User's Manual

32

RZ/T1-M

User's Manual: Hardware

RZ Family RZ/T Series

Renesas Electronics
www.renesas.com

Rev.1.80 Jan 2022

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 or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving 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, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
3. 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 shall be responsible for determining what licenses are required from any third parties, and obtaining such licenses for the lawful import, export, manufacture, sales, utilization, distribution or other disposal of any products incorporating Renesas Electronics products, if required.
5. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
6. Renesas Electronics products are classified according to the following two quality grades: “Standard” and “High Quality”. The intended 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; industrial robots; etc.
   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.
   Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or 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 damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user’s manual or other Renesas Electronics document.
7. No semiconductor product is absolutely secure. Notwithstanding any security measures or features that may be implemented in Renesas Electronics hardware or software products, Renesas Electronics shall have absolutely no liability arising out of any vulnerability or security breach, including but not limited to any unauthorized access to or use of a Renesas Electronics product or a system that uses a Renesas Electronics product. RENESAS ELECTRONICS DOES NOT WARRANT OR GUARANTEE THAT RENESAS ELECTRONICS PRODUCTS, OR ANY SYSTEMS CREATED USING RENESAS ELECTRONICS PRODUCTS WILL BE INVULNERABLE OR FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION (“Vulnerability Issues”). RENESAS ELECTRONICS DISCLAIMS ANY AND ALL RESPONSIBILITY OR LIABILITY ARISING FROM OR RELATED TO ANY VULNERABILITY ISSUES. FURTHERMORE, TO THE EXTENT PERMITTED BY APPLICABLE LAW, RENESAS ELECTRONICS DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT AND ANY RELATED OR ACCOMPANYING SOFTWARE OR HARDWARE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.
8. When using Renesas Electronics products, refer to the latest product information (data sheets, user’s manuals, application notes, “General Notes for Handling and Using Semiconductor Devices” in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, 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 and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. Renesas Electronics products and technologies shall 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 shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
12. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
13. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
14. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.
(Note2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

(Rev.5.0-1 October 2020)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

Contact Information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit:
www.renesas.com/contact/

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

General Precautions in the Handling of Microprocessing Unit and Microcontroller Unit Products

The following usage notes are applicable to all Microprocessing unit and Microcontroller unit products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Precaution against Electrostatic Discharge (ESD)
   A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.
2. Processing at power-on
   The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the time when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the time when power is supplied until the power reaches the level at which resetting is specified.
3. Input of signal during power-off state
   Do not input signals or an I/O pull-up power supply while the device is powered off. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Follow the guideline for input signal during power-off state as described in your product documentation.
4. Handling of unused pins
   Handle unused pins in accordance with the directions given under handling of unused pins in the manual. The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of the LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible.
5. Clock signals
   After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. When the clock signal is generated with an external resonator or from an external oscillator during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Additionally, when switching to a clock signal produced with an external resonator or by an external oscillator while program execution is in progress, wait until the target clock signal is stable.
6. Voltage application waveform at input pin
   Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between V_{IL} (Max.) and V_{IH} (Min.) due to noise, for example, the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between V_{IL} (Max.) and V_{IH} (Min.).
7. Prohibition of access to reserved addresses
   Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.
8. Differences between products
   Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

How to Use This Manual

1. Objective and Target Users

This manual was written to explain the hardware functions and electrical characteristics of this LSI to the target users, i.e. those who will be using this LSI in the design of application systems. Target users are expected to understand the fundamentals of electrical circuits, logic circuits, and microcomputers.
This manual is organized in the following items: an overview of the product, descriptions of the CPU, system control functions, and peripheral functions, electrical characteristics of the device, and usage notes.

When designing an application system that includes this LSI, take all points to note into account.
Points to note are given in their contexts and at the final part of each section, and in the section giving usage notes.

The list of revisions is a summary of major points of revision or addition for earlier versions. It does not cover all revised items. For details on the revised points, see the actual locations in the manual.

The following documents have been prepared for the RZ/T1-M Group. Before using any of the documents, please visit our web site to verify that you have the most up-to-date available version of the document.

Document Type	Contents	Document Title	Document No.
Data Sheet	Overview of hardware and electrical characteristics	—	—
User's manual: Hardware	Hardware specifications (pin assignments, memory maps, peripheral specifications, electrical characteristics, and timing charts) and descriptions of operation	RZ/T1-M Group User's manual: Hardware	This User's manual
User's manual: Software	Please find this information from the Arm® Ltd. home page.		
Application Note	Examples of applications and sample programs	—	—
Renesas Technical Update	Preliminary report on the specifications of a product, document, etc.	—	—

2. Description of Registers

Each register description includes a bit chart, illustrating the arrangement of bits, and a table of bits, describing the meanings of the bit settings. The standard format and notation for bit charts and tables are described below.

X.X.X ... Register

Address(es): xxxx xxxxxh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	... [1:0]		...4	—	—	—	...0
Value after reset:	x	0	0	0	0	0	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	R/W
b0	... 0	... Bit	0: ..... 1: Setting prohibited	R/W
b3 to b1	—	Reserved	The read value is 0. The write value should be 0.	R/W
b4	... 4	... Bit	0: 1:	R
b6, b5	... [1:0]	... Bit	0 0: ..... 0 1: ..... Settings other than above are prohibited.	R/(W)*
b7	—	Reserved	The read value is undefined. Writing to this bit has no effect.	R

(1) R/W: The bit or field is readable and writable.
R/(W): The bit or field is readable and writable. However, writing to this bit or field has some limitations. For details on the limitations, see the description or notes of respective registers.
R: The bit or field is readable. Writing to this bit or field has no effect.

(2) Reserved. Make sure to use the specified value when writing to this bit or field; otherwise, the correct operation is not guaranteed.

(3) Setting prohibited. The correct operation is not guaranteed if such a setting is performed.

3. List of Abbreviations and Acronyms

Abbreviation	Full Form
bps	bits per second
CRC	Cyclic Redundancy Check
DMA	Direct Memory Access
DMAC	Direct Memory Access Controller
Hi-Z	High Impedance
I/O	Input/Output
LSB	Least Significant Bit
MSB	Most Significant Bit
NC	Non-Connect
PLL	Phase Locked Loop
PWM	Pulse Width Modulation
UART	Universal Asynchronous Receiver/Transmitter

4. Description of the Access Size

Access size:
8 bits = Byte
16 bits = Word
32 bits = Longword

Table of Contents

450 MHz, MCU with Arm® Cortex®-R4, on-chip FPU, 747 DMIPS, up to 1 Mbyte of on-chip extended SRAM, MDIO I/F, various communications interfaces such as an SPI multi-I/O bus controller, safety functions, and security functions*1

Features

■ On-chip 32-bit Arm Cortex-R4 processor
- High-speed realtime control with maximum operating frequency of 450 MHz
  Capable of 747 DMIPS (in operation at 450 MHz)
- On-chip 32-bit Arm Cortex-R4 (revision r1p4)
- Tightly coupled memory (TCM) with ECC: 512 Kbytes/32 Kbytes
- Instruction cache/data cache with ECC: 8 Kbytes per cache
- High-speed interrupt
- The FPU supports addition, subtraction, multiplication, division, multiply-and-accumulate, and square-root operations at single-precision and double-precision.
- Harvard architecture with 8-stage pipeline
- Supports the memory protection unit (MPU)
- Arm CoreSight architecture, includes support for debugging through JTAG and SWD interfaces

■ Low power consumption
- Standby mode and module stop function

■ On-chip extended SRAM
- Up to 1 Mbyte of the on-chip extended SRAM with ECC
- 150 MHz

■ Data transfer
- DMAC: 16 channels × 2 units

■ Event link controller
- Module operations can be started by event signals rather than by interrupt handlers.
- Linked operation of modules is available even while the CPU is in the sleep state.

■ Reset and power supply voltage control
- Three reset sources including a pin reset
- Dual power-voltage configuration: 3.3 V, 1.2 V (I/O unit), 1.2 V (internal)

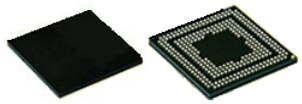
■ Clock functions
- Oscillator input frequency: 25 MHz
- CPU clock frequency: Up to 450 MHz
- Low-speed on-chip oscillator (LOCO): 240 kHz

■ Independent watchdog timer
- Operated by a clock signal obtained by frequency-dividing the clock signal from the low-speed on-chip oscillator: Up to 120 kHz

■ Safety functions
- Register write protection, input clock oscillation stop detection, CRC, IWDTa, and A/D self-diagnosis
- An error control module is incorporated to generate a pin signal output, interrupt, or internal reset in response to errors originating in the various modules.

■ Security functions (optional)*2
- Boot mode with security through encryption

■ Management data input/output interface (MDIO)
- An interface embedded in the optical transceiver modules which are compliant to the CFP MSA specifications: 1 channel (slave)
  Maximum operating frequency: 4 MHz
- Interface for DSP control: 2 channels (1 for slave operation, 1 for master operation*1)
  Maximum operating frequency: 10 MHz



PLBG0112KA-A

■ Various communications interfaces
- SCIFA with 16-byte transmission and reception FIFOs: 4 channels
- I^2C bus interface: 2 channels for transfer at up to 400 kbps
- RSPIa: 2 channels
- SPIBSC: Provides a single interface for multi-I/O compatible serial flash memory

■ Up to 12 extended-function timers
- 16-bit TPUa (6 channels): Input capture, output compare, PWM waveform output
- 16-bit CMT (4 channels), 32-bit CMTW (2 channels)

■ 12-bit A/D converters
- 12 bits × 2 units (max.)
  (8 channels for unit 0; 8 channels for unit 1)
- Self diagnosis
- Detection of analog input disconnection

■ Temperature sensor for measuring temperature within the chip

■ General-purpose I/O ports
- 5-V tolerance, open drain, input pull-up

■ Multi-function pin controller
- The locations of input/output functions for peripheral modules are selectable from among multiple pins.

■ Operating temperature range
- Tj = -40°C to +110°C
  Tj: Junction temperature

Note 1. Optional
Note 2. Details of these optional functions will only be disclosed after completion of a binding non-disclosure agreement. For details, contact our sales representative.

