3) Next-access delay period (SPND) setting value = B'000 (1 QSPCLK cycle)

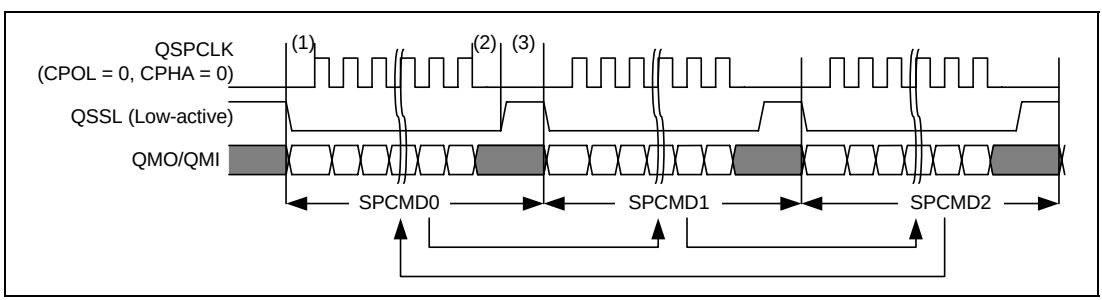


Figure 40.11 Sequence Control Operation

(d) Burst Transfer

This module can execute burst transfer with the following two methods in single-SPI mode.

One method uses the SPB[3:0] bits in SPCMD0 to SPCMD3 and SPBMUL0 to SPBMUL3. Setting SPB[3:0] to select 8, 16, or 32 bits and setting SPBMUL0 to SPBMUL3 to select one through 4,294,967,296 allows the specified length of data to be continuously transferred for the specified times, where the length is specified by SPB[3:0] and the number of times is specified by SPBMUL0 to SPBMUL3. However, if the transmit buffer (SPTXB) becomes empty during transfer, or the receive buffer (SPRXB) has no longer a space enough to receive the specified length of data defined by SPB[3:0], the clock is stopped until transfer is resumed. Figure 40.12 shows a burst transfer example in which SPB[3:0] are set to select 32 bits and SPBMUL to select four times thus specifying 128 bits as a total transfer data amount. The following describes operations (1) to (4) in the figure.

(1) First 32-bit data transfer

(2) Second 32-bit data transfer

(3) When the transmit buffer becomes empty or the receive buffer has no longer a space for 32 or more bits, the clock is stopped. Here, the QMO continues outputting the previous value. When data is written to the transmit buffer or an enough space is created in the receive buffer, the clock output is resumed to restart transfer.

(4) Third and fourth 32-bit data transfer

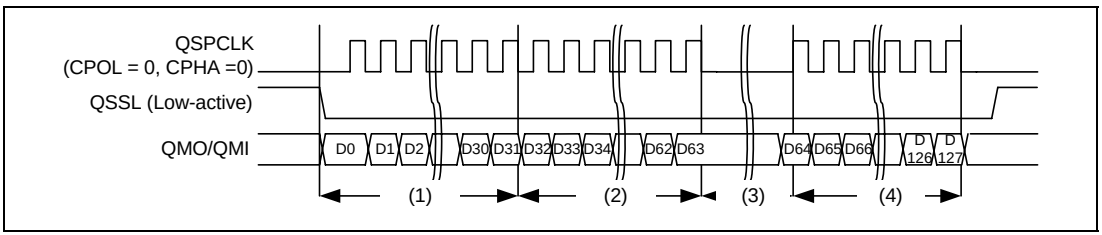


Figure 40.12 Burst Transfer Example in which Total Transfer Data Amount is 128 Bits (Single-SPI Mode Used)

In the other method, QSSL is kept asserted after a serial transfer is completed until the next serial transfer. Setting the QSSL signal level keeping bit (SSLKP) to 1 in SPCMD0 to SPCMD3 allows the QSSL signal to be kept asserted after the transfer corresponding to the pertinent command register is completed until the next transfer. Figure 40.13 shows a burst transfer example in which the QSSL signal level keeping function is used. The following describes operations (1) to (6) in the figure.

(1) Clock delay period according to the SPCMD0 setting. The setting must be made so that the delay period should be at least 1 QSPCLK cycle for the first transfer in the burst transfer.

(2) QSSL negation delay period according to the SPCMD0 setting. Since SSLKP is set to 1, QSSL is not negated even after QSSL negation delay period is over. The QSSL negation delay period depends on the SLNDEN bit setting in SPCMD0. When SLNDEN is 1, the QSSL negation delay period is determined by the SSLND setting, and the delay period is 0 QSPCLK cycle when SLNDEN is 0.

(3) Next-access delay period according to the SPCMD0 setting. Since SSLKP is set to 1, QSSL is not negated even during the next-access delay period. The next-access delay period depends on the SPNDEN bit setting in SPCMD0. When SPNDEN is 1, the next-access delay period is determined by the SPND setting, and the delay period is 0 QSPCLK cycle when SPNDEN is 0.

(4) Clock delay period according to the SPCMD1 setting. The clock delay period depends on the SCKDEN bit setting in SPCMD1. When SCKDEN is 1, the clock delay period is determined by the SPCKD setting, and the delay period is 0 QSPCLK cycle when SCKDEN is 0.

(5) QSSL negation delay period according to the SPCMD1 setting. The setting must be made so that the delay period should be at least one QSPCLK cycle for the last transfer in the burst transfer. Since SSLKP in SPCMD1 is set to 0, QSSL is negated after QSSL negation delay period is over.

(6) Next-access delay period according to the SPCMD1 setting. The setting must be made so that the delay period should be at least one QSPCLK cycle for the last transfer in the burst transfer. Be sure to set SSLKP to 0 to negate QSSL.

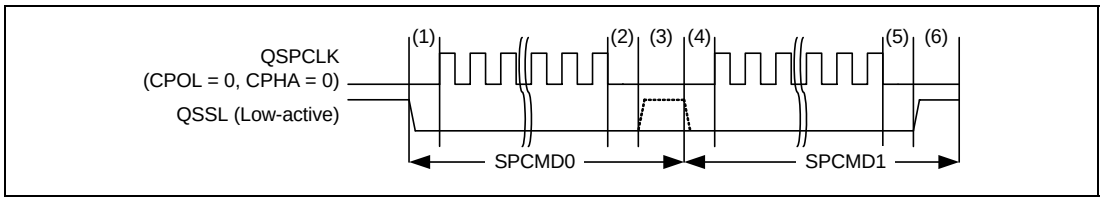


Figure 40.13 Burst Transfer Example in which QSSL Signal Level Keeping Function is Used (Single-SPI Mode)

Note the following when specifying a burst transfer using this method.

Periods (2) to (4) must be inserted without fail when changing the clock frequency division ratio or clock polarity through command update.

When the clock frequency division ratio is changed, period (4) may be advanced or delayed with respect to the set value.

At least period (2) must be inserted when changing the clock phase or transfer mode (single-/dual-/quad-SPI) through command update (changing dual-/quad-SPI includes changing read/write operation).

(e) Initialization Flowchart

Figure 40.14 is a flowchart illustrating an example of initialization in SPI operation when this module is used in single-SPI mode. For a description of how to set up the interrupt controller and direct memory access controller, see the descriptions given in the individual blocks.

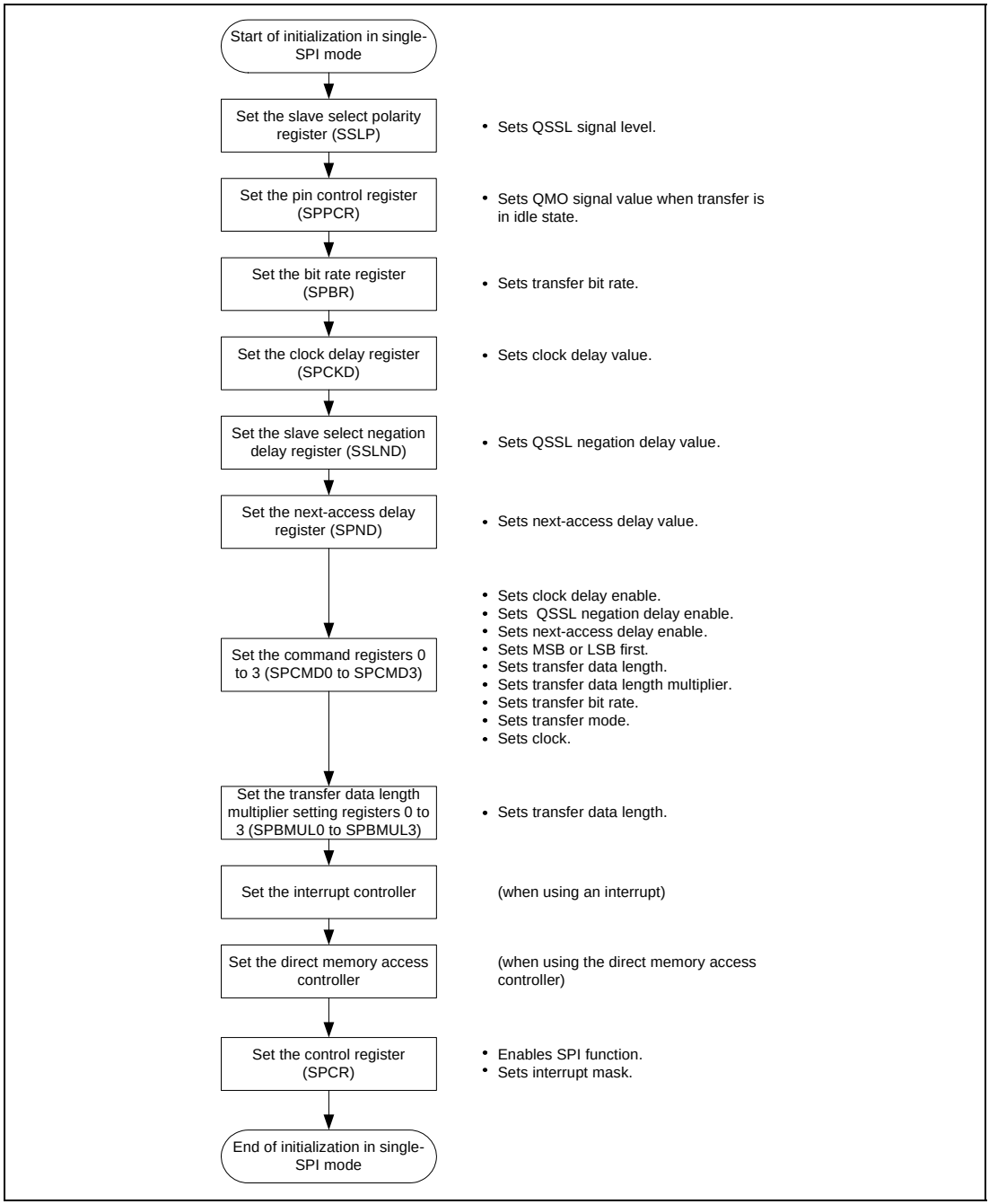


Figure 40.14 Example of Initialization Flowchart in Single-SPI Master Mode

(f) Transfer Operation Flowchart

Figure 40.15 is a flowchart illustrating a transfer in SPI operation when this module is used in single-SPI mode. Burst transfer by setting the transfer data length is also executed based on this flowchart.

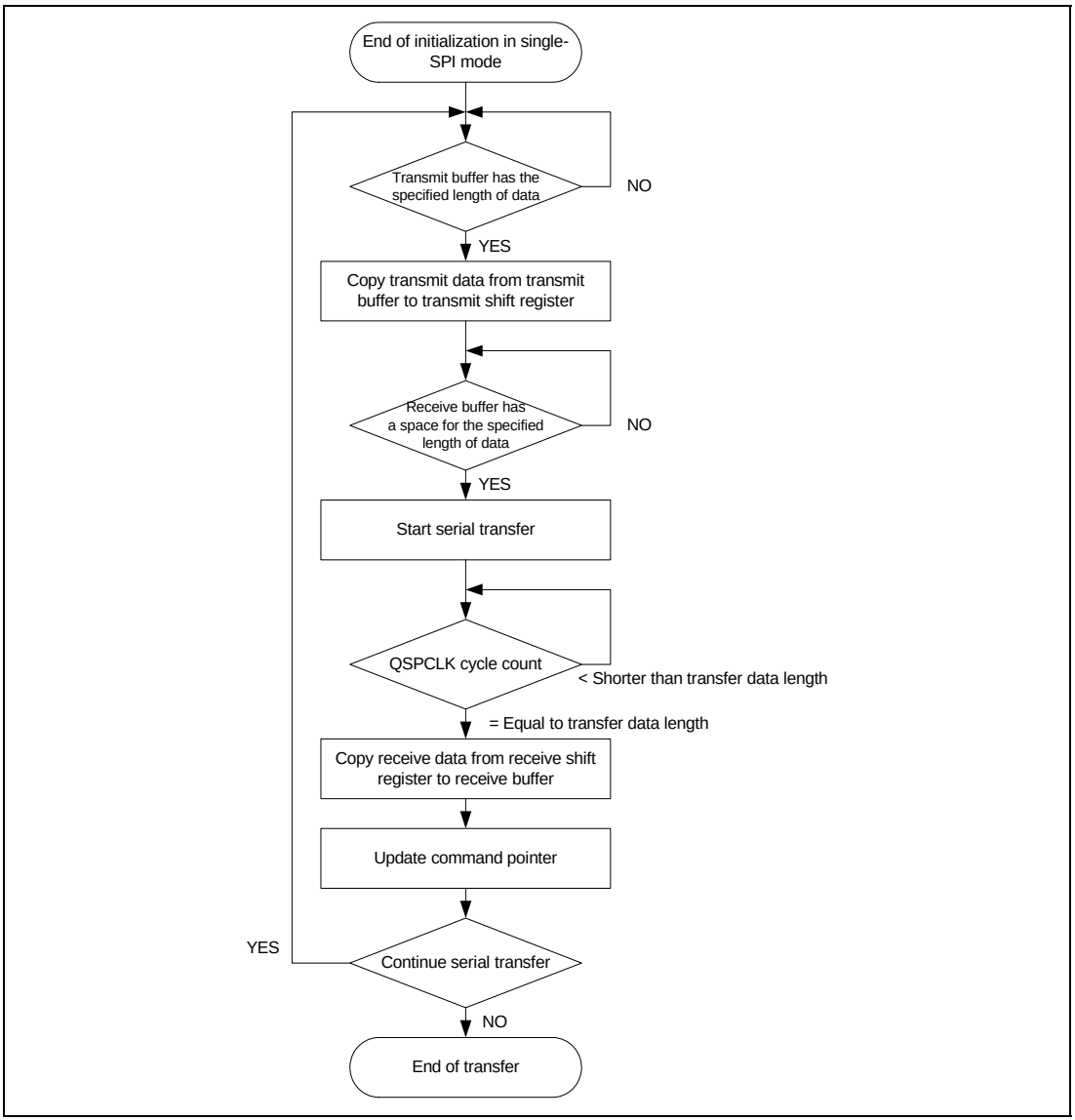


Figure 40.15 Transfer Operation Flowchart in Single-SPI Mode

(2) Dual-SPI/Quad-SPI Mode

(a) Starting Serial Transfer

In dual-/quad-SPI modes, the serial transfer start condition is different depending on the data transfer direction (transmission or reception).

In data transmission, the serial transfer start condition is that there is the specified length of data in the transmit buffer.

In data reception, the serial transfer start condition is that there is a space for the specified length of data in the receive buffer.

(b) Terminating Serial Transfer

Irrespective of data transmission or reception, this module terminates the serial transfer after transmitting an QSPCLK edge corresponding to the final sampling timing.

During idle cycles in dual-/quad-SPI modes, the IO pins are controlled differently depending on whether it is after write or read operation. Specifically, the IO pins output either the last output data or the fixed level depending on the register setting after write operation, whereas the IO pins are driven to the Hi-Z state after read operation. Figure 40.16 shows an example of the pin states after quad-SPI mode access is completed. The following describes operations (1) and (2) in the figure.

(1) During write operation, QIO0 to QIO3 serve as output pins. Thus, when QSSL is negated upon completion of write operation, QIO0 to QIO3 output different values depending on the value of MOIFE in the SPPCR. Specifically, the IO pins output the level specified by MOIFV when MOIFE is 1, whereas the IO pins output the last output data when MOIFE is 0.
(2) During read operation, QIO0 to QIO3 serve as input pins. Thus, when QSSL is negated upon completion of read operation, QIO0 to QIO3 are driven to Hi-Z state irrespective of the values of MOIFE and MOIFV.

For details on the pin control in dual-/quad-SPI modes, see section 40.4.2, Pin Control.

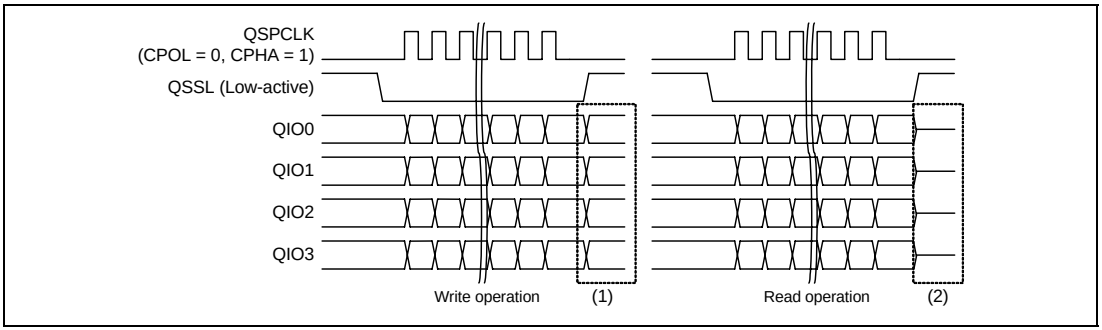


Figure 40.16 Example of Pin States after Quad-SPI Mode Access is Completed

(c) Sequence Control

As with the single-SPI mode, in dual-/quad-SPI modes, according to the sequence length that is assigned to SPSCR, this module makes up a sequence comprised of SPCMD0 to SPCMD3 and SPBMUL0 to SPBMUL3. For details on operation, see section 40.4.7 (1) (c), Sequence Control.

Dual-/quad-SPI modes only provide operation of a single direction, that is, either transmission or reception for serial transfer. Transmission or reception is set using the SPI read/write access setting bit (SPRW) in SPCMD0 to SPCMD3. One of the three operating modes including dual-SPI mode, quad-SPI mode, and single-SPI mode is set using the SPI operating mode setting bits (SPIMOD[1:0]) in SPCMD0 to SPCMD3. Combining these bits allow switching single-SPI mode, dual-SPI mode transmission/reception, and quad-SPI mode transmission/reception to control sequence. Figure 40.17 shows an example of sequence configuration with transfer mode switching.

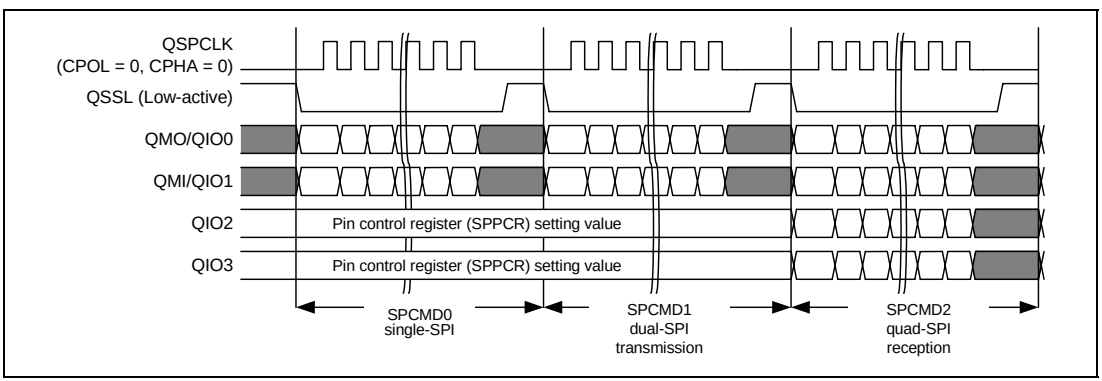


Figure 40.17 Example of Sequence Configuration with Transfer Mode Switching

Note the following when configuring a sequence in dual-/quad-SPI modes.

When all the commands configuring a sequence are dual-/quad-SPI read operations, the sequential operation is continued as long as the receive buffer has an enough space for the receive data.

To terminate read operation, clear the SPE bit to 0 in SPCR after receiving the required length of data, or execute write operation for the last sequence to empty the transmit buffer.

(d) Burst Transfer

This module can execute burst transfer with the following two methods in dual/quad SPI modes.

One method uses the SPB[3:0] bits in SPCMD0 to SPCMD3 and SPBMUL0 to SPBMUL3. As with the single-SPI mode, setting SPB[3:0] to select 8, 16, or 32 bits and setting SPBMUL0 to SPBMUL3 to select one through 4,294,967,296 allows the specified length of data to be continuously transferred for the specified times, where the length is specified by SPB[3:0] and the number of times is specified by SPBMUL0 to SPBMUL3. However, if the transmit buffer (SPTXB) becomes empty during transfer, or the receive buffer (SPRXB) has no longer a space enough to receive the specified length of data defined by SPB[3:0], the clock is stopped until transfer is resumed. This method is effective to transfer a large amount of data in dual-/quad-SPI modes. Figure 40.18 shows a burst transfer example in which SPB[3:0] are set to select 32 bits and SPBMUL to select four times thus specifying 128 bits as a total transfer data amount. The following describes operations (1) to (4) in the figure.

(1) First 32-bit data transfer

(2) Second 32-bit data transfer

(3) When the transmit buffer becomes empty or the receive buffer has no longer a space for 32 or more bits, the clock is stopped. Here, when QIO3 to QIO0 serve as output pins, QIO3 to QIO0 continue outputting the previous value. When QIO3 to QIO0 serve as input pins, the inputs to QIO3 to QIO0 depend on the output value of the device to communicate with. When data is written to the transmit buffer or an enough space is created in the receive buffer, the clock output is resumed to restart transfer.

(4) Third and fourth 32-bit data transfer

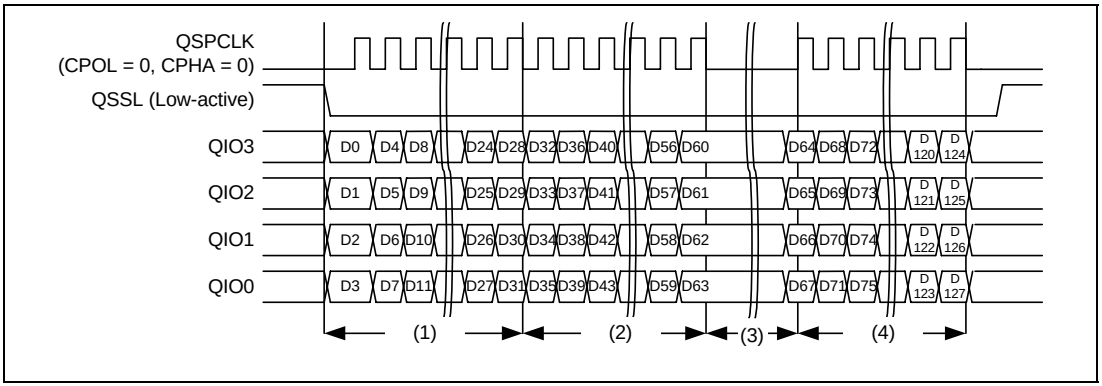


Figure 40.18 Burst Transfer Example in which Total Transfer Data Amount is 128 Bits (Quad-SPI Mode)

The other method uses the QSSL signal level keeping function as in single-SPI mode. Since this method allows switching the SPI transfer modes (single-/dual-/quad-SPI) during a transfer, it is particularly convenient, for example, when used with serial flash memory, where command data is written in single-SPI mode and data to be stored in memory is written in quad-SPI mode. Note, however, that at least one delay cycle should be inserted between transfers when switching the SPI transfer modes. Figure 40.19 shows a burst transfer example in which both single-SPI and quad-SPI modes are used. The following describes operations (1) to (6) in the figure.

(1) Clock delay period according to the SPCMD0 setting. The setting must be made so that the delay period should be at least 1 QSPCLK cycle for the first transfer in the burst transfer.

(2) QSSL negation delay period according to the SPCMD0 setting. Since SSLKP in SPCMD0 is set to 1, QSSL is not negated even after QSSL negation delay period is over. The QSSL negation delay period depends on the SLNDEN bit setting in SPCMD0. When SLNDEN is 1, the QSSL negation delay period is determined by the SSLND setting, and the delay period is 0 QSPCLK cycle when SLNDEN is 0.

(3) Next-access delay period according to the SPCMD0 setting. Since SSLKP is set to 1, QSSL is not negated even during the next-access delay period. The next-access delay period depends on the SPNDEN bit setting in SPCMD0. When SPNDEN is 1, the next-access delay period is determined by the SPND setting, and the delay period is 0 QSPCLK cycle when SPNDEN is 0. Up to this period, the data pin is driven according to the SPCMD0 setting.

(4) Clock delay period according to the SPCMD1 setting. The clock delay period depends on the SCKDEN bit setting in SPCMD1. When SCKDEN is 1, the clock delay period is determined by the SPCKD setting, and the delay period is 0 QSPCLK cycle when SCKDEN is 0.

(5) QSSL negation delay period according to the SPCMD1 setting. The setting must be made so that the delay period should be at least one QSPCLK cycle for the last transfer in the burst transfer. Since SSLKP in SPCMD1 is set to 0, QSSL is negated after QSSL negation delay period is over.

(6) Next-access delay period according to the SPCMD1 setting. The setting must be made so that the delay period should be at least one QSPCLK cycle for the last transfer in the burst transfer. Be sure to set SSLKP to 0 to negate QSSL.

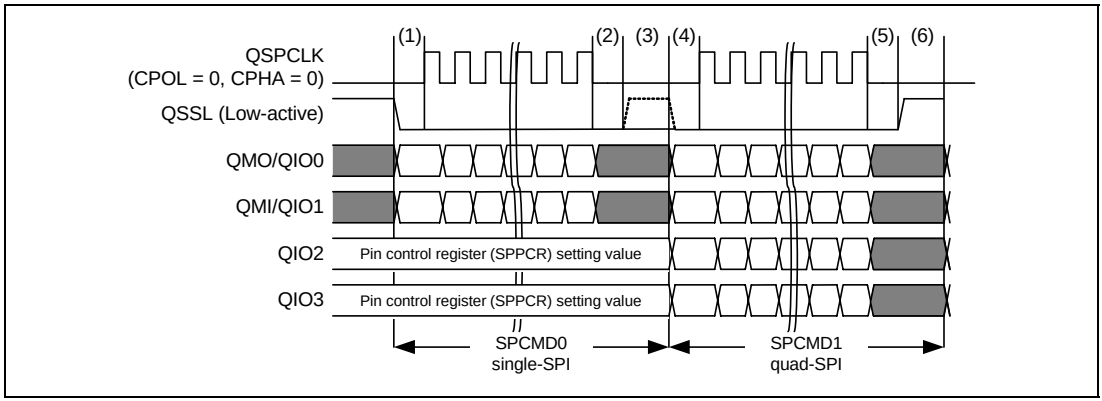


Figure 40.19 Burst Transfer Example in which QSSL Signal Level Keeping Function is Used (Single- and Quad-SPI Modes Used)

Note the following when specifying a burst transfer using this method.

Periods (2) to (4) must be inserted without fail when changing the clock frequency division ratio or clock polarity through command update.

When the clock frequency division ratio is changed, period (4) may be advanced or delayed with respect to the set value.

At least period (2) must be inserted when changing the clock phase or transfer mode (single-/dual-/quad-SPI) through command update (changing dual-/quad-SPI includes changing read/write operation).

(e) Initialization Flowchart

Figure 40.20 is a flowchart illustrating an example of initialization in SPI operation when this module is used in dual-/quad-SPI mode. For a description of how to set up the interrupt controller and direct memory access controller, see the descriptions given in the individual blocks.

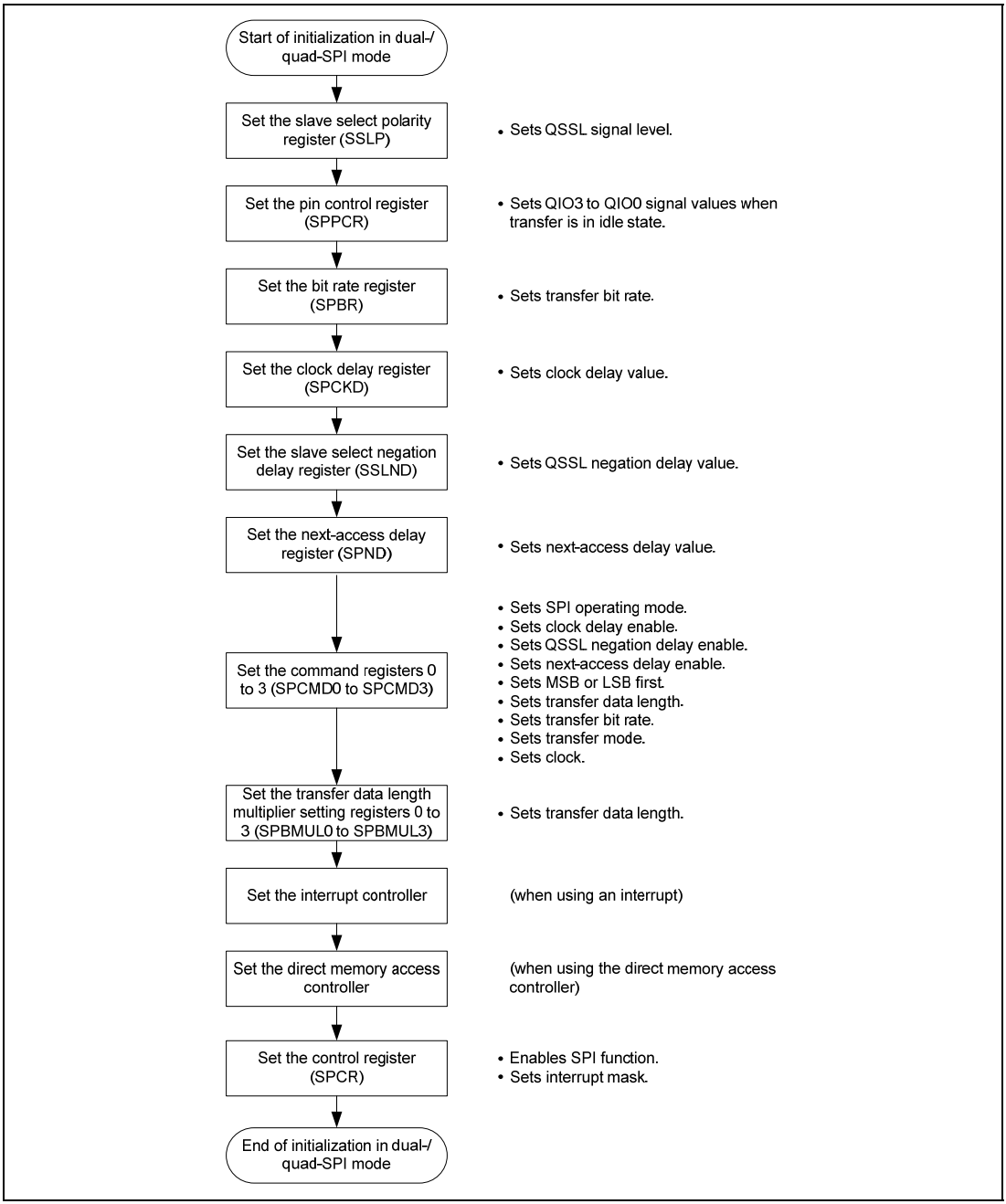


Figure 40.20 Example of Initialization Flowchart in Dual-/Quad-SPI Mode

(f) Transfer Operation Flowchart

Figure 40.21 is a flowchart illustrating a transfer in SPI operation when this module is used in dual-/quad-SPI mode.

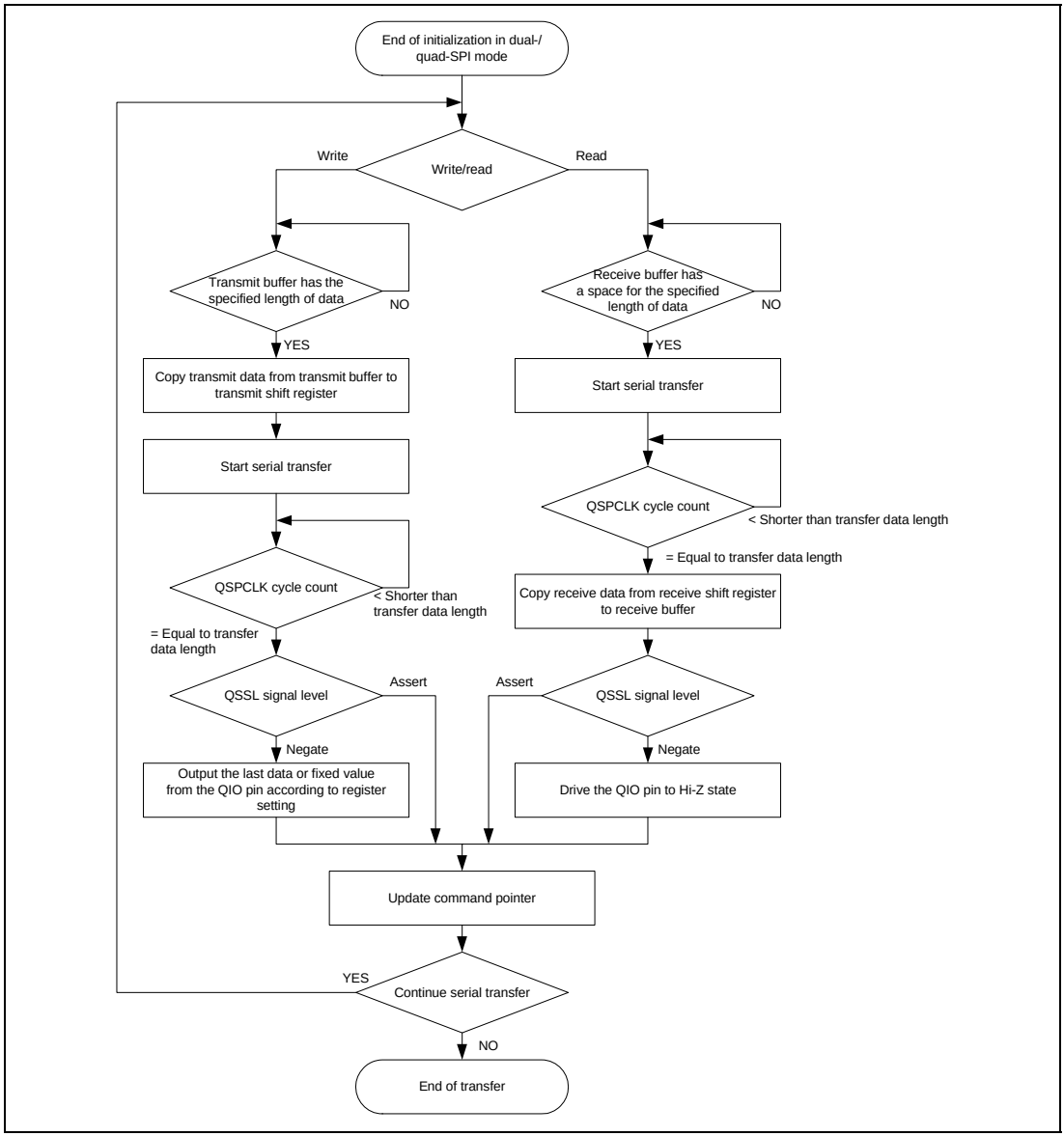


Figure 40.21 Transfer Operation Flowchart in Dual-/Quad-SPI Mode

40.4.8 Interrupt Sources

This module has interrupt sources of receive buffer full and transmit buffer empty. In addition, the direct memory access controller can be activated by the receive buffer full or transmit buffer empty interrupt for data transfer.

Table 40.9 shows the interrupt sources.

When any of the interrupt conditions in table 40.9 is met, an interrupt is generated. The interrupt sources should be cleared with data transfer by the CPU or direct memory access controller.

Table 40.9 Interrupt Sources

Name	Interrupt Source	Abbreviation	Interrupt Condition	Activation of Direct Memory Access Controller
SPRI	Receive buffer full	RXI	(SPRIE = 1) • (SPRFF = 1)	Possible
SPTI	Transmit buffer empty	TXI	(SPTIE = 1) • (SPTEF = 1)	Possible

40.4.9 Loopback Mode

This module provides loopback mode for testing. Writing 1 to the loopback mode bit (SPLP) in the pin control register (SPPCR) enables loopback mode. In loopback mode, this module disconnects the paths between the transmit/receive shift registers and the QMI/QMO and QIO3 to QIO0 pins, and connects the outputs from the transmit shift register to the inputs to the receive shift register instead. Figure 40.22 shows a schematic internal connection in loopback mode.

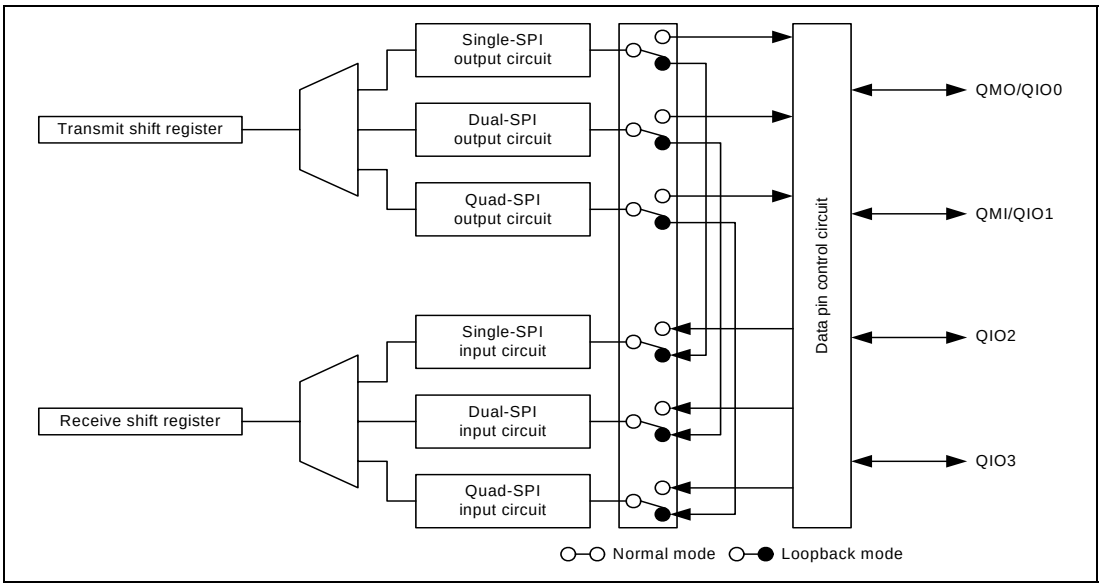


Figure 40.22 Schematic Internal Connection in Loopback Mode

Section 41 Electrical Characteristics

41.1 Absolute Maximum Ratings

Table 41.1 Absolute Maximum Ratings

Item		Value	Unit	Remarks
Power supply voltage (VCCQ, VCC, VCCQ-PLL, AV33)		-0.3 to +4.6	V	
Power supply voltage (VDD-DDR)		-0.3 to +2.6	V	
Power supply voltage (VDD, VDD-PLL, AV12)		-0.3 to +1.8	V	
Analog power supply voltage (AVCC)		-0.3 to +4.6	V	*1 *2
Analog power supply voltage (Avref)		-0.3 to AVCC +0.3	V	*1 *2
Input voltage	(3.3-V I/O)	-0.3 to VCCQ +0.3	V	*1 *2
	(DDR I/O)	-0.3 to VDD-DDR +0.3	V	*1 *3
	(OVC0/VBUS0 pin, OVC1/VBUS1 pin)	-0.3 to +5.5	V	*1
Output voltage	(3.3-V I/O)	-0.3 to VCCQ +0.3	V	*1 *2
	(DDR I/O)	-0.3 to VDD-DDR +0.3	V	*1 *3
Output current	(output pins)	22 (6 in a buffer)	mA	
		30 (8 in a buffer)	mA	
Storage temperature		-55 to +125	°C	
Operating temperature		-40 to +85	°C	Wide-range specifications

Notes: Permanent damage to the LSI may result if absolute maximum ratings are exceeded. In normal operation, this LSI should be used within the specifications described in the following descriptions. If this LSI is not used within the specifications, the reliability of this LSI may lower.