1. Overview

1.1 Outline of Specifications

This LSI circuit is a high-performance MCU equipped with the Arm® Cortex®-R4 (CR4) processor with FPU, and incorporating integrated peripheral functions necessary for system configuration. Table 1.1 lists the specifications in outline.

Table 1.1 Outline of Specifications (1 / 4)

Classification	Module/Function	Description
CPU	Central processing unit (Cortex-R4)	• Maximum operating frequency 112-pin FBGA: 450 MHz • 32-bit CPU Cortex-R4 designed by Arm (core revision r1p4) • Address space: 4 Gbytes • Instruction cache: 8 Kbytes (with ECC) • Data cache: 8 Kbytes (with ECC) • Tightly coupled memory (TCM) ATCM: 512 Kbytes (with ECC) BTCM: 32 Kbytes (with ECC) • Instruction set: Arm v7-R architecture, so support includes Thumb® and Thumb-2 • Data arrangement Instructions: Little endian Data: Little endian • Memory protection unit (MPU)
	FPU (Cortex-R4)	• Supports addition, subtraction, multiplication, division, multiply-and-accumulate, and square-root operations at single- and double-precision. • Registers 32-bit single-word registers: 32 bits × 32 (can be used as 16 double-word registers: 64 bits × 16)
Memory	On-chip extended SRAM with ECC	• Capacity: Up to 1 Mbyte • Operating frequency: 150 MHz • SEC-DED (single error correction/double error detection)
Operating modes		• Boot mode SPI boot mode (for booting up from serial flash memory)
Clock	Clock generation circuit	• An external resonator can be used as the input clock. • Detection of input clock oscillation stopping • The following clocks are generated. CPU clock: 450 MHz (max.) System clock: 150 MHz (fixed) High-speed peripheral module clock: 150 MHz (fixed) Low-speed peripheral module clock: 75 MHz (fixed) ADCCLK in the 12-bit A/D converter (S12ADCa): 60 MHz (max.) Low-speed on-chip oscillator: 240 kHz (fixed)
Reset		RES# pin reset, error control module (ECM) reset, software reset
Low power	Low-power consumption function	• Standby mode • Module stop function
Interrupt	Vector interrupt controller (VIC)	• Peripheral function interrupts: 96 sources • External interrupts: 8 sources (NMI, IRQ0 to IRQ4, IRQ6, and IRQ7 pins) • Non-maskable interrupts: 2 sources • Sixteen levels specifiable for the order of priority
Data transfer	Direct memory access controller (DMAC)	• 2 units (16 channels for unit 0, 16 channels for unit 1) • Transfer modes: Single transfer mode and block transfer mode • Transfer size Unit 0: 1/2/4/16/32/64 bytes Unit 1: 1/2/4/16 bytes • Activation sources: External interrupts, on-chip peripheral module requests, and software requests

Table 1.1 Outline of Specifications (2 / 4)

Classification	Module/Function	Description
I/O ports	General-purpose I/O ports	• 112-pin FBGA I/O pins: 51 (including seven 1.2-V I/O pins) Input pins: 4 Pull-up/pull-down resistors: 44 5-V tolerance: 4
Event link controller (ELC)		• Event signals can be interlinked with the operation of modules. • In particular, the operation of timer modules can be started by input event signals. • Event link operation is possible for port E.
Multi-function pin controller (MPC)		The locations of input/output functions are selectable from among multiple pins.
Timers	16-bit timer pulse unit (TPUa)	• 16 bits × 6 channels • Maximum of 16 pulse-input/output possible • Select from among seven or eight counter-input clock signals for each channel (with maximum operating frequency of 75 MHz) • Input capture/output compare function • Counter clear operation (synchronous clearing by compare match/input capture) • Simultaneous writing to multiple timer counters (TCNT) • Simultaneous register input/output by synchronous counter operation • Output of PWM waveforms in up to 15 phases in PWM mode • Support for buffered operation, phase-counting mode (two phase encoder input) and cascade-connected operation (32 bits × 2 channels) depending on the channel. • Capable of generating conversion start triggers for the A/D converters • Digital noise filtering of signals from the input capture pins • Event linking by the ELC
	Compare match timer (CMT)	• (16 bits × 2 channels) × 2 units • Select from among four counter-input clock signals for each channel (with maximum operating frequency of 75 MHz) • Event linking by the ELC (channel 1 of unit 0 only)
	Compare match timer W (CMTW)	• (32 bits × 1 channel) × 2 units • Compare-match, input-capture input, and output-comparison output are available. • Select from among four counter-input clock signals for each channel (with maximum operating frequency of 75 MHz) • Interrupt requests can be output in response to compare-match, input-capture, and output-comparison events. • Digital noise filter function for signals on the input capture pins • Event linking by the ELC
	Watchdog timer (WDTA)	• 14 bits × 1 channel • Select from among six counter-input clock signals for each channel (with maximum operating frequency of 75 MHz)
	Independent watchdog timer (IWDTa)	• 14 bits × 1 channel • Counter-input clock: Low-speed on-chip oscillator (LOCO)/2 Dedicated clock/1, dedicated clock/16, dedicated clock/32, dedicated clock/64, dedicated clock/128, dedicated clock/256 (with maximum operating frequency of 120 kHz)

Table 1.1 Outline of Specifications (3 / 4)

Classification	Module/Function	Description
Communication function	Serial communication interface with FIFO (SCIFA)	• 4 channels • Serial communications modes: Asynchronous, clock synchronous*2 • On-chip baud rate generator allows selection of the desired bit rate • Choice of LSB-first or MSB-first transfer • Both the transmission and reception sections are equipped with 16-byte FIFO buffers, allowing continuous transmission and reception. • Bit rate modulation
	I^2C bus interface (RIICa)	• 2 channels • Supports I^2C bus format • Supports the multi-master • Max. transfer rate: 400 kbps • Event linking by the ELC
	Management data input/ output interface (MDIO)	• 2 channels (1 for slave operation, 1 for master operation*3) Slave: Interface for an optical transceiver module compliant with the CFP MSA specification Master: Interface for DSP control
	Serial peripheral interface (RSPIa)	• 2 channels • RSPI transfer facility Using the MOSI (master out slave in), MISO (master in slave out), SSL (slave select), and RSPCK (RSPI clock) signals enables serial transfer through SPI operation (four lines) or clock-synchronous operation (three lines) Capable of handling serial transfer as a master or slave • Data formats Switching between MSB first and LSB first The number of bits in each transfer can be changed to any number of bits from 8 to 16, or 20, 24, or 32 bits. 128-bit buffers for transmission and reception Up to four frames can be transmitted or received in a single transfer operation (with each frame having up to 32 bits) • Buffered structure Double buffers for both transmission and reception • RSPCK can be stopped automatically with the reception buffer full for master reception • Event linking by the ELC
	SPI multi I/O bus controller (SPIBSC)	• 1 channel • One serial flash memory with multiple I/O bus sizes (single/dual/quad) can be connected. • External address space read mode (built-in read cache) • SPI operating mode • Clock polarity and clock phase can be selected. • Maximum transfer rate: 300 Mbps (for quad)

Table 1.1 Outline of Specifications (4 / 4)

Classification	Module/Function	Description
12-bit A/D converter (S12ADCa)		• 12 bits × 2 units (unit 0: 8 channels, unit 1: 8 channels) • 12-bit resolution • Conversion time VREFH0, VREFH1 = 3.0 to 3.6 V Unit 0: 0.483 µs per channel Unit 1: 0.883 µs per channel VREFH0, VREFH1 = 2.5 to 3.0 V Unit 0: 0.883 µs per channel Unit 1: 0.883 µs per channel • Operating mode Scan mode (single scan mode, continuous scan mode, or group scan mode) Group A priority control (only for group scan mode) • Sample-and-hold function Common sample-and-hold circuit included In addition, channel-dedicated sample-and-hold function (4 channels: in unit 0 only) included • Sampling variable Sampling time can be set up for each channel • Self-diagnostic function The self-diagnostic function internally generates three analog input voltages (unit 0: VREFL0, VREFH0 × 1/2, VREFH0; unit 1: VREFL1, VREFH1 × 1/2, VREFH1) • Double trigger mode (A/D conversion data duplicated) • Detection of analog input disconnection • Three ways to start A/D conversion Software trigger, timer (TPUa) trigger, external trigger • Event linking by the ELC
Temperature sensor		• 1 channel • Relative precision: ±1°C • The voltage of the temperature is converted into a digital value by the 12-bit A/D converter (unit 0).
Safety	Register write protection function	Protects important registers from being overwritten in cases where a program runs out of control.
	CRC calculator (CRC)	• CRC code generation for arbitrary amounts of data in 8-, 16-, or 32-bit units • Select any of four generating polynomials: $X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$ (32-Ethernet), $X^{16} + X^{12} + X^5 + 1$ (16-CCITT), $X^8 + X^4 + X^3 + X^2 + 1$ (8-SAEJ1850), $X^8 + X^5 + X^3 + X^2 + X + 1$ (8-0x2F)
	Input clock oscillation stop function	Input clock oscillation stop detection: Available
	Clock monitor circuit (CLMA)	Monitors the abnormal output clock frequency from the PLL circuit or low-speed on-chip oscillator.
	Data operation circuit (DOC)	The function to compare, add, or subtract 16-bit data
	Error control module (ECM)	• Generates an interrupt or internal reset for the error signal input from each module. • Time-out function • The error control is duplicated in the master and the checker.
Security	Secure boot mode*1	As an option, a boot mode with encryption as a security function is available.
Power supply voltage		VDD = PLLVDD0 = PLLVDD1 = VCCQ12 = 1.14 to 1.26 V VCCQ33 = AVCC0 = AVCC1 = 3.0 to 3.6 V VREFH0 = VREFH1 = 2.5 to 3.6 V
Operating temperature		Tj = -40 to +110°C
Package		112-pin FBGA: 6 × 6 mm, 0.5-mm pitch PLBG0112KA-A
Debugging interface		• CoreSight architecture designed by Arm • Debugging function by the JTAG/SWD interface, and trace function by the trace port/ SWV interface

Note 1. See Table 1.3, List of Products, for the products that have the secure boot mode. Details of these optional functions will only be disclosed after completion of a binding non-disclosure agreement. For details, contact our sales representative.

Note 2. Channel 4 is used only in asynchronous mode.
Note 3. The MDIO master is optional. For the products which have it, see Table 1.3, List of Products.

Table 1.2 List of Functions

Module/Function		RZ/T1-M Group 112 Pins
Interrupt	External interrupt	NMI, IRQ0 to IRQ4, IRQ6, IRQ7
DMA	DMA controller (DMAC)	ch0 to ch31
Timers	16-bit timer pulse unit (TPUa)	ch0 to ch5
	Compare match timer (CMT)	ch0 to ch3
	Compare match timer W (CMTW)	ch0, ch1
	Watchdog timer (WDTA)	ch0
	Independent watchdog timer (IWDTa)	Available
Communication function	Serial communications interface with FIFO (SCIFA)	ch0 to ch2, ch4*3
	Management data input/output interface (MDIO master*1/MDIO slave)	Available
	I^2C bus interface (RIICa)	ch0, ch1
	Serial peripheral interface (RSPIa)	ch0, ch1
	SPI multi I/O bus controller (SPIBSC)	ch0
12-bit A/D converter (S12ADCa)		AN000 to AN007 (unit 0) AN100 to AN107 (unit 1)
Temperature sensor		Available
CRC calculator (CRC)		Available
Data operation circuit (DOC)		Available
Clock monitor circuit (CLMA)		Available
Secure boot mode*2		Optional
Event link controller (ELC)		Available

Note 1. The MDIO master is optional. For the products which have it, see Table 1.3, List of Products.
Note 2. See Table 1.3, List of Products, for the products that have the secure boot mode. Details of this function will only be disclosed after completion of a binding non-disclosure agreement. For details, contact our sales representative.
Note 3. Channel 4 is used only in asynchronous mode.

1.2 List of Products

Table 1.3 is a list of products.

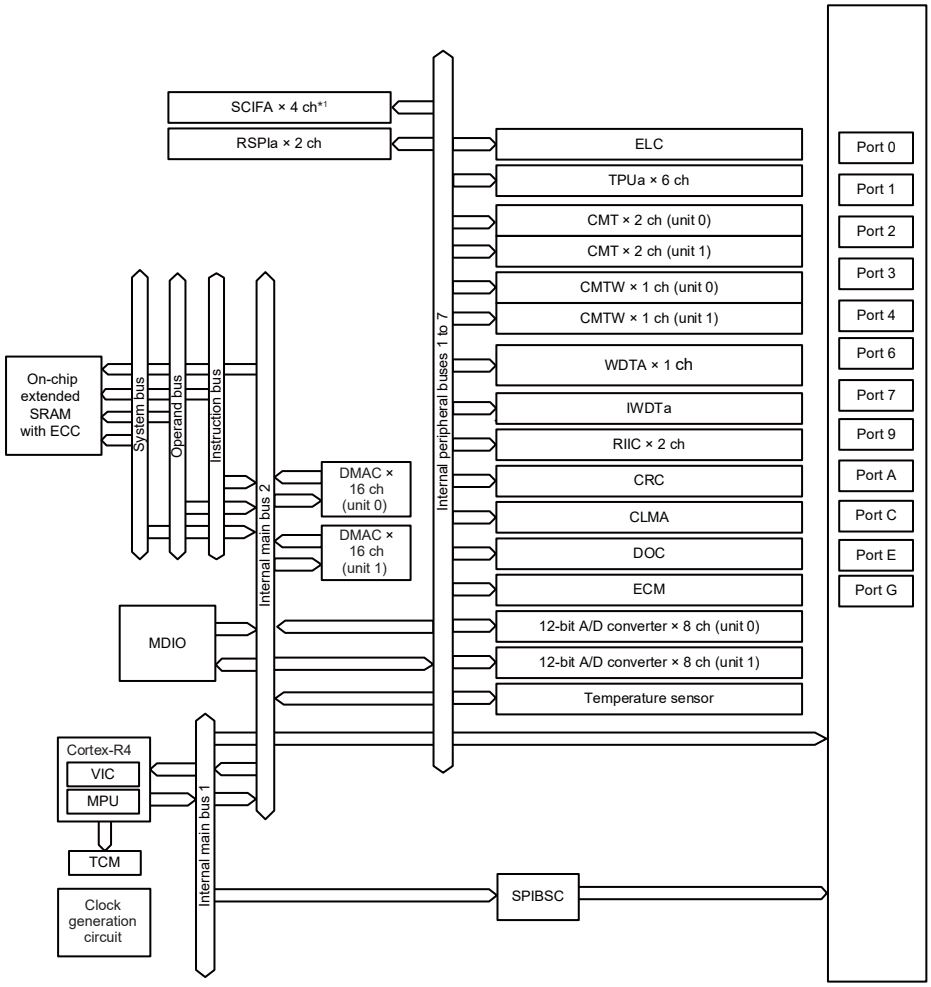
Table 1.3 List of Products

Group	Part No.	Package	CPU	On-Chip Extended SRAM Capacity	Operating Frequency (max.)	Security Function*1	Option
RZ/T1-M	R7S910020CBG	112 pins	Cortex-R4	Not supported	450 MHz	Not supported	Not supported
	R7S910021CBG	112 pins	Cortex-R4	1 Mbyte	450 MHz	Not supported	Not supported
	R7S910120CBG	112 pins	Cortex-R4	Not supported	450 MHz	Available	Not supported
	R7S910121CBG	112 pins	Cortex-R4	1 Mbyte	450 MHz	Available	Not supported
	R7S910022CBG	112 pins	Cortex-R4	Not supported	450 MHz	Not supported	MDIO master
	R7S910023CBG	112 pins	Cortex-R4	1 Mbyte	450 MHz	Not supported	MDIO master
	R7S910122CBG	112 pins	Cortex-R4	Not supported	450 MHz	Available	MDIO master
	R7S910123CBG	112 pins	Cortex-R4	1 Mbyte	450 MHz	Available	MDIO master

Note 1. Details of these functions will only be disclosed after completion of a binding non-disclosure agreement. For details, contact our sales representative.

1.3 Block Diagram

Figure 1.1 shows a block diagram.



DMAC:	DMA controller	ELC:	Event link controller
SPIBSC:	SPI multi I/O bus controller	TPUa:	16-bit timer pulse unit
WDTA:	Watchdog timer	CMT:	Compare match timer
IWDTa:	Independent watchdog timer	CMTW:	Compare match timer W
MDIO:	Management data input/output interface	RIICa:	I2C bus interface
SCIFA:	Serial communication interface with FIFO	CRC:	CRC (cyclic redundancy check) calculator
RSPIa:	Serial peripheral interface	CLMA:	Clock monitor circuit
VIC:	Vector interrupt controller	DOC:	Data operation circuit
MPU:	Memory protection unit	ECM:	Error control module

Note 1. Channel 4 is used only in asynchronous mode.

Figure 1.1 Block Diagram

1.4 Pin Functions

Table 1.4 lists the pin functions.

Table 1.4 Pin Functions (1 / 3)

Classifications	Pin Name	I/O	Description
Power supply	VDD	Input	Power supply pin. Connect this pin to the system power supply.
	VSS	Input	Ground pin. Connect this pin to the system power supply (0 V).
	VCCQ12	Input	Power supply pin for MDIO pins
	VCCQ33	Input	Power supply pin for I/O pins
	PLLVDD0, PLLVDD1	Input	Power supply pins for the on-chip PLL oscillator
	PLLVSS0, PLLVSS1	Input	Ground pins for the on-chip PLL oscillator. Connect these pins to the system power supply (0 V).
Clock	XTAL	Output	Connected to a crystal resonator.
	EXTAL	Input	
Operating mode control	MD0, MD1	Input	Input the operating mode select signal.
System control	RES#	Input	Reset signal input pin. This LSI enters the reset state when this signal goes low.
	RSTOUT#	Output	Outputs the reset signal externally.
Debugging interface	TRST#	Input	Test reset pin for on-chip emulator
	TMS	I/O	Test mode select pin for on-chip emulator
	TDI	Input	Test data input pin for on-chip emulator
	TDO	Output	Test data output pin for on-chip emulator
	TCK	Input	Test clock pin for on-chip emulator
	TRACECLK	Output	Outputs the clock for synchronization with the trace data.
	TRACECTL	Output	Outputs the enable signal for trace control.
	TRACEDATA0 to TRACEDATA7	Output	Output the trace data.
Interrupt	NMI	Input	Inputs the non-maskable interrupt request signal.
	IRQ0 to IRQ4, IRQ6, IRQ7	Input	Input the external interrupt request signal.
16-bit timer pulse unit (TPUa)	TIOCA0, TIOCB0, TIOCC0, TIOCD0	I/O	TGRA0 to TGRD0 input capture input/output compare output/ PWM output pins
	TIOCA1, TIOCB1	I/O	TGRA1 and TGRB1 input capture input/output compare output/ PWM output pins
	TIOCA2, TIOCB2	I/O	TGRA2 and TGRB2 input capture input/output compare output/ PWM output pins
	TIOCA3, TIOCB3, TIOCC3, TIOCD3	I/O	TGRA3 to TGRD3 input capture input/output compare output/ PWM output pins
	TIOCA4, TIOCB4	I/O	TGRA4 and TGRB4 input capture input/output compare output/ PWM output pins
	TIOCA5, TIOCB5	I/O	TGRA5 and TGRB5 input capture input/output compare output/ PWM output pins
	TCLKA, TCLKB, TCLKC, TCLKD	Input	External clock input pins for TPUa
Compare match timer W (CMTW)	TIC0 to TIC3	Input	CMTW input capture input pins
	TOC0 to TOC3	Output	CMTW output compare output pins

Table 1.4 Pin Functions (2 / 3)