Voltages are referenced at GND = VSS = 0 V.

1. Must be higher than or equal to –0.3 V.
2. Do not exceed 4.6 V.
3. Do not exceed 2.6 V.

41.2 Power Supply

Table 41.2 Power Supply*[1]

Item	Symbol	Min.	Typ.	Max.	Unit	Remarks
Power supply (Internal)	VDD	1.15	—	1.30	V	VDD for power supply and VSS for ground
Power supply (3.3-V I/O)	VCCQ, VCC*[2]	3.0	3.3	3.6	V	VCCQ for power supply and VSSQ for ground
Power supply (DDR I/O)	VDD-DDR (DDR2)	1.7	1.8	1.9	V	VDD-DDR for power supply and VSS for ground
	VDD-DDR (only for products supporting DDR3)	1.425	1.5	1.575	V	
	MVREFCA	—	VSS	—	V	MVREFCA is a test pin.
	MVREFDQ	—	VDD-DDR/2	—	V	
Power supply (PLL)	VCCQ-PLL	3.0	3.3	3.6	V	VCCQ-PLL for power supply and VSSQ-PLL for ground
Power supply (PLL)	VDD-PLL	1.15	—	1.30	V	VDD-PLL for power supply and VSS-PLL for ground
Power supply (AVCC)	AVCC	3.0	3.3	3.6	V	AVCC for power supply and AVSS for ground
Power supply (AVREF)	AVREF	—	—	AVREF ≤ AVCC + 0.3	V	AVREF for power supply and AVSS for ground
Power supply (USB)	AV33	3.0	3.3	3.6	V	AV33 for power supply and AG for ground
	AV12	1.15	—	1.30	V	AV12 for power supply and AG for ground

Notes: 1. Whether or not they are to be used, supply the specified voltages for all of the power supplies. Not doing so creates a risk of permanent damage to the LSI.
2. Power-supply lines for VCC and VCCQ must be separated widely on the PCB.

41.3 Sequence of Turning On/Off Power Supplies

41.3.1 Sequence of Turning On/Off Power Supplies with Different Voltages

When supplying power, turn on AVCC and then AVREF in that order. When shutting down the power, switch off AVREF and then AVCC in that order. Turn the other power supplies on or off as follows.

This subsection specifies the sequence of turning on/off power supplies with different voltages: 1.2-V power supplies (hereinafter referred to as VDD12 and the LSI pins are VDD, VDD-PLL, and AV12), DDR power supplies (referred to as VDD-DDR, and the LSI pin is VDD-DDR), and 3.3-V power supplies (referred to as VDD33, and the LSI pins are VCCQ, VCC, VCCQ-PLL, and AV33).

VSS** indicates the GND level of each power supply.

(1) Turning On Power Supply

There are no restrictions on the power-on sequence. Ensure that all other power supplies rise from the ground (VSS**) level within 300 ms of any single power supply rising from the ground (VSS**) level.

(2) Turning Off Power Supply

There are no restrictions on the power-off sequence. Ensure that all other power supplies fall to the ground (VSS**) level within 300 ms of any single power supply being turned off.

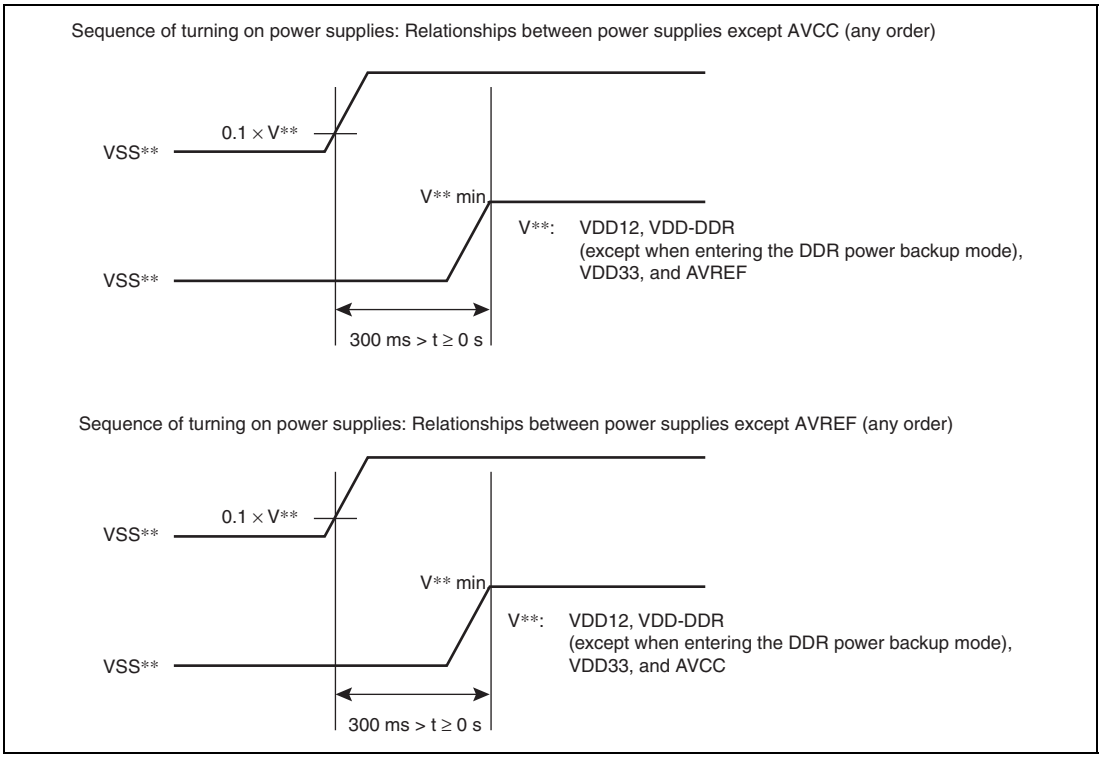


Figure 41.1 Sequence of Turning On Power Supplies with Different Voltages

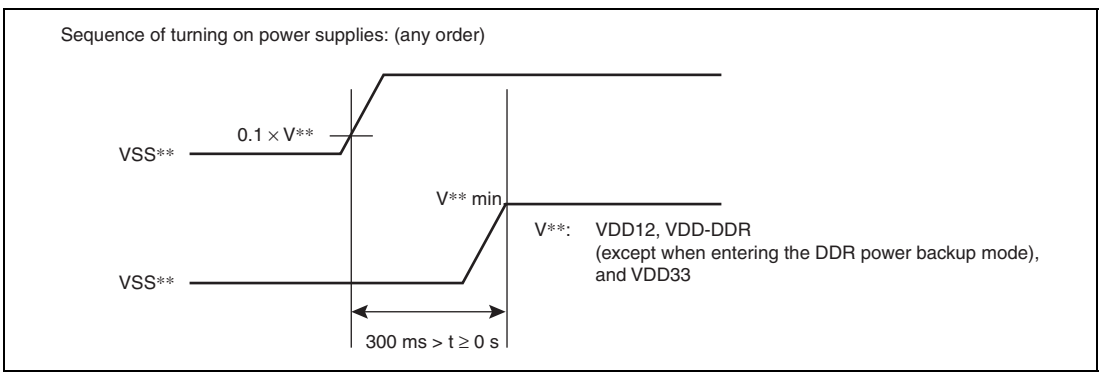


Figure 41.2 Sequence of Turning On Power Supplies with Different Voltages

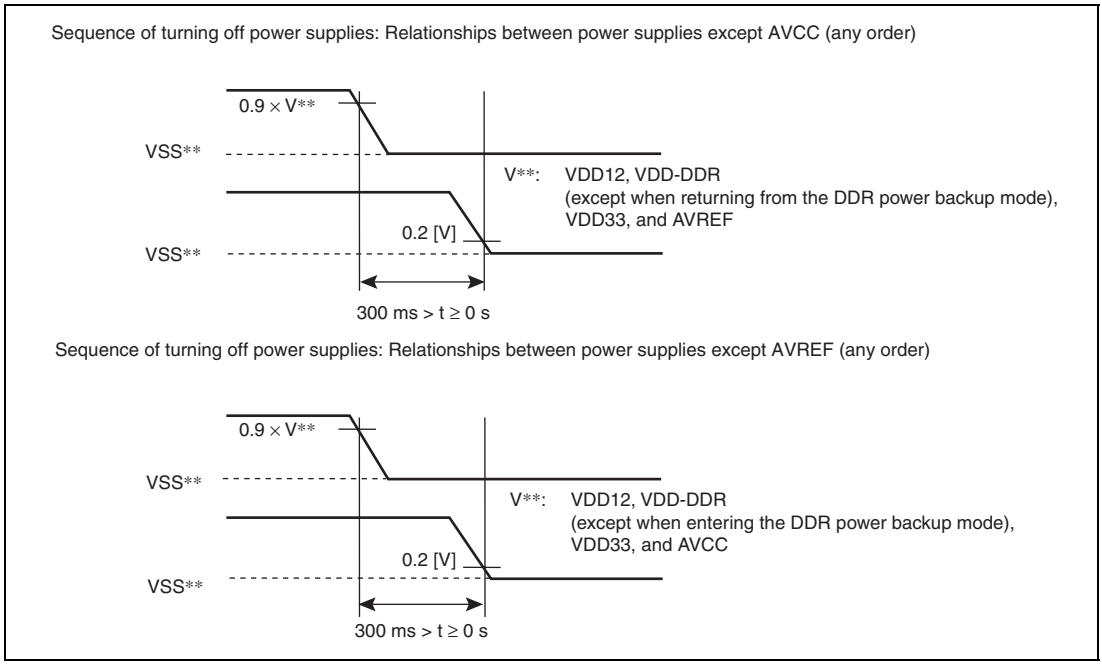


Figure 41.3 Sequence of Turning Off Power Supplies with Different Voltages

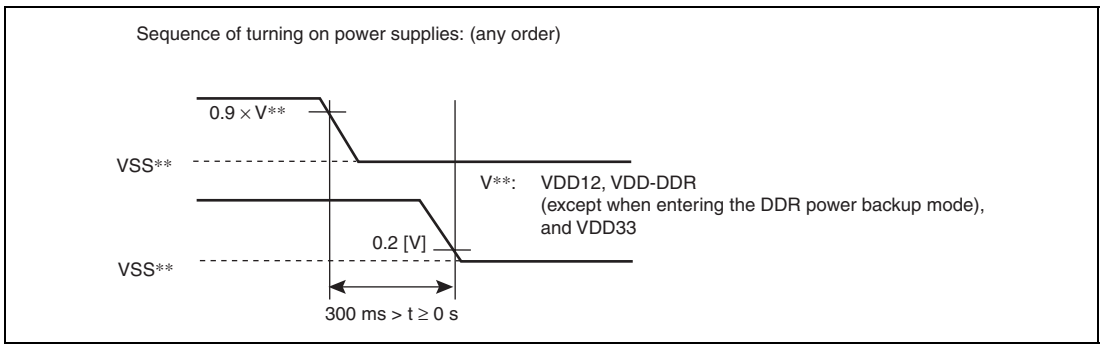


Figure 41.4 Sequence of Turning On Power Supplies with Different Voltages [except AD]

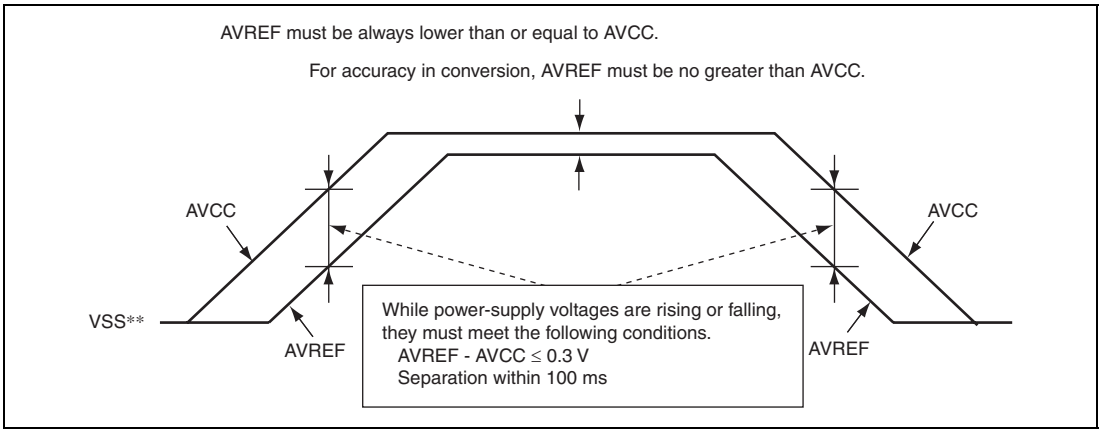


Figure 41.5 Sequences of Turning On/Off AVCC and AVREF

41.3.2 Sequence of Turning On/Off Power Supplies with Same Voltage

Power-supply voltages other than VCC are turned on and off as follows. VCC and VCCQ are separate, but should be at the same voltage on the board.

This subsection specifies the sequence of turning on/off several power supplies with the same voltage level (VDD12 power supplies, VDD-DDR power supplies, and VDD33 power supplies). The sequence of turning on/off VDD12 power supplies is shown in figure 41.2. The potential difference specified for the VDD12 supplies also applies to the VDD-DDR and VDD33 power supplies.

(1) Turning On Power Supply

There are no restrictions on the power-on sequence. Ensure that the power supplies rise from the ground (VSS**) level. Note that the potential difference between the power supplies with the same voltage must be 0.3 V or lower.

(2) Turning Off Power Supply

There are no restrictions on the power-off sequence. Note that the potential difference between the power supplies with the same voltage must be 0.3 V or lower. Ensure that the power supplies fall to the ground (VSS**) level.

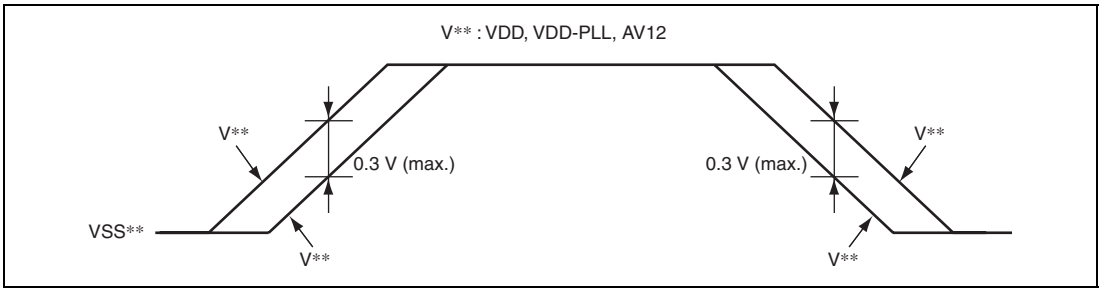


Figure 41.6 Sequence of Turning On/Off Power Supplies with Same Voltage

41.4 DC Characteristics (Common)

Table 41.3 Supply Current (1)

Ta-max is a common temperature condition (see the conditions for measuring the AC characteristics)

Item		Symbol	Min.	Typ.	Max.	Unit	Remarks
Supply current (internal)	Normal	IDD	—	0.6	1.2	A	VDD = 1.30 V, Ta-max, including the USB digital (applicable to a CPU at 400 MHz and SHwy at 200 MHz, and to a CPU at 533.3 MHz and SHwy at 177.7 MHz)
Supply current (3.3-V IO, including ADC)	Normal	ICCQ	—	200	310	mA	VCCQ = 3.6 V, Ta-max, including the USB digital
Supply current (DDR IO)	Normal	IDD-DDR	—	400	600	mA	VDD-DDR = 1.9 V, (DDR2), Ta-max (DDR2: 600 Mbps)
			—	300	400	mA	VDD-DDR = 1.9 V, (DDR2), Ta-max (DDR2: 400 Mbps)
			—	250	333	mA	VDD-DDR = 1.9 V, (DDR2), Ta-max (DDR2: 333 Mbps)
			—	400	500	mA	VDD-DDR = 1.575 V, (DDR3), Ta-max (DDR3: 600 Mbps)
	DDR power supply backup		—	—	55	mA	VDD-DDR = 1.9 V, (DDR2), Ta = 70 °C No Vtt termination
			—	—	50	mA	VDD-DDR = 1.575 V, (DDR3), Ta = 70 °C No Vtt termination
Supply current (PLL)	Normal	ICCQ-PLL	—	—	15	mA	VCCQ-PLL = 3.6 V, Ta-max
Supply current (PLL)		IDD-PLL	—	—	20	mA	VDD-PLL = 1.30 V, Ta-max
Supply current (ADC)		AICC	—	—	5	mA	AVCC = 3.6 V, Ta-max
Supply current (USB)		AI33	—	—	5	mA	AV33 = 3.6 V, Ta-max
		AI12	—	—	15	mA	AV12 = 1.30 V, Ta-max

Table 41.4 Supply Current (2)

Ta-max is a common temperature condition (see the conditions for measuring the AC characteristics)

Item		Symbol	Min.	Typ.	Max.	Unit	Remarks
Supply current (internal)	Software standby	IDD	—	—	450	mA	VDD = 1.30 V, Ta-max, with USB stopped (fixed to USB_EXTAL) (applicable to a CPU at 400 MHz and SHwy at 200 MHz, and to a CPU at 533.3 MHz and SHwy at 177.7 MHz) DDR-SDRAM is in the self-refresh state*.
	Deep standby (when the RTC, HIFRAM, and GEther are all stopped or shut down)		—	—	50		
	Deep standby (when only the RTC is operating)		—	—	51		
	Deep standby (when only the retention RAM, i.e. the HIFRAM, is not shut down but operating)		—	—	51		
	Deep standby (when only the GEther is not shut down but operating)		—	—	60		

Item		Symbol	Min.	Typ.	Max.	Unit	Remarks
Supply current (3.3-V IO, including ADC)	Software standby or deep standby	ICCQ	—	—	5	mA	VCCQ = 3.6 V, Ta-max, with USB stopped (fixed to USB_EXTAL) All modules are stopped.
Supply current (DDR IO)		IDD-DDR	—	—	55	mA	VDD-DDR = 1.9 V, Ta-max, DDR-SDRAM is in the self-refresh state*.
Supply current (PLL)		ICCQ-PLL	—	—	9	µA	VCCQ-PLL = 3.6 V, Ta-max
Supply current (PLL)		IDD-PLL	—	—	9	µA	VDD-PLL = 1.30 V, Ta-max
Supply current (USB)		AI33	—	—	4	µA	(Fixed to USB_EXTAL) AV33 = 3.6 V, Ta-max
		AI12	—	—	4	µA	(Fixed to USB_EXTAL) AV12 = 1.30 V, Ta-max

Note: * DDR-SDRAM: Transitions to self-refresh state.
(DBPDCNT3 settings: db_stby_n = 0, db_comhiz = 1, db_add2cyc_mode = 0, db_iobackup = 1, db_dllenable2 = 0, db_dllenable1 = 0, db_dllreset_n = 0, db_add_strength = 0, db_dqdm_strength = 0, db_ck_strength = 0, db_ioenable2 = 1, db_ioenable1 = 1, db_calib_start = 1)

Table 41.5 DC Characteristics (3.3-V IO)

Item		Symbol	Min.	Typ.	Max.	Unit	Measurement conditions	Remarks
Input high voltage	AN0 to AN7	VIH	2.4	—	AVCC + 0.3	V	AVCC = 3.0 V to 3.6 V, VDD = 1.15 V to 1.30 V	
	EXTAL, USB_EXTAL, RTC_X1		VCCQ × 0.8	—	VCCQ + 0.3	V	VCCQ = 3.0 V to 3.6 V, VDD = 1.15 V to 1.30 V	
	Other input pins (except Schmitt pins)		2.0	—	VCCQ + 0.3	V	VCCQ = 3.0 V to 3.6 V, VDD = 1.15 V to 1.30 V	
Input low voltage	AN0 to AN7	VIL	-0.3	—	0.8	V	AVCC = 3.0 V to 3.6 V, VDD = 1.15 V to 1.30 V	
	EXTAL, USB_EXTAL, RTC_X1		-0.3	—	VCCQ × 0.2	V	VCCQ = 3.0 V to 3.6 V, VDD = 1.15 V to 1.30 V	
	Other input pins (except Schmitt pins)		-0.3	—	0.8	V	VCCQ = 3.0 V to 3.6 V, VDD = 1.15 V to 1.30 V	
Schmitt input high voltage		VT+	—	—	2.2	V	VCCQ = 3.0 V to 3.6 V, VDD = 1.15 V to 1.30 V	$\overline{\text{PRESET}}$, BSMODE, NMI
Schmitt input low voltage		VT-	0.8	—	—	V		
Rising input slope time (recommended value)		Tr	—	—	10	ns/V		Vin = 0.7 V to 1.7 V
Falling input slope time (recommended value)		Tf	—	—	10	ns/V		Vin = 1.7 V to 0.7 V
Output high voltage		VOH	2.4	—	VCCQ + 0.3	V		IOH = -2 mA (TTL)
Output low voltage		VOL	-0.3	—	0.4	V		IOL = 2 mA (TTL)
Pin capacitance		CL	—	—	10	pF	—	All pins*2
		CL	—	—	20	pF	—	DP/DM only
		CL	—	—	20	pF	—	AN0 to AN7
Input leakage current		ILI	—	—	1	µA	VCCQ = 3.0 V to 3.6 V, VDD = 1.15 V to 1.30 V	All input pin*1
Output leakage current		ILO	—	—	1	µA		Hi-Z output *1
Pull-up current		IPU	10	—	200	µA		Vin = VSS

Notes: 1. This excludes pins with pull-up resistors, which are off.
2. Except power supply, USB, and AD conversion pins

Table 41.6 DC Characteristics (MIM Pins [MLB_CLK/MLB_SIG/MLB_DAT])

Item	Symbol	Min.	Typ.	Max.	Unit	Measurement conditions	Remarks
Input high voltage	VIH	1.8	—	VCCQ + 0.3	V	VCCQ = 3.0 V to 3.6 V, VDD = 1.15 V to 1.30 V	
Input low voltage	VIL	-0.3	—	0.7	V		
Output high voltage	VOH	2.4	—	VCCQ + 0.3	V	VCCQ = 3.0 V to 3.6 V, VDD = 1.15 V to 1.30 V	IOH = -2 mA (TTL)
Output low voltage	VOL	-0.3	—	0.4	V		IOL = 2 mA (TTL)
Pin capacitance	CL	—	—	10	pF	—	MLB_SIG, MLB_DAT, MLB_CLK
Input leakage current	ILI	—	—	1	µA	VCCQ = 3.0 V to 3.6 V, VDD = 1.15 V to 1.30 V	MLB_SIG, MLB_DAT, MLB_CLK
Output leakage current	ILO	—	—	1	µA		Hi-Z output

Table 41.7 DC Characteristics (DDR IO Pin)

Note: VREF indicates the MVREFDQ pin.

Item	Symbol	Min.	Typ.	Max.	Unit	Measurement conditions	Remarks
Input high voltage	VIH	0.8 × VDD-DDR	—	VDD-DDR + 0.3	V	VDD-DDR = 1.425 V to 1.575 V (DDR3), VDD-DDR = 1.7 V to 1.9 V (DDR2), VDD = 1.15 V to 1.30 V	$\overline{\text{MBKPRST}}$, SDBUP
Input low voltage	VIL	-0.3	—	0.2 × VDD-DDR	V		
Input high voltage	VIH (DC)	VREF + 0.125	—	VDD-DDR + 0.3	V	VDD-DDR = 1.7 V to 1.9 V (DDR2), VDD = 1.15 V to 1.30 V, VREF = 0.5 × VDD-DDR	MDQ15 to MDQ0
Input low voltage	VIL (DC)	-0.3	—	VREF – 0.125	V		
Input high voltage	VIH (AC)	VREF + 0.200	—	—	V		
Input low voltage	VIL (AC)	—	—	VREF – 0.200	V		
Input high voltage	VIH (DC)	VREF + 0.100	—	VDD-DDR + 0.3	V	VDD-DDR = 1.425 V to 1.575 V (DDR3), VDD = 1.15 V to 1.30 V, VREF = 0.5 × VDD-DDR	MDQ15 to MDQ0
Input low voltage	VIL (DC)	-0.3	—	VREF – 0.100	V		
Input high voltage	VIH (AC)	VREF + 0.175	—	—	V		
Input low voltage	VIL (AC)	—	—	VREF – 0.175	V		
Input signal slew rate	SLEW	1.0	—	—	V/ns	VDD-DDR = 1.425 V to 1.575 V (DDR3), VDD-DDR = 1.7 V to 1.9 V (DDR2), VDD = 1.15 V to 1.30 V	DDR IO other than $\overline{\text{MBKPRST}}$ and SDBUP*4 DDR IO other than $\overline{\text{MBKPRST}}$*5
AC differential input voltage*1	VID (AC)	0.500	—	VDD-DDR + 0.6	V	VDD-DDR = 1.7 V to 1.9 V (DDR2), VDD = 1.15 V to 1.30 V	MDQS1 and MDQS0
AC differential input cross point voltage*2	VIX (AC)	0.5 × VDD-DDR – 0.300	—	0.5 × VDD-DDR + 0.300	V		
AC differential input voltage*1	VID (AC)	0.400	—	VDD-DDR + 0.6	V	VDD-DDR = 1.425 V to 1.575 V (DDR3), VDD = 1.15 V to 1.30 V	MDQS1 and MDQS0
AC differential input cross point voltage*2	VIX (AC)	0.5 × VDD-DDR – 0.250	—	0.5 × VDD-DDR + 0.250	V		

Item	Symbol	Min.	Typ.	Max.	Unit	Measurement conditions	Remarks
Output source DC current	IOH (DC)	-13.36	—	-4.82	mA	VDD-DDR = 1.7 V to 1.9 V (DDR2) VDD = 1.15 V to 1.30 V	DDR output pin IOH: Vout = VDD-DDR – 0.28 V
Output sink DC current	IOL (DC)	4.82	—	13.36	mA		IOL: Vout = 0.28 V
AC differential output cross point voltage	VOX (AC)	0.5 × VDD-DDR – 0.125	—	0.5 × VDD-DDR + 0.125	V	VDD-DDR = 1.425 V to 1.575 V (DDR3), VDD-DDR = 1.7 V to 1.9 V (DDR2), VDD = 1.15 V to 1.30 V	MCK, MDQS1 and MDQS0
Differential input reference voltage*6	VREF	0.49 × VDD-DDR	0.50 × VDD-DDR	0.51 × VDD-DDR	V	VDD-DDR = 1.425 V to 1.575 V (DDR3), VDD-DDR = 1.7 V to 1.9 V (DDR2), VDD = 1.15 V to 1.30 V VREF = 0.5 × VDD-DDR	
Termination voltage	Vtt	VREF – 40	VREF	VREF + 40	mV		
ODT resistor value (75 Ω)	Rtt1	50	75	100	Ω	VDD-DDR = 1.7 V to 1.9 V (DDR2), VDD = 1.15 V to 1.30 V	
ODT resistor value (150 Ω)	Rtt2	100	150	200	Ω		
VM deviation (DDR2)*5	ΔVM	-6	—	6	%		
ODT resistor value (60 Ω)	Rtt1	54	60	96	Ω	VDD-DDR = 1.5 V, VDD = 1.25 V (DDR3)	
		46.8	60	102.6		Other than above	
VM deviation (DDR3)*5	ΔVM	-5	—	5	%	VDD-DDR = 1.425 V to 1.575 V (DDR3), VDD = 1.15 V to 1.30V (DDR3)	
Pin capacitance	CL	—	—	4	pF	VDD-DDR = 1.425 V to 1.575 V (DDR3), VDD-DDR = 1.7 V to 1.9 V (DDR2)	All pins*3
Input leakage current	ILI	—	—	7	μA		All input pin
Output leakage current	ILO	—	—	7	μA		Hi-Z output

Notes:
1. VID (AC) indicates the size of the differential input voltage, i.e. |Vtr-Vcp|. Vtr indicates the positive voltage signals (MDQS0 and MDQS1) and Vcp indicates the negative voltage signals ($\overline{\text{MDQS0}}$ and $\overline{\text{MDQS1}}$). The minimum value is VIHD (AC) - VILD (AC).
2. The VIX (AC) indicates the voltage at which differential input signals cross each other. The typical value of VIX (AC) is expected to be 0.5 × VDD-DDR.
3. Except power supply pins.
4. SLEW indicates a period from VREF to VIH (AC) on the rising edge or a period from VREF to VIL (AC) on the falling edge.

5. VM is a voltage value measured with the ODT turned on without any load applied to this LSI chip. ΔVM is obtained by the following formula:
   ΔVM = (2 × VM / VDD-DDR – 1) × 100
6. Peak to peak ac noise on VREF may not exceed ±2 % VREF.

Table 41.8 DC Characteristics (I^2C Bus Interface 3-Related Pins*)

Conditions: Common temperature and voltage conditions (see the measurement conditions)

Item	Symbol	Min.	Typ.	Max.	Unit	Measurement conditions
Input high voltage	V_{IH}	VCCQ × 0.7	—	VCCQ + 0.3	V	
Input low voltage	V_{IL}	-0.3	—	VCCQ × 0.3	V	
Schmitt trigger input characteristics	V_{IH} - V_{IL}	VCCQ × 0.05	—	—	V	
Output low voltage	V_{OL}	—	—	0.4	V	I_{OL} = 3.0 mA

Note: * The SCLn and SDAn (n = 0 or 1) pins are open-drain pins.

Table 41.9 DC Characteristics [USB-Related Pins (1)]

Conditions: Common temperature and voltage conditions

Item	Symbol	Min.	Typ.	Max.	Unit	Measurement conditions
Reference resistor value	R_{REF}	5.6 kΩ ±1 %			V	
Input high voltage (VBUS)	V_{IH}	4.02	—	5.25	V	
Input low voltage (VBUS)	V_{IL}	0.0	—	1.0	V	
Input high voltage (XIN)	V_{IH}	V_{DDQ} - 0.5	—	V_{DDQ} + 0.3		
Input low voltage (XIN)	V_{IL}	-0.3	—	0.5	V	

Table 41.10 DC Characteristics [USB-Related Pins (2) Full-Speed and High-Speed Common Items]

Conditions: Common temperature and voltage conditions

Item	Symbol	Min.	Typ.	Max.	Unit	Measurement conditions
DP pull-up resistance (when function is selected)	R_{pu}	0.900	—	1.575	kΩ	Idle
		1.425	—	3.090	kΩ	Transmission/reception
DP and DM pull-down resistance (when host is selected)	R_{pd}	14.25	—	24.80	kΩ	

Table 41.11 DC Characteristics [USB-Related Pins (3) Full Speed]

Conditions: Common temperature and voltage conditions

Item	Symbol	Min.	Typ.	Max.	Unit	Measurement conditions
Input high voltage	V_{IH}	2.0	—	—	V	
Input low voltage	V_{IL}	—	—	0.8	V	
Differential input sensitivity	V_{DI}	0.2	—	—	V	|(DP) – (DM)|
Differential common mode range	V_{CM}	0.8	—	2.5	V	
Output high voltage	V_{OH}	2.8	—	3.6	V	I_{OH} = 5 mA
Output low voltage	V_{OL}	0.0	—	0.3	V	I_{OL} = 5 mA
Single-ended receiver threshold voltage	V_{SE}	0.8	—	2.0	V	
Cross over voltage	V_{CRS}	1.3	—	2.0	V	C_L = 50 pF

Note: The DP and DM pins are USB-related pins.

Table 41.12 DC Characteristics [USB-Related Pins (4) High Speed]

Conditions: Common temperature and voltage conditions

Item	Symbol	Min.	Typ.	Max.	Unit	Measurement conditions
Differential input sensitivity	V_{HSDI}	0.15	—	—	V	
Squelch detection threshold voltage (differential voltage)	V_{HSSQ}	100	—	150	mV	
Common mode voltage range	V_{HSCM}	-50	—	500	mV	
Idle state	V_{HSOI}	-10.0	—	10.0	mV	
Output high voltage	V_{HSOH}	360	—	440	mV	
Output low voltage	V_{HSOL}	-10.0	—	10.0	mV	
Chirp J output voltage (difference)	V_{CHIRPJ}	700	—	1000	mV	
Chirp K output voltage (difference)	V_{CHIRPK}	-900	—	-500	mV	

Note: The DP and DM pins are USB-related pins.

Table 41.13 DC Characteristics [USB-Related Pins (5) Low Speed]

Conditions: Common temperature and voltage conditions

Item	Symbol	Min.	Typ.	Max.	Unit	Measurement conditions
Output high voltage	V_{LSOH}	2.8	—	—	V	I_{OH} = 200 µA
Output low voltage	V_{LSOL}	—	—	0.3	V	I_{OL} = 2 mA

Note: The DP and DM pins are USB-related pins.

Table 41.14 Permissible Output Current of Pins Driven by VccQ

Conditions: Common temperature and voltage conditions

Item	Symbol	Min.	Typ.	Max.	Unit
Permissible output low current (6 mA in buffer)	I_{OL}	—	6	22	mA
Permissible output low current (8 mA in buffer)	I_{OL}	—	8	30	
Permissible output low current (total, reference value)	ΣI_{OL}	—	—	120	
Permissible output high current (6 mA in buffer)	$-I_{OH}$	—	6	22	mA
Permissible output high current (8 mA in buffer)	$-I_{OH}$	—	8	30	
Permissible output high current (total, reference value)	$\Sigma \lvert -I_{OH} \rvert$	—	—	120	

Note: To protect the LSI's reliability, do not exceed the output current values per pin in this table. Targets for the totals are averages over time. Exceeding the values over the long term is not directly connected with damage to the LSI chip.

41.5 Reset and Watchdog Timer (RESET and WDT)

Table 41.15 Clock and Reset Timings

Conditions: Common temperature and voltage conditions (see the measurement conditions)

Pin	Item	Symbol	Min.	Max.	Unit	Figures
$\overline{\text{PRESET}}$, EXTAL	Power-on oscillation settling time	tOSC	20	—	ms	Figure 41.7
Mode signal*	MD reset setup time	tMDRS	20	—	ms	
Mode signal*	MD reset hold time	tMDRH	0	—	ns	
$\overline{\text{TRST}}$	$\overline{\text{TRST}}$ reset hold time	tTRSTRH	20	—	ns	
VCCQ	Real time clock oscillation settling time	tROSC	3	—	s	Figure 41.8

Note: * Mode signals are defined as follows: MD (0 to 19) and MPMD
Rising and falling edges of the reset signal (time for the input level to make transitions between ViH and ViL) must take no more than 20 ns.

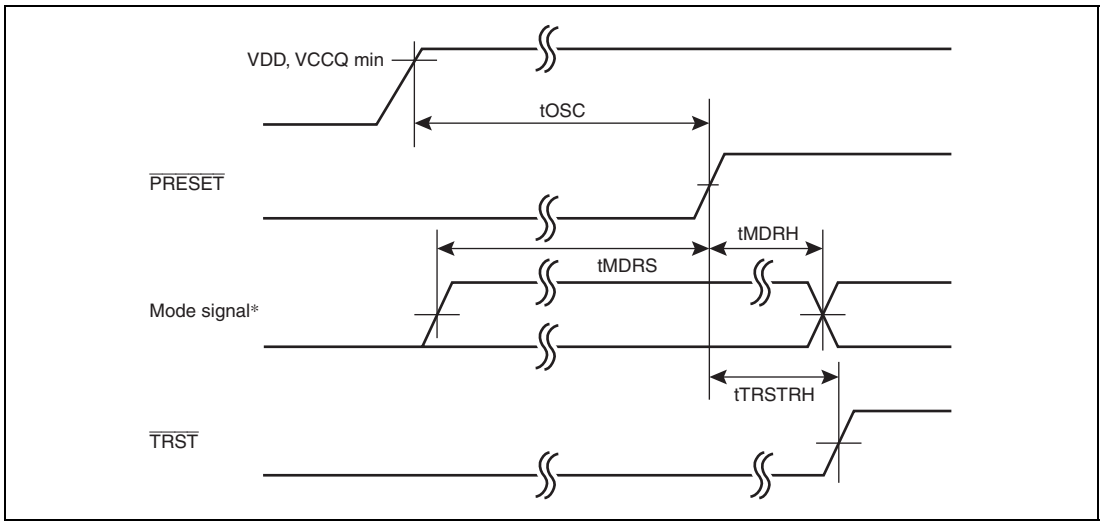


Figure 41.7 Reset when Turning on Power Supply

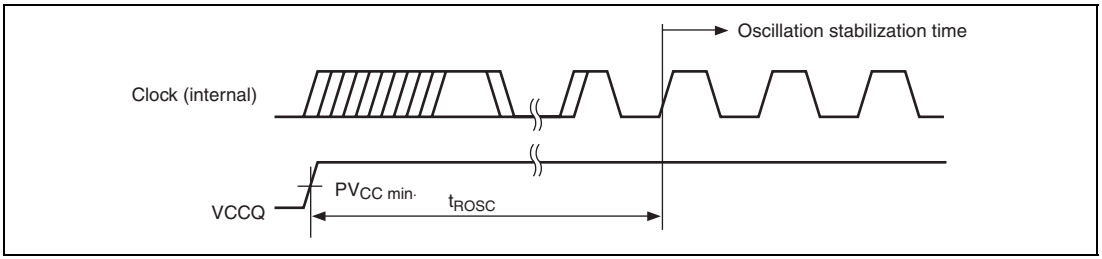


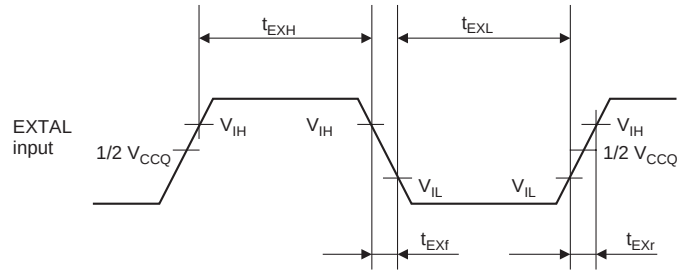
Figure 41.8 Real-time Clock Oscillation Stabilization Time

41.6 Clock Timing

Table 41.16 Clock Timing

Conditions: Common temperature and voltage conditions

Item		Symbol	Min.	Max.	Unit	Figures
EXTAL clock input frequency (400-MHz mode (1))	PLL1 Multiplication ratio × 12	f_{EX}	50.00	51.00	MHz	
	PLL1 Multiplication ratio × 16		37.50	38.25		
	PLL1 Multiplication ratio × 24		25.00	25.50		
	PLL1 Multiplication ratio × 32		18.75	19.12		
EXTAL clock input frequency (400-MHz mode (2))	PLL1 Multiplication ratio × 12		27.77	33.33		
	PLL1 Multiplication ratio × 16		20.83	25.00		
	PLL1 Multiplication ratio × 24		13.88	16.66		
	PLL1 Multiplication ratio × 32		10.41	12.50		
EXTAL clock input frequency (533-MHz mode)	PLL1 Multiplication ratio × 12		41.66	44.44		
	PLL1 Multiplication ratio × 16		31.25	33.33		
	PLL1 Multiplication ratio × 24		20.83	22.22		
	PLL1 Multiplication ratio × 32		15.62	16.66		
EXTAL clock input low pulse width		t_{EXL}	3.84	—	ns	Figure 41.9
EXTAL clock input high pulse width		t_{EXH}	3.84	—	ns	Figure 41.9
EXTAL clock input rise time		t_{EXr}	—	3	ns	Figure 41.9
EXTAL clock input fall time		t_{EXf}	—	3	ns	Figure 41.9
CLKOUT clock output		t_{OP}	—	51	MHz	
CLKOUT clock output cycle time		$t_{CLKOUTcyc}$	19.61	—	ns	Figure 41.10
CLKOUT clock output low pulse width		$t_{CLKOUTL}$	5	—	ns	Figure 41.10
CLKOUT clock output high pulse width		$t_{CLKOUTH}$	5	—	ns	Figure 41.10
CLKOUT clock output rise time		$t_{CLKOUTr}$	—	3	ns	Figure 41.10
CLKOUT clock output fall time		$t_{CLKOUTf}$	—	3	ns	Figure 41.10



Note: This applies when a clock signal is being input on the EXTAL pin.