Classifications	Pin Name	I/O	Description
Serial communication interface with FIFO (SCIFA)	SCK0 to SCK2	I/O	Clock I/O pins
	RXD0 to RXD2, RXD4*1	Input	Input the receive data.
	TXD0 to TXD2, TXD4*1	Output	Output the transmit data.
	CTS0# to CTS2#	I/O	Hardware flow control input (transmission enable signal)/general output
	RTS0# to RTS2#	Output	Hardware flow control output (transmission request signal)/general output
I2C bus interface (RIICa)	SCL0, SCL1	I/O	Clock I/O pins. The bus can be directly driven by the N-channel open drain.
	SDA0, SDA1	I/O	Data I/O pins. The bus can be directly driven by the N-channel open drain.
Management data input/ output interface (MDIOM/ MDIO)	MDC	Input	MDIO clock input pin for slave operation (up to 4 MHz)
	MDIO	I/O	MDIO data I/O pin for slave operation
	MMDC1	Output	MDIO clock output pins for master operation (up to 10 MHz)
	MMDIO1	I/O	MDIO data I/O pins for master operation
	PRTADR0	Input	Input pin to select the optical transceiver module for slave operation
	PRTADR1	Input	Input pin to select the optical transceiver module for slave operation
	PRTADR2	Input	Input pin to select the optical transceiver module for slave operation
	PRTADR3	Input	Input pin to select the optical transceiver module for slave operation
	PRTADR4	Input	Input pin to select the optical transceiver module for slave operation
Serial peripheral interface (RSPIa)	RSPCK0, RSPCK1	I/O	Clock I/O pins
	MOSI0, MOSI1	I/O	Master transmit data I/O pins
	MISO0, MISO1	I/O	Slave transmit data I/O pins
	SSL00, SSL10	I/O	Slave select signal I/O pins
	SSL01, SSL02, SSL03, SSL11	Output	Slave select signal output pins
SPI multi I/O bus controller (SPIBSC)	SPBCLK	Output	Clock output pin
	SPBSSL	Output	Slave select signal output pin
	SPBMO/SPBIO0	I/O	Master transmit data/data 0 I/O pin
	SPBMI/SPBIO1	I/O	Master input data/data 1 I/O pin
	SPBIO2, SPBIO3	I/O	Data 2, data 3 I/O pins
12-bit A/D converter (S12ADCa)	AN000 to AN007, AN100 to AN107	Input	Analog input pins for A/D converter
	ADTRG0, ADTRG1	Input	External trigger input pins for the start of A/D conversion
Analog power supply	AVCC0	Input	Analog power supply input pin for the 12-bit A/D converter (unit 0). Connect this pin to the VCCQ33 pin if the 12-bit A/D converter is not to be used.
	AVSS0	Input	Analog ground input pin for the 12-bit A/D converter (unit 0). Connect this pin to the VSS pin if the 12-bit A/D converter is not to be used.
	VREFH0	Input	Reference power supply input pin for the 12-bit A/D converter (unit 0). Connect this pin to the VCCQ33 pin if the 12-bit A/D converter is not to be used.
	VREFL0	Input	Reference ground pin for the 12-bit A/D converter (unit 0). Connect this pin to the VSS pin if the 12-bit A/D converter is not to be used.

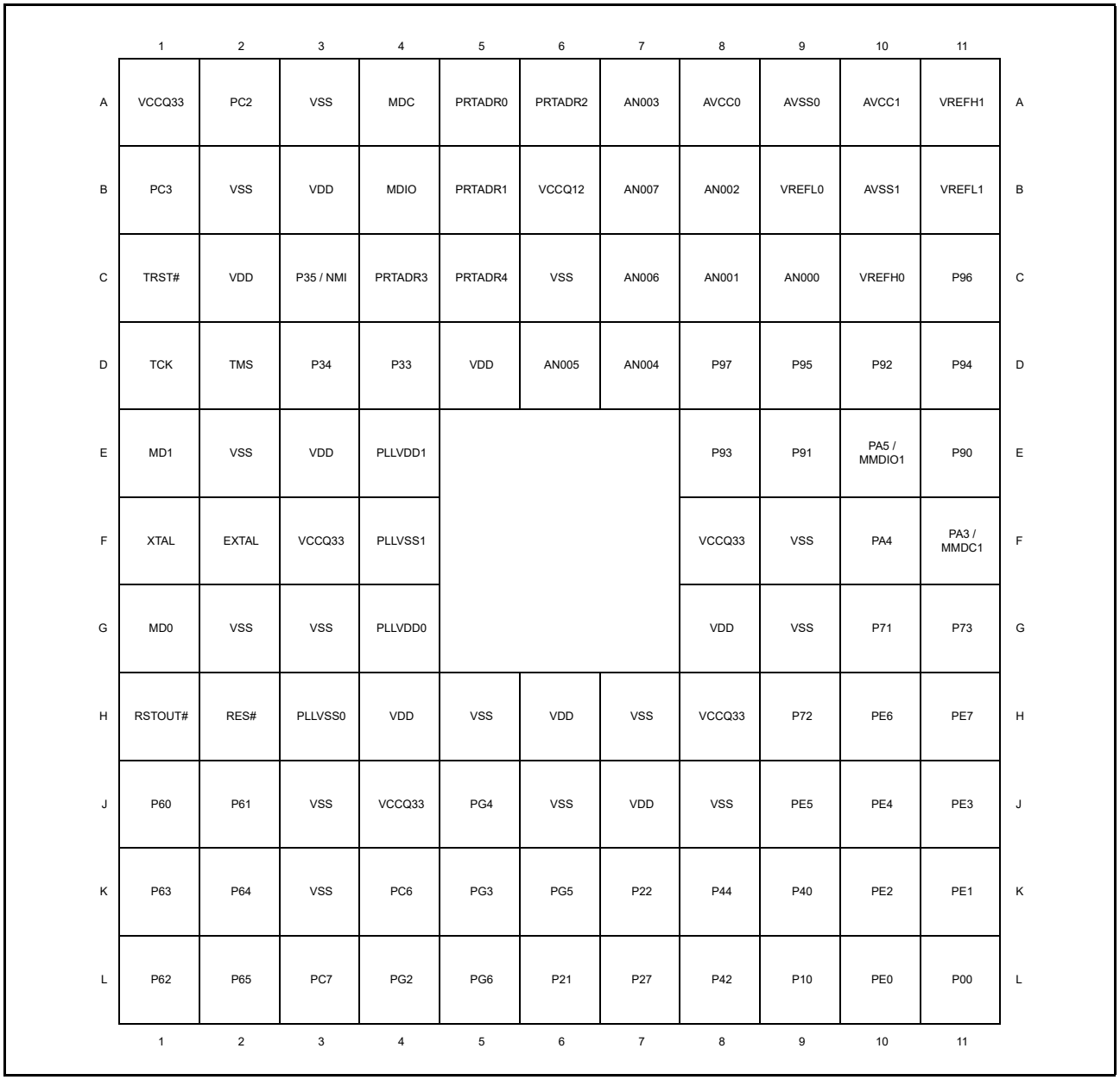
Table 1.4 Pin Functions (3 / 3)

Classifications	Pin Name	I/O	Description
Analog power supply	AVCC1	Input	Analog power supply input pin for the 12-bit A/D converter (unit 1). Connect this pin to the VCCQ33 pin if the 12-bit A/D converter is not to be used.
	AVSS1	Input	Analog ground input pin for the 12-bit A/D converter (unit 1). Connect this pin to the VSS pin if the 12-bit A/D converter is not to be used.
	VREFH1	Input	Reference power supply input pin for the 12-bit A/D converter (unit 1). Connect this pin to the VCCQ33 pin if the 12-bit A/D converter is not to be used.
	VREFL1	Input	Reference ground pin for the 12-bit A/D converter (unit 1). Connect this pin to the VSS pin if the 12-bit A/D converter is not to be used.
I/O ports	P00	I/O	1-bit I/O pin
	P10	I/O	1-bit I/O pin
	P21, P22, P27	I/O	3-bit I/O pins
	P33, P34, P35	I/O	3-bit I/O pins
	P40, P42, P44	I/O	3-bit I/O pins
	P50 to P56*2	I/O	7-bit I/O pins
	P60 to P65	I/O	6-bit I/O pins
	P71 to P73	I/O	3-bit I/O pins
	P90 to P97	I/O	8-bit I/O pins
	PA3 to PA5	I/O	3-bit I/O pins
	PC2, PC3, PC6, PC7	Input	4-bit input pins
	PE0 to PE7	I/O	8-bit I/O pins
	PG2 to PG6	I/O	5-bit I/O pins

Note 1. Channel 4 is used only in asynchronous mode.
Note 2. 1.2-V pins

1.5 Pin Assignments

Figure 1.2 shows the pin arrangement. Table 1.5 shows the pin assignments. Table 1.6 shows the list of pin functions.



	1	2	3	4	5	6	7	8	9	10	11	
A	VCCQ33	PC2	VSS	MDC	PRTADR0	PRTADR2	AN003	AVCC0	AVSS0	AVCC1	VREFH1	A
B	PC3	VSS	VDD	MDIO	PRTADR1	VCCQ12	AN007	AN002	VREFL0	AVSS1	VREFL1	B
C	TRST#	VDD	P35 / NMI	PRTADR3	PRTADR4	VSS	AN006	AN001	AN000	VREFH0	P96	C
D	TCK	TMS	P34	P33	VDD	AN005	AN004	P97	P95	P92	P94	D
E	MD1	VSS	VDD	PLLVDD1				P93	P91	PA5 / MMDIO1	P90	E
F	XTAL	EXTAL	VCCQ33	PLLVSS1				VCCQ33	VSS	PA4	PA3 / MMDC1	F
G	MD0	VSS	VSS	PLLVDD0				VDD	VSS	P71	P73	G
H	RSTOUT#	RES#	PLLVSS0	VDD	VSS	VDD	VSS	VCCQ33	P72	PE6	PE7	H
J	P60	P61	VSS	VCCQ33	PG4	VSS	VDD	VSS	PE5	PE4	PE3	J
K	P63	P64	VSS	PC6	PG3	PG5	P22	P44	P40	PE2	PE1	K
L	P62	P65	PC7	PG2	PG6	P21	P27	P42	P10	PE0	P00	L
	1	2	3	4	5	6	7	8	9	10	11	

Figure 1.2 Pin Arrangement (112-pin FBGA) (Top View)

Table 1.5 Pin Assignments (112-Pin FBGA) (1 / 3)

Pin Number	Pin Name
A1	VCCQ33
A2	PC2 / SDA0
A3	VSS
A4	P56 / MDC
A5	P54 / PRTADR0
A6	P51 / PRTADR2
A7	AN003
A8	AVCC0
A9	AVSS0
A10	AVCC1
A11	VREFH1
B1	PC3 / RXD4 / SCL0
B2	VSS
B3	VDD
B4	P55 / MDIO
B5	P52 / PRTADR1
B6	VCCQ12
B7	AN007
B8	AN002
B9	VREFL0
B10	AVSS1
B11	VREFL1
C1	TRST#
C2	VDD
C3	P35 / NMI
C4	P53 / PRTADR3
C5	P50 / PRTADR4
C6	VSS
C7	AN006
C8	AN001
C9	AN000
C10	VREFH0
C11	P96 / AN106
D1	TCK
D2	TMS
D3	P34 / TDI
D4	P33 / TDO
D5	VDD
D6	AN005
D7	AN004
D8	P97 / AN107 / IRQ7 / ADTRG1
D9	P95 / AN105 / CTS2#
D10	P92 / AN102 / TOC3 / RXD2
D11	P94 / AN104 / IRQ4 / RTS2#

Table 1.5 Pin Assignments (112-Pin FBGA) (2 / 3)

Pin Number	Pin Name
E1	MD1
E2	VSS
E3	VDD
E4	PLLVDD1
E8	P93 / AN103 / TIC3 / SCK2
E9	P91 / AN101 / TXD2
E10	PA5 / TIOCA4 / TXD2 / MMDIO1
E11	P90 / AN100 / TIOCA5 / TXD4
F1	XTAL
F2	EXTAL
F3	VCCQ33
F4	PLLVSS1
F8	VCCQ33
F9	VSS
F10	PA4 / TIOCA3 / ADTRG0 / RXD2
F11	PA3 / TIOCA2 / SCK2 / MMDC1
G1	MD0
G2	VSS
G3	VSS
G4	PLLVDD0
G8	VDD
G9	VSS
G10	P71 / TOC2 / SCK1 / TRACECTL
G11	P73 / IRQ3 / RXD1 / TRACEDATA1
H1	RSTOUT#
H2	RES#
H3	PLLVSS0
H4	VDD
H5	VSS
H6	VDD
H7	VSS
H8	VCCQ33
H9	P72 / TIC2 / TXD1 / TRACEDATA0
H10	PE6 / IRQ6 / TIOCD0 / RXD1 / MISO0 / TRACEDATA6
H11	PE7 / TIOCD3 / SCK1 / RSPCK0 / TRACEDATA7
J1	P60 / SPBSSL
J2	P61 / SPBIO3
J3	VSS
J4	VCCQ33
J5	PG4 / TOC1 / MOSI1
J6	VSS
J7	VDD
J8	VSS
J9	PE5 / TIOCC3 / TXD1 / MOSI0 / TRACEDATA5
J10	PE4 / TIOCC0 / RTS1# / SSL00 / TRACEDATA4

Table 1.5 Pin Assignments (112-Pin FBGA) (3 / 3)

Pin Number	Pin Name
J11	PE3 / IRQ3 / TIOCB5 / CTS1# / SSL01 / TRACEDATA3
K1	P63 / SPBMO/SPBIO0
K2	P64 / SPBMI/SPBIO1
K3	VSS
K4	PC6 / TCLKC / SCL1
K5	PG3 / TIC1 / MISO1
K6	PG5 / TCLKA / SSL10
K7	P22 / IRQ2 / TIOCD0 / SCK0
K8	P44 / TCLKD / ADTRG0 / CTS0#
K9	P40 / TXD0
K10	PE2 / IRQ2 / TIOCB4 / SSL02 / TRACEDATA2
K11	PE1 / TIOCB3 / SSL03 / TRACEDATA1
L1	P62 / SPBCLK
L2	P65 / SPBIO2
L3	PC7 / TIC0 / SDA1
L4	PG2 / TOC0 / RSPCK1
L5	PG6 / TCLKB / SSL11
L6	P21 / IRQ1 / TIOCB1 / CTS0#
L7	P27 / TIOCB0 / RTS0#
L8	P42 / RXD0
L9	P10 / IRQ0 / TIOCA0 / TRACECLK
L10	PE0 / TIOCB2 / TRACEDATA0
L11	P00 / TIOCA1 / ADTRG1 / TRACECTL

Table 1.6 List of Pins and Pin Functions (112-Pin FBGA) (1 / 3)

Pin Number			Timer	Communication			
112-Pin FBGA	Power Supply Clock System Control	I/O Port	(TPUa, CMT2)	(MDIO, SCIFA, RSPIa, RIICa, SPIBSC)	Others	Interrupt	S12ADC
A1	VCCQ33						
A2		PC2		SDA0			
A3	VSS						
A4		P56		MDC			
A5		P54		PRTADR0			
A6		P51		PRTADR2			
A7							AN003
A8	AVCC0						
A9	AVSS0						
A10	AVCC1						
A11	VREFH1						
B1		PC3		RXD4 / SCL0			
B2	VSS						
B3	VDD						
B4		P55		MDIO			
B5		P52		PRTADR1			
B6	VCCQ12						
B7							AN007
B8							AN002
B9	VREFL0						
B10	AVSS1						
B11	VREFL1						
C1	TRST#						
C2	VDD						
C3		P35				NMI	
C4		P53		PRTADR3			
C5		P50		PRTADR4			
C6	VSS						
C7							AN006
C8							AN001
C9							AN000
C10	VREFH0						
C11		P96					AN106
D1	TCK						
D2	TMS						
D3	TDI	P34					
D4	TDO	P33					
D5	VDD						
D6							AN005
D7							AN004
D8		P97				IRQ7	ADTRG1 / AN107

Table 1.6 List of Pins and Pin Functions (112-Pin FBGA) (2 / 3)

Pin Number 112-Pin FBGA	Power Supply Clock System Control	I/O Port	Timer (TPUa, CMT2)	Communication (MDIO, SCIFA, RSPIa, RIICa, SPIBSC)	Others	Interrupt	S12ADC
D9		P95		CTS2#			AN105
D10		P92	TOC3	RXD2			AN102
D11		P94		RTS2#		IRQ4	AN104
E1	MD1						
E2	VSS						
E3	VDD						
E4	PLLVDD1						
E8		P93	TIC3	SCK2			AN103
E9		P91		TXD2			AN101
E10		PA5	TIOCA4	TXD2 / MMDIO1			
E11		P90	TIOCA5	TXD4			AN100
F1	XTAL						
F2	EXTAL						
F3	VCCQ33						
F4	PLLVSS1						
F8	VCCQ33						
F9	VSS						
F10		PA4	TIOCA3	RXD2			ADTRG0
F11		PA3	TIOCA2	SCK2 / MMDC1			
G1	MD0						
G2	VSS						
G3	VSS						
G4	PLLVDD0						
G8	VDD						
G9	VSS						
G10	TRACECTL	P71	TOC2	SCK1			
G11	TRACEDATA1	P73		RXD1		IRQ3	
H1	RSTOUT#						
H2	RES#						
H3	PLLVSS0						
H4	VDD						
H5	VSS						
H6	VDD						
H7	VSS						
H8	VCCQ33						
H9	TRACEDATA0	P72	TIC2	TXD1			
H10	TRACEDATA6	PE6	TIOCD0	RXD1 / MISO0		IRQ6	
H11	TRACEDATA7	PE7	TIOCD3	SCK1 / RSPCK0			
J1		P60		SPBSSL			
J2		P61		SPBIO3			
J3	VSS						
J4	VCCQ33						

Table 1.6 List of Pins and Pin Functions (112-Pin FBGA) (3 / 3)

Pin Number 112-Pin FBGA	Power Supply Clock System Control	I/O Port	Timer (TPUa, CMT2)	Communication (MDIO, SCIFA, RSPIa, RIICa, SPIBSC)	Others	Interrupt	S12ADC
J5		PG4	TOC1	MOSI1			
J6	VSS						
J7	VDD						
J8	VSS						
J9	TRACEDATA5	PE5	TIOCC3	TXD1 / MOSI0			
J10	TRACEDATA4	PE4	TIOCC0	RTS1# / SSL00			
J11	TRACEDATA3	PE3	TIOCB5	CTS1# / SSL01		IRQ3	
K1		P63		SPBMO / SPBIO0			
K2		P64		SPBMI / SPBIO1			
K3	VSS						
K4		PC6	TCLKC	SCL1			
K5		PG3	TIC1	MISO1			
K6		PG5	TCLKA	SSL10			
K7		P22	TIOCD0	SCK0		IRQ2	
K8		P44	TCLKD	CTS0#			ADTRG0
K9		P40		TXD0			
K10	TRACEDATA2	PE2	TIOCB4	SSL02		IRQ2	
K11	TRACEDATA1	PE1	TIOCB3	SSL03			
L1		P62		SPBCLK			
L2		P65		SPBIO2			
L3		PC7	TIC0	SDA1			
L4		PG2	TOC0	RSPCK1			
L5		PG6	TCLKB	SSL11			
L6		P21	TIOCB1	CTS0#		IRQ1	
L7		P27	TIOCB0	RTS0#			
L8		P42		RXD0			
L9	TRACECLK	P10	TIOCA0			IRQ0	
L10	TRACEDATA0	PE0	TIOCB2				
L11	TRACECTL	P00	TIOCA1				ADTRG1

2. CPU

This LSI includes a Cortex-R4 CPU. The revision of the module is r1p4.

2.1 Overview

Table 2.1 Specifications of CPU

Item		Specification
Cortex-R4 (r1p4)	Minimum instruction execution time	One clock per instruction
	Address space	4 Gbytes
	Instruction cache size	8 Kbytes (with ECC)
	Data cache size	8 Kbytes (with ECC)
	Tightly coupled memory (TCM) size	ATCM: 512 Kbytes (with ECC) BTCM: 32 Kbytes (with ECC)
	Instruction set	Arm v7-R architecture supporting Thumb®/Thumb-2
	Data arrangement	Instruction: Little endian Data: Little endian
	Memory protection	Memory protection unit (MPU)
	FPU	• Supports addition, subtraction, multiplication, division, product-sum operation, and square-root operation in single-precision and double-precision • 32-bit single word register: 32 registers Can also be used as 16 64-bit double word registers

For details, refer to the following documents supplied by Arm.