Figure 41.9 EXTAL Clock Input Timing

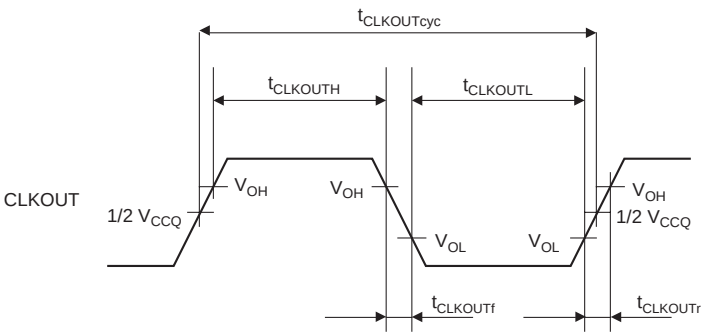


Figure 41.10 CLKOUT Clock Output Timing

41.7 Memory Controller (DBSC3)

Table 41.17 DDR2-SDRAM Access Timing

Note: VREF means the MVREFDQ.

Conditions: VDD-DDR = 1.8 V ± 0.1 V, GND = Vss = 0 V, measurements are under a common temperature condition (see the conditions for measuring the AC characteristics), see table 41.19.

Item	Symbol	Min.	Max.	Unit	Figures*1	Remarks
CK cycle	tCK (avg.)	5.0	6.0	ns	Figure 41.12	DDR2-400
		3.75	5.0			DDR2-533
		3.27	3.75			DDR2-667*2
CK high time	tCH (avg.)	0.48	0.52	tCK (avg.)	Figure 41.12	—
CK low time	tCL (avg.)	0.48	0.52	tCK (avg.)	Figure 41.12	—
Control signal setup time to CK	tIS	1.29	—	ns	Figure 41.13	DDR2-400
		0.98	—			DDR2-533
		0.88	—			DDR2-667*2
Control signal hold time to CK	tIH	1.29	—	ns	Figure 41.13	DDR2-400
		0.98	—			DDR2-533
		0.88	—			DDR2-667*2
Control/address signal width	tIPW	0.7	—	tCK (avg.)	Figure 41.13	—
Skew between CK and DQS (read)	tRDQSCK	-0.2	1.4	ns	Figure 41.14	—
DQS high time (read)	tRDQSH	0.35	0.65	tCK (avg.)	Figure 41.15	—
DQS low time (read)	tRDQSL	0.35	0.65	tCK (avg.)	Figure 41.15	—
DQS preamble time (read)	tRPRE	0.9	1.1	tCK (avg.)	Figure 41.15	—
DQS postamble time (read)	tRPST	0.4	0.6	tCK (avg.)	Figure 41.15	—
Skew between DQS and DQ (read)	tRDQSQ	—	0.59	ns	Figure 41.16	DDR2-400
		—	0.45			DDR2-533
		—	0.39			DDR2-667*2
DQ hold time to DQS (read)	tRQH	0.48tCK - 0.63	—	ns	Figure 41.16	DDR2-400
		0.48tCK - 0.53	—			DDR2-533
		0.48tCK - 0.47	—			DDR2-667*2
DQS latching rising transitions to associated clock edges (write)	tWDQSS	-0.15	0.15	tCK (avg.)	Figure 41.17	—

Item	Symbol	Min.	Max.	Unit	Figures*1	Remarks
DQS falling edge setup time to CK (write)	tWDSS	0.27	—	tCK (avg.)	Figure 41.17	—
DQS falling edge hold time to CK (write)	tWDSH	0.27	—	tCK (avg.)	Figure 41.17	—
DQS high time (write)	tWDQSH	0.35	0.9	tCK (avg.)	Figure 41.18	—
DQS low time (write)	tWDQSL	0.35	0.9	tCK (avg.)	Figure 41.18	—
DQS preamble time (write)	tWPRE	0.35	—	tCK (avg.)	Figure 41.18	—
DQS postamble time (write)	tWPST	0.4	0.6	tCK (avg.)	Figure 41.18	—
DQ/DM setup time to DQS (write)	tWDS	0.63	—	ns	Figure 41.19	DDR2-400
		0.48	—			DDR2-533
		0.43	—			DDR2-667*2
DQ/DM hold time to DQS (write)	tWDH	0.63	—	ns	Figure 41.19	DDR2-400
		0.48	—			DDR2-533
		0.43	—			DDR2-667*2
DQ/DM signal width (write)	tWDIPW	0.35	—	tCK (avg.)	Figure 41.19	—
DQ transition time to Hi-Z (write)	tHZ	tWDH	tCK	ns	Figure 41.20	—

Notes: 1. The signal timing is based on the following electric potential:
For MCK output, MDQS input/output: Cross point of complementary signals
For MDQ input: VREF
For outputs other than MCK or MDQS: 0.5 × VDD-DDR
2. Although the remarks indicate DDR2-667 memory, this cannot be used at frequencies above 612 MHz.

Table 41.18 DDR3-SDRAM Access Timing

Note: VREF means the MVREFDQ.

Conditions: VDD-DDR = 1.5 V ± 0.075 V, GND = VSS = 0 V, measurements are under a common temperature condition (see the conditions for measuring the AC characteristics), see table 41.19.

Item	Symbol	Min.	Max.	Unit	Figures*1	Remarks
CK cycle	tCK (avg.)	3.27	5.0	ns	Figure 41.12	DDR3-800*2
CK high time	tCH (avg.)	0.47	0.53	tCK (avg.)	Figure 41.12	DDR3-800*2
CK low time	tCL (avg.)	0.47	0.53	tCK (avg.)	Figure 41.12	DDR3-800*2
Control signal setup time to CK	tIS	0.52	—	ns	Figure 41.13	DDR3-800*2
Control signal hold time to CK	tIH	0.52	—	ns	Figure 41.13	DDR3-800*2
Control/address signal width	tIPW	—	—	tCK (avg.)	Figure 41.13	DDR3-800*2
Skew between CK and DQS (read)	tRDQSCK	-0.3	1.3	ns	Figure 41.14	DDR3-800*2

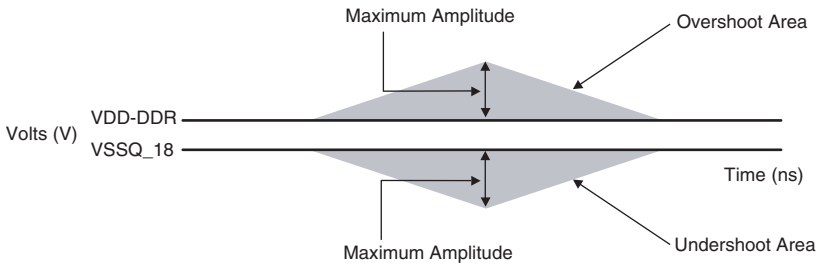
Item	Symbol	Min.	Max.	Unit	Figures*1	Remarks
DQS high time (read)	tRDQSH	0.38	—	tCK (avg.)	Figure 41.15	DDR3-800*2
DQS low time (read)	tRDQSL	0.38	—	tCK (avg.)	Figure 41.15	DDR3-800*2
DQS preamble time (read)	tRPRE	0.9	—	tCK (avg.)	Figure 41.15	DDR3-800*2
DQS postamble time (read)	tRPST	0.3	—	tCK (avg.)	Figure 41.15	DDR3-800*2
Skew between DQS and DQ (read)	tRDQSQ	—	0.22	ns	Figure 41.16	DDR3-800*2
DQ hold time to DQS (read)	tRQH	0.32	—	tCK (avg.)	Figure 41.16	DDR3-800*2
DQS latching rising transitions to associated clock edges (write)	tWDQSS	-0.18	0.18	tCK (avg.)	Figure 41.17	DDR3-800*2
DQS falling edge setup time to CK (write)	tWDSS	0.27	—	tCK (avg.)	Figure 41.17	DDR3-800*2
DQS falling edge hold time to CK (write)	tWDSH	0.27	—	tCK (avg.)	Figure 41.17	DDR3-800*2
DQS high time (write)	tWDQSH	0.45	0.55	tCK (avg.)	Figure 41.18	DDR3-800*2
DQS low time (write)	tWDQSL	0.45	0.55	tCK (avg.)	Figure 41.18	DDR3-800*2
DQS preamble time (write)	tWPRE	0.9	—	tCK (avg.)	Figure 41.18	DDR3-800*2
DQS postamble time (write)	tWPST	0.3	—	tCK (avg.)	Figure 41.18	DDR3-800*2
DQ/DM setup time to DQS (write)	tWDS	0.28	—	ns	Figure 41.19	DDR3-800*2
DQ/DM hold time to DQS (write)	tWDH	0.28	—	ns	Figure 41.19	DDR3-800*2
DQ/DM signal width (write)	tWDIPW	0.35	—	tCK (avg.)	Figure 41.19	DDR3-800*2
DQ transition time to Hi-Z (write)	tHZ	tWDH	tCK	ns	Figure 41.20	DDR3-800*2

Notes: 1. The signal timing is based on the following electric potential:
For MCK output, MDQS input/output: Cross point of complementary signals
For MDQ input: VREF
For outputs other than MCK or MDQS: 0.5 × VDD-DDR

2. The memory which is written as DDR3-800 in remarks cannot be used at over 612 MHz.

Table 41.19 Input Overshoot/Undershoot Conditions

	Value	Unit
Maximum peak amplitude allowed for overshoot area. (See the figure below.)	0.3	V
Maximum peak amplitude allowed for undershoot area. (See the figure below.)	0.3	V
Maximum overshoot area above VDD-DDR (See the figure below.)	0.19	V-ns
Maximum undershoot area below VSSQ_18 (See the figure below.)	0.19	V-ns



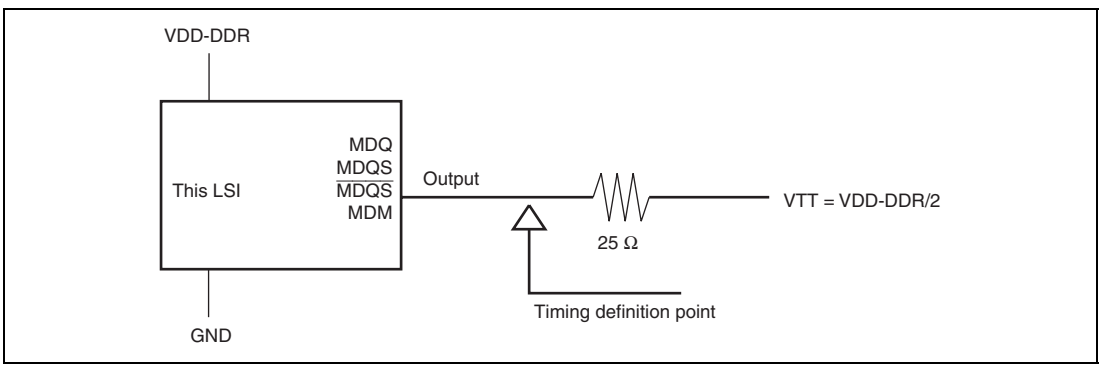


Figure 41.11 Output Clock AC Timing Measurement Conditions

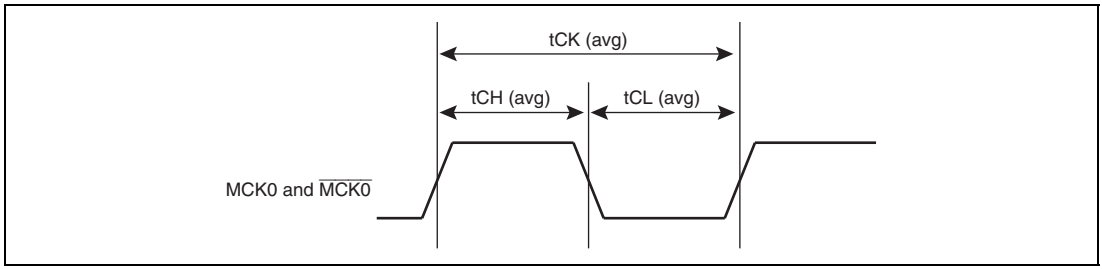


Figure 41.12 Output Clock

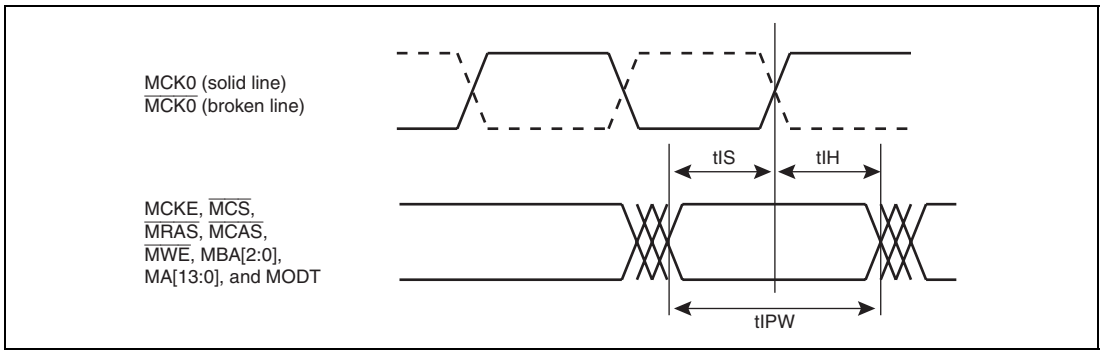


Figure 41.13 Command Pins in Relation to Output Clock

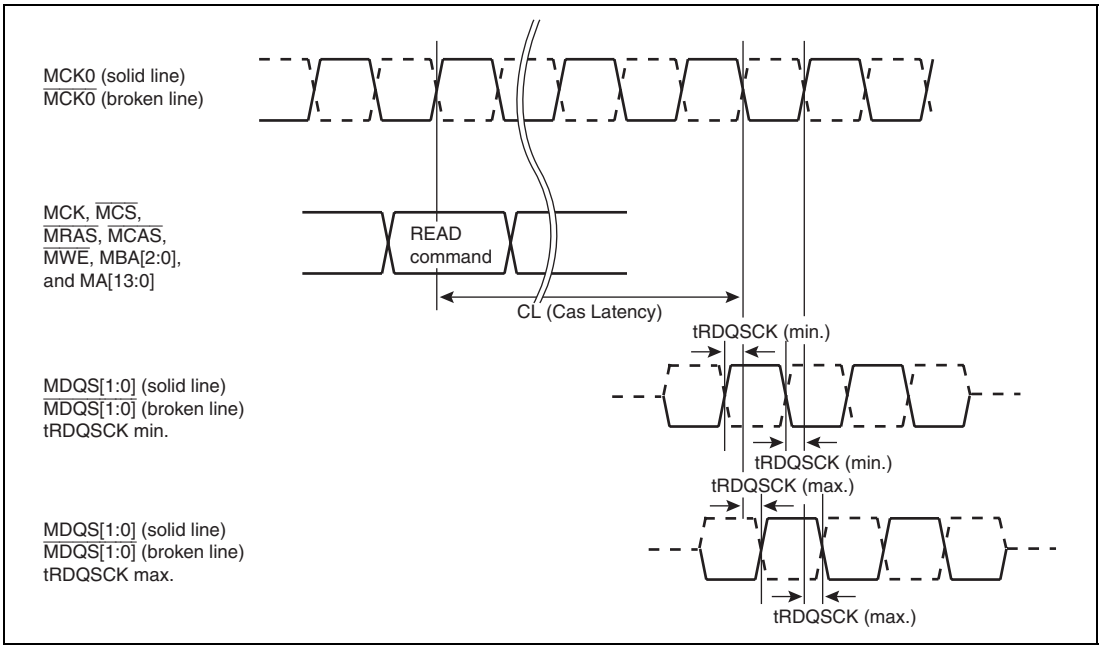


Figure 41.14 DQS Input for Data Read

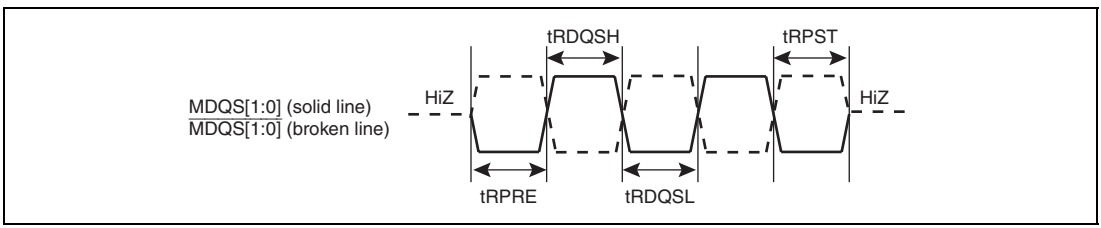


Figure 41.15 Constraints on DQS Input Waveform (for Read)

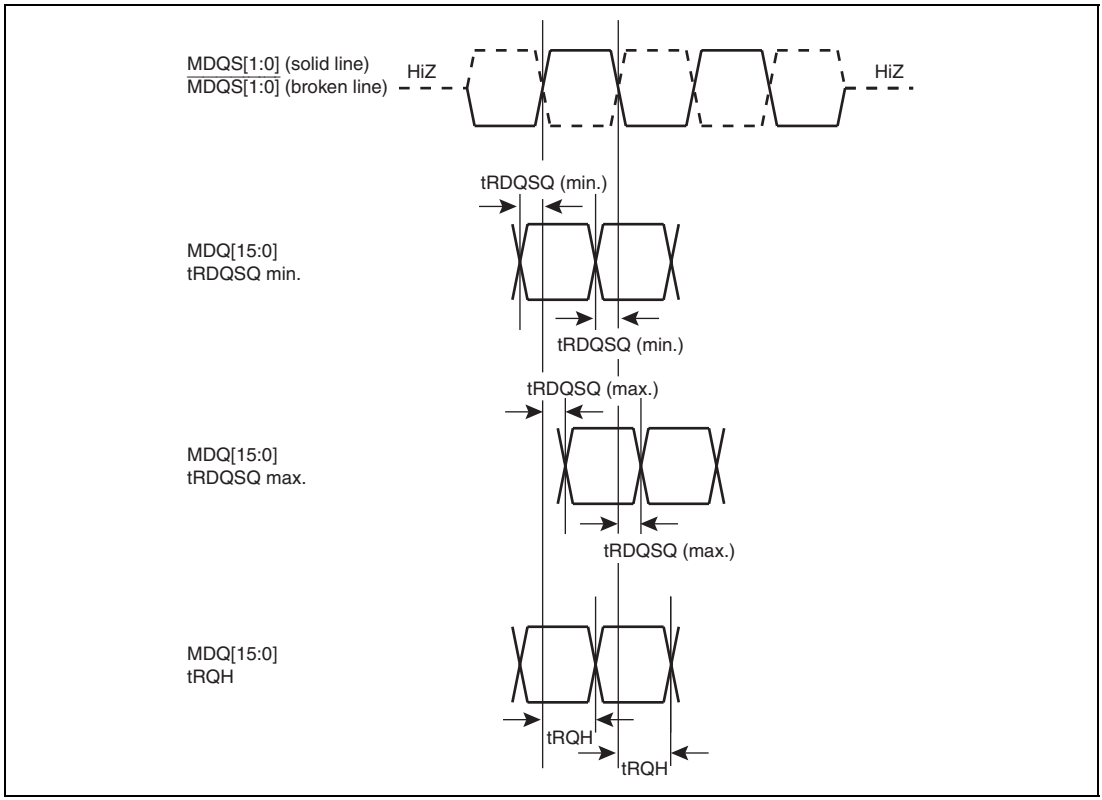


Figure 41.16 Constraints on DQ Input Waveform in Relation to DQS (for Read)

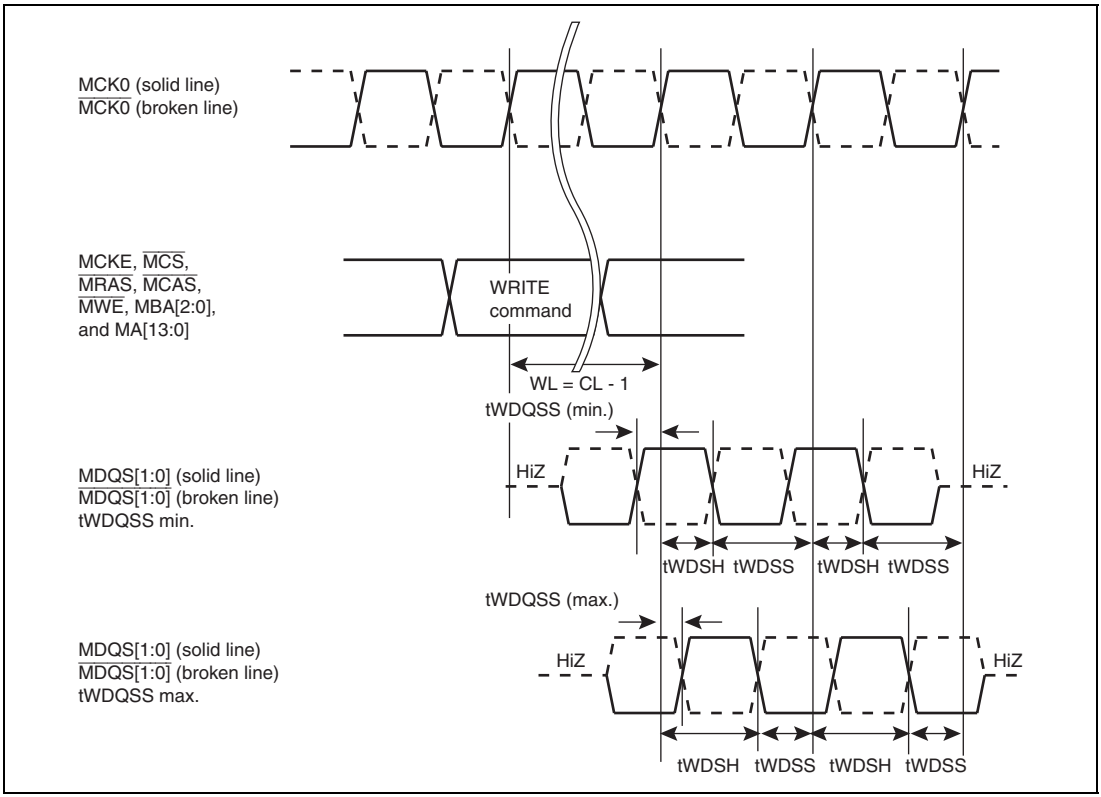


Figure 41.17 DQS Output Waveform in Relation to CK (for Write)

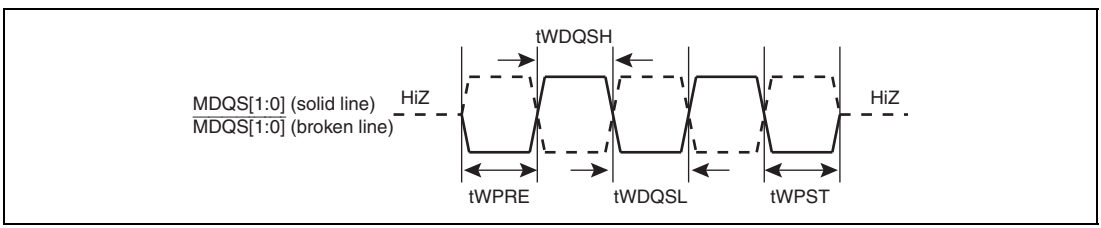


Figure 41.18 DQS Output Waveform (for Write)

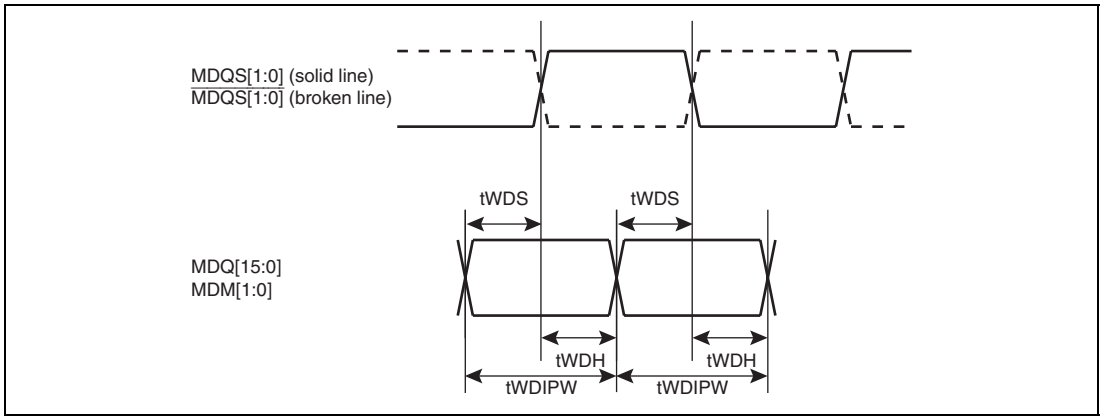


Figure 41.19 DQS, DQ, and DQM Output Waveform (for Write)

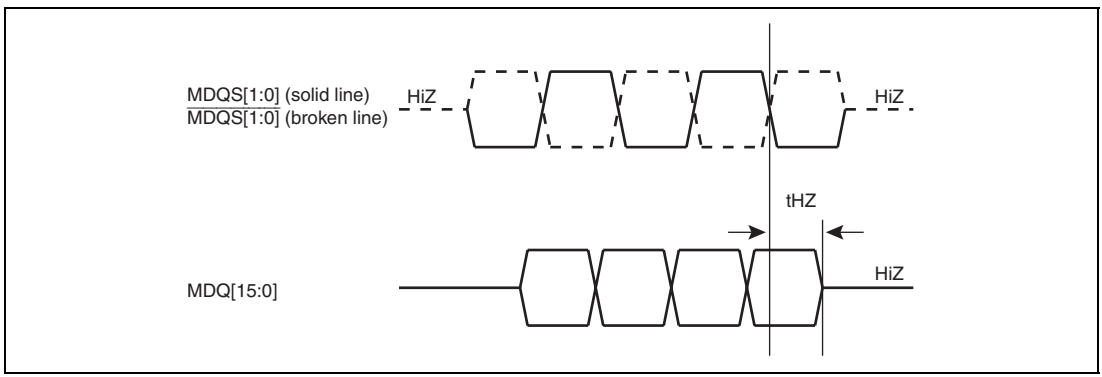


Figure 41.20 DQ Output Transition Time to Hi-Z State (for Write)

41.8 Local Bus State Controller (LBSC)

Table 41.20 Normal Read/Write Access Timing

Conditions: Common temperature and voltage conditions (see the measurement conditions), CL = 40 pF

Item	Symbol	Min.	Typ.	Max.	Unit	Figures
Address output delay time	tDA	0.0	—	6.0	ns	Figure 41.21
CS output delay time	tDCS	0.0	—	6.0	ns	
BS output delay time	tDBS	0.0	—	6.0	ns	
RD output delay time	tDRD	0.0	—	6.0	ns	
RDWR output delay time	tDRW	0.0	—	6.0	ns	
Read data setup time	tSD	11.0	—	—	ns	
Read data hold time	tHD	0.0	—	—	ns	
WE output delay time	tDWE	0.0	—	6.0	ns	
Write data output delay time	tDD	0.0	—	6.0	ns	
External wait signal setup time	tSEW	11.0	—	—	ns	
External wait signal hold time	tHEW	0.0	—	—	ns	
ATADIR output delay time	tDATAD	0.0	—	6.0	ns	
ATAG output delay time	tDATAG	0.0	—	6.0	ns	
DIOR output delay time	tDDIOR	0.0	—	6.0	ns	
DIOW output delay time	tDDIOW	0.0	—	6.0	ns	

Table 41.21 Burst ROM Read Access timing

Conditions: Common temperature and voltage conditions, CL = 40 pF

Item	Symbol	Min.	Typ.	Max.	Unit	Figures
Address output delay time	tDABST	0.0	—	6.0	ns	Figure 41.22
CS output delay time	tDCSBST	0.0	—	6.0	ns	
RD output delay time	tDRDBST	0.0	—	6.0	ns	
Read data setup time	tSDBST	11.0	—	—	ns	
Read data hold time	tHDBST	0.0	—	—	ns	

Table 41.22 DMA Signal Access Timing

Conditions: Common temperature and voltage conditions, CL = 40 pF

Item	Symbol	Min.	Typ.	Max.	Unit	Figures
DMA transfer request signal setup time	tSDRQ	11.0	—	—	ns	Figure 41.23
DMA transfer request signal hold time	tHDRQ	0.0	—	—	ns	
Delay time for DMA transfer end acknowledge signal output	tDDAK	0.0	—	6.0	ns	
Delay time for DMA acceptance signal output	tDDRQ	0.0	—	6.0	ns	

Table 41.23 Timing of Ultra ATA Transfer for ATA-IF

Conditions: Common temperature and voltage conditions, CL = 40 pF

		Mode 0		Mode 1		Mode 2		Mode 3		Mode 4			
Item	Symbol	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Unit	Figures
Typical average two-cycle time	t2CYCTYP	240	—	160	—	120	—	90	—	60	—	ns	Figures 41.24 to 41.33
Cycle time	tCYC	112	—	73	—	54	—	39	—	25	—	ns	
Minimum two-cycle time	t2CYC	230	—	153	—	115	—	86	—	57	—	ns	
Data setup time (reception)	tDS	15	—	10	—	7	—	7	—	5	—	ns	
Data hold time (reception)	tDH	5	—	5	—	5	—	5	—	5	—	ns	
Data setup time (transmission)	tDVS	70	—	48	—	31	—	20	—	6.7	—	ns	
Data hold time (transmission)	tDVH	6.2	—	6.2	—	6.2	—	6.2	—	6.2	—	ns	
CRC data setup time (transmission)	tCVS	70	—	48	—	31	—	20	—	6.7	—	ns	
CRC data hold time (transmission)	tCVH	6.2	—	6.2	—	6.2	—	6.2	—	6.2	—	ns	
First DSTROBE time (transmission)	tZFS	0	—	0	—	0	—	0	—	0	—	ns	
Data enabled to the first DSTROBE edge time (transmission)	tDZFS	70	—	48	—	31	—	20	—	6.7	—	ns	
First STROBE time	tFS	—	230	—	200	—	170	—	130	—	120	ns	

Item	Symbol	Mode 0		Mode 1		Mode 2		Mode 3		Mode 4		Unit	Figures
		Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.		
Limited interlock time	tLI	0	150	0	150	0	150	0	100	0	100	ns	Figures 41.24 to 41.33
Minimum interlock time	tMLI	20	—	20	—	20	—	20	—	20	—	ns	
Unlimited interlock time	tUI	0	—	0	—	0	—	0	—	0	—	ns	
Output release time	tAZ	—	10	—	10	—	10	—	10	—	10	ns	
Output delay time	tZAH	20	—	20	—	20	—	20	—	20	—	ns	
Minimum output assert/negate time (from release timing)	tZAD	0	—	0	—	0	—	0	—	0	—	ns	
Envelope time	tENV	20	70	20	70	20	70	20	55	20	55	ns	
Final STROBE time	tRFS	—	75	—	70	—	60	—	60	—	60	ns	
STOP assertion or DMARQ negation time	tRP	160	—	125	—	100	—	100	—	100	—	ns	
IORDY release time	tIORDYZ	—	20	—	20	—	20	—	20	—	20	ns	
STROBE driven time	tZIORDY	0	—	0	—	0	—	0	—	0	—	ns	
$\overline{\text{DMACK}}$ setup/hold time	tACK	20	—	20	—	20	—	20	—	20	—	ns	
STROBE STOP time	tSS	50	—	50	—	50	—	50	—	50	—	ns	

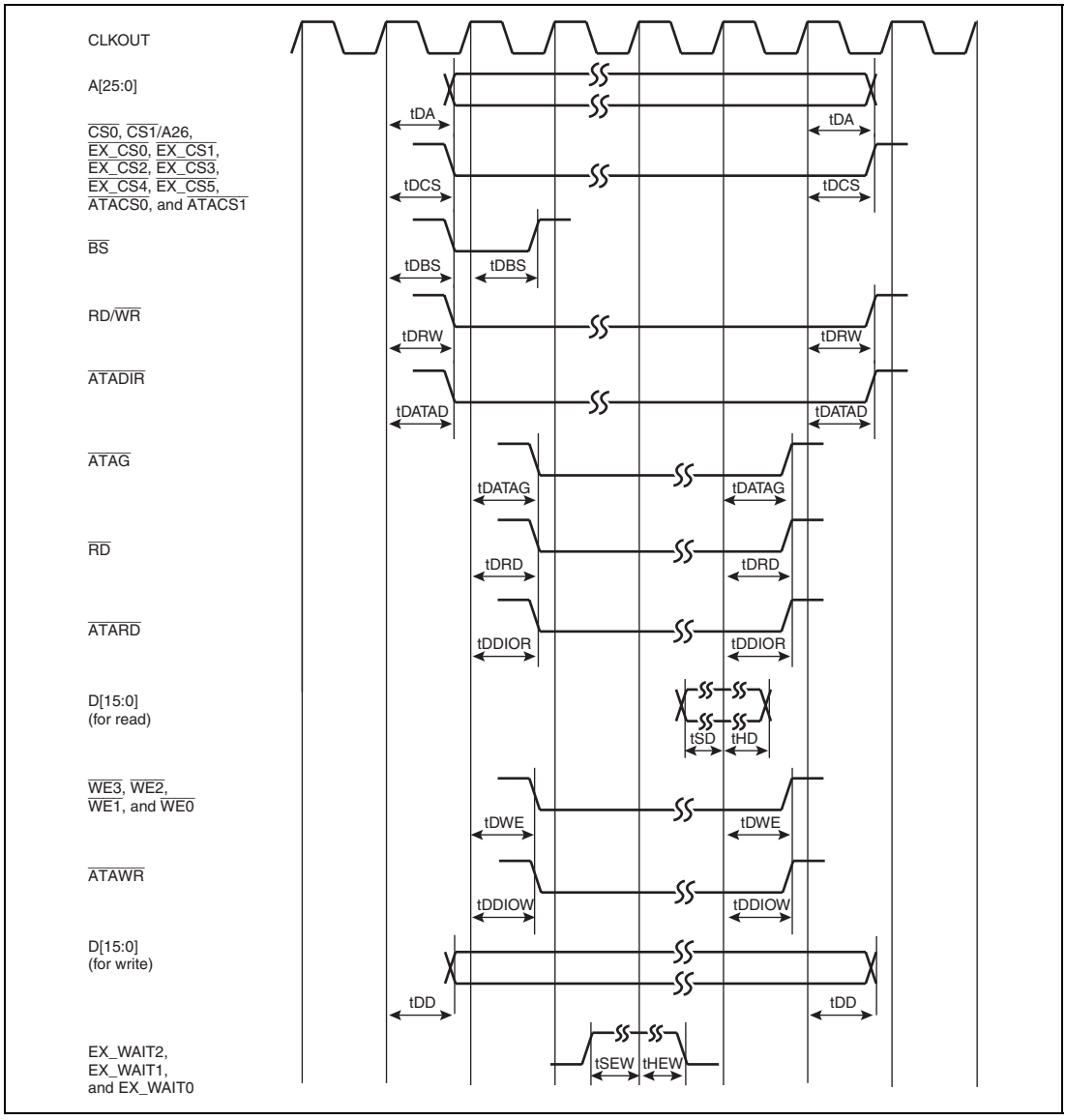


Figure 41.21 Normal Read/Write Access Timing

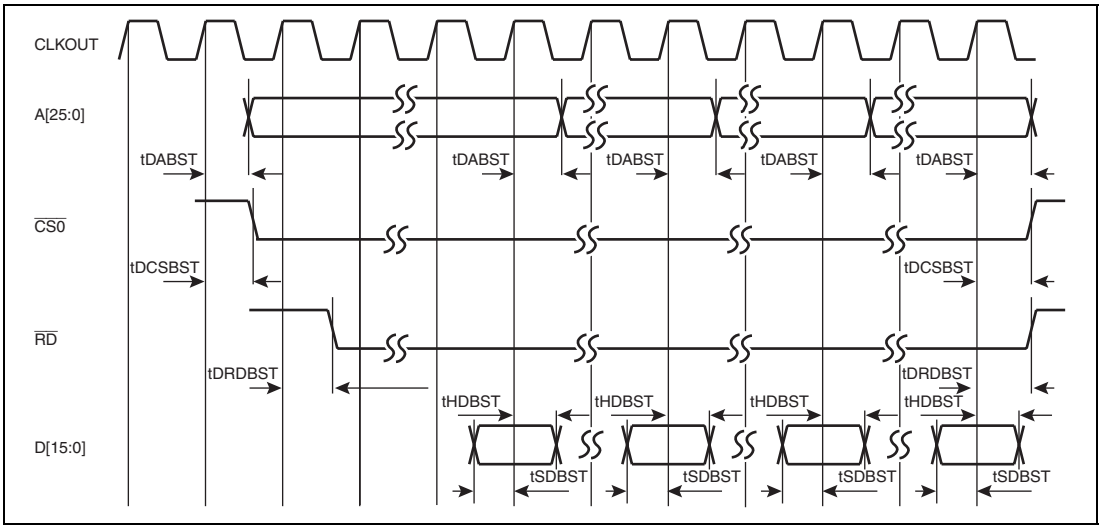


Figure 41.22 Burst ROM Read Access Timing

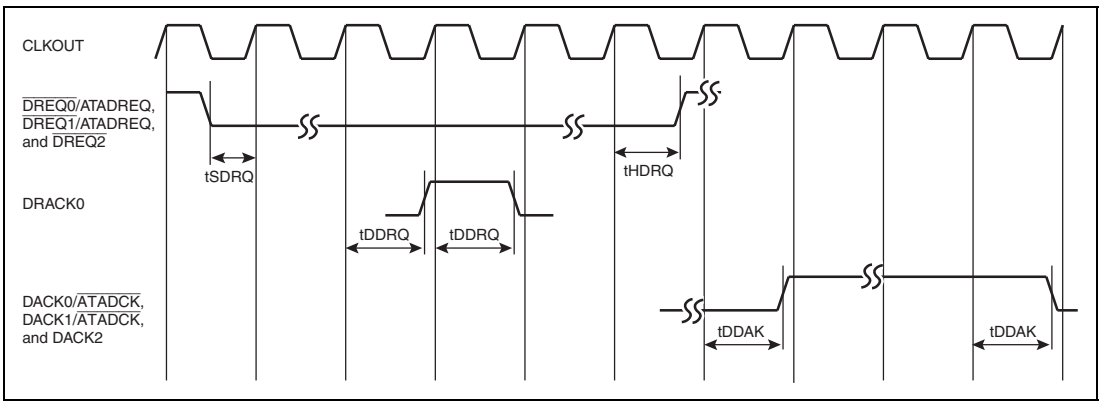


Figure 41.23 DMA Signal Access Timing

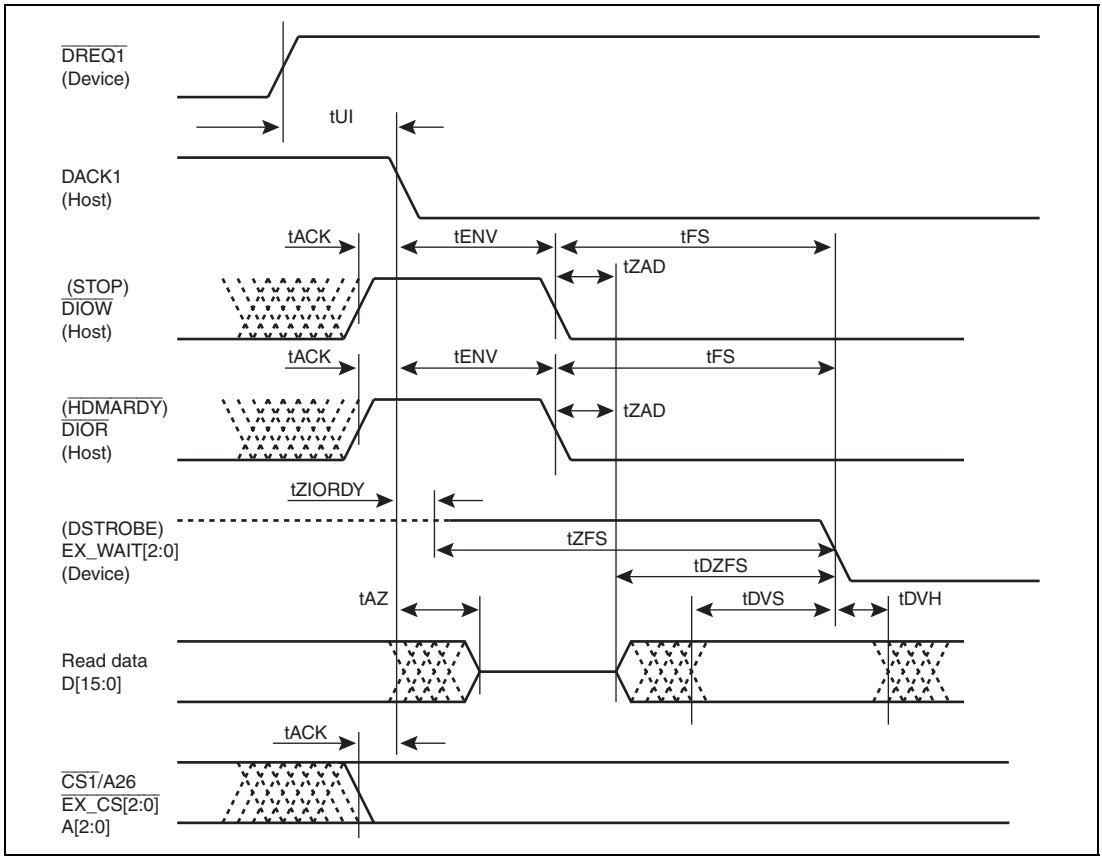


Figure 41.24 Initiating UltraDMA Transfer (for Read)

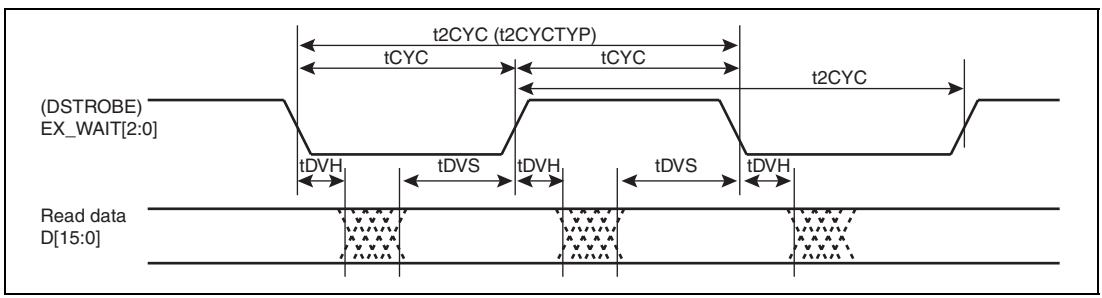


Figure 41.25 UltraDMA Transfer (for Read)

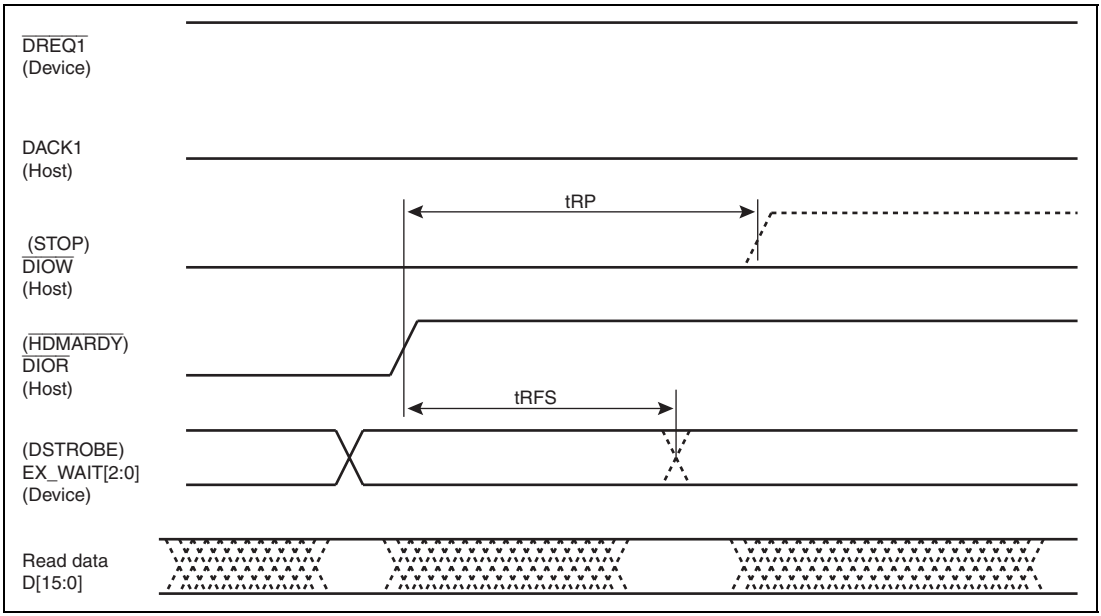


Figure 41.26 Host Pausing UltraDMA (for Read)

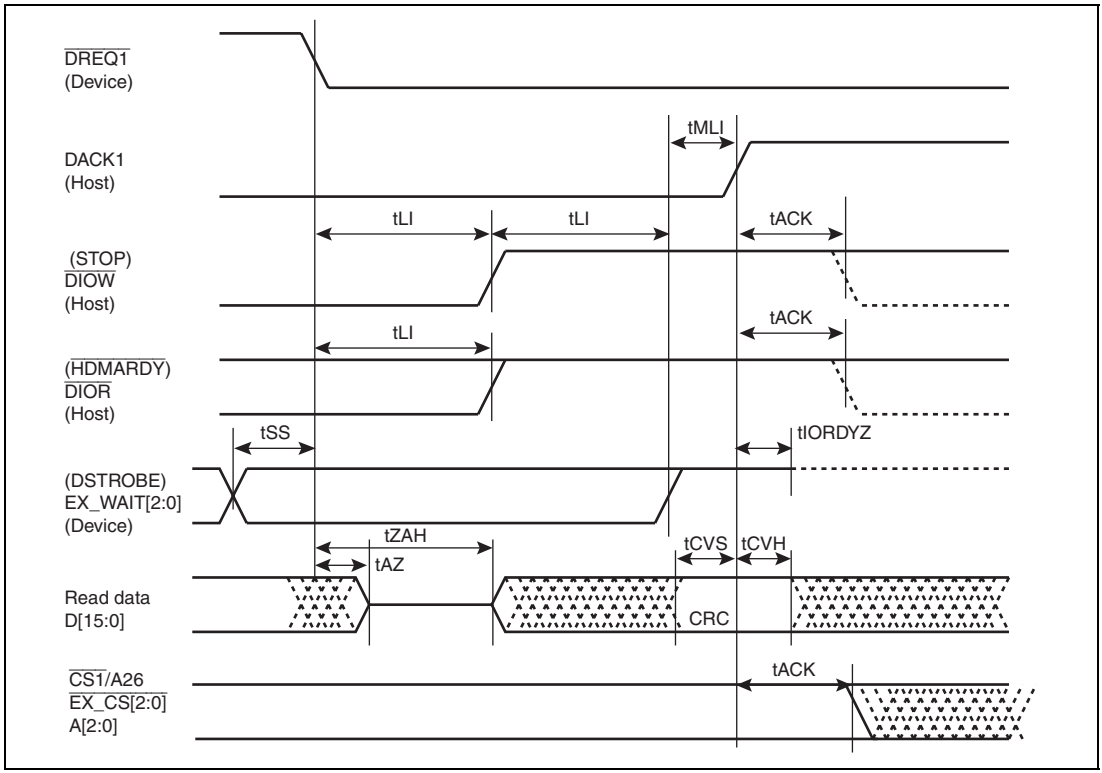


Figure 41.27 Device Terminating UltraDMA Transfer (for Read)

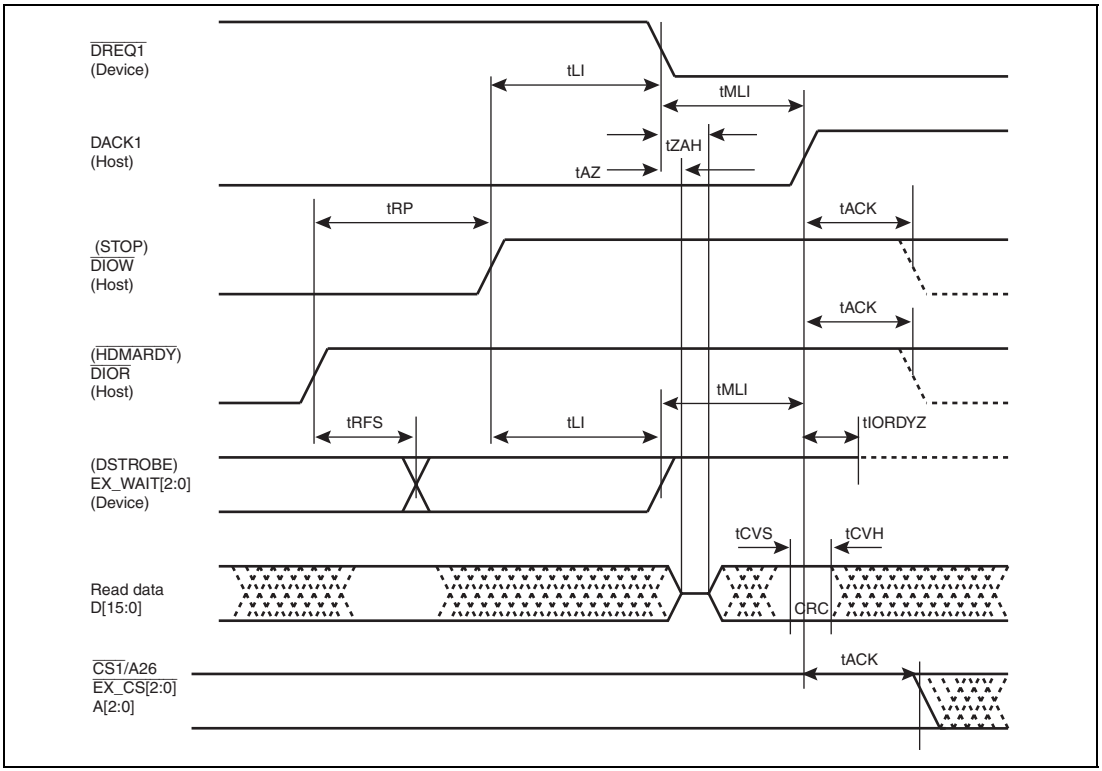


Figure 41.28 Host Terminating UltraDMA Transfer (for Read)

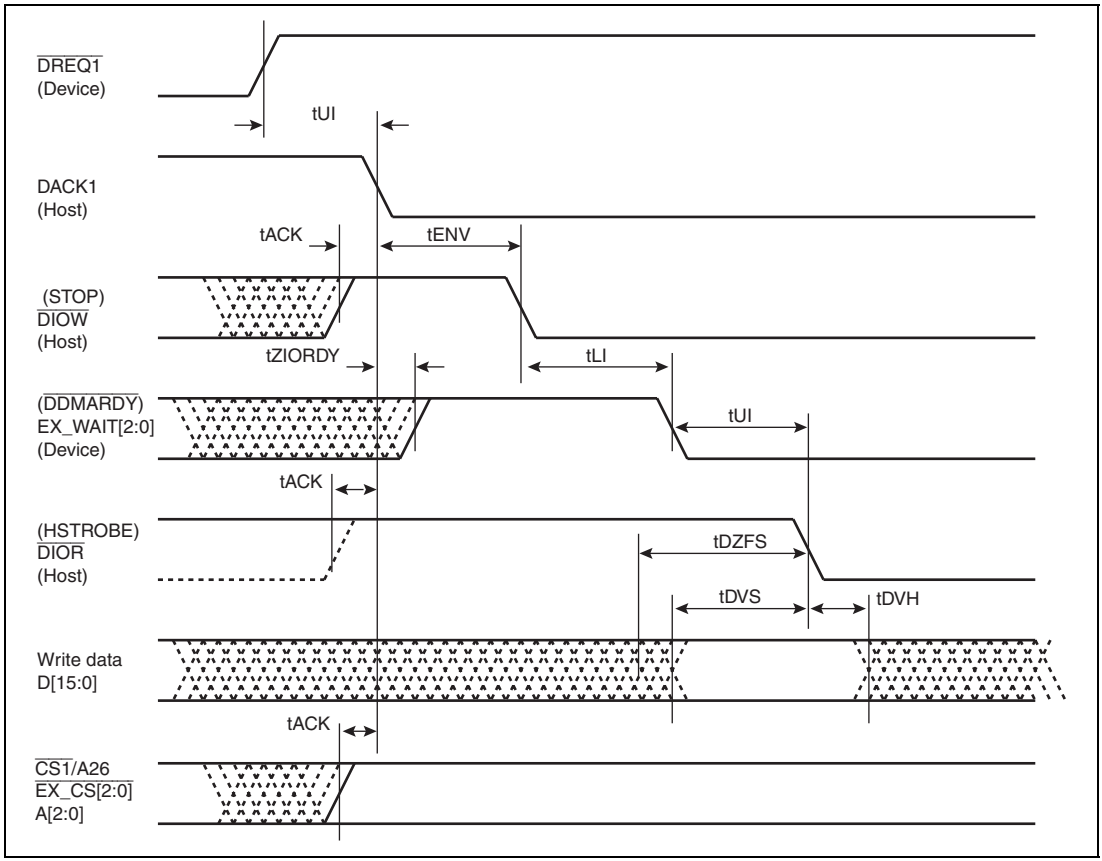


Figure 41.29 Initiating UltraDMA Transfer (for Write)

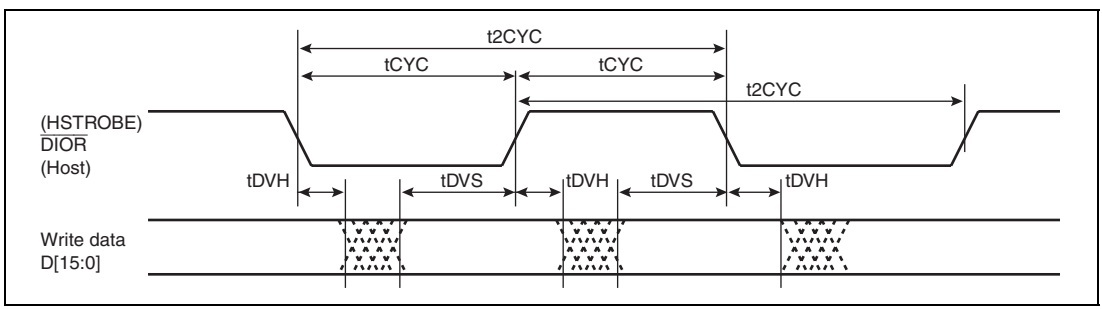


Figure 41.30 UltraDMA Transfer (for Write)

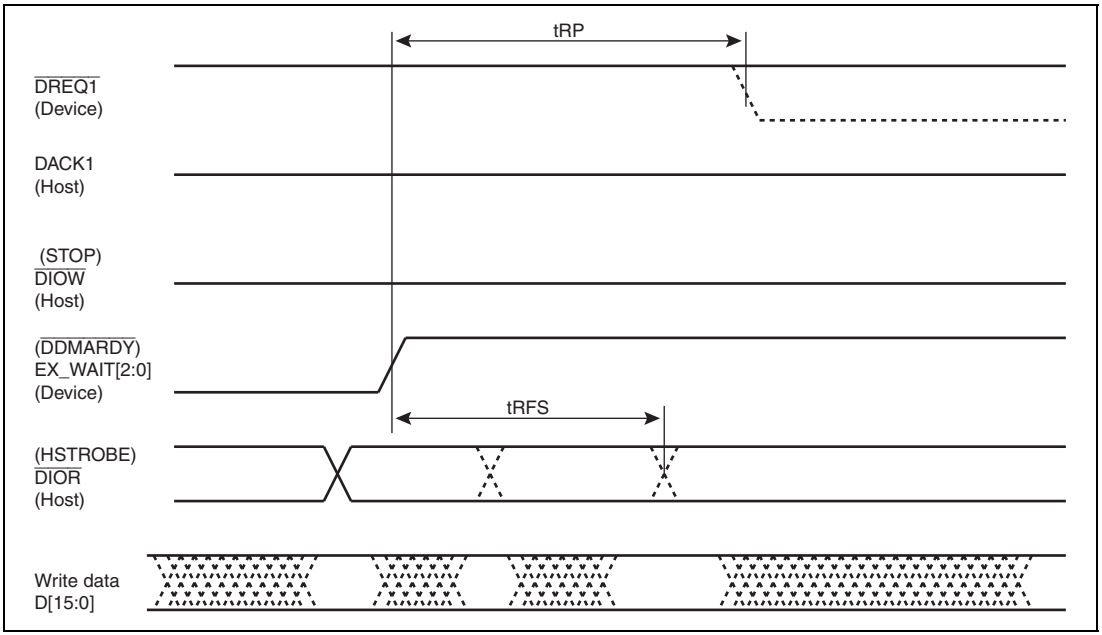


Figure 41.31 Device Pausing UltraDMA (for Write)

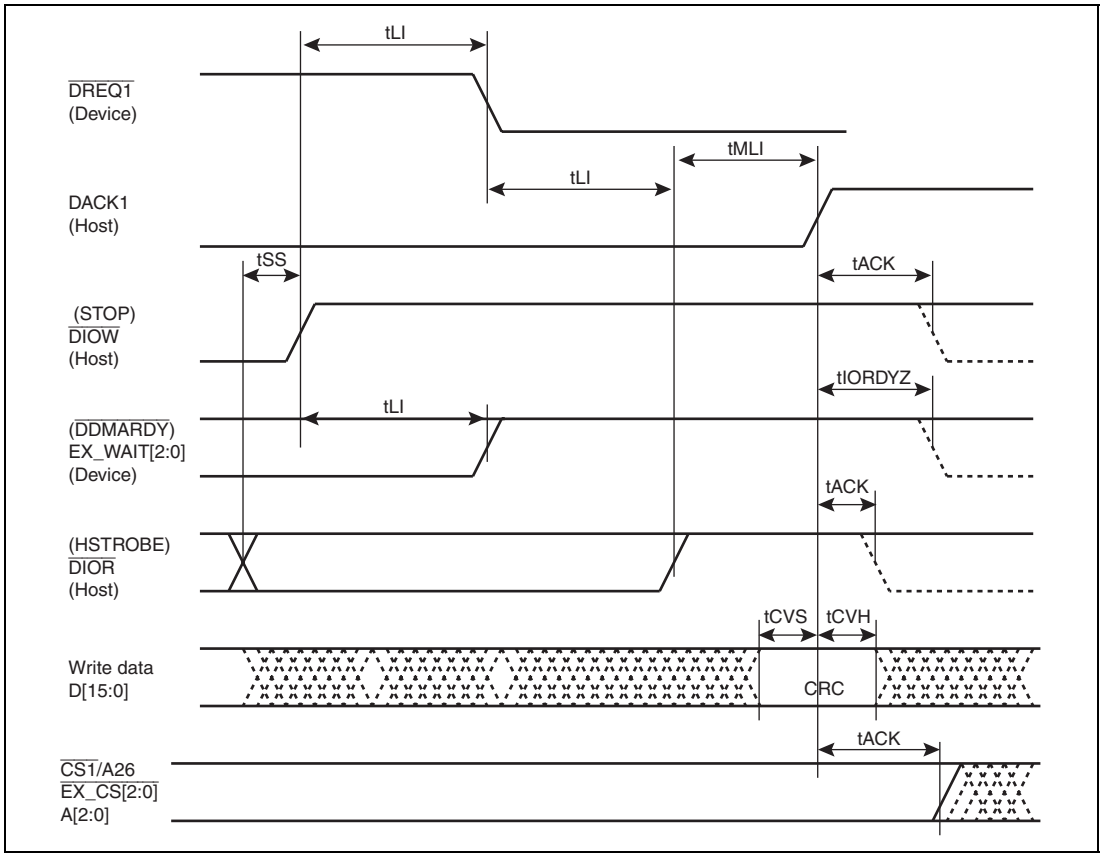


Figure 41.32 Host Terminating UltraDMA Transfer (for Write)

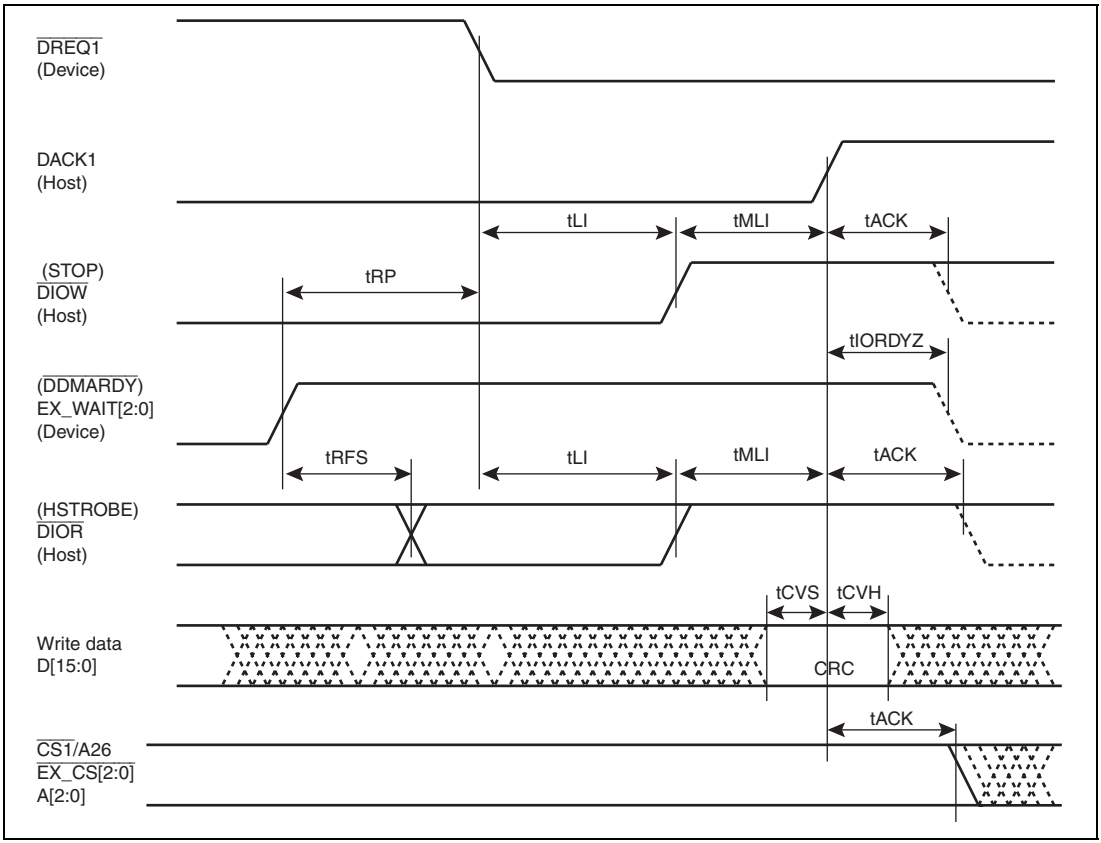


Figure 41.33 Device Terminating UltraDMA Transfer (for Write)

41.9 Interrupt Controller (INTC, INTC2)

Table 41.24 INTC Module Signal Timing

Conditions: Common temperature and voltage conditions

Item	Symbol	Min	Max.	Unit	Figures	Remarks
NMI/IRQn (n = 0 to 3) pulse width at high level	t_{NMIH}/t_{IRQH}	5	—	tcyc	Figure 41.34	This applies during normal operation and during sleep.
NMI/IRQn (n = 0 to 3) pulse width at low level	t_{NMIL}/t_{IRQL}	5	—	tcyc	Figure 41.34	This applies during normal operation and during sleep.

Note: tcyc indicates the period of one cycle of clks1 (the internal clock).
An interrupt may not be detectable if the pulse width is shorter than the above minimum value.

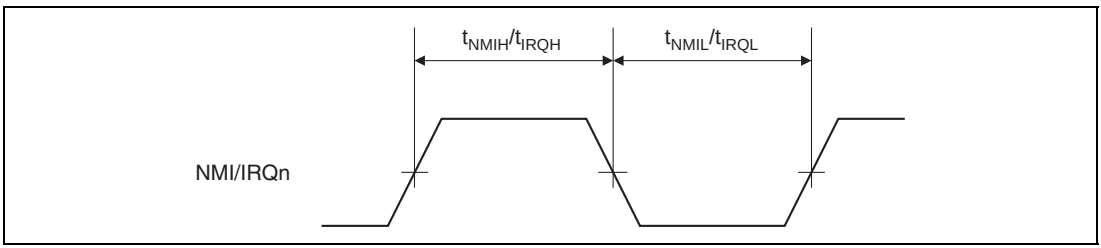


Figure 41.34 NMI/IRQn Input Timing

41.10 Video Input 0 (VIN0)

Table 41.25 Video Input Timing

Conditions: Common temperature and voltage conditions, CL= 0 pF

Item	Symbol	Min.	Typ.	Max.	Unit	Figures
VI_CLK cycle time	tVCcyc	12.5	37	—	ns	Figure 41.35
Data setup time	tVCDs	5	—	—	ns	
Data hold time	tVCDh	3	—	—	ns	
Sync signal setup time	tVCCEs	5	—	—	ns	
Sync signal hold time	tVCCEh	3	—	—	ns	
$\overline{\text{VI\_HSYNC}}$ hold cycle	tVCMcyc	8	—	—	cycle	

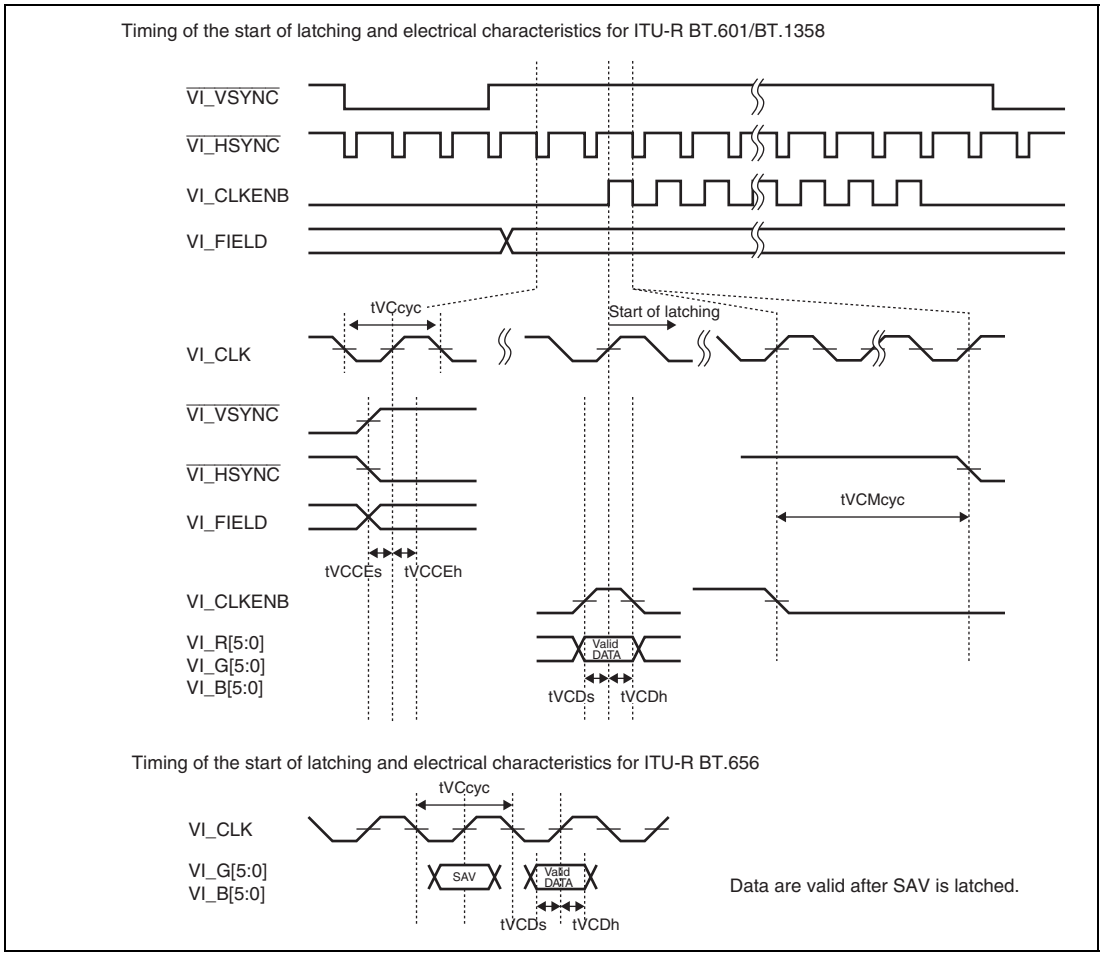


Figure 41.35 Latch Start Timing and Electrical Characteristics

41.11 Video Input 1 (VIN1)

Table 41.26 VIN Module Signal Timing

Conditions: Common temperature and voltage conditions

Item	Symbol	Min.	Typ.	Max.	Unit	Figures
VINn input hold time	t_{VIDH}	5	—	—	ns	Figure 41.36
VINn input setup time	t_{VIDS}	5	—	—	ns	
Sync signal hold time	t_{VCCEH}	5	—	—	ns	
Sync signal setup time	t_{VCCES}	5	—	—	ns	
VICLK clock cycle	t_{VICYC}	34	37	40	ns	

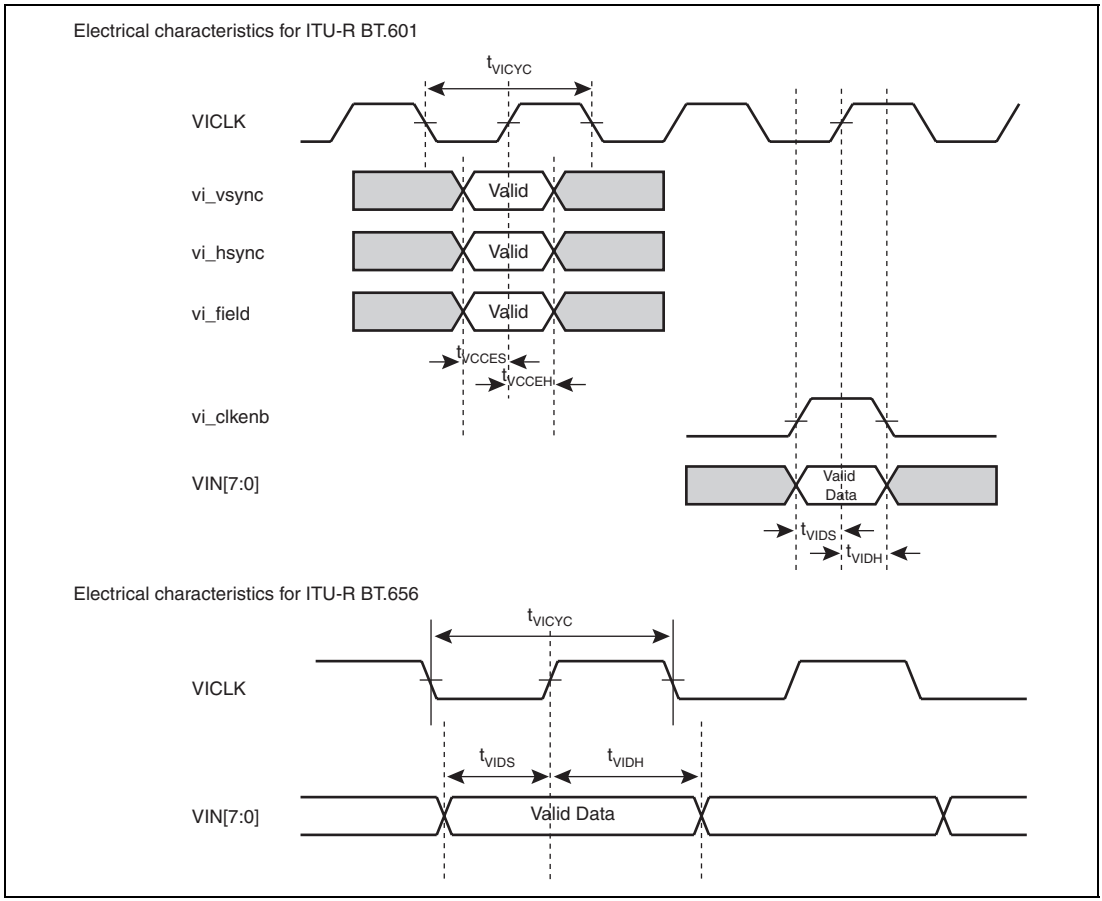


Figure 41.36 Video Input Timing

41.12 Display Unit (DU)

Table 41.27 DOTCLKIN Timing

Conditions: Common temperature and voltage conditions, CL= 20 pF

Item	Symbol	Min.	Typ.	Max.	Unit	Figures
DOTCLKIN cycle time	tDICYC	8.4	—	200	ns	Figure 41.37
DOTCLKIN high level time	tDCKIH	3	—	—	ns	
DOTCLKIN low level time	tDCKIL	3	—	—	ns	

Table 41.28 Display Timing

Conditions: Common temperature and voltage conditions, CL= 20 pF

Item	Symbol	Min.	Typ.	Max.	Unit	Figures
Display input control signal*[1] setup time	tDS1	5	—	—	ns	Figure 41.38
Display input control signal*[1] hold time	tDH1	3	—	—	ns	(DOTCLKIN used as reference)
DOTCLKOUT output cycle time	tDCYC	10	—	200	ns	Figure 41.39
DOTCLKOUT output high level width	tDCKH	2	—	—	ns	(DOTCLKOUT used as reference)
Display input control signal*[1] setup time*[4]	tDS2	16	—	—	ns	
Display input control signal*[1] hold time*[4]	tDH2	-3	—	—	ns	
Display output control signal*[2] output delay time	tDD	0	—	7	ns	
Display digital data*[3] output delay time	tDD	0	—	7	ns	
$\overline{\text{EXHSYNC}}$ input low level width	tEXHLW	4tDCYC	—	—	ns	Figure 41.40
$\overline{\text{EXHSYNC}}$ input high level width	tEXHHW	4tDCYC	—	—	ns	
$\overline{\text{EXVSYNC}}$ input low level width	tEXVLW	3HC	—	—	tDCYC	
$\overline{\text{ODDF}}$ setup time for VSYNC input	tOD1	(ys + yw) × HC	—	—	tDCYC	
$\overline{\text{ODDF}}$ hold time for VSYNC input	tOD2	1HC	—	—	tDCYC	

Notes: ys: From rise of VSYNC to display start position in the vertical direction of the display screen (unit: raster line)

yw: Vertical display period of display screen (unit: raster line)

HC: Horizontal scan period (unit: dot clock)

1. For correspondence between the display input control signals and pin names, see table 41.29.
2. For correspondence between the display output control signals and pin names, see table 41.29.
3. For correspondence between the display digital data and pin names, see table 41.29.