- Arm Architecture Reference Manual Arm v7-A and Arm v7-R edition Issue C
- Arm v7-M Architecture Reference Manual

2.2 Configuration Information

Table 2.2 lists the configuration information for the Cortex-R4 of this LSI.

Table 2.2 Setting Values for Cortex-R4 Configuration Signals

Item		Setting Value
Endian	CFGEE	0
	CFGIE	0
Interrupt	CFGNMFI	1
Exception vector	TEINIT	0
	VINITHI	1
TCM configuration	INITRAMA	1
	INITRAMB	1
	LOCZRAMA	1
	CFGATCMSZ[3:0]	Ah
	CFGBTCMSZ[3:0]	6h
	ENTCM1IF	0
	SLBTCMSB	1 (don't care)
ECC, etc.	PARECCENRAM[2:0]	000b
	ERRENRAM[2:0]	000b
	RMWENRAM[1:0]	00b
	PARLVRAM	0 (don't care)

2.3 Restrictions on CPU

For details on restrictions on Cortex-R4 mounted on this LSI, refer to the information provided at the website of Arm.

2.4 Register Descriptions

2.4.1 ATCM Wait Control Register (SYTATCMWAIT)

SYTATCMWAIT is a register that controls ATCM access wait.
This register can be protected by the register write protection function. When writing to this register, cancel the write protection of bit 3 in the protection register (PRCR). For details, see section 11, Register Write Protection Function.

Address(es): A00B 0800h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ATCMWAIT[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b1, b0	ATCMWAIT[1:0]	ATCM Wait Setting *1, *2	b1 b0 0 0: 1-wait with optimization 0 1: 1-wait without optimization 1 0: 0-wait 1 1: Setting prohibited	R/W
b31 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. When the CPU clock frequency is 450 MHz, set these bits for “1-wait with optimization” or “1-wait without optimization”. “0-wait” can be set only when the CPU clock frequency is 150 MHz or 300 MHz. It cannot be set when the frequency is 450 MHz.
Note 2. If the ATCMWAIT[1:0] setting changes, operation cannot be guaranteed when the bus maser such as CPU accesses ACTM (including instruction fetch). To prevent fetch access from CPU, these bits should be handled by programs allocated in memory areas other than ATCM.

ATCMWAIT[1:0] Bits (ATCM Wait Setting)

These bits specify the number of memory access waits for ATCM.
In case of “with optimization”, memory access speed can be increased practically to 0-wait by prefetching the next address when instructions are fetched from sequential addresses in ATCM.

3. Operating Modes

3.1 Overview

This LSI chip is intended for booting up from an external serial flash memory. The SPI boot mode is available for the serial flash memory that supports the operating mode. In SPI boot mode, the user program stored in the corresponding external serial flash memory is booted up and then runs.
Secure boot mode (in which a user program is protected by encryption) can also be selected for the products that support security function*1.

Note 1. This function is provided upon signing a nondisclosure agreement. For details, contact Renesas Electronics Corporation's sales office.

3.2 Types and Selection of Operating Modes

The SPI boot mode can be selected, depending on the method of connecting to an external serial flash memory. An operating mode is selected based on the input levels of the mode setting pins (MD1 and MD0) at the time pin reset is released.

Table 3.1 describes the relationship between the input levels of the mode setting pins (MD1 and MD0) at the time reset is released and the selected operating mode. For details on individual operating modes, see section 3.5, Operating Mode Descriptions.

Table 3.1 Selection of Operating Mode for Each Combination of Levels of Mode Setting Pins (MD1 and MD0)

Mode Setting Pins		
MD1	**MD0**	**Operating Mode**
Low	Low	SPI boot mode (Serial flash) Boots a program from a serial flash memory connected to the SPI multi-I/O bus space.
Other than above		Reserved (Setting prohibited)

3.3 Hardware Used in Individual Operating Modes

Table 3.2 describes hardware used in individual operating modes.
“Pins to be Used” indicates pins required to execute each operating mode. The functions for these pins are automatically configured at boot.

Table 3.2 Hardware Used in Individual Operating Modes

Operating Mode	Peripheral Module	Pins to be Used
SPI boot mode (Serial flash)	SPI Multi-I/O bus controller (SPIBSC)	SPBCLK, SPBSSL SPBMO, SPBMI

3.4 Register Descriptions

3.4.1 Mode Monitor Register (MDMONR)

Mode monitor register (MDMONR) indicates input levels of the MD1 and MD0 pins.

Address(es): A00B 0A60h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MD1	MD0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0/1 *1	0/1 *1

Bit	Symbol	Bit Name	Description	R/W
b0	MD0	MD0 pin status flag	0: The MD0 pin is “Low”. 1: The MD0 pin is “High”.	R
b1	MD1	MD1 pin status flag	0: The MD1 pin is “Low”. 1: The MD1 pin is “High”.	R
b31 to b2	—	Reserved	These bits are read as 0.	R

Note 1. This value differs depending on the pin level at the time reset is released. For details, see section 6, Reset.

3.5 Operating Mode Descriptions

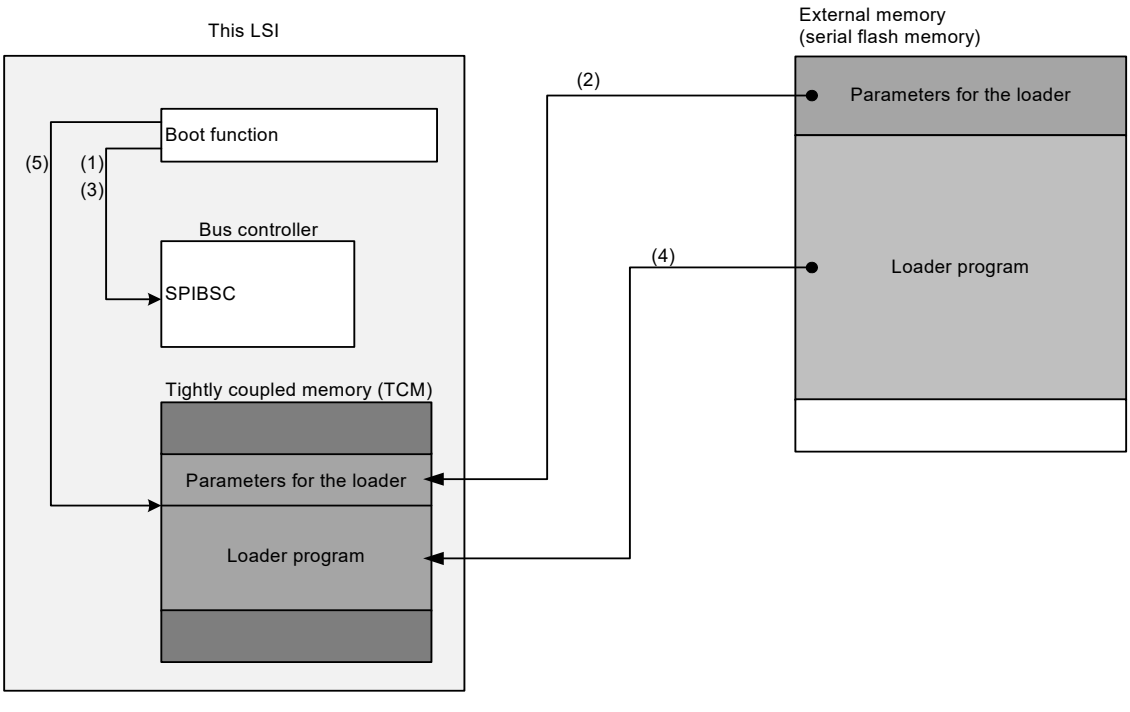
3.5.1 Boot Function

After reset is released on this LSI, the boot function executes the boot processing described below. The boot processing can extract a loader program that was stored in an external memory in advance by a user, to the internal tightly coupled memory (TCM) area, and hand over the processing to the loader program at the start address of that program.

(1) Setting the bus controller (SPIBSC) specified by the mode setting pins (MD1 and MD0)
(2) Loading parameters for the loader from an external memory, and executing checksum
(3) Setting for speeding up the bus controller (SPIBSC) by using parameters for the loader
(4) Loading the loader program from an external memory
(5) Branching off to the start address of the loader program extracted to the tightly coupled memory (TCM)

Parameters for the loader can have configuration information that suite for the user system, such as, loader program information, cache setting for speeding up the boot processing, and bus controller (SPIBSC) settings. Parameters for the loader must be stored in an external memory in advance by a user.

Figure 3.1 shows the operating overview of boot processing.