4. Clock signal is input from DOTCLKIN and then output from DOTCLKOUT without being frequency-divided.

Table 41.29 Correspondence between Signals in Notes and Pin Names

Signal in Note	Pin name
Display input control signals	DU0_EXVSYNC/DU0_VSYNC
	DU0_EXHSYNC/DU0_HSYNC
	DU0_EXODDF/DU0_ODDF
Display output control signals	DU0_EXVSYNC/DU0_VSYNC
	DU0_EXHSYNC/DU0_HSYNC
	DU0_EXODDF/DU0_ODDF
	DU0_DISP
	DU0_CDE
Display digital data	DU0_DR7
	DU0_DR6
	DU0_DR5
	DU0_DR4
	DU0_DR3
	DU0_DR2
	DU0_DR1
	DU0_DR0
	DU0_DG7
	DU0_DG6
	DU0_DG5
	DU0_DG4
	DU0_DG3
	DU0_DG2
	DU0_DG1
	DU0_DG0
	DU0_DB7
	DU0_DB6
	DU0_DB5

Signal in Note	Pin name
Display digital data	DU0_DB4
	DU0_DB3
	DU0_DB2
	DU0_DB1
	DU0_DB0

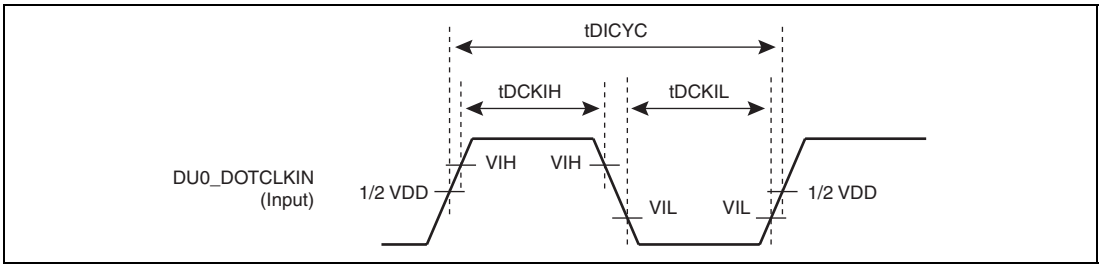


Figure 41.37 DOTCLKIN Clock Input Timing

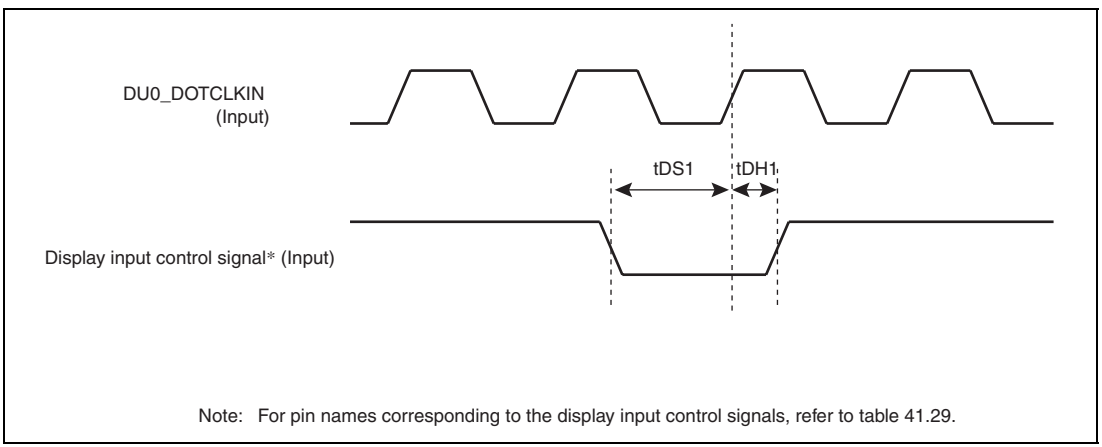
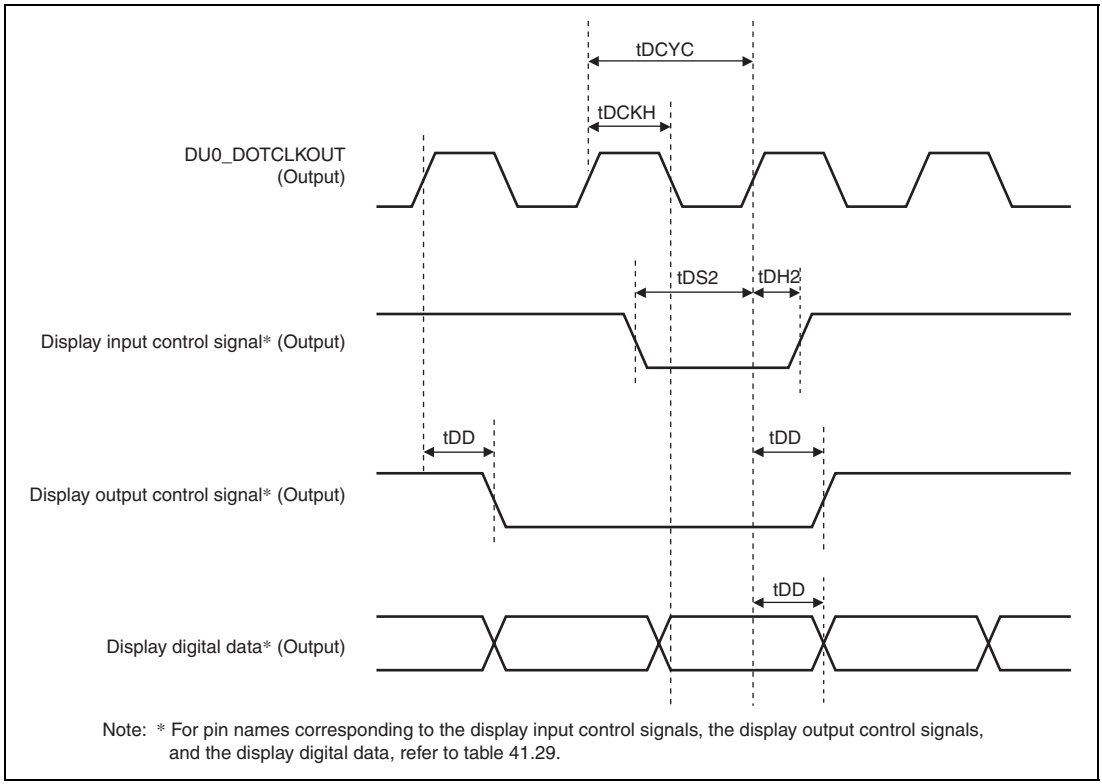


Figure 41.38 Display Timing (DOTCLKIN Used as Reference)



Note: * For pin names corresponding to the display input control signals, the display output control signals, and the display digital data, refer to table 41.29.

Figure 41.39 Display Timing (DOTCLKOUT Used as Reference)

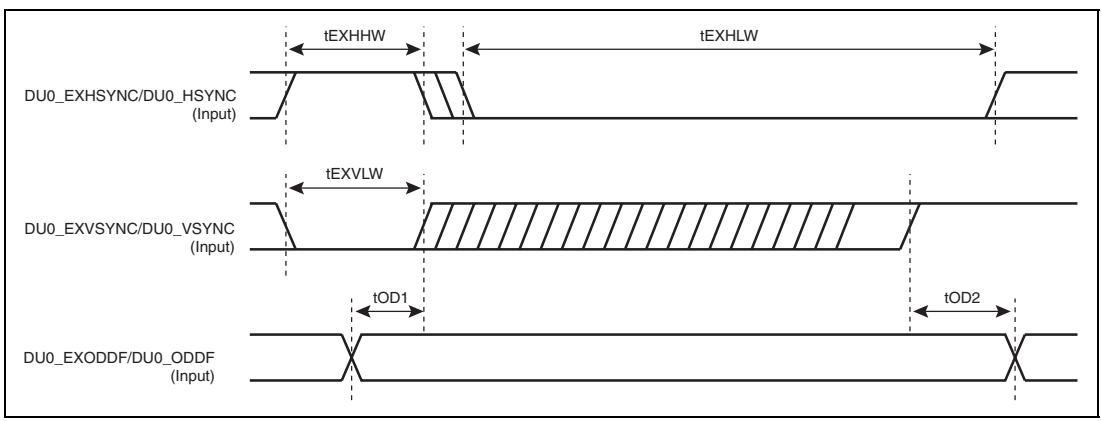


Figure 41.40 TV Sync Mode Display Timing

41.13 LCD Controller (LCDC)

Table 41.30 LCDC Module Signal Access Timing

Conditions: Common temperature and voltage conditions

Item	Symbol	Min.	Max.	Unit	Figures
LCD_CLK input clock frequency	t_{FREQ}	—	54	MHz	
LCD_CLK input clock rise time	t_r	—	3	ns	
LCD_CLK input clock fall time	t_f	—	3	ns	
LCD_CLK input clock duty	t_{DUTY}	90	110	%	
Clock (LCD_CL2) cycle time	t_{CC}	25	—	ns	Figure 41.41
Clock (LCD_CL2) high level pulse width	t_{CHW}	7	—	ns	
Clock (LCD_CL2) low level pulse width	t_{CLW}	7	—	ns	
Clock (LCD_CL2) signal transition time (for rising and falling edges)	t_{CT}	—	3	ns	
Data (LCD_DATA) delay time	t_{DDdo}	–3.5	3	ns	
Display enable (LCD_M_DISP) delay time	t_{IDdo}	–3.5	3	ns	
Horizontal sync (LCD_CL1) delay time	t_{HDdo}	–3.5	3	ns	
Vertical sync signal (LCD_FLM) delay time	t_{VDdo}	–3.5	3	ns	

Note: Pck indicates a peripheral clock frequency.

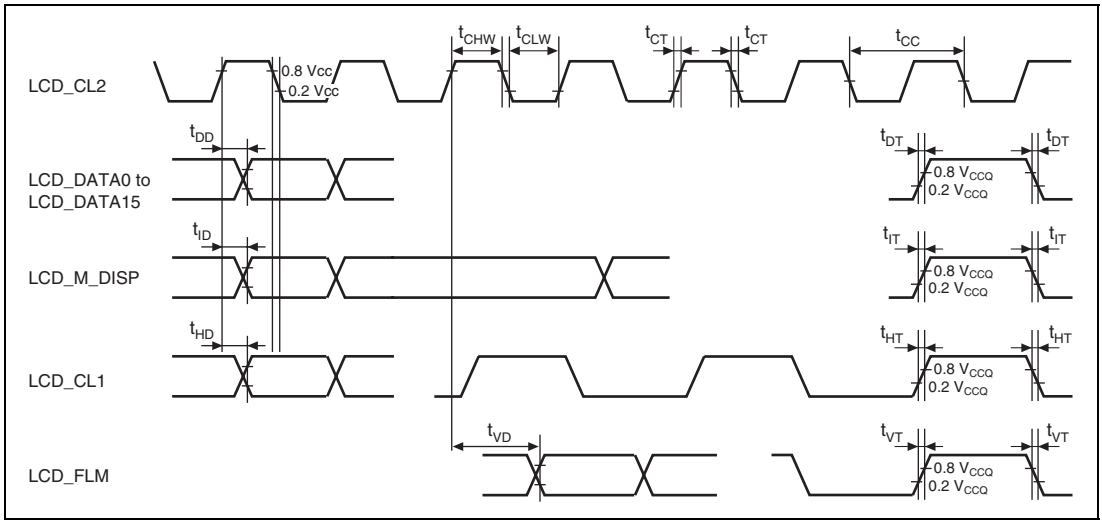


Figure 41.41 LCDC Module Signal Timing

41.14 Serial Sound Interface (SSI)

Table 41.31 SSI Signal Timing

Conditions: Common temperature and voltage conditions, CL = 30 pF

Item	Symbol	Min.	Typ.	Max.	Unit	Remarks	Figures
Output clock cycle	t_O	80	—	3364	ns	—	Figure 41.42
Input clock cycle	t_I	80	—	3364	ns	—	
Clock high time	t_{HC}	35	—	—	ns	Bidirectional	
Clock low time	t_{LC}	35	—	—	ns	Bidirectional	
Clock rise time	t_{RC}	—	—	20	ns	Output (100 pF)	
Output delay time	t_D	0	—	25	ns	—	Figures 41.43 to 41.48
Setup time	t_S	17	—	—	ns	—	
Hold time	t_H	5	—	—	ns	—	
AUDIO_CLK frequency	f_{AUDIO}	3.072	—	24.576	MHz	—	Figure 41.49

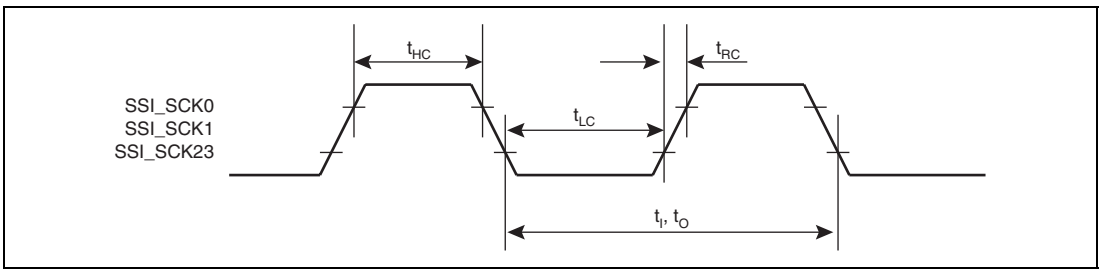


Figure 41.42 Clock Input/Output Timing

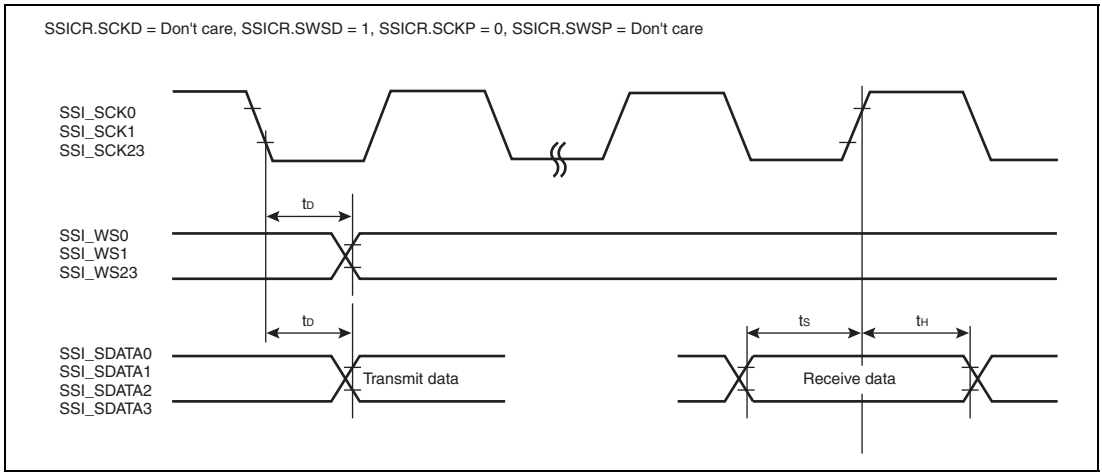


Figure 41.43 SSI Timing (1) for Master Mode (when Sampled at the Rising Edge of SSI_SCK)

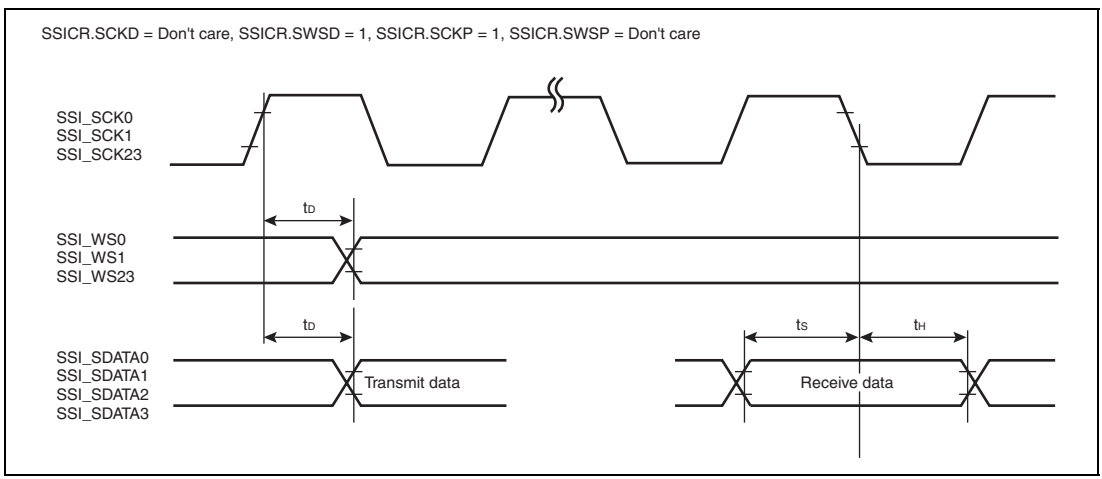


Figure 41.44 SSI Timing (2) for Master Mode (when Sampled at the Falling Edge of SSI_SCK)

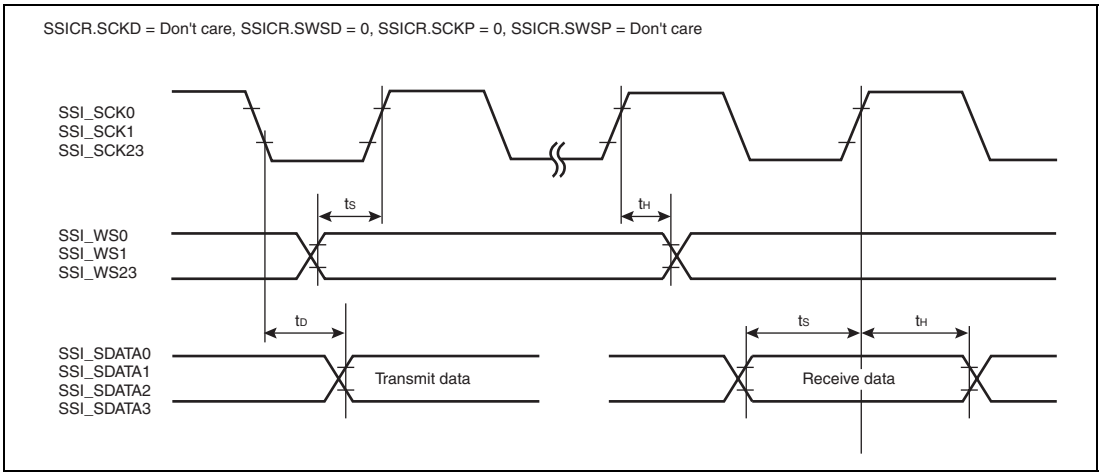


Figure 41.45 SSI Timing (3) for Slave Mode (when Sampled at the Rising Edge of SSI_SCK)

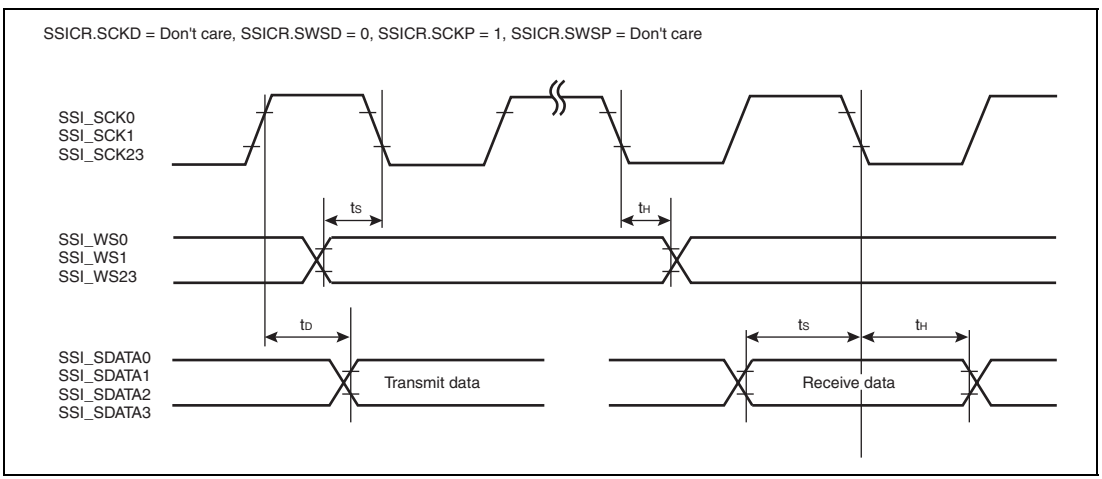


Figure 41.46 SSI Timing (4) for Slave Mode (when Sampled at the Falling Edge of SSI_SCK)

MSB output timing for the uncompressed data and slave transmission
(In Sony or Panasonic format with no padding bit)
SSICR.SCKD = 0, SSICR.SWSD = 0, SSICR.DEL = 1, SSICR.CPEN = 0, SSICR.TRMD = 1, SSITDM.TDM = 0
SSICR.SDTA = 0, or SSICR.SDTA = 1 and SSICR.SWL = SSICR.DWL

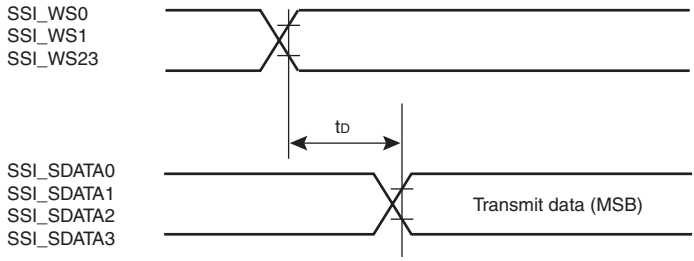


Figure 41.47 SSI Timing (5) for Slave Mode (Sony/Panasonic Format)

MSB output timing for slave transmission in TDM mode (no serial data delay)
SSICR.SCKD = 0, SSICR.SWSD = 0, SSICR.DEL = 1, SSICR.CPEN = 0, SSICR.TRMD = 1, SSITDM.TDM = 1
SSICR.SDTA = 0, or SSICR.SDTA = 1 and SSICR.SWL = SSICR.DWL

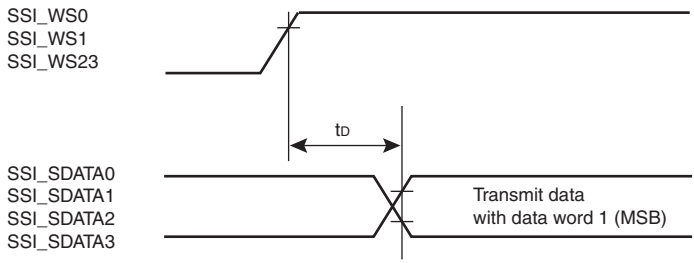


Figure 41.48 SSI Timing (6) for TDM Mode

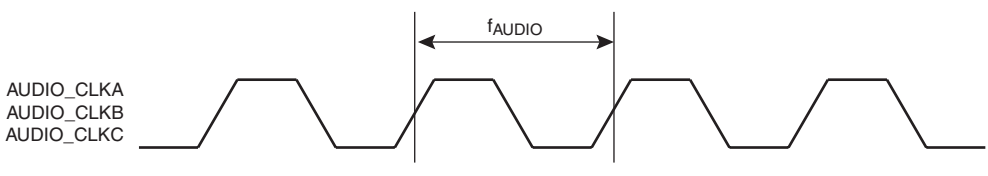


Figure 41.49 AUDIO_CLK Timing

41.15 I²C Bus Interface 3

Table 41.32 I²C Signal Timing

Conditions: Common temperature and voltage conditions, CL = 400 pF

Item	Symbol	Min.	Typ.	Max.	Unit	Figures
I²C_SCL frequency	t_{ICYC}	—	—	400	kHz	Figure 41.50
I²C_SCL low level time	t_{LOW}	$1/(2 \times t_{ICYC}) - 100$	—	—	ns	
I²C_SCL high level time	t_{HIGH}	600	—	—	ns	
I²C_SCL/I²C_SDA falling time	t_{ICF}	—	—	250	ns	
I²C_SDA bus free time	t_{ICBF}	1300	—	—	ns	
I²C_SCL start condition hold time	t_{ICH}	600	—	—	ns	
I²C_SCL repeat-start condition setup time	t_{ICS}	600	—	—	ns	
I²C_SDA stop condition setup time	t_{ICST}	600	—	—	ns	
I²C_SDA setup time	t_{DAS}	100	—	—	ns	
I²C_SDA hold time	t_{ICDH}	0	—	900	ns	

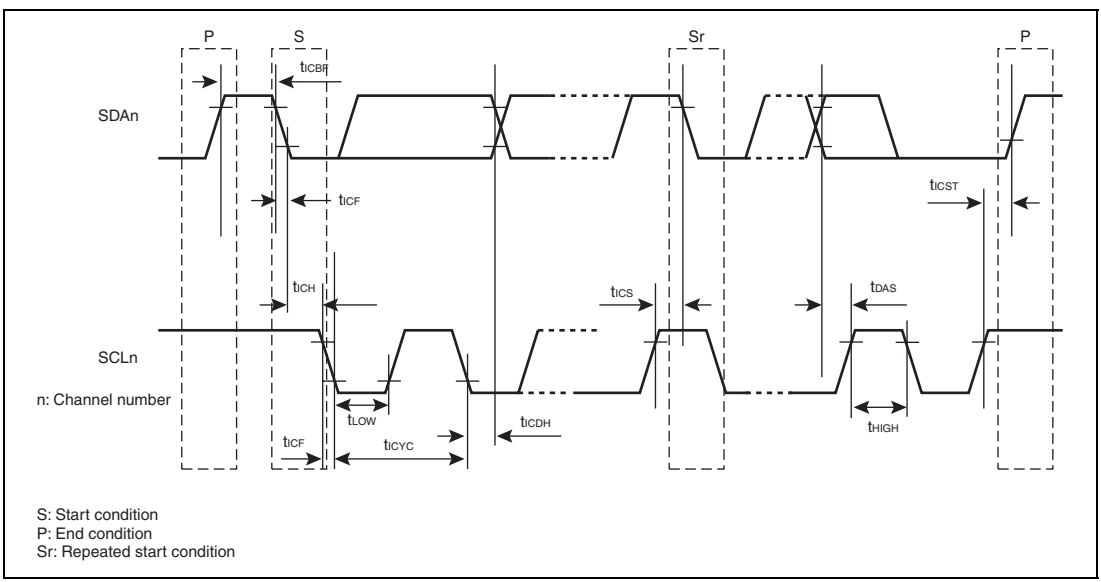


Figure 41.50 I²C Signal Timing

41.16 Serial Peripheral Interface (HSPI)

Table 41.33 HSPI Timing

Conditions: Common temperature and voltage conditions, CL = 30 pF

Mode	Item	Symbol	Min.	Typ.	Max.	Unit	Figures
MASTER	HSPI clock cycle	t_{SPICYC}	—	—	clkp/8	MHz	Figure 41.51
	HSPI clock high level width	t_{SPIHW}	4clkp	—	—	ns	
	HSPI clock low level width	t_{SPILW}	4clkp	—	—	ns	
	HSPI TX setup time	$t_{SUSPITX}$	20	—	—	ns	
	HSPI TX delay time	t_{DSPITX}	—	—	20	ns	
	HSPI RX setup time	$t_{SUSPIRX}$	20	—	—	ns	
	HSPI RX hold time	$t_{HLSPIRX}$	20	—	—	ns	
SLAVE	HSPI clock cycle	t_{SPICYC}	—	—	clkp/8	MHz	
	HSPI clock high level width	t_{SPIHW}	4clkp	—	—	ns	
	HSPI clock low level width	t_{SPILW}	4clkp	—	—	ns	
	HSPI TX setup time	t_{DSPITX}	—	—	3clkp + 20	ns	
	HSPI TX delay time	$t_{SUSPIRX}$	20	—	—	ns	
	HSPI RX setup time	$t_{HLSPIRX}$	1clkp + 5	—	—	ns	
	HSPI RX hold time	t_{CSLEAD}	—	—	3clkp + 20	ns	

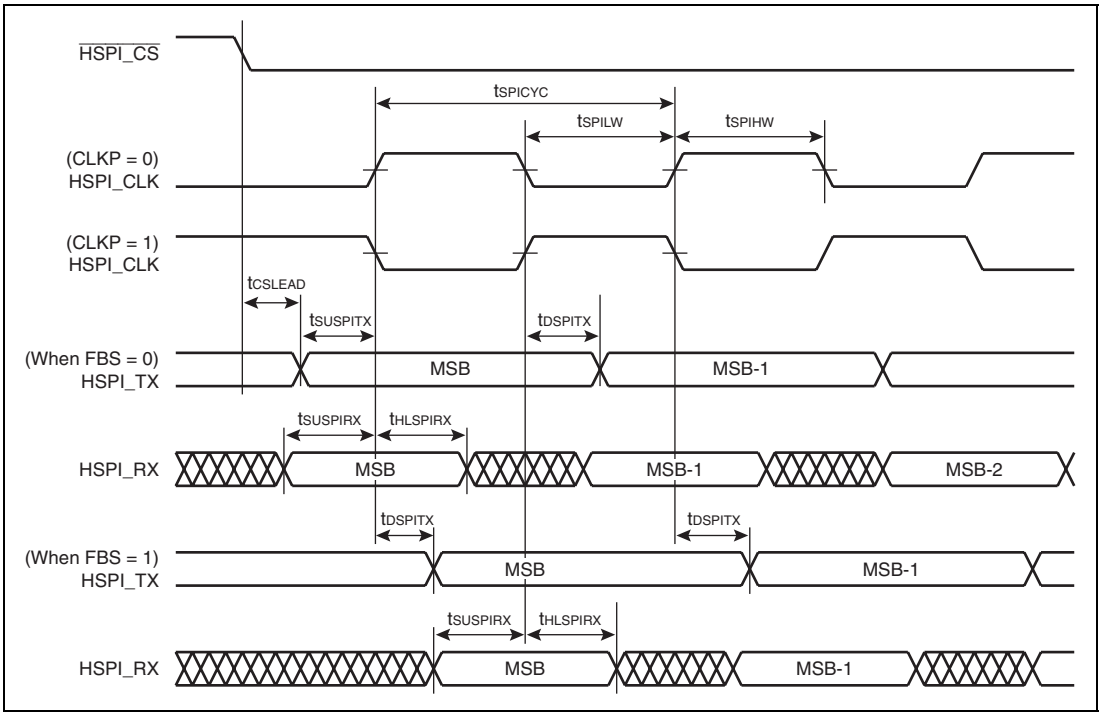


Figure 41.51 HSPI Timing

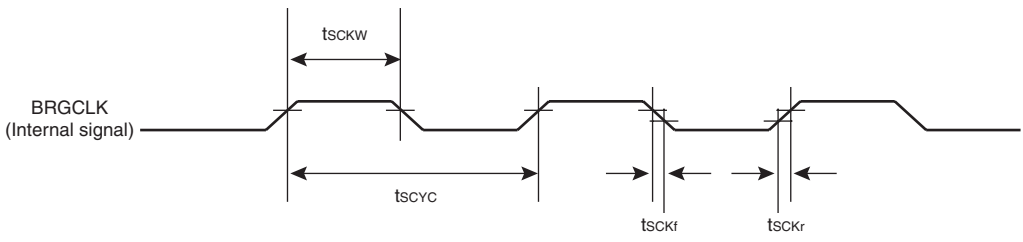
41.17 FIFO Serial Communication Interface (SCIF)

Table 41.34 SCIF Signal Timing

Conditions: Common temperature and voltage conditions, CL = 30 pF

Item	Symbol	Min.	Typ.	Max.	Unit	Figures
Input clock cycle (asynchronous)	t_{SCYC}	4	—	—	t_{CYC}	Figure 41.52
Input clock cycle (synchronous)	t_{SCYC}	8	—	—	t_{CYC}	
Input clock pulse width	t_{SCKW}	0.4	—	0.6	t_{SCYC}	
Input clock rise time	t_{SCKr}	—	—	0.8	t_{CYC}	
Input clock fall time	t_{SCKf}	—	—	0.8	t_{CYC}	
Transmit data delay time	t_{TXD}	—	—	4	t_{CYC}	Figure 41.53
Receive data setup time (synchronous)	t_{RXS}	4	—	—	t_{CYC}	
Receive data hold time (synchronous)	t_{RXH}	1	—	—	t_{CYC}	

Note: t_{CYC} is for one cycle of the IO clock (clkp).



Note: The output signal (BRGCLK) of the external baud rate generator (BRG) is an SCIF internal signal. Set the input clock signal and the DL and CKS registers of the BRG so that the BRGCLK signal can satisfy the above specification.

Figure 41.52 Input Clock Timing

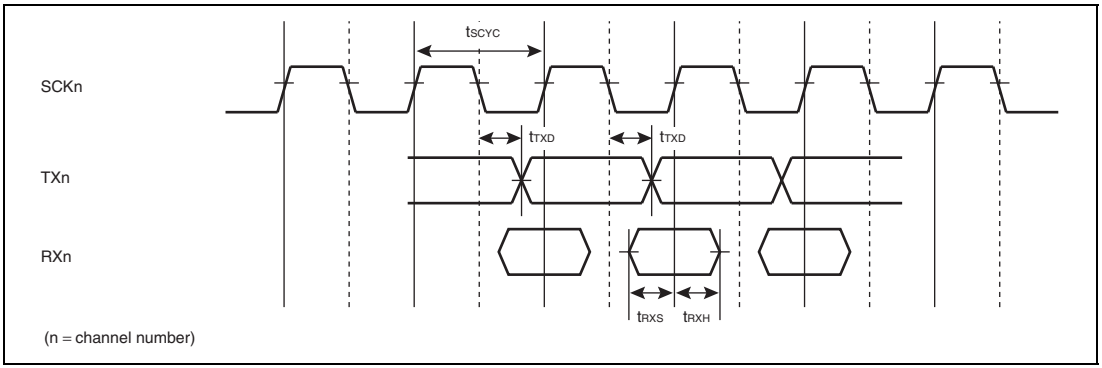


Figure 41.53 Input/Output Timing in Synchronous Mode

41.18 IrDA

For electrical characteristics, see section 41.17, FIFO Serial Communication Interface (SCIF).

41.19 Renesas Serial Peripheral Interface

Table 41.35 RSPI Timing

Conditions: Common temperature and voltage conditions

Item		Symbol	Min.	Max.	Unit	Figures
RSPCK clock cycle	Master	t_{SPcyc}	1	2048	t_{clkp}	Figure 41.54
	Slave		4	2048		
RSPCK clock high-level pulse width	Master	t_{SPCKWH}	0.4	—	t_{SPcyc}	
	Slave		0.4	—		
RSPCK clock low-level pulse width	Master	t_{SPCKWL}	0.4	—	t_{SPcyc}	
	Slave		0.4	—		
Data input setup time	Master	t_{SU}	15	—	ns	Figures 41.55 to 41.58
	Slave		0	—	t_{clkp}	
Data input hold time	Master	t_{H}	0	—	ns	
	Slave		2	—	t_{clkp}	
SSL setup time	Master	t_{LEAD}	1	8	t_{SPcyc}	
	Slave		2	—	t_{clkp}	
SSL hold time	Master	t_{LAG}	1	8	t_{SPcyc}	
	Slave		2	—	t_{clkp}	
Data output delay time	Master	t_{OD}	—	11	ns	
	Slave		—	2	t_{clkp}	
Data output hold time	Master	t_{OH}	0	—	ns	
	Slave		1	—	t_{clkp}	
Continuous transmit delay time	Master	t_{TD}	$1 \times t_{SPcyc} + 1 \times t_{clkp}$	$8 \times t_{SPcyc} + 1 \times t_{clkp}$	ns	
	Slave		$2 \times t_{clkp}$	—		
Slave access time		t_{SA}	—	2	t_{clkp}	Figures 41.57 and 41.58
Slave out release time		t_{REL}	—	2	t_{clkp}	

Note: tclkp: Min. = 20 ns (50 MHz)

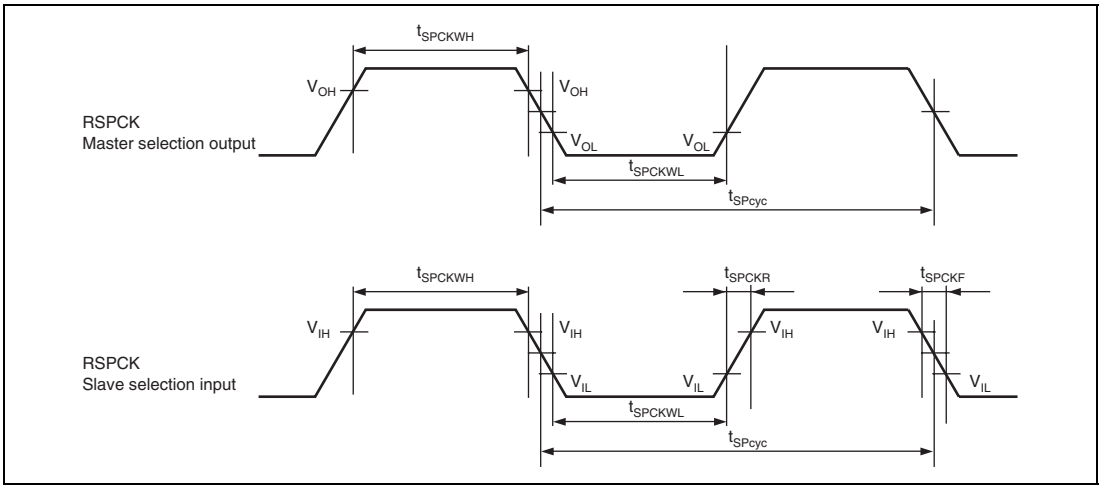


Figure 41.54 Clock Timing

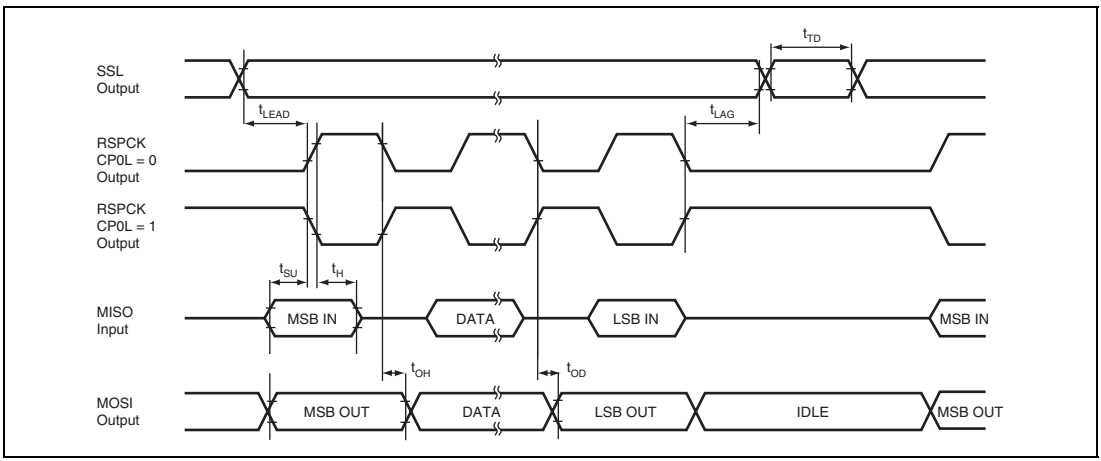


Figure 41.55 Transmit/Receive Timing (Master, CPHA = 0)

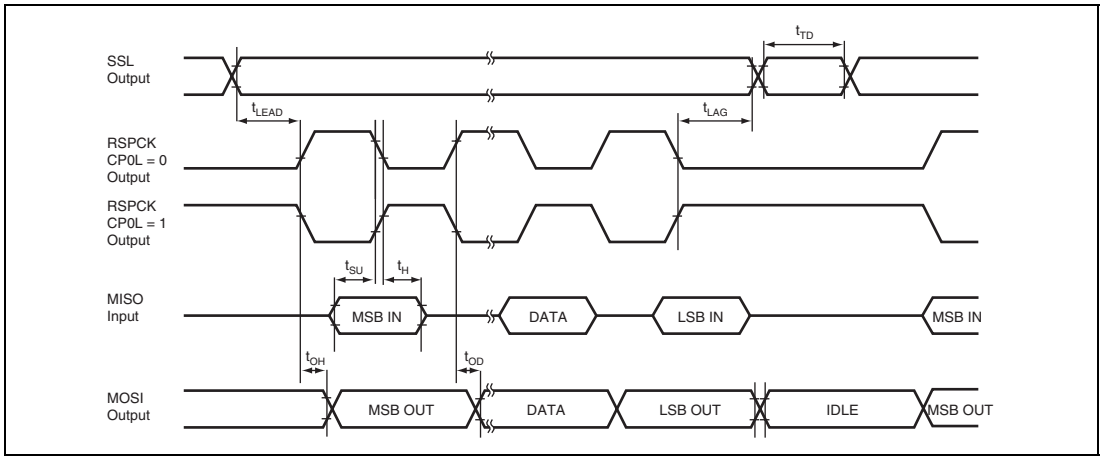


Figure 41.56 Transmit/Receive Timing (Master, CPHA = 1)

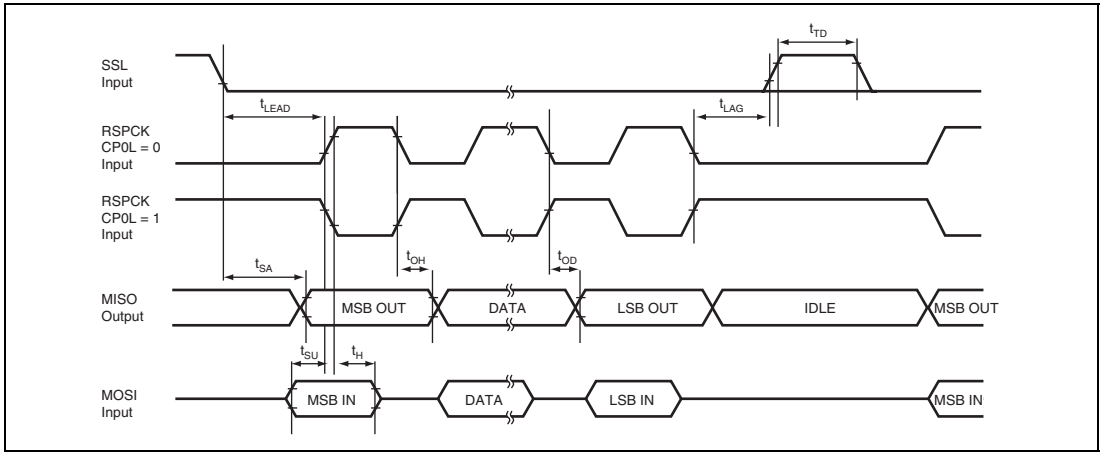


Figure 41.57 Transmit/Receive Timing (Slave, CPHA = 0)

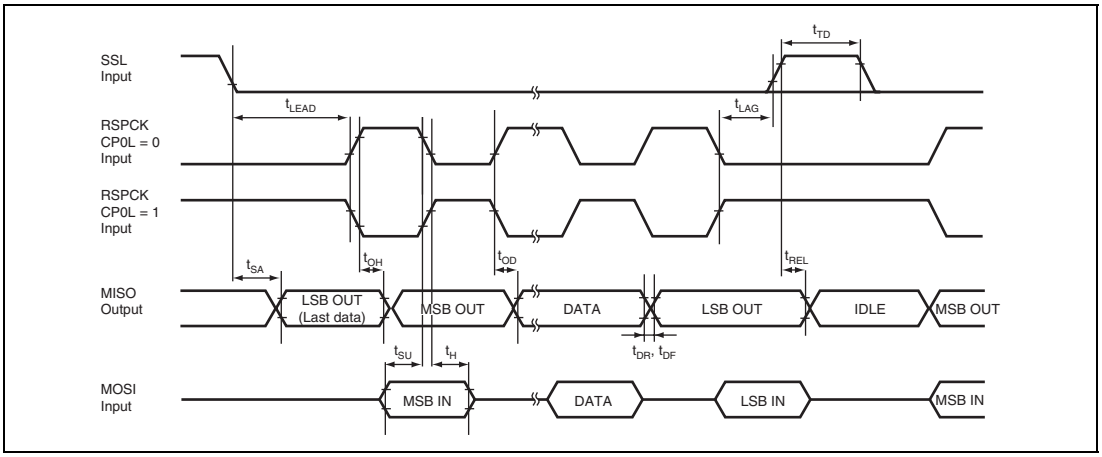


Figure 41.58 Transmit/Receive Timing (Slave, CPHA = 1)

41.20 Host Interface (HIF)

Table 41.36 HIF Module Signal Timing

Conditions: Common temperature and voltage conditions

Item	Symbol	Min.	Max.	Unit	Figures
Read bus cycle time	$t_{HIFCYCR}$	6.0	—	t_{pcyc}	Figure 41.59
Write bus cycle time	$t_{HIFCYCW}$	6.0	—	t_{pcyc}	
Address setup time	t_{HIFAS}	1.0	—	t_{pcyc}	
Address hold time	t_{HIFAH}	1.0	—	t_{pcyc}	
Read low width (read)	t_{HIFWRL}	3.0	—	t_{pcyc}	
Write low width (write)	t_{HIFWWL}	3.0	—	t_{pcyc}	
Read/write high width	$t_{HIFWRWH}$	3.0	—	t_{pcyc}	
Read data delay time	t_{HIFRDD}	—	$2 \times t_{pcyc} + 10$	ns	
Read data hold time	t_{HIFRDH}	0	—	ns	
Write data setup time	t_{HIFWDS}	$t_{pcyc} + 10$	—	ns	
Write data hold time	t_{HIFWDH}	10	—	ns	
$\overline{\text{HIFINT}}$ output delay time	t_{HIFITD}	—	20	ns	Figure 41.60
HIFDREQ output delay time	t_{HIFDQD}	—	20	ns	
HIFRDY output delay time (MD0 = 0)	t_{HIFRYD}	—	3100	t_{pcyc}	Figure 41.61
HIFRDY output delay time (MD0 = 1)	t_{HIFRYD}	—	61000	t_{pcyc}	
HIF pin enable time	t_{HIFEBD}	—	20	ns	
HIF pin disable delay time	t_{HIFDBD}	—	20	ns	

Notes: 1. t_{pcyc} is for a peripheral clock (Pϕ) cycle.