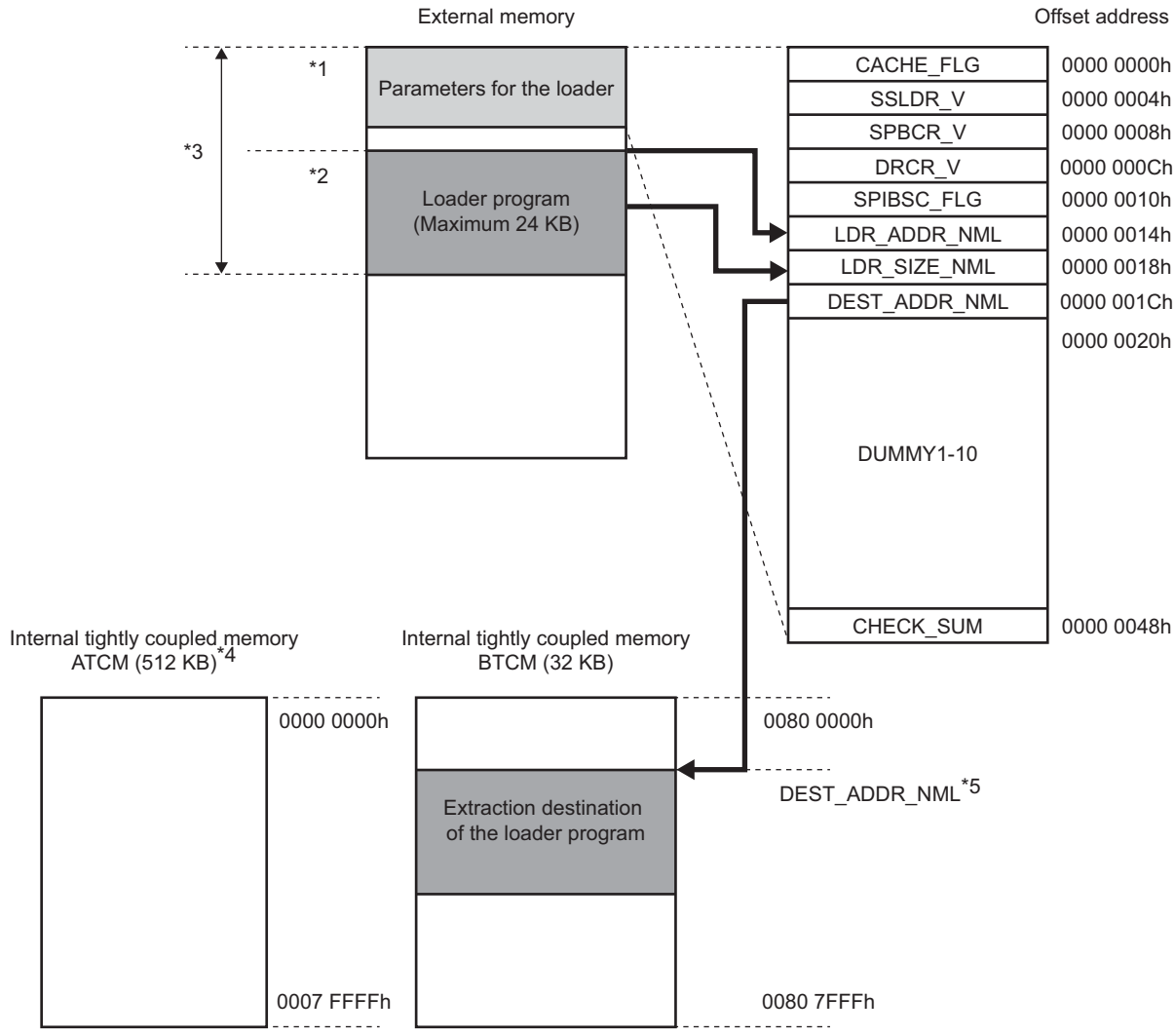
(1) Set the bus controller specified by the operating mode.
(2) Transfer the parameters for the loader from an external memory.
(3) Configure the bus controller based on the parameters for the loader.
(4) Transfer the loader program from an external memory.
(5) Hand over the processing to the loader program in the TCM area.

Figure 3.1 Operating Overview of Boot Processing

3.5.2 Parameters for the Loader

The parameters for the loader are setting parameters for boot processing, which are loaded from an external memory during boot processing and used by the boot function. The parameters for the loader specify information, such as the cache settings during boot processing in individual operating modes, setting of the bus controller (SPIBSC) used for communication with an external memory, and the size of the loader program.

Figure 3.2 shows memory assignment of the loader program and parameters for the loader.



Note 1. The parameters for the loader must be stored in the following address range:
SPI boot mode: SPI multi-I/O bus space (3000 0000h and higher)
Note 2. The loader program can be stored in any address (LDR_ADDR_NML). Note that, however, the program size (LDR_SIZE_NML) must be multiples of 512 bytes and no more than 24 Kbytes.
Note 3. In SPI boot mode, address output is 3 bytes, so only use addresses in the range from 3000 0000h to 30FF FFFFh.
Note 4. The ATCM area is assumed to be used for storing user application programs.
Note 5. Specify the extraction destination area of the loader program so that the area falls in the range from 0080 2000h to 0080 7FFFh.

Figure 3.2 Memory Assignment of the Loader Program and Parameters for the Loader

Table 3.3 describes parameter information for the loader in SPI boot mode.

Table 3.3 Parameter Information for the Loader in SPI Boot Mode

Offset Address	Parameter Name	Description
0000 0000h	CACHE_FLG	Selects whether to enable the I1 cache and D1 cache of Cortex-R4 at boot processing (for speeding up). 0000 0001h: Enables the I1 and D1 caches. Other setting values than above: Disables the I1 and D1 caches.
0000 0004h	SSLDR_V	Setting value of the SSL delay register (SSLDR) This parameter value is set to the SSLDR register during the setting for speeding up of SPIBSC in (3) of section 3.5.1, Boot Function.*1
0000 0008h	SPBCR_V	Setting value of the bit-rate configuration register (SPBCR) This parameter value is set to the SPBCR register during the setting for speeding up of SPIBSC in (3) of section 3.5.1, Boot Function.*1
0000 000Ch	DRCR_V	Setting value of the data read control register (DRCR) This parameter value is set to the DRCR register during the setting for speeding up of SPIBSC in (3) of section 3.5.1, Boot Function.*1
0000 0010h	SPIBSC_FLG	Selects whether to change the SPIBSC setting back to the initial value after the boot processing finishes. 2236 0679h: Changes the SPIBSC setting value back to the initial value after the boot processing finishes. Other setting values than above: Retains the SPIBCS setting value used during boot processing.*2
0000 0014h	LDR_ADDR_NML	Sets the start address of the loader program stored in the external memory.*3
0000 0018h	LDR_SIZE_NML	Specifies the size of the loader program. Note that the program size must be multiples of 512 bytes and no more than 24 Kbytes.*3
0000 001Ch	DEST_ADDR_NML	Specifies the start address of the tightly coupled memory (BTCM) that is used as the extraction destination of the loader program. Specify the extraction destination area of the loader program so that the area falls in the range from 0080 2000h to 0080 7FFFh.
0000 0020h	DUMMY1	Option (Not used in this mode.)
0000 0024h	DUMMY2	Option (Not used in this mode.)
0000 0028h	DUMMY3	Option (Not used in this mode.)
0000 002Ch	DUMMY4	Option (Not used in this mode.)
0000 0030h	DUMMY5	Option (Not used in this mode.)
0000 0034h	DUMMY6	Option (Not used in this mode.)
0000 0038h	DUMMY7	Option (Not used in this mode.)
0000 003Ch	DUMMY8	Option (Not used in this mode.)
0000 0040h	DUMMY9	Option (Not used in this mode.)
0000 0044h	DUMMY10	Option (Not used in this mode.)
0000 0048h	CHECK_SUM	Checksum value of the parameters for the loader This parameter specifies the sum of the higher-order 16 bits and the lower-order 16 bits of the parameters (in unsigned long (32-bit) format) in the range of the offset addresses 0000h to 0044h.*4

Note 1. For details about the SSLDR, SPBCR, and DRCR registers, see section 27, SPI Multi I/O Bus Controller (SPIBSC).
Note 2. For details on the settings of the individual peripheral modules after boot processing finishes, see section 3.5.4.1, Operation Settings in SPI Boot Mode.
Note 3. LDR_ADDR_NML must be in the range from 3000 004Ch and LDR_ADDR_NML + LDR_SIZE_NML ≤ 3100 0000h in the external address space (SPI).
Note 4. An example for calculating CHECK_SUM is given below.
If SSLDR_V = 0007 0707h,
SPBCR_V = 0000 0003h,
LDR_ADDR_NML = 3000 004Ch,
LDR_SIZE_NML = 0000 6000h,
DEST_ADDR_NML = 0080 2000h, and
others = 0000 0000h,
CHECK_SUM is calculated as below ((0000h) is omitted in the formula):
CHECK_SUM = (0007h) + (0707h) + (0003h) + (3000h) + (004Ch) + (6000h) + (0080h) + (2000h) = (0000 B7DDh)

3.5.3 Loader Program

The loader program is a user program that is transferred from an external memory to the internal tightly coupled memory (TCM) by the boot function, and starts its processing after the boot processing finishes. The loader program can execute such processing that suites the user system, for example, extracting a user application program from an external memory to the internal TCM area and executing it at high speed.
Set the loader program so that the following conditions are satisfied:

- Program size (LDR_SIZE_NML): Multiples of 512 bytes and no more than 24 Kbytes
- Storage address in the external memory (LDR_ADDR_NML) in SPI boot mode: Address range from 3000 004Ch and LDR_ADDR_NML + LDR_SIZE_NML ≤ 3100 0000h

These setting values must be stored in an external memory as parameters for the loader. For details, see section 3.5.2, Parameters for the Loader.

3.5.4 SPI Boot Mode (Serial Flash)

In SPI boot mode, this LSI boots a program from an external serial flash memory connected to the SPI multi-I/O bus space.
In this mode, the SPI multi-I/O bus controller is set to the mode of reading the external address space, and the SPBCLK, SPBSSL, SPBMO, and SPBMI pins are enabled.
After the reset is released, this LSI executes the boot processing. The loader program stored in a serial flash memory connected to the SPI multi-I/O bus space is extracted to the internal memory (TCM), and then the processing is executed.
Figure 3.3 shows the connection diagram of this LSI with a serial flash memory.