2. t_{HIFAS} is set at the start of the period of overlap between $\overline{\text{HIFCS}}$ and $\overline{\text{HIFRD}}$ being at the low level or of the period of overlap between $\overline{\text{HIFCS}}$ and $\overline{\text{HIFWR}}$ being at the low level.
3. t_{HIFAH} is set at the end of the period of overlap between $\overline{\text{HIFCS}}$ and $\overline{\text{HIFRD}}$ being at the low level or of the period of overlap between $\overline{\text{HIFCS}}$ and $\overline{\text{HIFWR}}$ being at the low level.
4. t_{HIFWRI} is set in the period of overlap between $\overline{\text{HIFCS}}$ and $\overline{\text{HIFRD}}$ being at the low level.
5. t_{HIFWWI} is set in the period of overlap between $\overline{\text{HIFCS}}$ and $\overline{\text{HIFWR}}$ being at the low .

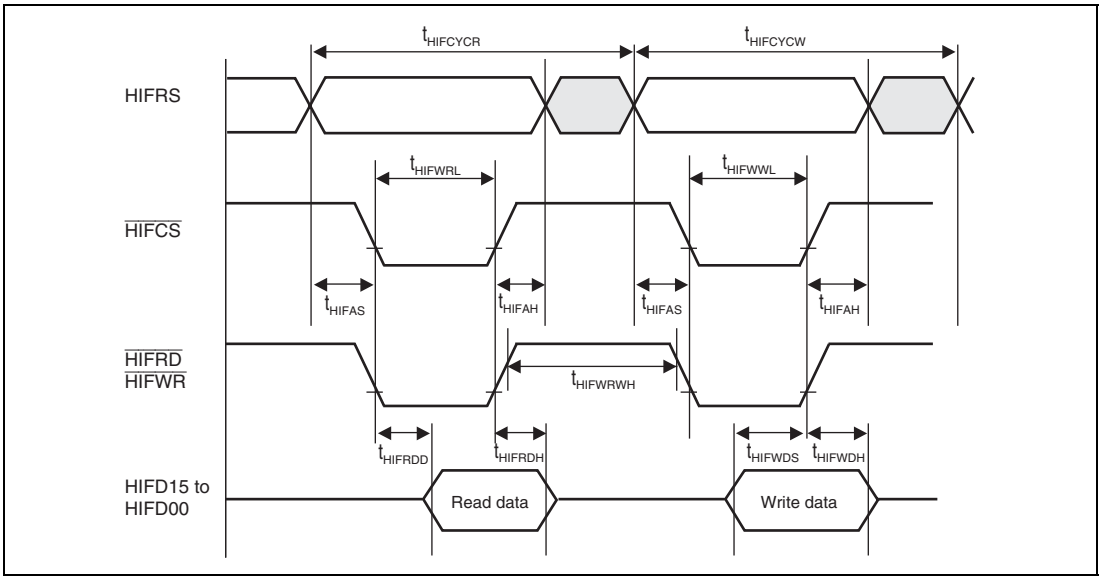


Figure 41.59 HIF Access Timing

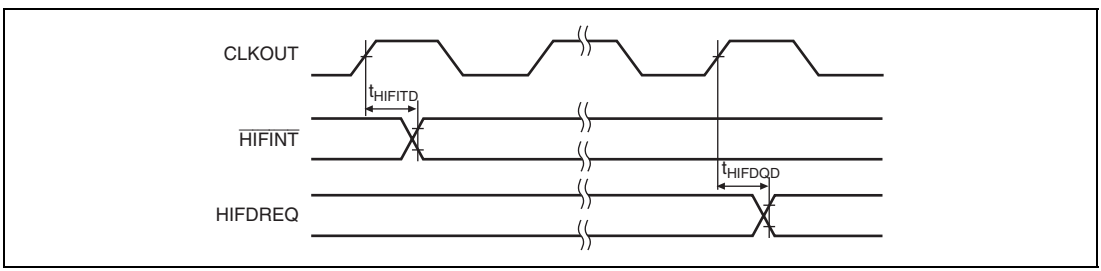


Figure 41.60 Timing of $\overline{\text{HIFINT}}$ and HIFDREQ

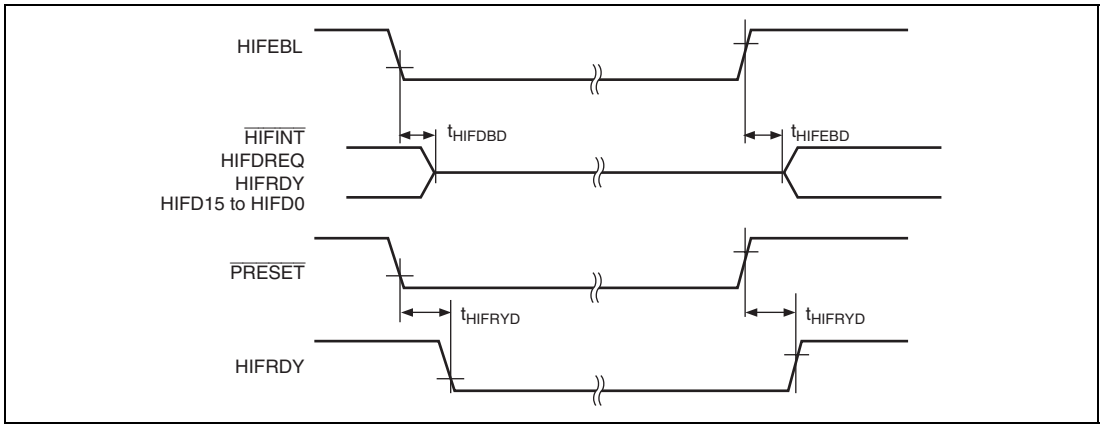


Figure 41.61 Timing of HIFRDY and Enabling or Disabling of the HIF Pin

41.21 USB

41.21.1 High-Speed Transceiver Characteristics

Table 41.37 High-Speed Transceiver Characteristics

Conditions: VDD = VDD-PLL = AV12 = 1.15 to 1.3 V, VCCQ = VCC = VCCQ-PLL = AV33 = AVCC = AVREF = 3.3V, VSS = VSS-PLL = AG = AVSS = 0 V
Measurements are under a common temperature condition.

Item		Symbol	Min.	Typ.	Max.	Unit	Remarks
Input characteristics	Input common-mode voltage range	VHSCM	-50	—	500	mV	
	Differential input sensitivity	Vdiff	150	—	—	mV	
	Squelch detection voltage	VHSSQ	100	—	150	mV	
	Disconnect voltage	VHSDSQ	525	—	625	mV	
Output characteristics	Idle state	VHSOI	-10	—	10	mV	
	High-level output voltage	VHSOH	360	—	440	mV	
	Low-level output voltage	VHSOL	-10	—	10	mV	
	Chirp-J output voltage (differential)	VCHIRPJ	700	—	1100	mV	
	Chirp-K output voltage (differential)	VCHIRPK	-900	—	-500	mV	
	Data transfer rate	THSDRAT	479.76	480	480.24	Mb/s	

41.21.2 Full/Low Speed Transceiver Characteristics

Table 41.38 Full/Low Speed Transceiver Characteristics (Input)

Conditions: VDD = VDD-PLL = AV12 = 1.15 to 1.3 V, VCCQ = VCC = VCCQ-PLL = AV33 = AVCC = AVREF = 3.3 V, VSS = VSS-PLL = AG = AVSS = 0 V
Measurements are under a common temperature condition

Item		Symbol	Min.	Typ.	Max.	Unit	Remarks
Input characteristics	Input voltage	V_{IH}	20	—	—	V	
		V_{IL}	—	—	0.8	V	
	Differential input sensitivity	V_{DI}	0.2	—	—	V	| D+ — D- |
	Differential input common-mode range	V_{CM}	0.8	—	2.5	V	

Table 41.39 Full/Low Speed Transceiver Characteristics (Output)

Conditions: VDD = VDD-PLL = AV12 = 1.15 to 1.3 V, VCCQ = VCC = VCCQ-PLL = AV33 = AVCC = AVREF - 3.3 V, VSS = VSS-PLL = AG = AVSS = 0 V
Measurements are under a common temperature condition

Item			Symbol	Min.	Typ.	Max.	Unit	Remarks
Output characteristics	Output voltage		V_{OH}	2.8	—	VCC	V	
			V_{OL}	—	—	0.3	V	
	Low-speed mode (See figure 41.62.)	Rise time*	T_R	75	—	300	ns	
		Fall time*	T_F	75	—	300	ns	
		Matching	T_{RFM}	80	—	125	%	
		Crossover voltage	T_{CRS}	1.3	—	2.0	V	
	Full-speed mode (See figure 41.63.)	Rise time*	T_R	4	—	20	ns	
		Fall time*	T_F	4	—	20	ns	
		Matching	T_{RFM}	90	—	111.1	%	
		Crossover voltage	T_{CR}	1.3	—	2.0	V	

Note: * Time 10 to 90 % (figure 41.64)

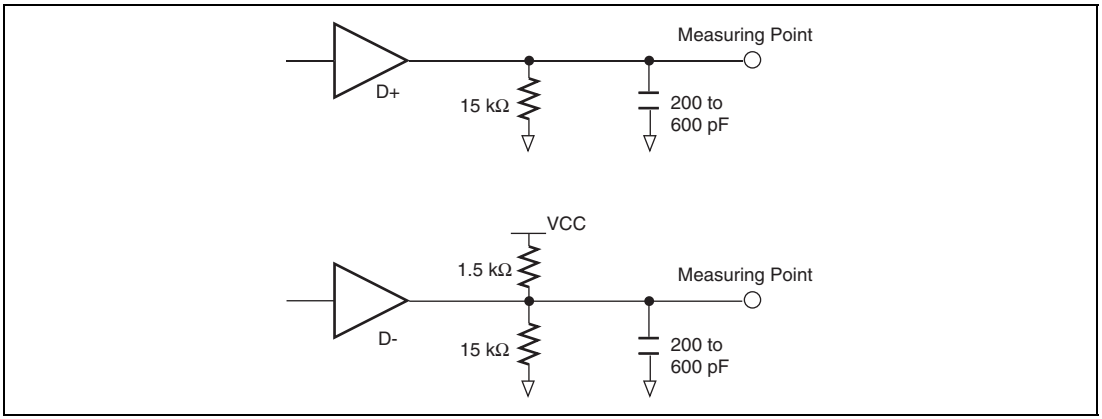


Figure 41.62 Load Conditions in Low-Speed Mode

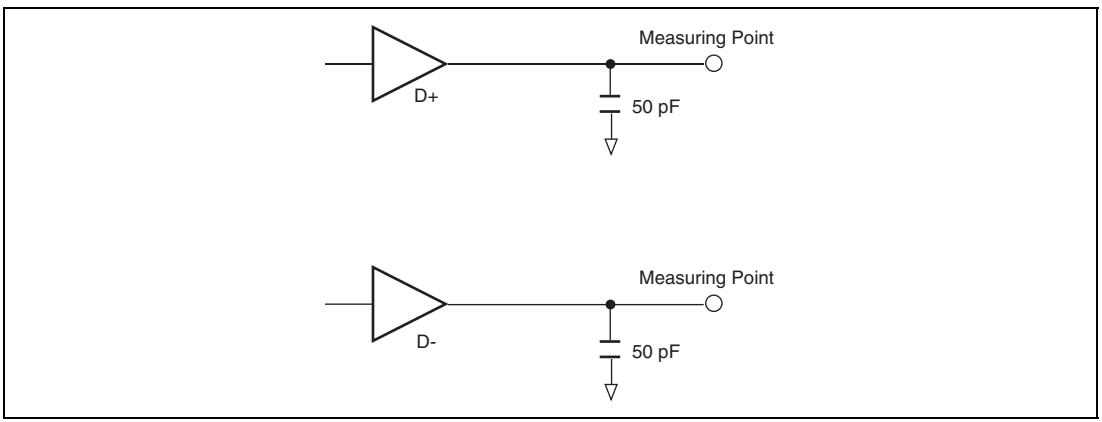


Figure 41.63 Load Conditions in Full-Speed Mode

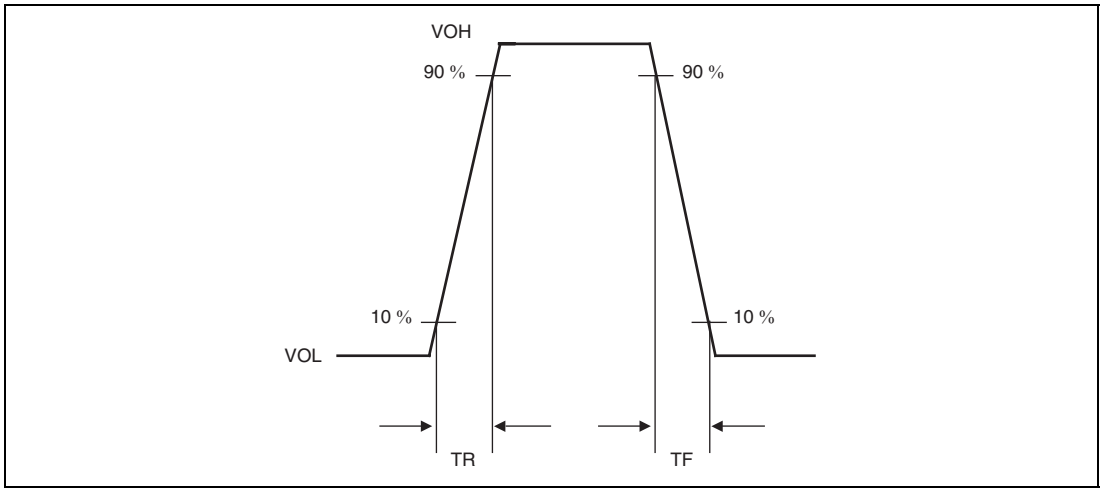


Figure 41.64 TR and TF Measurement Condition

41.21.3 Driver Output Impedance Characteristics

Table 41.40 Driver Output Impedance Characteristics

Conditions: VDD = VDD-PLL = AV12 = 1.15 to 1.3 V, VCCQ = VCC = VCCQ-PLL = AV33 = AVCC = AVREF = 3.3V, VSS = VSS-PLL = AG = AVSS = 0 V
Measurements are under a common temperature condition.

Item		Symbol	Min.	Typ.	Max.	Unit	Remarks
Output impedance		R_O	40.5	45.0	49.5	Ω	
On-chip DP pull-up resistor	For idle state	R_{PU}	0.9	—	1.575	kΩ	
	For reception	R_{PU}	1.425	—	3.09	kΩ	
On-chip pull-down resistor		R_{PD}	14.25	—	24.80	kΩ	
External reference resistor		R_{REF}	5.544	5.6	5.656	kΩ	±1 %

41.21.4 External Clock Accuracy

Table 41.41 External Clock Accuracy

Conditions: VDD = VDD-PLL = AV12 = 1.15 to 1.3 V, VCCQ = VCC = VCCQ-PLL = AV33 = AVCC = AVREF = 3.3V, VSS = VSS-PLL = AG = AVSS = 0 V
Measurements are under a common temperature condition.

Item	Symbol	Min.	Typ.	Max.	Unit	Remarks
External clock accuracy (USBCLK)	—	47.981	48.000	48.019	MHz	±Frequency deviation: ±100 ppm or less

41.21.5 DC Characteristics of VBUS Pin

Table 41.42 DC Characteristics of VBUS Pin

Conditions: VDD = VDD-PLL = AV12 = 1.15 to 1.3 V, VCCQ = VCC = VCCQ-PLL = AV33 = AVCC = AVREF = 3.3V, VSS = VSS-PLL = AG = AVSS = 0 V
Measurements are under a common temperature condition.

Item	Symbol	Min.	Typ.	Max.	Unit	Remarks
VBUS input voltage	VIH	4.35	—	5.25	V	
	VIL	—	—	1.0	V	

41.22 GETHER Module Signal Timing

(1) Ethernet Control Timing (MII)

Table 41.43 Ethernet Control Timing (MII)

Conditions: Common temperature and voltage conditions, 6 mA IO cell selected

Item	Symbol	Min.	Typ.	Max.	Unit	Figures
ETn_TX-CLK cycle time	t_{Tcyc}	40	—	—	ns	Figure 41.65
ETn_TX-EN output delay time	t_{TEND}	3	—	20		
ETn_ETXD3 to ETn_ETXD0 output delay time	t_{ETDD}	3	—	20		
ETn_RX-CLK cycle time	t_{Rcyc}	40	—	—		Figure 41.66
ETn_RX-DV setup time	t_{RDVS}	10	—	—		
ETn_RX-DV hold time	t_{RDVH}	3	—	—		
ETn_ERXD3 to ETn_ERXD0 setup time	t_{ERDS}	10	—	—		
ETn_ERXD3 to ETn_ERXD0 hold time	t_{ERDH}	3	—	—		
ETn_RX-ER setup time	t_{RERS}	10	—	—		Figure 41.67
ETn_RX-ER hold time	t_{RERH}	3	—	—		
ETn_WOL output delay time	t_{WOLD}	1	—	18		Figure 41.68

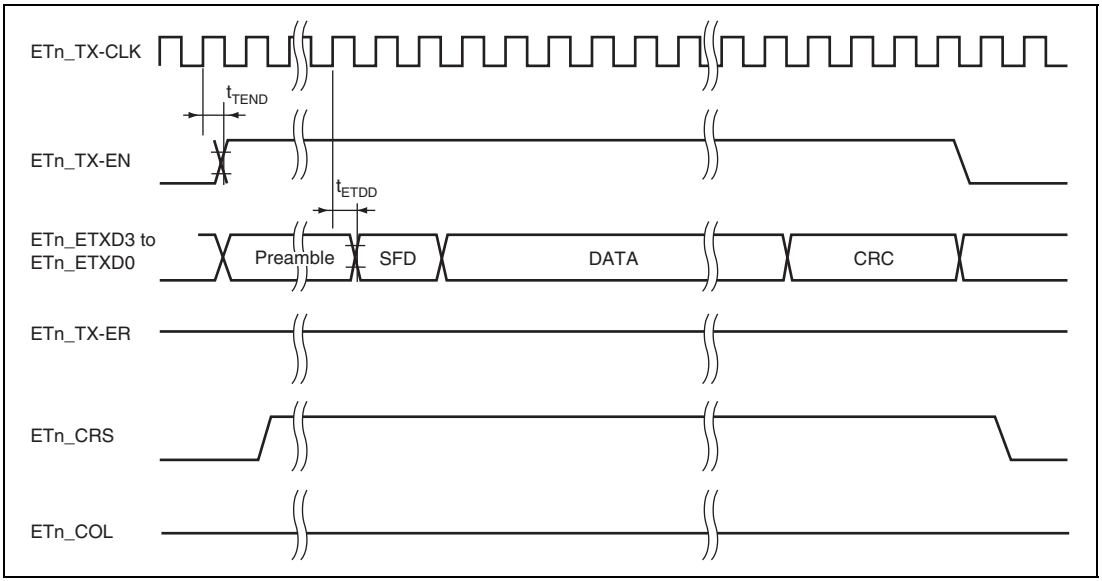


Figure 41.65 MII Transmission Timing (Normal Operation)

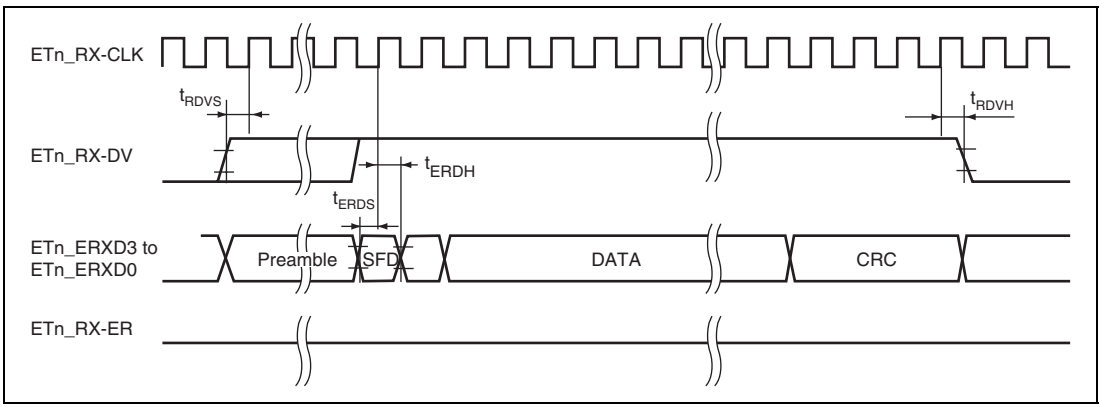


Figure 41.66 MII Reception Timing (Normal Operation)

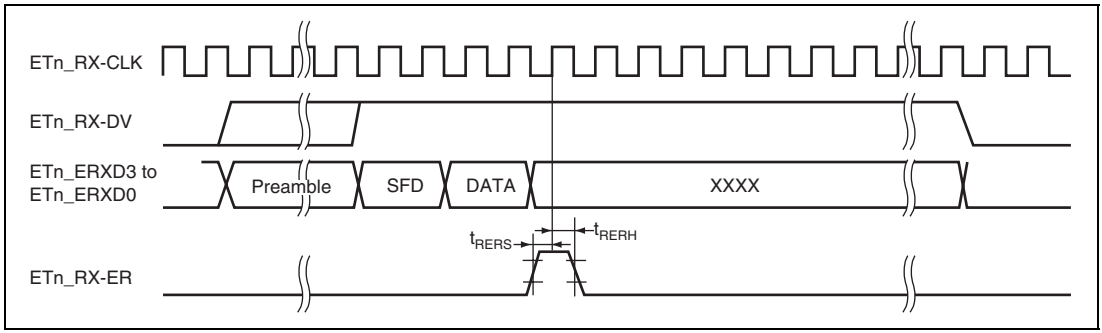


Figure 41.67 MII Reception Timing (Case When Error Occurs)

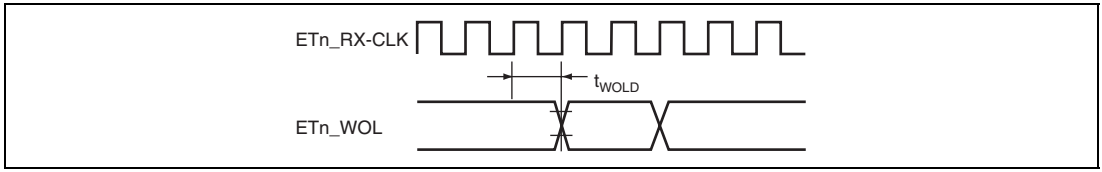


Figure 41.68 WOL Output Timing

(2) Ethernet Control Timing (GMII)

Table 41.44 Ethernet Control Timing (GMII)

Conditions: Common temperature and voltage conditions, 8 mA IO cell selected

Item	Symbol	Min.	Typ.	Max.	Unit	Figures
REF125CK clock input frequency	$f_{REF125CK}$	125 – 100 ppm	—	125 + 100 ppm	MHz	
GETn_TX-CLK cycle time	t_{GTcyc}	8	—	—	ns	Figure 41.69
ETn_TX-EN output delay time	t_{GTEND}	0.5	—	5.5		
GETn_ETXD7 to GETn_ETXD4, ETn_ETXD3 to ETn_ETXD0 output delay time	t_{GETDD}	0.5	—	5.5		
ETn_RX-CLK cycle time	t_{GRcyc}	8	—	—		Figure 41.70
ETn_RX-DV setup time	t_{GRDVS}	2	—	—		
ETn_RX-DV hold time	t_{GRDVH}	0	—	—		
GETn_ERXD7 to GETn_ERXD4, ETn_ERXD3 to ETn_ERXD0 setup time	t_{GERDS}	2	—	—		
GETn_ERXD7 to GETn_ERXD4 ETn_ERXD3 to ETn_ERXD0 hold time	t_{GERDH}	0	—	—		
ETn_RX-ER setup time	t_{GRERS}	2	—	—		Figure 41.71
ETn_RX-ER hold time	t_{GRERH}	0	—	—		
ETn_WOL output delay time	t_{GWOLD}	0	—	18		Figure 41.72

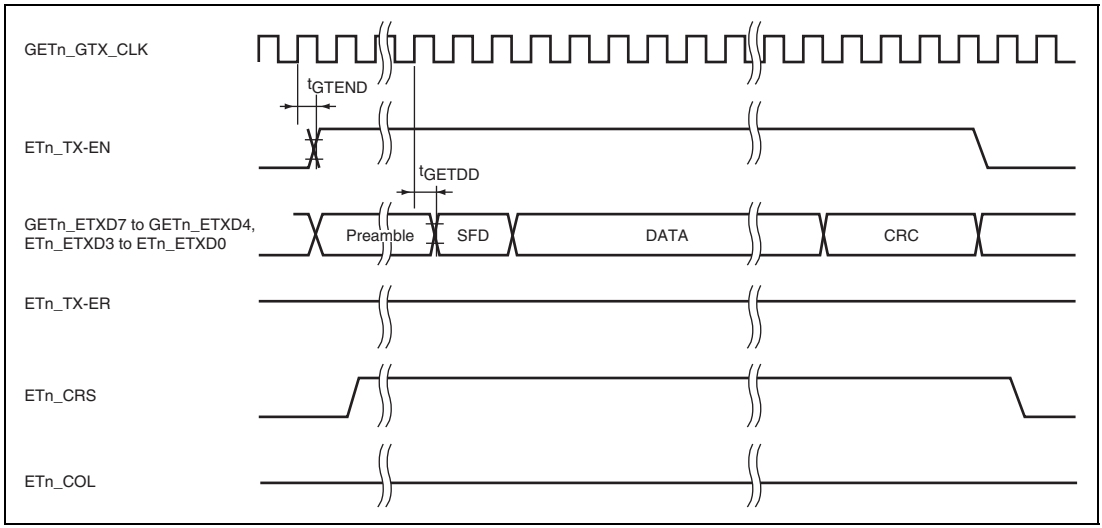


Figure 41.69 GMII Transmission Timing (Normal Operation)

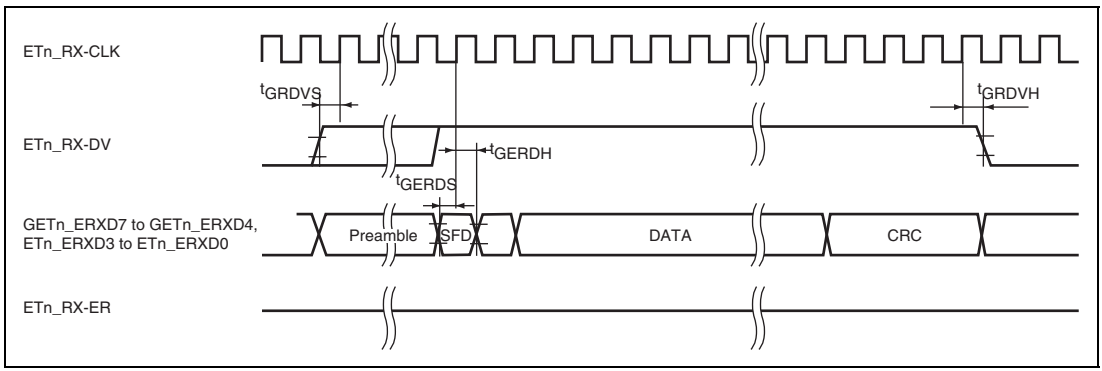


Figure 41.70 GMII Reception Timing (Normal Operation)

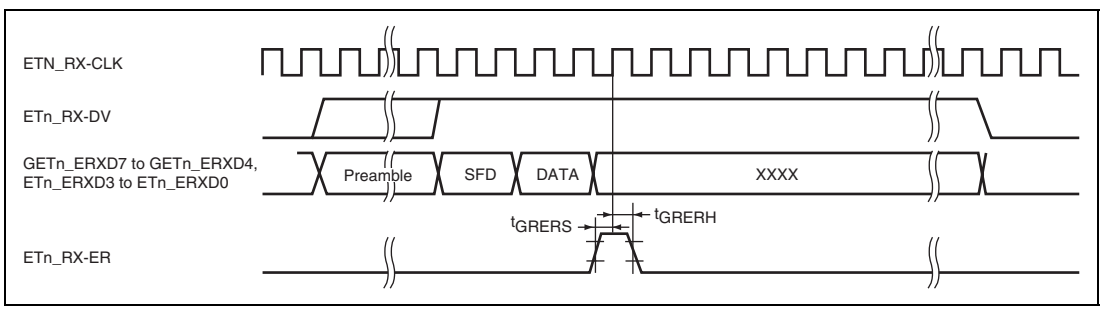


Figure 41.71 GMII Reception Timing (Case When Error Occurs)

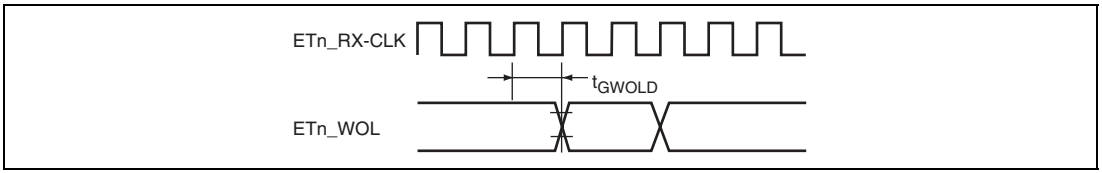


Figure 41.72 WOL Output Timing

(3) Ethernet Control Timing (RMII)

Table 41.45 Ethernet Control Timing (RMII)

Conditions: Common temperature and voltage conditions, 6 mA IO cell selected

Item	Symbol	Min.	Typ.	Max.	Unit	Figures
REF50CK clock input frequency	f_{RTcyc}	50 – 50 ppm	—	50 + 50 ppm	MHz	Figure 41.73
RMIIn_TXD_EN, RMII1M_TXD_EN output delay time	t_{RTEND}	2.5	—	11	ns	
RMIIn_TXD1, RMIIn_TXD0, RMII1M_TXD1, RMII1M_TXD0 output delay time	t_{RETDD}	2.5	—	11		
RMIIn_CRS_DV, RMII1M_CRS_DV setup time	t_{RRDVS}	4	—	—		Figure 41.74
RMIIn_CRS_DV, RMII1M_CRS_DV hold time	t_{RRDVH}	2.5	—	—		
RMIIn_RXD1, RMIIn_RXD0, RMII1M_RXD1, RMII1M_RXD0 setup time	t_{RERDS}	4	—	—		
RMIIn_RXD1, RMIIn_RXD0, RMII1M_RXD1, RMII1M_RXD0 hold time	t_{RERDH}	2.5	—	—		
RMIIn_RX_ER setup time	t_{RRERS}	4	—	—		Figure 41.75
RMIIn_RX_ER hold time	t_{RRERH}	2.5	—	—		

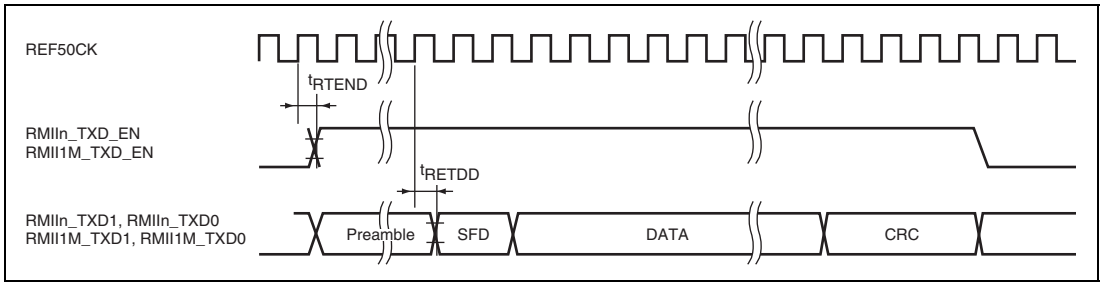


Figure 41.73 RMII Transmission Timing

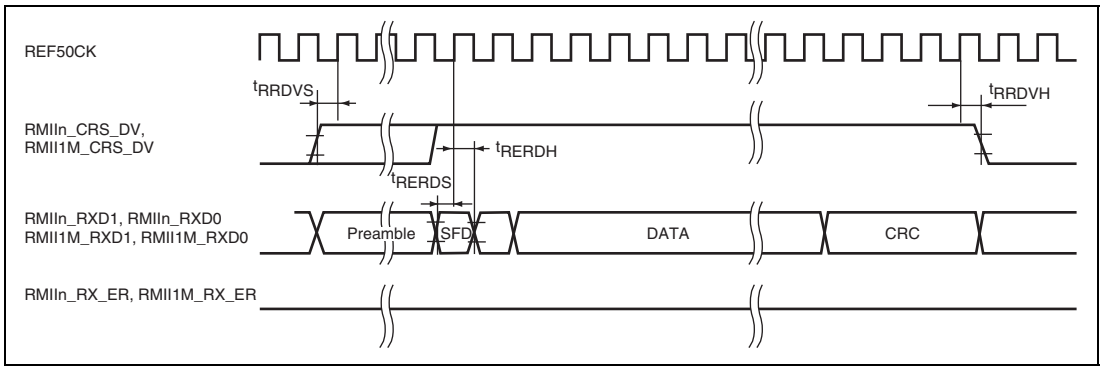


Figure 41.74 RMII Reception Timing (Normal Operation)

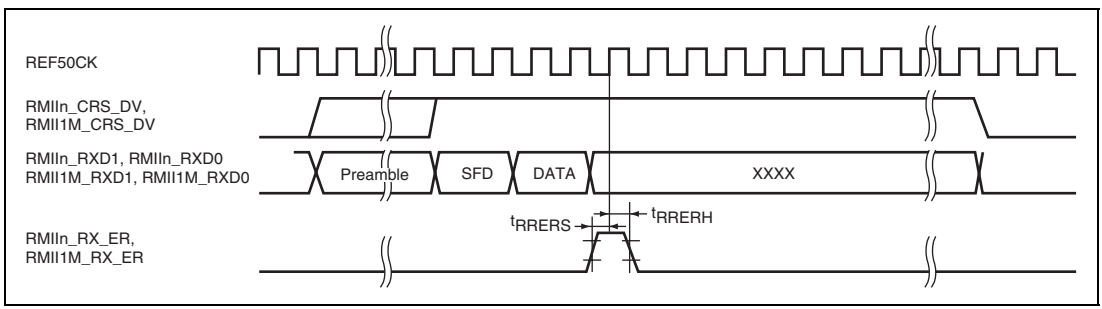


Figure 41.75 RMII Reception Timing (Case When Error Occurs)

41.23 TMU

Table 41.46 TMU Signal Timing

Conditions: Common temperature and voltage conditions

Item	Symbol	Min.	Typ.	Max.	Unit	Figures
TCLK clock cycle	t_{TCLKCY}	4	—	16.37	t_{CYC}	Figure 41.76

Note: * t_{CYC} is for one cycle of the IO clock (clkp).

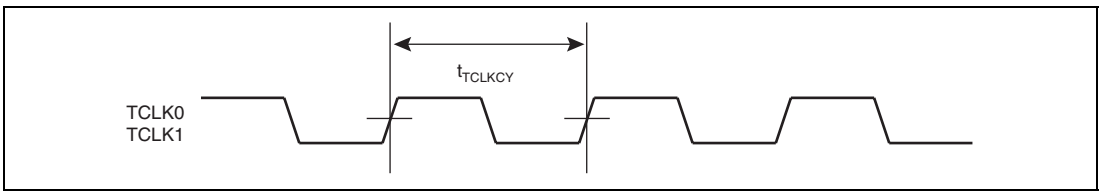


Figure 41.76 TMU Signal Timing

41.24 Multifunction Timer Pulse Unit 2 Timing

Table 41.47 Multifunction Timer Pulse Unit 2 Timing

Conditions: Common temperature and voltage conditions

Item	Symbol	Min.	Max.	Unit	Figures
Output compare output delay time	t_{TOCD}	—	20	ns	Figure 41.77
Input capture input setup time	t_{TICS}	20	—	ns	
Timer input setup time	t_{TCKS}	20	—	ns	Figure 41.78
Timer clock pulse width (single edge)	$t_{TCKWH/L}$	1.5	—	t_{pcyc}	
Timer clock pulse width (both edges)	$t_{TCKWH/L}$	2.5	—	t_{pcyc}	
Timer clock pulse width (phase count mode)	$t_{TCKWH/L}$	2.5	—	t_{pcyc}	

Note: t_{pcyc} indicates a peripheral clock frequency.

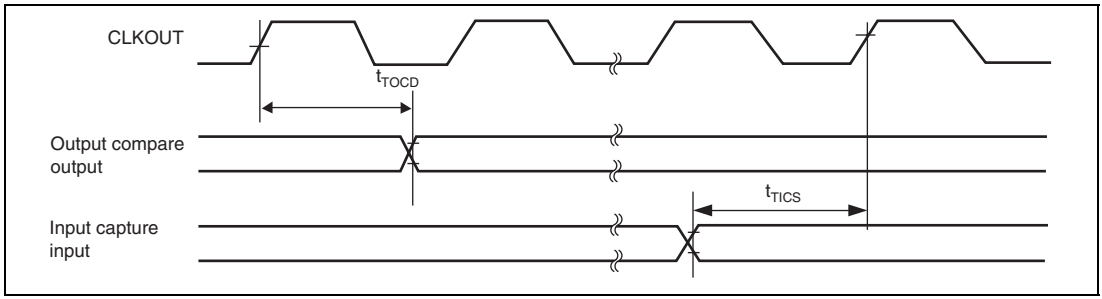


Figure 41.77 Pulse I/O Timing

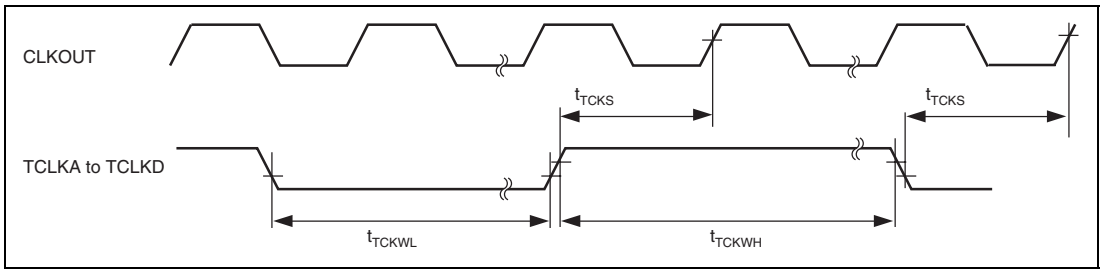


Figure 41.78 Clock Input Timing

41.25 A/D Converter Timing

Table 41.48 A/D Converter Timing

Conditions: Common temperature and voltage conditions

Module	Item		Symbol	Min.	Max.	Unit	Figures
A/D converter	Trigger input setup time	S clock : P clock = 1:1	t_{TRGS}	17	—	ns	Figure 41.79
		S clock : P clock = 2:1		$t_{cyc} + 17$	—		
		S clock : P clock = 4:1		$3 \times t_{cyc} + 17$	—		

Note: S = clks1, P = clkp1

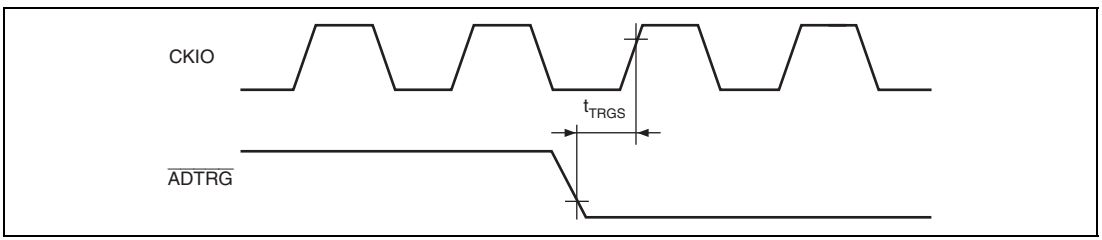


Figure 41.79 Timing of External Trigger Input for the A/D Converter

41.26 A/D Converter Characteristics

Table 41.49 A/D Converter Characteristics

Conditions: Common temperature and voltage conditions

Item	Min.	Typ.	Max.	Unit
Resolution	10	10	10	Bit
Conversion time	6	—	—	µs
Analog input capacitance	—	—	20	pF
Permissible signal-source impedance	—	—	5	kΩ
Nonlinearity error	—	—	±5.0*	LSB
Offset error	—	—	±2.0*	LSB
Full-scale error	—	—	±2.0*	LSB
Quantization error	—	—	±0.5*	LSB
Absolute accuracy	—	—	±5.0	LSB

Note * Reference values

41.27 Multi Media Card Interface (MMC)

Table 41.50 MMC Signal Timing

Conditions: Common temperature and voltage conditions, CL = Max.30 pF

Item	Symbol	Min.	Max.	Unit	Figures
MMC_CLK clock cycle	t_{MMCCYC}	$2 \times t_{PCYC}$	—	ns	Figure 41.80
MMC_CMD output data set up time	$t_{MMCCMDS}$	4	—	ns	
MMC_CMD output data hold time	$t_{MMCCMDH}$	4	—	ns	
MMC_D output data set up time	$t_{MMCDADS}$	4	—	ns	
MMC_D output data hold time	$t_{MMCDADH}$	4	—	ns	
MMC_CMD input data set up time	t_{MMCCMS}	5	—	ns	
MMC_CMD input data hold time	t_{MMCCMH}	2	—	ns	
MMC_D input data set up time	t_{MMCDAS}	5	—	ns	
MMC_D input data hold time	t_{MMCDAH}	2	—	ns	

Note: t_{PCYC} is for one cycle of clks1.

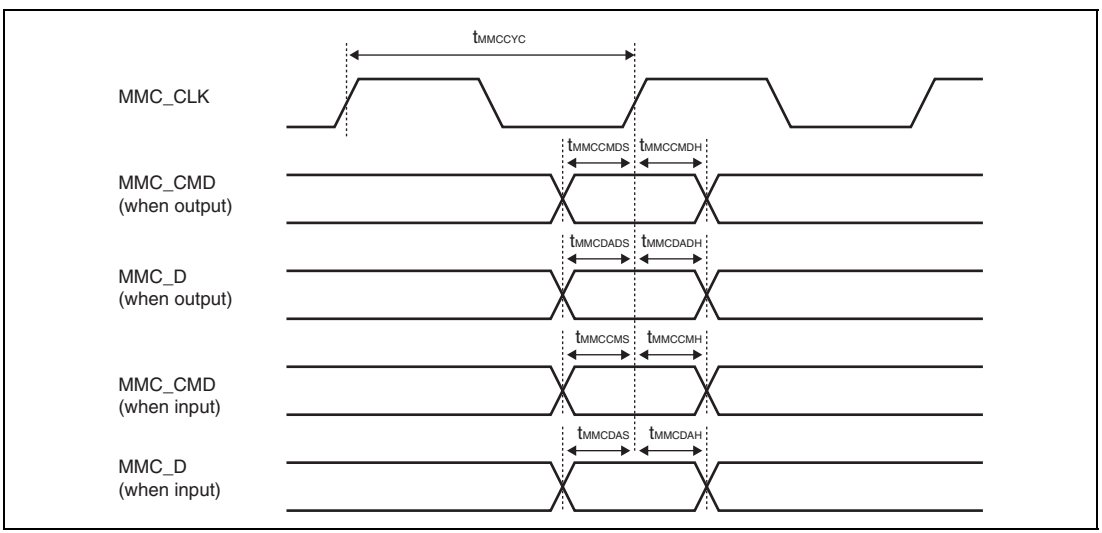


Figure 41.80 MMC Signal Timing

41.28 NAND Type Flash Memory Controller Timing

Table 41.51 NAND Type Flash Memory Interface Timing

Conditions: Common temperature and voltage conditions

Item	Symbol	Min.	Max.	Unit	Figures
Command output setup time	t_{NCDS}	$2 \times t_{fcyc}$ - 10	—	ns	Figures 41.81 and 41.85
Command output hold time	t_{NCDH}	$1.5 \times t_{fcyc}$ - 5	—	ns	
Data output setup time	t_{NDOS}	$0.5 \times t_{wfcyc}$ - 5	—	ns	Figures 41.81, 41.82, 41.84 and 41.85
Data output hold time	t_{NDOH}	$0.5 \times t_{wfcyc}$ - 10	—	ns	
Command to address transition time 1	t_{NCDAD1}	$1.5 \times t_{fcyc}$ - 10	—	ns	Figures 41.81 and 41.82
Command to address transition time 2	t_{NCDAD2}	$2 \times t_{fcyc}$ - 10	—	ns	Figure 41.82
$\overline{FWE}$ cycle time	t_{NWC}	T_{wfcyc} - 5	—	ns	Figures 41.82 and 41.84
$\overline{FWE}$ low pulse width	t_{NWP}	$0.5 \times t_{wfcyc}$ - 5	—	ns	Figures 41.81, 41.82, 41.84 and 41.85
$\overline{FWE}$ high pulse width	t_{NWH}	$0.5 \times t_{wfcyc}$ - 5	—	ns	Figures 41.82 and 41.84
Address to ready/busy transition time	t_{NADRB}	—	$32 \times t_{pcyc}$	ns	Figures 41.82 and 41.83
Command to ready/busy transition time	t_{NCDRB}	—	$10 \times t_{pcyc}$	ns	Figures 41.82 and 41.83
Ready/busy to data read transition time 1	t_{NRBDR1}	$1.5 \times t_{fcyc}$	—	ns	Figure 41.83
Ready/busy to data read transition time 2	t_{NRBDR2}	$32 \times t_{pcyc}$	—	ns	
$\overline{FRE}$ cycle time	t_{NSCC}	T_{wfcyc} - 5	—	ns	
$\overline{FRE}$ low pulse width	t_{NSP}	$0.5 \times t_{wfcyc}$ - 5	—	ns	Figures 41.83 and 41.85
$\overline{FRE}$ high pulse width	t_{NSPH}	$0.5 \times t_{wfcyc}$ - 5	—	ns	Figure 41.83
Read data setup time	t_{NRDS}	16	—	ns	Figures 41.83 and 41.85
Read data hold time	t_{NRDH}	5	—	ns	Figures 41.83 and 41.85
Data write setup time	t_{NDWS}	$32 \times t_{pcyc}$	—	ns	Figure 41.84

Item	Symbol	Min.	Max.	Unit	Figures
Command to status read transition time	t_{NCDSR}	$4 \times t_{fcyc}$	—	ns	Figure 41.85
Command output off to status read transition time	t_{NCDFSR}	$3.5 \times t_{fcyc}$	—	ns	
Status read setup time	t_{NSTS}	$2.5 \times t_{fcyc}$	—	ns	
$\overline{FCE}$ output setup time	t_{NCES}	$8 \times t_{pcyc}$	—	ns	Figure 41.81
$\overline{FCE}$ output hold time	t_{NCEH}	t_{pcyc}	—	ns	Figure 41.84
$\overline{FCE}$ output access time	t_{NCEA}	$6 \times t_{pcyc}$	—	ns	Figure 41.83
$\overline{FCE}$ output high hold time	t_{NCEOH}	$2 \times t_{pcyc}$	—	ns	

Notes t_{fcyc} is for one cycle of FCLK.

t_{wfcyc} is for one cycle of FCLK when the NANDWF bit is set to 0 and for two cycles of FCLK when the NANDWF bit is set to 1.

t_{pcyc} is for one cycle of a peripheral clock (Pϕ).

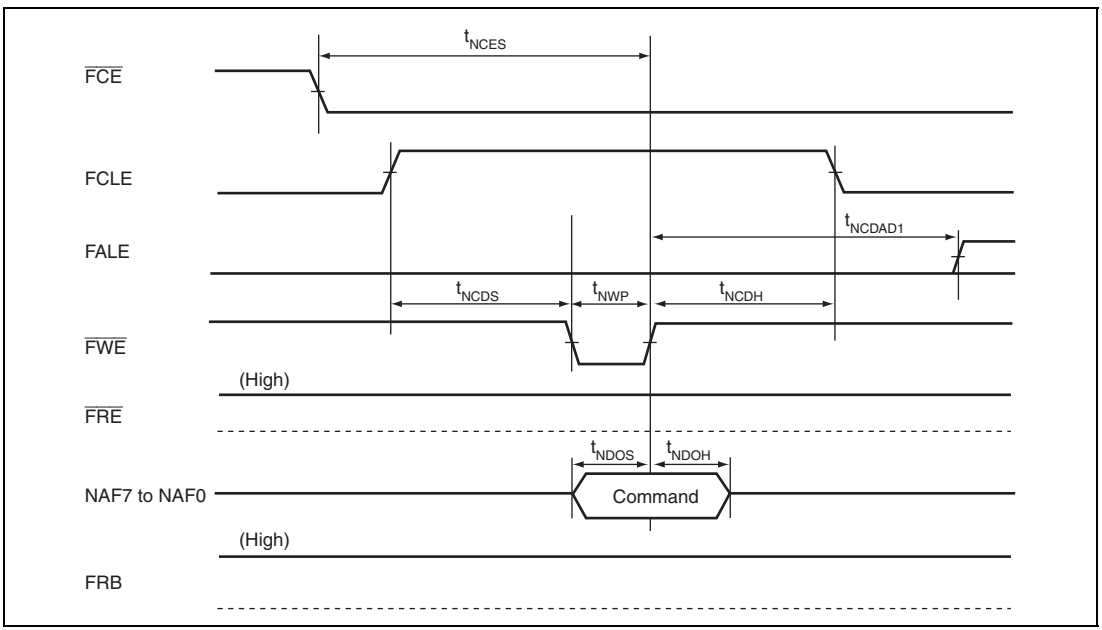


Figure 41.81 NAND Type Flash Memory Command Issuance Timing

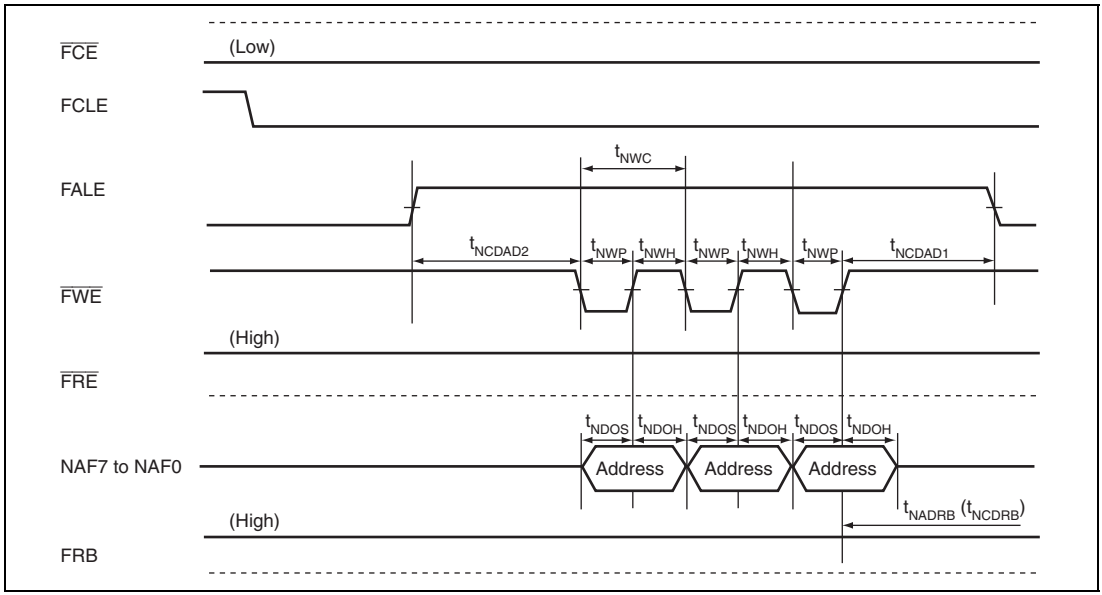


Figure 41.82 NAND Type Flash Memory Address Issuance Timing

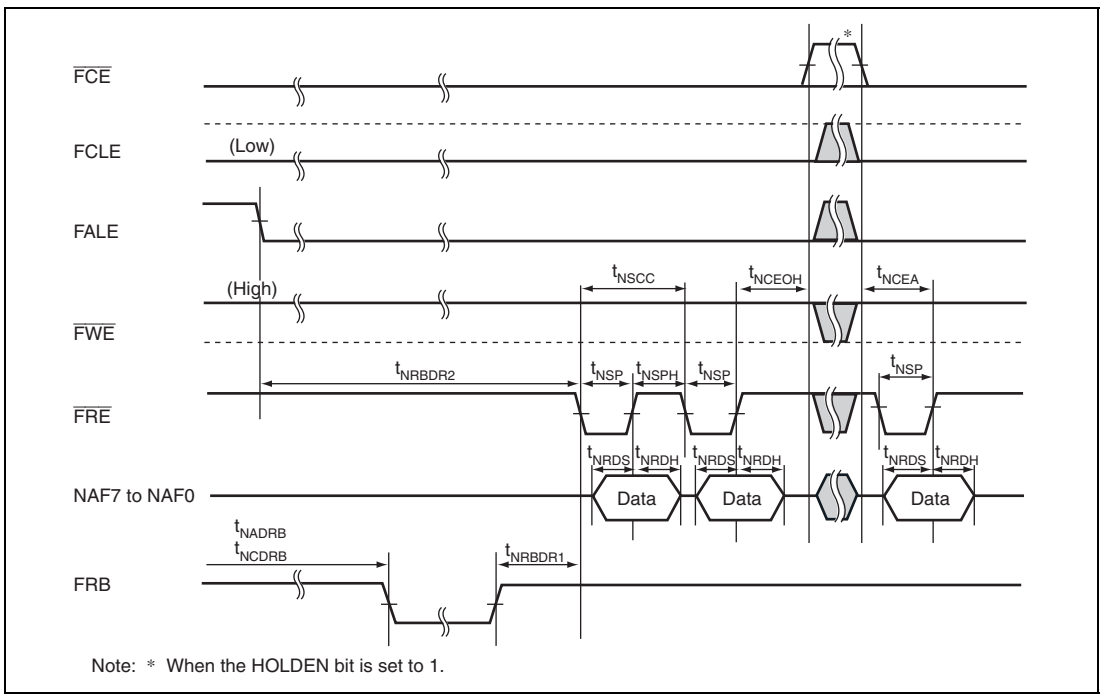


Figure 41.83 NAND Type Flash Memory Data Read Timing

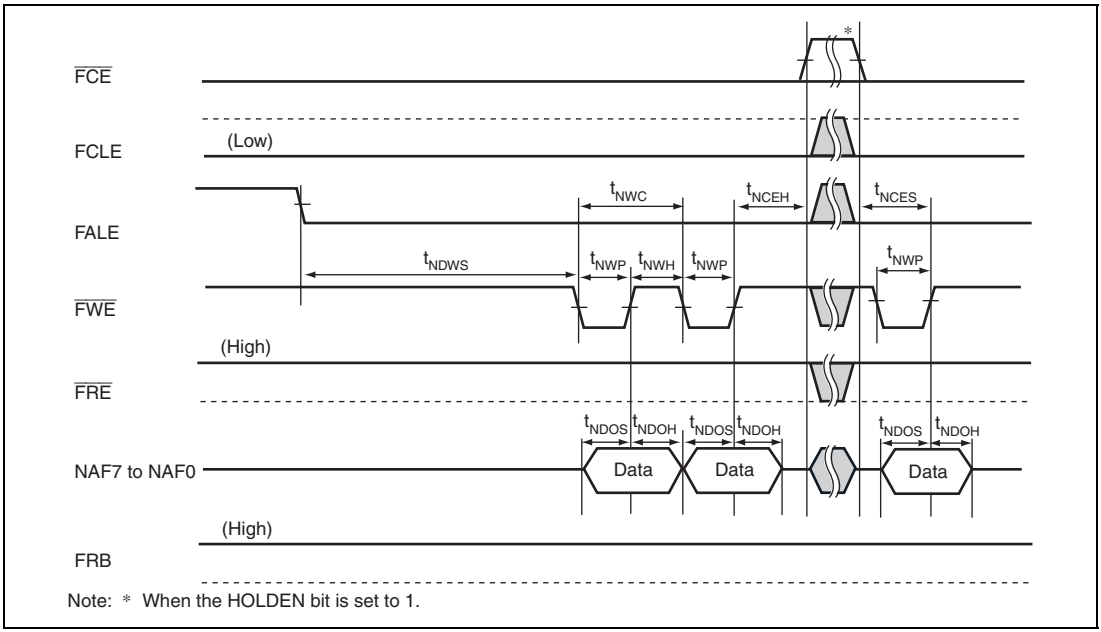


Figure 41.84 NAND Type Flash Memory Data Write Timing

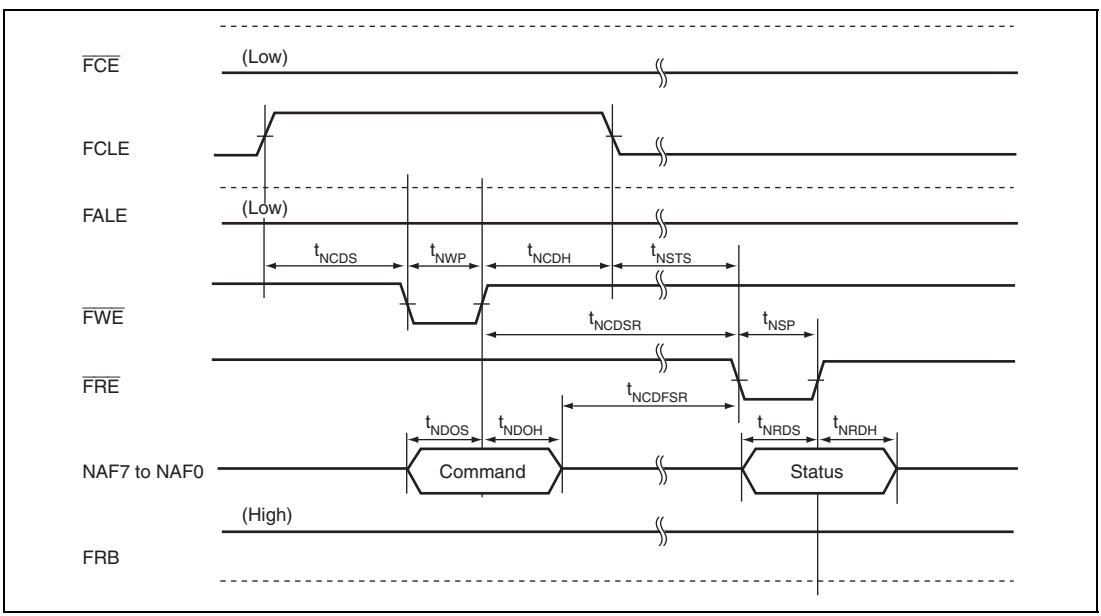


Figure 41.85 NAND Type Flash Memory Status Read Timing

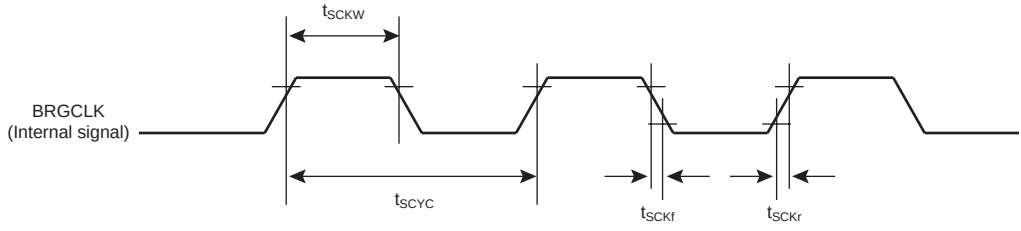
41.29 High Speed Serial Communication Interface with FIFO (HSCIF)

Table 41.52 HSCIF Signal Timing

Conditions: Common temperature and voltage conditions

Item	Symbol	Min.	Typ.	Max.	Unit	Figures
Input clock cycle (asynchronous)	t_{SCYC}	4	—	—	t_{CYC}	Figure 41.86
Input clock pulse width	t_{SCKW}	0.4	—	0.6	t_{SCYC}	
Input clock rise time	t_{SCKr}	—	—	0.8	t_{CYC}	
Input clock fall time	t_{SCKf}	—	—	0.8	t_{CYC}	

Note: tcyc indicates the period of one cycle of the SHwy-bus frequency (clks).



Note: The output signal (BRGCLK) of the external baud rate generator (BRG) is an HSCIF internal signal. Set the input clock signal and the DL and CKS registers of the BRG so that the BRGCLK signal can satisfy the above specification.

Figure 41.86 Input Clock Timing

41.30 I/O Port Timing

Table 41.53 I/O Port Timing

Conditions: Common temperature and voltage conditions

Item	Symbol	Min.	Max.	Unit	Figures
Output data delay time	t_{PORTD}	—	100	ns	Figure 41.87
Input data setup time	t_{PORTS}	100	—	ns	
Input data hold time	t_{PORTH}	100	—	ns	

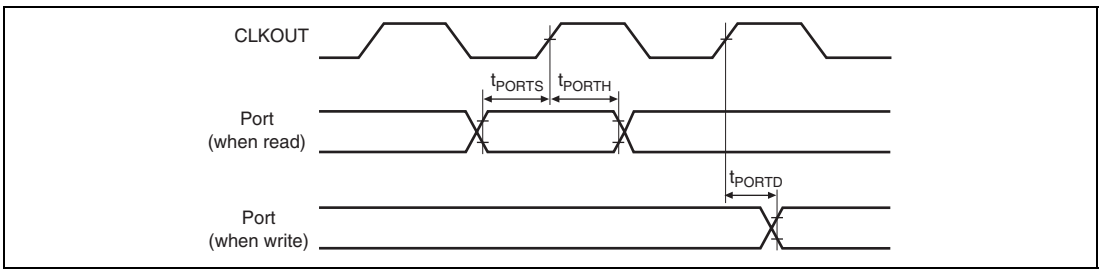


Figure 41.87 I/O Port Timing

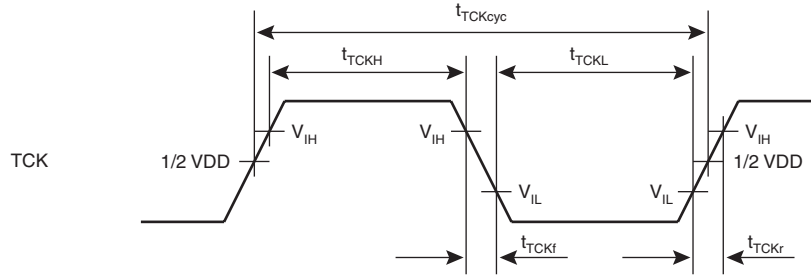
41.31 H-UDI

Table 41.54 H-UDI Timing

Conditions: Common temperature and voltage conditions, CL = 30 pF

Item	Symbol	Min.	Typ.	Max.	Unit	Figures
Input clock cycle	t_{TCKcyc}	50*	—	—	ns	Figure 41.88
Input clock pulse width (high level)	t_{TCKH}	15	—	—	ns	
Input clock pulse width (low level)	t_{TCKL}	15	—	—	ns	
Input clock rise time	t_{TCKr}	—	—	10	ns	
Input clock fall time	t_{TCKf}	—	—	10	ns	
TDI/TMS setup time	t_{TDIS}	15	—	—	ns	Figure 41.89
TDI/TMS hold time	t_{TDIH}	15	—	—	ns	
TDO output delay time	t_{TDO}	0	—	14	ns	

Note: The cycle is 500 ns (2 MHz) during boundary scan operation.



Note: When the clock is input from the TCK pin.

Figure 41.88 TCK Input Timing

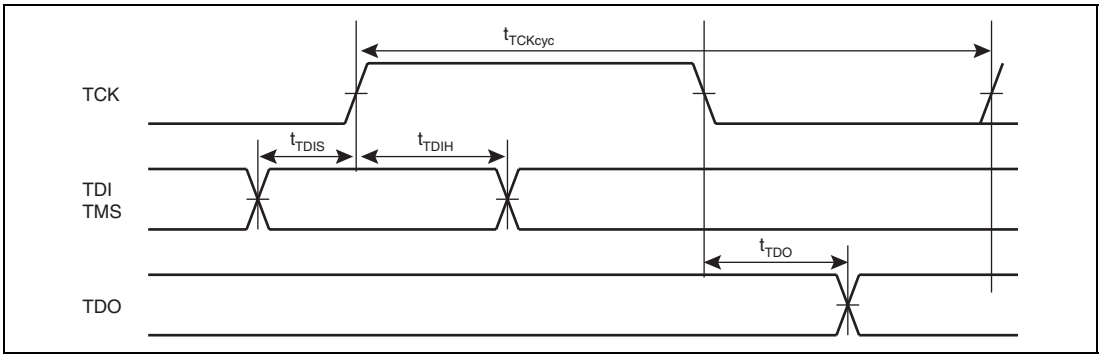


Figure 41.89 H-UDI Data Transfer Timing

41.32 Renesas Quad Serial Peripheral Interface Timing

Table 41.55 Renesas Quad Serial Peripheral Interface Timing

Conditions: Common temperature and voltage conditions

Item	Symbol	Min.	Max.	Unit	Figures
QSPCLK clock cycle	t_{QScyc}	2	4080	t_{cyc}	Figure 41.90
QSPCLK clock high-level pulse width	t_{SPCKWH}	0.4	—	t_{QScyc}	
QSPCLK clock low-level pulse width	t_{SPCKWL}	0.4	—	t_{QScyc}	
Data input setup time	t_{SU}	7	—	ns	Figures 41.91 and 41.92
Data input hold time	t_{H}	0.0	—	ns	
SSL setup time	t_{LEAD}	1.5	8.5	t_{QScyc}	
SSL hold time	t_{LAG}	1	8	t_{QScyc}	
Data output delay time	t_{OD}	—	10.0	ns	
Data output hold time	t_{OH}	-1.4	—	ns	
Continuous transmit delay time	t_{TD}	1	8	t_{QScyc}	

Note: t_{cyc} is for one cycle of clks1.

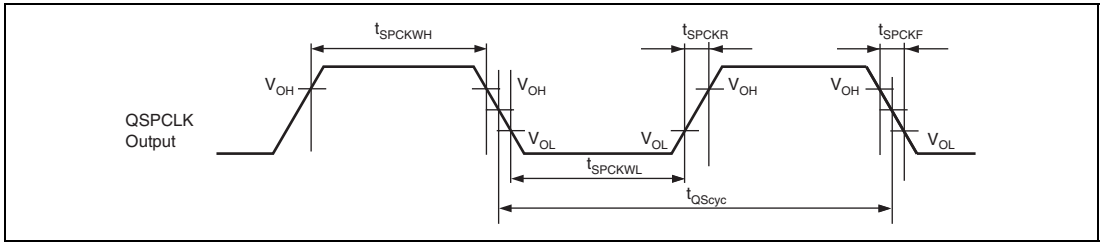


Figure 41.90 Clock Timing

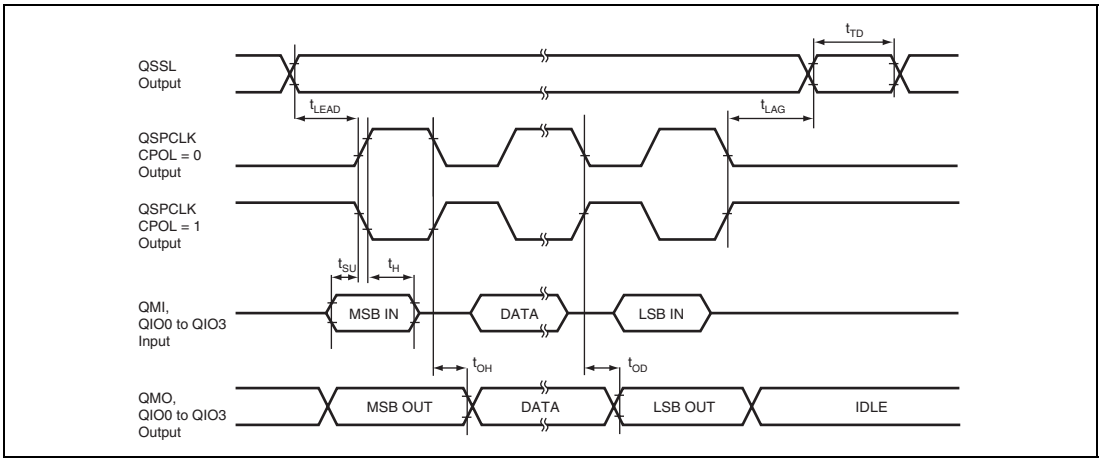


Figure 41.91 Transmit/Receive Timing (CPHA = 0)

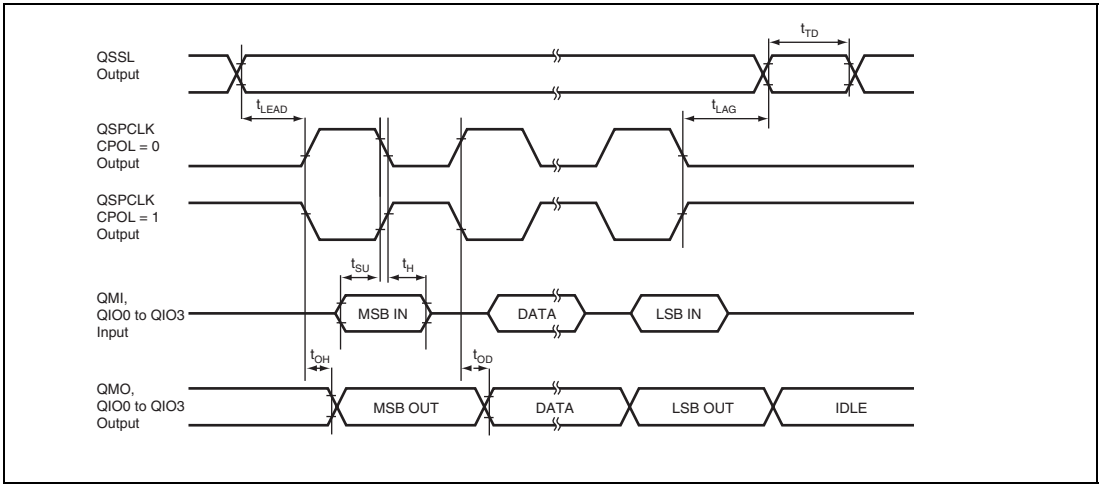


Figure 41.92 Transmit/Receive Timing (CPHA = 1)

41.33 MIMLB

Table 41.56 MediaLB Interface Timing

Conditions: VCCQ = 3.3 V ± 0.3 V, Ta = -40 to 85 °C, GND = VSS = 0 V, CL = 10 pF

Item	Symbol	Min.	Typ.	Max.	Unit	Figures
MLB_CLK cycle	fCK	45.056	49.152	49.2544	MHz	Figure 41.93
MLB_CLK cycle time	tCK	—	20.3	—	ns	
MLB_CLK high period	tCH	9.7	10.6	—	ns	
MLB_CLK low period	tCL	6.5	7.7	—	ns	
MLB_SIG output delay time (clock signal rising)	tDR	—	—	7.5	ns	
MLB_SIG output delay time (clock signal falling)	tDF	0	—	—	ns	
MLB_SIG input setup time	tSD	2.7	—	—	ns	
MLB_SIG input hold time	tHD	0	—	—	ns	
MLB_DAT output delay (clock signal rising)	tDR	—	—	7.5	ns	
MLB_DAT output delay (clock signal falling)	tDF	0	—	—	ns	
MLB_DAT input setup time	tSD	2.7	—	—	ns	
MLB_DAT input hold time	tHD	0	—	—	ns	

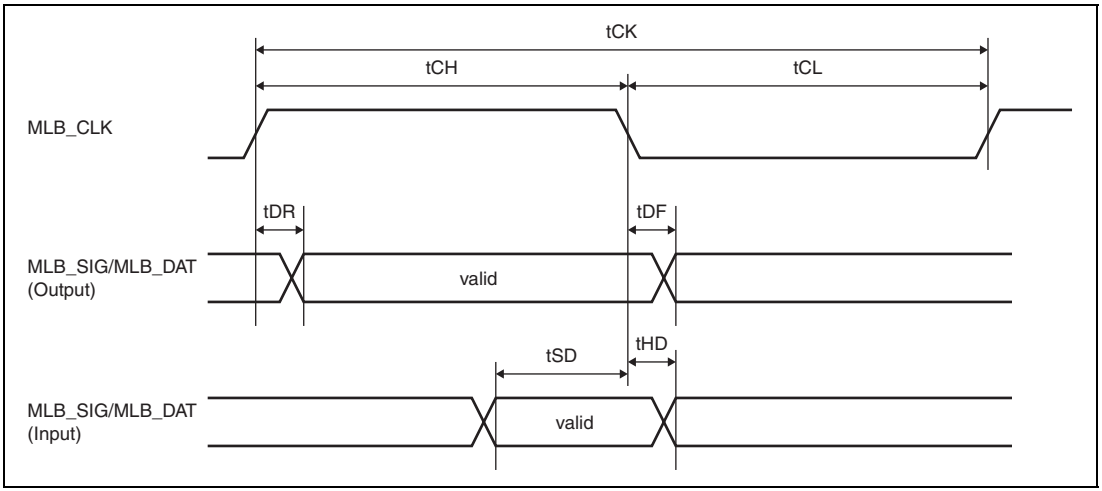


Figure 41.93 MediaLB Interface Timing

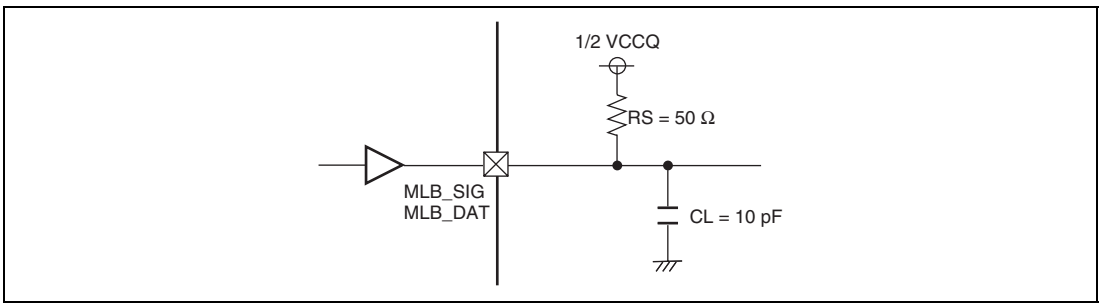


Figure 41.94 Conditions for Measuring Output Delay for MLB_SIG and MLB_DAT

41.34 Usage Notes

41.34.1 USB I/O Buffer

Connect a USB cable with impedance in the prescribed range to the USB I/O buffer of this LSI chip.

For mounting of this buffer on an actual system, refer to the hardware manual, evaluate the system, and determine whether the board design guideline is applicable or not.

41.34.2 I^2C I/O Buffer

The I^2C I/O buffer does not support 5-V input tolerance. Do not apply 5 V to the buffer.

For the open-drain buffers of this LSI, the power supply for the I^2C bus and VCCQ should be in the same state. Applying only the I^2C bus power supply, i.e. only ViH, which is the case when VCCQ is off, may damage the LSI.

41.34.3 Measure against Latchup

Exposing this IC to excessive external noise may lead to latchup. Latchup causes a large current to flow through the IC and the emission of heat and smoke by the board on which it is mounted. To deal with latchup, place a current limiter in the power system.

41.34.4 Input of Intermediate Voltages

Note that the input of intermediate voltages to pins* in use as input pins may affect the long-term reliability of the product. To leave input pins open-circuit when they are not in use, connect them to the power supply via external pull-up resistors or internal programmable pull-up resistors.

Note: * This includes I/O pins set as inputs as well as input-only pins.

41.34.5 Regarding the AC Characteristics

Inputs to this LSI chip are basically in synchronization with a clock. Be sure to secure the setup and hold times for all inputs unless there is a specific note stating that this is not required.

41.35 Notes on Designing the Board

41.35.1 Notes on Using a Crystal Oscillator

Place crystal oscillators and the capacitors as close to the EXTAL and XTAL pins as is possible. Do not place other signal lines across the signal lines for these pins. If this is ignored, induction may prevent accurate oscillation.

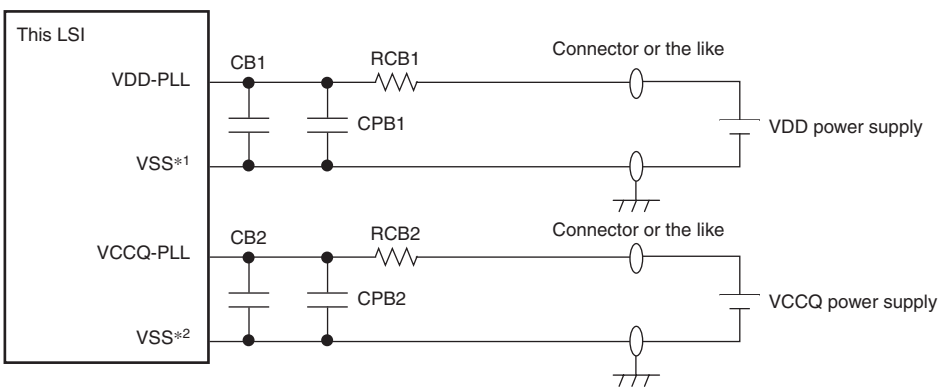
Determine the load capacitance and the damping resistance near the crystal oscillators on the bases of consultation with the manufacturer of the oscillator and evaluation.

41.35.2 Note on the Input of External Clock Signals through the EXTAL Pin

Do not connect the XTAL pin to anything.

41.35.3 Note on Using the PLL Oscillation Circuit

Separate VDD for the PLL from the main VDD lines and VCCQ for the PLL from the main VCCQ lines on the board. Place resistors RCB1 and RCB2 and bypass capacitors CPB and CB to form noise filters near these pins.



Recommended values: RCB1 = RCB2 = 4.7 Ω
CPB1 = CPB2 = 1 μF (capacitor)
CB1 = CB2 = 0.1 μF

Notes: 1. Proximate VSS for a BGA440 is K12 or the like
2. Either of proximate VSS: A BGA440 for C23, C24, B24, or the like

Figure 41.95 Note on Using PLL Oscillation Circuit

41.36 Measurement Conditions

41.36.1 Common Temperature Condition

Unless otherwise specified, the temperature condition is the common temperature condition.

Ta = -20 to 75 °C (regular specifications), -40 to 85 °C (wide-range specifications)

Ta-max = 75 °C (regular specifications), Ta-max = 85 °C (wide-range specifications)

41.36.2 Conditions for Measuring AC Characteristics

Unless otherwise specified, the conditions for measuring the AC characteristics are as follows.

(1) Temperature Voltage Common Conditions

VDD = VDD-PLL = AV12 = 1.15 to 1.3 V

VCCQ = VCC = VCCQ-PLL = AV33

= AVCC = AVREF = 3.0 to 3.6 V

VSS = VSS-PLL = AG = AVSS = 0V

Measurements are under a common temperature condition.

(2) Conditions for Measuring AC Characteristics

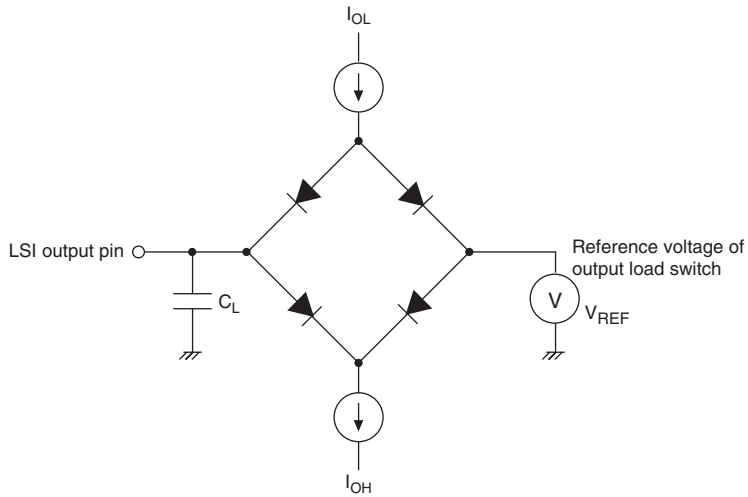
- I/O signal reference level: V*/2
- Input pulse level: VSS to V*
- Input rise and fall times: 1 ns

Note: * V: VCCQ = VCC = VCCQ-PLL = AV33 = AVCC = AVREF (VCCQ = 3.0 to 3.6 V)

Figure 41.96 shows the output load circuit for a 3.3-V I/O cell.

Unless otherwise specified, the current for an I/O cell is set to 6 mA.

If the table of AC characteristics specifies a load capacitance, this is set to the value in the table.



Notes: 1. C_L is the total value that includes the capacitance of measurement instruments, and is set 30 pF for each pin.
2. I_{OL} = 3 mA (the IIC pin), I_{OL} = 2 mA (all other pins), I_{OH} = -2 mA

Figure 41.96 Output Load Circuit

Appendix

A. Pin States (for the BGA440)

Pin Name	Power-on Reset*[1]	Manual Reset	Sleep	Software Standby	Deep Standby: While Stopped	Deep Standby: On Return*[2]
MCK0	L/O	O	O	K	K	O
$\overline{\text{MCK0}}$	L/O	O	O	K	K	O
MCKE	L	O	O	K	K	L
$\overline{\text{MCS}}$	Z/H	O	O	K	K	H*[13]
$\overline{\text{MWE}}$	Z/H	O	O	K	K	H*[13]
$\overline{\text{MRAS}}$	Z/H	O	O	K	K	H*[13]
$\overline{\text{MCAS}}$	Z/H	O	O	K	K	H*[13]
MA0	Z/L	O	O	K	K	L*[13]
MA1	Z/L	O	O	K	K	L*[13]
MA2	Z/L	O	O	K	K	L*[13]
MA3	Z/L	O	O	K	K	L*[13]
MA4	Z/L	O	O	K	K	L*[13]
MA5	Z/L	O	O	K	K	L*[13]
MA6	Z/L	O	O	K	K	L*[13]
MA7	Z/L	O	O	K	K	L*[13]
MA8	Z/L	O	O	K	K	L*[13]
MA9	Z/L	O	O	K	K	L*[13]
MA10	Z/L	O	O	K	K	L*[13]
MA11	Z/L	O	O	K	K	L*[13]
MA12	Z/L	O	O	K	K	L*[13]
MA13	Z/L	O	O	K	K	L*[13]
MBA0	Z/L	O	O	K	K	L*[13]
MBA1	Z/L	O	O	K	K	L*[13]
MBA2	Z/L	O	O	K	K	L*[13]
MDQ0	Z	IO	IO	K	K	Z
MDQ1	Z	IO	IO	K	K	Z

Pin Name	Power-on Reset*1	Manual Reset	Sleep	Deep Standby: Software Standby	Deep Standby: While Stopped	Deep Standby: On Return*2
MDQ2	Z	IO	IO	K	K	Z
MDQ3	Z	IO	IO	K	K	Z
MDQ4	Z	IO	IO	K	K	Z
MDQ5	Z	IO	IO	K	K	Z
MDQ6	Z	IO	IO	K	K	Z
MDQ7	Z	IO	IO	K	K	Z
MDQ8	Z	IO	IO	K	K	Z
MDQ9	Z	IO	IO	K	K	Z
MDQ10	Z	IO	IO	K	K	Z
MDQ11	Z	IO	IO	K	K	Z
MDQ12	Z	IO	IO	K	K	Z
MDQ13	Z	IO	IO	K	K	Z
MDQ14	Z	IO	IO	K	K	Z
MDQ15	Z	IO	IO	K	K	Z
MDQS0	Z	IO	IO	K	K	Z
$\overline{\text{MDQS0}}$	Z	IO	IO	K	K	Z
MDQS1	Z	IO	IO	K	K	Z
$\overline{\text{MDQS1}}$	Z	IO	IO	K	K	Z
MDM0	L/O	O	O	K	K	O
MDM1	L/O	O	O	K	K	O
MODT	L	O	O	K	K	L
$\overline{\text{MBKPRST}}$	I	I	I	I	I	I
$\overline{\text{MRESET}}$	O	O	O	K	K	K
SDBUP	I	I	I	I	I	I
SDSELF	ZU	P	P	K	K	K
BSMODE	I	I	I	I	I	I
EXTAL	I	I	I	I	I	I
XTAL	O	O	O	O	O	O
$\overline{\text{PRESET}}$	I	I	I	I	I	I
$\overline{\text{PRESETOUT}}$	L/O	P	P	H*15 K*15	H*15 K*15	O*18 K*8

Pin Name			Power-on Reset*[1]	Manual Reset	Sleep	Software Standby	Deep Standby: While Stopped	Deep Standby: On Return*[2]
A0	In boot mode	0, 1	L/O	P	P	K	K	O*[18] K*[6]
		2, 3, 4, 5, 6	L	P	P	K	K	K
A1	In boot mode	0, 1	L/O	P	P	K	K	O*[18] K*[6]
		2, 3, 4, 5, 6	L	P	P	K	K	K
A2	In boot mode	0, 1	L/O	P	P	K	K	O*[18] K*[6]
		2, 3, 4, 5, 6	L	P	P	K	K	K
A3	In boot mode	0, 1	L/O	P	P	K	K	O*[18] K*[6]
		2, 3, 4, 5, 6	L	P	P	K	K	K
A4	In boot mode	0, 1	L/O	P	P	K	K	O*[18] K*[6]
		2, 3, 4, 5, 6	L	P	P	K	K	K
A5	In boot mode	0, 1	I/O	P	P	K	K	O*[3], *[18] K*[6]
		2, 3, 4, 5, 6	I/L	P	P	K	K	K
A6	In boot mode	0, 1	I/O	P	P	K	K	O*[3], *[18] K*[6]
		2, 3, 4, 5, 6	I/L	P	P	K	K	K
A7	In boot mode	0, 1	I/O	P	P	K	K	O*[3], *[18] K*[6]
		2, 3, 4, 5, 6	I/L	P	P	K	K	K
A8	In boot mode	0, 1	I/O	P	P	K	K	O*[3], *[18] K*[6]
		2, 3, 4, 5, 6	I/L	P	P	K	K	K
A9	In boot mode	0, 1	I/O	P	P	K	K	O*[3], *[18] K*[6]
		2, 3, 4, 5, 6	I/L	P	P	K	K	K

Pin Name			Power-on Reset*1	Manual Reset	Sleep	Software Standby	Deep Standby: While Stopped	Deep Standby: On Return*2
A10	In boot mode	0, 1	I/O	P	P	K	K	O*3, *18 K*6
		2, 3, 4, 5, 6	I/L	P	P	K	K	K
A11	In boot mode	0, 1	I/O	P	P	K	K	O*3, *18 K*6
		2, 3, 4, 5, 6	I/L	P	P	K	K	K
A12	In boot mode	0, 1	I/O	P	P	K	K	O*3, *18 K*6
		2, 3, 4, 5, 6	I/L	P	P	K	K	K
A13	In boot mode	0, 1	I/O	P	P	K	K	O*3, *18 K*6
		2, 3, 4, 5, 6	I/L	P	P	K	K	K
A14	In boot mode	0, 1	I/O	P	P	K	K	O*3, *18 K*6
		2, 3, 4, 5, 6	I/L	P	P	K	K	K
A15	In boot mode	0, 1	I/O	P	P	K	K	O*3, *18 K*6
		2, 3, 4, 5, 6	I/L	P	P	K	K	K
A16	In boot mode	0, 1	I/O	P	P	K	K	O*3, *18 K*6
		2, 3, 4, 5, 6	I/L	P	P	K	K	K
A17	In boot mode	0, 1	I/O	P	P	K	K	O*3, *18 K*6
		2, 3, 4, 5, 6	I/L	P	P	K	K	K
A18	In boot mode	0, 1	I/O	P	P	K	K	O*3, *18 K*6
		2, 3, 4, 5, 6	I/L	P	P	K	K	K

Pin Name			Power-on Reset*[1]	Manual Reset	Sleep	Software Standby	Deep Standby: While Stopped	Deep Standby: On Return*[2]
A19	In boot mode	0, 1	I/O	P	P	K	K	O*[3], *[18]
								K*[6]
		2, 3, 4, 5, 6	I/L	P	P	K	K	K
A20	In boot mode	0, 1	L/O	P	P	K	K	O*[18]
								K*[6]
		2, 3, 4, 5, 6	L	P	P	K	K	K
A21	In boot mode	0, 1	L/O	P	P	K	K	O*[18]
								K*[6]
		2, 3, 4, 5, 6	L	P	P	K	K	K
A22	In boot mode	0, 1	L/O	P	P	K	K	O*[18]
								K*[6]
		2, 3, 4, 5, 6	L	P	P	K	K	K
A23	In boot mode	0, 1	L/O	P	P	K	K	O*[18]
								K*[6]
		2, 3, 4, 5, 6	L	P	P	K	K	K
A24			I	P	P	K	K	K
A25			I	P	P	K	K	K
D0	In boot mode	0, 1	Z/IO	P	P	K	K	IO*[18]
								K*[6]
		2, 4, 5	Z/IO	P*[16]	P	K	K	IO*[18]
								K*[7]
		3, 6	Z	P*[16]	P	K	K	K
D1	In boot mode	0, 1	Z/IO	P	P	K	K	IO*[18]
								K*[6]
		2, 4, 5	Z/IO	P*[16]	P	K	K	IO*[18]
								K*[7]
		3, 6	Z	P*[16]	P	K	K	K

Pin Name			Power-on Reset*[1]	Manual Reset	Sleep	Software Standby	Deep Standby: While Stopped	Deep Standby: On Return*[2]
D2	In boot mode	0, 1	Z/IO	P	P	K	K	IO*[18]
								K*[6]
		2, 4, 5	Z/IO	P*[16]	P	K	K	IO*[18]
								K*[7]
		3, 6	Z	P*[16]	P	K	K	K
D3	In boot mode	0, 1	Z/IO	P	P	K	K	IO*[18]
								K*[6]
		2, 4, 5	Z/IO	P*[16]	P	K	K	IO*[18]
								K*[7]
		3, 6	Z	P*[16]	P	K	K	K
D4	In boot mode	0, 1	Z/IO	P	P	K	K	IO*[18]
								K*[6]
		2, 4, 5	Z/IO	P*[16]	P	K	K	IO*[18]
								K*[7]
		3, 6	Z	P*[16]	P	K	K	K
D5	In boot mode	0, 1	Z/IO	P	P	K	K	IO*[18]
								K*[6]
		2, 4, 5	Z/IO	P*[16]	P	K	K	IO*[18]
								K*[7]
		3, 6	Z	P*[16]	P	K	K	K
D6	In boot mode	0, 1	Z/IO	P	P	K	K	IO*[18]
								K*[6]
		2, 3, 4	Z/IO	P*[16]	P	K	K	IO*[18]
								K*[7]
		5, 6	Z	P*[16]	P	K	K	K

Pin Name			Power-on Reset*[1]	Manual Reset	Sleep	Software Standby	Deep Standby: While Stopped	Deep Standby: On Return*[2]
D7	In boot mode	0, 1	Z/IO	P	P	K	K	IO*[18]
								K*[6]
		2, 3, 4	Z/IO	P*[16]	P	K	K	IO*[18]
								K*[7]
		5, 6	Z	P*[16]	P	K	K	K
D8	In boot mode	0	Z	P	P	K	K	K
							L*[10]	L*[10]
							O*[10a]	O*[10a]
		1	Z/IO	P	P	K	K	IO*[18]
								K*[6]
		2, 4, 5	Z/IO	P*[16]	P	K	K	IO*[18]
								K*[7]
		3, 6	Z	P*[16]	P	K	K	K
D9	In boot mode	0	Z	P	P	K	K	K
							O*[11]	O*[11]
		1	Z/IO	P	P	K	K	IO*[18]
								K*[6]
		2, 4, 5	Z/IO	P*[16]	P	K	K	IO*[18]
								K*[7]
		3, 6	Z	P*[16]	P	K	K	K
D10	In boot mode	0	Z	P	P	K	K	K
							O*[11]	O*[11]
		1	Z/IO	P	P	K	K	IO*[18]
								K*[6]
		2, 3	Z/IO	P*[16]	P	K	K	IO*[18]
								K*[7]
		4, 5, 6	Z	P*[16]	P	K	K	K

Pin Name			Power-on Reset*[1]	Manual Reset	Sleep	Software Standby	Deep Standby: While Stopped	Deep Standby: On Return*[2]
D11	In boot mode	0	Z	P	P	K	K	K
							O*[11]	O*[11]
		1	Z/IO	P	P	K	K	IO*[18]
								K*[6]
		2, 3	Z/IO	P*[16]	P	K	K	IO*[18]
								K*[7]
		4, 5, 6	Z	P*[16]	P	K	K	K
D12	In boot mode	0	Z	P	P	K	K	K
							O*[11]	O*[11]
		1	Z/IO	P	P	K	K	IO*[18]
								K*[6]
		2	Z/IO	P*[16]	P	K	K	IO*[18]
								K*[7]
		3, 4, 5, 6	Z	P*[16]	P	K	K	K
D13	In boot mode	0	Z	P	P	K	K	K
							O*[11]	O*[11]
		1	Z/IO	P	P	K	K	IO*[18]
								K*[6]
		2	Z/IO	P*[16]	P	K	K	IO*[18]
								K*[7]
		3, 4, 5, 6	Z	P*[16]	P	K	K	K
D14	In boot mode	0	Z	P	P	K	K	K
							I*[11]	I*[11]
		1	Z/IO	P	P	K	K	IO*[18]
								K*[6]
		2	Z/IO	P*[16]	P	K	K	IO*[18]
								K*[7]
		3, 4, 5, 6	Z	P*[16]	P	K	K	K

Pin Name			Power-on Reset*[1]	Manual Reset	Sleep	Software Standby	Deep Standby: While Stopped	Deep Standby: On Return*[2]
D15	In boot mode	0, 2, 3, 4, 5, 6	Z	P	P	K	K	K
		1	Z/IO	P	P	K	K	IO*[18]
								K*[6]
CLKOUT			O	P	P	L*[14]	L*[14]	O*[18]
						K*[14]	K*[14]	K*[9]
$\overline{BS}$	In boot mode	0, 1	H/O	P	P	K	K	O*[18]
								K*[6]
		2, 3, 4, 5, 6	H	P	P	K	K	K
$\overline{CS0}$	In boot mode	0, 1	H/O	P	P	K	K	O*[18]
								K*[6]
		2, 3, 4, 5, 6	H	P	P	K	K	K
$\overline{CS1}$/A26			H/O	P	P	K	K	K
$\overline{EX\_CS0}$			ZU	P	P	K	K	K
$\overline{EX\_CS1}$			ZU	P	P	K	K	K
							O*[11]	O*[11]
$\overline{EX\_CS2}$			ZU	P	P	K	K	K
							L*[10b]	L*[10b]
$\overline{EX\_CS3}$			I	P	P	K	K	K
$\overline{EX\_CS4}$			I	P	P	K	K	K
$\overline{EX\_CS5}$			ZU	P	P	K	K	K
$\overline{RD}$	In boot mode	0, 1	H/O	P	P	K	K	O*[18]
								K*[6]
		2, 3, 4, 5, 6	H	P	P	K	K	K
RD/$\overline{WR}$			ZU	P	P	K	K	K
							O*[11]	O*[11]
$\overline{WE0}$			H	P	P	K	K	K
$\overline{WE1}$			H	P	P	K	K	K
EX_WAIT0			I	P	P	K	K	K
EX_WAIT1			ZU	P	P	K	K	K
EX_WAIT2			ZU	P	P	K	K	K

Pin Name	Power-on Reset*1	Manual Reset	Sleep	Software Standby	Deep Standby: While Stopped	Deep Standby: On Return*2
DRACK0	ZU	P	P	K	K	K
					O*11	O*11
DREQ0	ZU	P	P	K	K	K
					O*11	O*11
DACK0	Z	P	P	K	K	K
					O*11	O*11
DREQ1	ZU	P	P	K	K	K
DACK1	ZU	P	P	K	K	K
$\overline{\text{TRST}}$	ZU	I	I	I	K	K*19
TCK	ZU	I	I	I	K	K*19
TMS	ZU	I	I	I	K	K*19
TDI	ZU	I	I	I	K	K*19
TDO	Z	O	O	K	K	K*19
MPMD	I	I	I	I	I	I
$\overline{\text{ASEBRK}}$/ACK	ZU	IO	IO	K*12	K*12	IO*19
NMI	I	I	I	I	I	I
IRQ0_A	ZU	P	P	K	K	K
				I*17	I*11, *17	I*11, *17
IRQ1_A	ZU	P	P	K	K	K
				I*17	I*11, *17	I*11, *17
IRQ2_A	ZU	P	P	K	K	K
				I*17	I*17	I*17
IRQ3_A	ZU	P	P	K	K	K
				I*17	I*17	I*17
SCIF_CLK_A	ZU	P	P	K	K	K
					I*11a	I*11a
					O*11b	O*11b
SCK0_A	ZU	P	P	K	K	K
					I*11a	I*11a
					O*11b	O*11b

Pin Name	Power-on Reset*1	Manual Reset	Sleep	Software Standby	Deep Standby: While Stopped	Deep Standby: On Return*2
RX0_A	ZU	P	P	K	K	K
					I*11	I*11
TX0_A	I	P	P	K	K	K
$\overline{\text{HCTS0\_A}}$	ZU	P	P	K	K	K
					I*11	I*11
$\overline{\text{HRTS0\_A}}$	ZU	P	P	K	K	K
					I*11a	I*11a
					O*11b	O*11b
HSCK0_A	ZU	P	P	K	K	K
					I*11	I*11
HRX0_A	ZU	P	P	K	K	K
					I*11	I*11
HTX0_A	ZU	P	P	K	K	K
					I*11	I*11
$\overline{\text{CTS0\_B}}$	ZU	P	P	K	K	K
$\overline{\text{RTS0\_B}}$	ZU	P	P	K	K	K
SCK1_B	ZU	P	P	K	K	K
RX1_B	ZU	P	P	K	K	K
TX1_B	ZU	P	P	K	K	K
$\overline{\text{CTS1\_B}}$	ZU	P	P	K	K	K
$\overline{\text{RTS1\_B}}$	ZU	P	P	K	K	K
SCK2_A	ZU	P	P	K	K	K
SD2_CLK_A	ZU	P	P	K	K	K
SD2_CMD_A	ZU	P	P	K	K	K
SD2_DAT0_A	ZU	P	P	K	K	K
SD2_DAT1_A	ZU	P	P	K	K	K
SD2_DAT2_A	ZU	P	P	K	K	K
SD2_DAT3_A	ZU	P	P	K	K	K
					O*11	O*11
SD2_CD_A	ZU	P	P	K	K	K
SD2_WP_A	ZU	P	P	K	K	K

Pin Name			Power-on Reset*1	Manual Reset	Sleep	Software Standby	Deep Standby: While Stopped	Deep Standby: On Return*2
REF125CK			Z	P	P	K	K	K
REF50CK			Z	P	P	K	K	K
DU0_DR0	In boot mode	0 to 3, 4, 5	Z	P	P	K	K	K
		6	Z/IO	P	P	K	K	IO
DU0_DR1	In boot mode	0 to 3, 4, 5	Z	P	P	K	K	K
		6	Z/IO	P	P	K	K	IO
DU0_DR2	In boot mode	0 to 3, 4, 5	Z	P	P	K	K	K
		6	Z/IO	P	P	K	K	IO
DU0_DR3	In boot mode	0 to 3, 4, 5	Z	P	P	K	K	K
		6	Z/IO	P	P	K	K	IO
DU0_DR4	In boot mode	0 to 3, 4, 5	Z	P	P	K	K	K
		6	Z/IO	P	P	K	K	IO
DU0_DR5	In boot mode	0 to 3, 4, 5	Z	P	P	K	K	K
		6	Z/IO	P	P	K	K	IO
DU0_DR6	In boot mode	0 to 3, 4, 5	Z	P	P	K	K	K
		6	Z/IO	P	P	K	K	IO
DU0_DR7	In boot mode	0 to 3, 4, 5	Z	P	P	K	K	K
		6	Z/IO	P	P	K	K	IO
DU0_DG0	In boot mode	0 to 3, 4, 5	Z	P	P	K	K	K
		6	Z/IO	P	P	K	K	IO
DU0_DG1	In boot mode	0 to 3, 4, 5	Z	P	P	K	K	K
		6	Z/IO	P	P	K	K	IO
DU0_DG2	In boot mode	0 to 3, 4, 5	Z	P	P	K	K	K
		6	Z/IO	P	P	K	K	IO
DU0_DG3	In boot mode	0 to 3, 4, 5	Z	P	P	K	K	K
		6	Z/IO	P	P	K	K	IO
DU0_DG4	In boot mode	0 to 3, 4, 5	Z	P	P	K	K	K
		6	Z/IO	P	P	K	K	IO
DU0_DG5	In boot mode	0 to 3, 4, 5	Z	P	P	K	K	K
		6	Z/IO	P	P	K	K	IO

Pin Name			Power-on Reset*1	Manual Reset	Sleep	Software Standby	Deep Standby: While Stopped	Deep Standby: On Return*2
DU0_DG6	In boot mode	0 to 3, 4, 5	Z	P	P	K	K	K
		6	Z/IO	P	P	K	K	IO
DU0_DG7	In boot mode	0 to 3, 4, 5	Z	P	P	K	K	K
		6	Z/IO	P	P	K	K	IO
DU0_DB0	In boot mode	0 to 3, 4, 5	Z	P	P	K	K	K
		6	Z/I	P	P	K	K	I
DU0_DB1	In boot mode	0 to 3, 4, 5	Z	P	P	K	K	K
		6	Z/I	P	P	K	K	I
DU0_DB2	In boot mode	0 to 3, 4, 5	Z	P	P	K	K	K
		6	Z/I	P	P	K	K	I
DU0_DB3	In boot mode	0 to 3, 4, 5	Z	P	P	K	K	K
		6	Z/I	P	P	K	K	I
DU0_DB4	In boot mode	0 to 3, 4, 5	Z	P	P	K	K	K
		6	Z/O	P	P	K	K	O
DU0_DB5	In boot mode	0 to 3, 4, 5	Z	P	P	K	K	K
		6	Z/O	P	P	K	K	O
DU0_DB6	In boot mode	0 to 3, 4, 5	Z	P	P	K	K	K
		6	L/O	P	P	K	L	O
DU0_DB7			Z	P	P	K	K	K
DU0_DOTCLKIN			ZU	P	P	K	K	K
DU0_DOTCLKOUT			ZU	P	P	K	K	K
DU0_EXHSYNC/DU0_HSYNC			ZU	P	P	K	K	K
DU0_EXVSYNC/DU0_VSYNC			ZU	P	P	K	K	K
DU0_EXODDF/DU0_ODDF			ZU	P	P	K	K	K
DU0_DISP			ZU	P	P	K	K	K
DU0_CDE			ZU	P	P	K	K	K
VI1_CLK_A			Z	P	P	K	K	K
VI1_0_A			ZU	P	P	K	K	K
VI1_1_A			ZU	P	P	K	K	K
VI1_2_A			ZU	P	P	K	K	K
VI1_3_A			ZU	P	P	K	K	K

Pin Name			Power-on Reset*1	Manual Reset	Sleep	Software Standby	Deep Standby: While Stopped	Deep Standby: On Return*2
VI1_4_A			ZU	P	P	K	K	K
VI1_5_A			ZU	P	P	K	K	K
VI1_6_A			ZU	P	P	K	K	K
VI1_7_A			ZU	P	P	K	K	K
SSI_SCK0_A			ZU	P	P	K	K	K
SSI_WS0_A			ZU	P	P	K	K	K
SSI_SDATA0_A			ZU	P	P	K	K	K
SSI_SCK1_A			ZU	P	P	K	K	K
SSI_WS1_A			ZU	P	P	K	K	K
SSI_SDATA1_A			ZU	P	P	K	K	K
SSI_SCK23			ZU	P	P	K	K	K
SSI_WS23			ZU	P	P	K	K	K
SSI_SDATA2			ZU	P	P	K	K	K
SSI_SDATA3			ZU	P	P	K	K	K
AUDIO_CLKA_A			ZU	P	P	K	K	K
AUDIO_CLKB_A			Z	P	P	K	K	K
AUDIO_CLKC			Z	P	P	K	K	K
AUDIO_CLKOUT			ZU	P	P	K	K	K
SCL0			Z	P	P	K	Z	K*4
SDA0	In boot mode	0 to 3, 4, 5	Z	P	P	K	Z	K*4
		6	Z/I	P	P	K	Z	I
SCL1			Z	P	P	K	Z	K*4
SDA1			Z	P	P	K	Z	K*4
USB_EXTAL			I	I	I	I	I	I
USB_XTAL			O	O	O	O	O	O
PENC0			O	O	O	K	K	K
PENC1			ZU	P	P	K	K	K
USB_OVC0			Z	I	I	K	K	K
USB_OVC1			ZU	P	P	K	K	K
DP0			Z	IO	IO	K	Z	Z*5
DM0			Z	IO	IO	K	Z	Z*5

Pin Name	Power-on Reset*1	Manual Reset	Sleep	Deep Standby: Software Standby	Deep Standby: While Stopped	Deep Standby: On Return*2
DP1	Z	IO	IO	K	Z	Z*5
DM1	Z	IO	IO	K	Z	Z*5
OVC0/VBUS0	Z	I	I	I	I	I
OVC1/VBUS1	Z	I	I	I	I	I
RTC_X1	I	I	I	I	I	I
RTC_X2	O	O	O	O	O	O
CAN_CLK_A	ZU	P	P	K	K	K
CAN0_TX_A	Z	P	P	K	K	K
CAN0_RX_A	Z	P	P	K I*17	K I*17	K
CAN1_TX_A	Z	P	P	K	K	K
CAN1_RX_A	ZU	P	P	K I*17	K I*17	K
AN0	Z	P	P	K I*17	Z	I
AN1	Z	P	P	K I*17	Z	I
AN2	Z	P	P	K	Z	I
AN3	Z	P	P	K	Z	I
AN4	Z	P	P	K	Z	I
AN5	Z	P	P	K	Z	I
AN6	Z	P	P	K	Z	I
AN7	Z	P	P	K	Z	I

[Legend]

K: The prior state of the port pin is retained after a transition to deep standby or to software standby.

I: Input

O: Output

P: Port function (Switching between input and output and control of pulling up and driving ability depend on register settings.)

Z: High-impedance (Both input and output buffers are switched off.)

ZU: High-impedance (Internal pulling up is active.)

H: High-level output

L: Low-level output

IO: Input or output (Settings of the pin function selection module determine which is selected.)

/: Under power-on reset—the symbol to the left of "/" indicates the state when the low level is being input on the $\overline{\text{PRESET}}$ pin.

The symbol to the right indicates the state after the level on the $\overline{\text{PRESET}}$ pin is changed from low to high.

[Boot mode]

0: CS0 boot (8 bits)

1: CS0 boot (16 bits)

2: NAND Flash boot

3: Serial boot

4: MMC boot

5: eSD boot

6: HIF boot

Notes: 1. Input of the low-level on the $\overline{\text{PRESET}}$ pin is the trigger for a power-on reset. A power-on reset due to a reset assertion command being issued through the user debugging interface or an overflow of the watchdog timer leads to all pins being restored to their initial functions and being in the same state as in normal operation.

2. The entries under this label are applicable until clearing of the IOKEEP flag.

When pull-up control or driving ability is switched, K is applicable as the state of all pins. The DBSC3 being in the self-refresh state is a prerequisite for transitions to deep standby. Furthermore, when booting up is via the HIF, the $\overline{\text{HIFRDY}}$ pin being fixed to the low level is a prerequisite for transitions to deep standby.

3. This pin operates as an input.

4. The state of control before the transition to deep standby is retained.

5. This may be made usable by processing to initialize the USB module.

6. The port state is retained if 1 is written to the EBUSKEEPE bit in the DSCTR register before the transition to deep standby.
7. The port state is retained if 1 is written to the EBUSKEEPE bit or the RAMBOOT bit in the DSCTR register before the transition to deep standby.
8. The port state is retained if 1 is written to the PRSTOKEEPE bit in the DSCTR register before the transition to deep standby.
9. The port state is retained if 1 is written to the CKOKEEPE bit in the DSCTR register before the transition to deep standby.
10. Writing 0 to the GET bit while ET0_GTX_CLK_B is selected outputs the low level on this pin.

10a. Writing 1 to the GET bit while ET0_GTX_CLK_B is selected makes this pin work as an output.

10b. If ET0_GTX_CLK_A is the selected pin function, the low level is output on this pin.

11. When the value of the GET bit in the DSSSR register is 1 and the selected pin function is for the GMII_B, MII_B, or RMII_A, levels input on the pin are reflected by GEtherC and levels output from GEtherC are reflected by the pin.

11a. When the value of the GET bit in the DSSSR register is 1 and the selected pin function is for the GMII_B or MII_B, levels input on the pin are reflected by GEtherC and levels output from GEtherC are reflected by the pin.

11b. When the value of the GET bit in the DSSSR register is 1 and the selected pin function is for the RMII_A, levels input on the pin are reflected by GEtherC and levels output from GEtherC are reflected by the pin.

12. When this pin is functioning as an output and the low level is being input on the $\overline{\text{TRST}}$ pin, this pin becomes an input. If mpmd is 0, the input will change from high to low, leading to return from standby.
13. This pin remains in the Z state until the start of the first fetch.
14. If CLKOUT is the selected pin function, the low level will be output on this pin.
    If a pin function other than CLKOUT is selected, the state prior to a transition to standby is retained after the transition.
15. If $\overline{\text{PRESETOUT}}$ is the selected pin function, the high level will be output on this pin.
    If a pin function other than $\overline{\text{PRESETOUT}}$ is selected, the state prior to a transition to standby is retained after the transition.
16. If operation is in boot mode 2, 3, 4, or 5, the initial state of the pin is the same after a manual reset as after a power-on reset.
    Apply the same processing for initialization as required after a power-on reset has been generated.
17. If pins IRQ0 to IRQ3 are in use as triggers for return from standby, return from standby starts in accord with a signal transition that corresponds to the settings.
18. If the port-pin state is not retained, changing the pin function switches the pin between GPIO and operation with a peripheral module.
19. The pin has its debugging function while debugging is in use (mpmd = 0).

Index

Q

R

S

T

U

V

W

SH7734 User's Manual: Hardware

Publication Date: Rev.1.00 Jun 12, 2012

Published by: Renesas Electronics Corporation

SALES OFFICES Renesas Electronics Corporation http://www.renesas.com

Refer to "http://www.renesas.com/" for the latest and detailed information.

Renesas Electronics America Inc.
2880 Scott Boulevard Santa Clara, CA 95050-2554, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

Renesas Electronics Canada Limited
1101 Nicholson Road, Newmarket, Ontario L3Y 9C3, Canada
Tel: +1-905-898-5441, Fax: +1-905-898-3220

Renesas Electronics Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-585-100, Fax: +44-1628-585-900

Renesas Electronics Europe GmbH
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-65030, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
7th Floor, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100083, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 204, 205, AZIA Center, No.1233 Lujiazui Ring Rd., Pudong District, Shanghai 200120, China
Tel: +86-21-5877-1818, Fax: +86-21-6887-7858 / -7898

Renesas Electronics Hong Kong Limited
Unit 1601-1613, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2886-9318, Fax: +852 2886-9022/9044

Renesas Electronics Taiwan Co., Ltd.
13F, No. 363, Fu Shing North Road, Taipei, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.
1 harbourFront Avenue, #06-10, keppel Bay Tower, Singapore 098632
Tel: +65-6213-0200, Fax: +65-6278-8001

Renesas Electronics Malaysia Sdn.Bhd.
Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics Korea Co., Ltd.
11F., Samik Lavied' or Bldg., 720-2 Yeoksam-Dong, Kangnam-Ku, Seoul 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

Colophon 1.1

SH7734

Renesas Electronics Corporation

R01UH0233EJ0100
(Previous Number:REJ09B0632-0060)