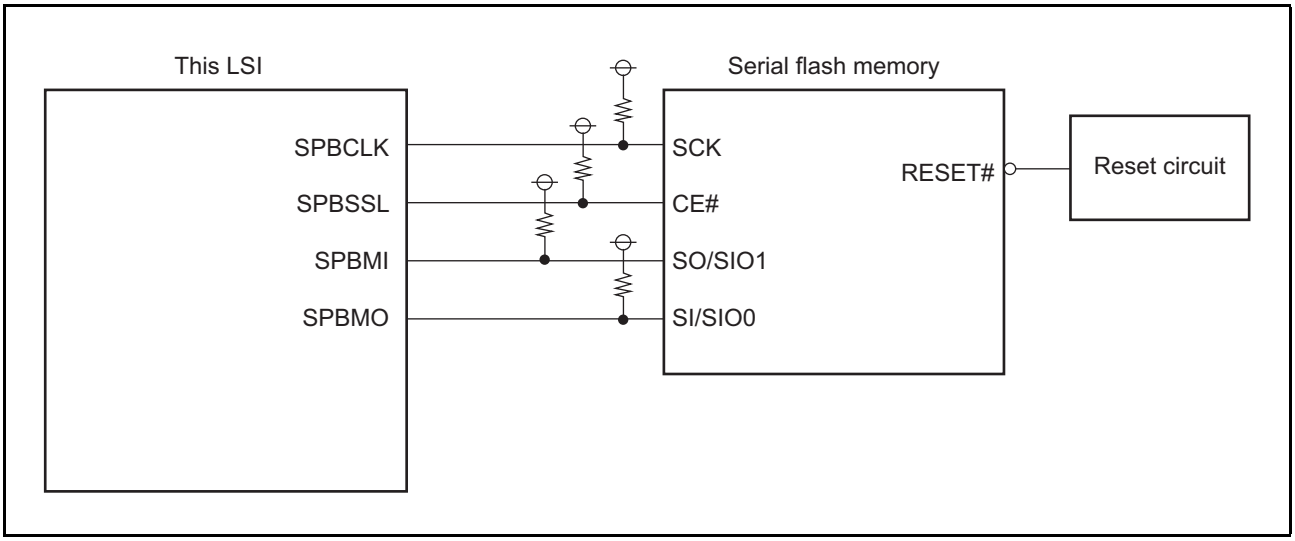


Figure 3.3 Connection Diagram of This LSI with a Serial Flash Memory

3.5.4.1 Operation Settings in SPI Boot Mode

Immediately after the boot processing starts in SPI boot mode after the reset is released, this LSI operates with the following initial setting values, and executes processing until transferring parameters for the loader.

- CPU clock (CPUCLK): 150 MHz
- SPIBSC bit rate (SPBCLK): 18.75 MHz
- Supported command: Read (03h)
- Address output: 3 bytes
- Dummy cycles: None
- Data read width: 1 bit
- SPI mode: CPOL = 0 (positive pulse)
  CPHAR = 0 (data reception at odd edge)
  CPHAT = 0 (data transmission at even edge)

After the parameters for the loader are loaded, the settings for the I1 and D1 caches of Cortex-R4 and for the SSLDR, SPBCR, and DRCR registers are performed based on the values of parameters CACHE_FLG, SSLDR_V, SPBCR_V, and DRCR_V, so that the processing can be speed up.

Table 3.4 describes the setting values of the individual peripheral modules and registers at the time SPI boot mode finishes.

Also, Table 3.5 describes the setting values of Arm general-purpose registers at the time the boot processing finishes, and Table 3.6 describes the status of the Arm CP15 registers at the time the boot finishes.

Table 3.4 Setting Values of the Individual Peripheral Modules and Registers at the Time SPI Boot Mode Finishes

		Setting Value at the Time the Boot Processing Finishes	
Peripheral Module	**Register**	**When SPIBSC is initialized (SPIBSC_FLG = 2236 0679h)**	**When SPIBSC is not initialized (SPIBSC_FLG ≠ 2236 0679h)**
Low power consumption	MSTPCRC	0000 7DFEh (Initial value)	0000 7DFEh
SPIBSC	SSLDR	0007 0707h (Initial value)	Setting value of SSLDR_V
	SPBCR	0000 0003h (Initial value)	Setting value of SPBCR_V
	DRCR	0000 0000h (Initial value)	Setting value of DRCR_V
I/O ports	PORT6 .PMR	1Dh*1	1Dh*1
	MPC.PmnPFS	1Bh*1	1Bh*1

Note 1. Bits corresponding to the SPBCLK, SPBSSL, SPBMI, and SPBMO pins

Table 3.5 Setting Values of the Arm General-Purpose Registers at the Time the Boot Processing Finishes

No.	Register Name	Setting Values for Individual Processor Modes					
		User Mode/Current Mode	IRQ	FIQ	Undef	Abort	SVC
1	R0	Undefined	—	—	—	—	—
2	R1	Undefined	—	—	—	—	—
3	R2	Undefined	—	—	—	—	—
4	R3	Undefined	—	—	—	—	—
5	R4	Undefined	—	—	—	—	—
6	R5	Undefined	—	—	—	—	—
7	R6	Undefined	—	—	—	—	—
8	R7	Undefined	—	—	—	—	—
9	R8	Undefined	—	Undefined	—	—	—
10	R9	Undefined	—	Undefined	—	—	—
11	R10	Undefined	—	Undefined	—	—	—
12	R11	Undefined	—	Undefined	—	—	—
13	R12	Undefined	—	Undefined	—	—	—
14	R13(sp)	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
15	R14(lr)	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
16	R15(pc)	Arbitrary	—	—	—	—	—
17	cpsr	xxxx xx93h ([31:8] is undefined.) [7]I = 1 [6]F = 0*1 [5]T = 0 [4:0]MD = 10011b(SVC)	—	—	—	—	—
18	spsr	—	Undefined	Undefined	Undefined	Undefined	Undefined

—: Non-existent register

sp: Stack pointer

lr: Link register (which stores the return address when calling a subroutine call)

pc: Program counter

cpsr: Abbreviation of “current program status register”. It monitors or controls internal operations.

spsr: Abbreviation of “saved program status register”. It saves cpsr in the previous mode.

Note 1. In this product, the non-maskable interrupt has been assigned to FIQ. The boot processing changes the [6]F bit of the CPSR register from 1 to 0 to enable non-maskable interrupt after the boot processing finishes.

Note: After the boot processing sets the [6]F bit in the CPSR register to 0, if a non-maskable interrupt (FIQ exception) occurs before the processing reaches the branch to the loader program, the processing is jumped to the FIQ exception handler address and goes into an infinite loop. For details, see section 3.5.7, Note.

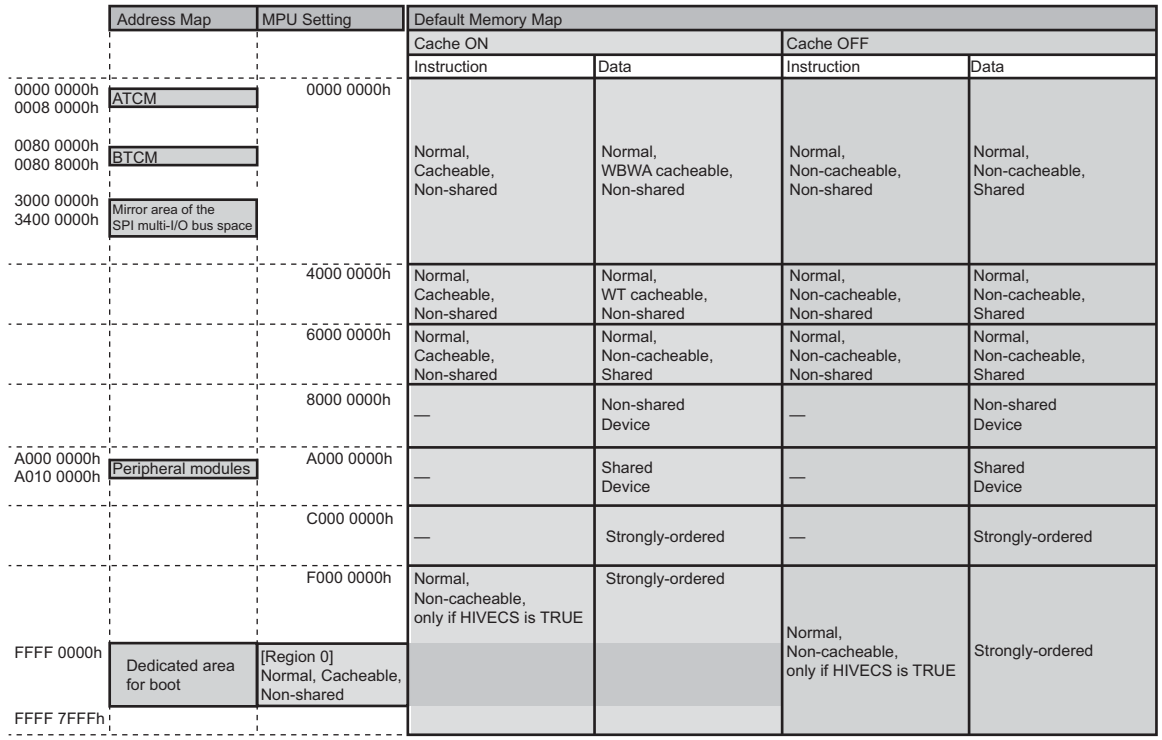
Table 3.6 Status of the Arm CP15 Registers at the Time the Boot Finishes

Register Name	Symbol	Setting Value at the Time the Boot Processing Finishes	Remarks
System control register	SCTLR	09E5 2878h*1	[24]VE = 1: Sets the IRQ exception vector address in VIC.
System control auxiliary register	ACTLR	0E00 0020h (When ATCM and BTCM are used)	ECC enable for TCM All areas of ATCM and BTCM are written and processed in 32-bit units during boot processing and initialized.
Invalidate all Instruction Caches Register	—	—	The I1 cache entry is not invalidated after the boot processing finishes.
Invalidate all Data Caches Register	—	—	The D1 cache entry is not invalidated after the boot processing finishes.
MPU Memory Region Number Register	RGNR	0000 0000h	All MPU settings are initialized even when the cache is enabled by the parameters for the loader.
Data Region Base Address Register	DRBAR	0000 0000h	
Data Region Size and Enable Register	DRSR	0000 0000h	
Data Region Access Control Register	DRACR	0000 0000h	

Note 1. When the boot processing finishes, the register is in high vector status, where V[13] = 1 (FFFF 0000h). Use the loader program to write appropriate processing in the low vectors (0000 0000h), and then to change the register to low vector status, where V[13] = 0 (0000 0000h).

3.5.5 MPU Setting

The boot function uses the primary instruction cache (I1) and primary data cache (D1) of Cortex-R4 when the parameter CACHE_FLG for the loader is set to 0000 0001h.
However, the dedicated area for boot processing (FFFF 0000h to FFFF 7FFFh), which is used by the boot function, is set as the non-cache area in the default map of Cortex-R4, so the MPU (memory protection unit) redefines the cache area during the boot processing.
The boot function defines the high-vector area (FFFF 0000h to FFFF 7FFFh) as the cache area of Region 0, and uses other areas for the default memory map.
When the boot processing finishes, the I1 and D1 caches are invalidated, and all areas are initialized to the default memory map.
Figure 3.4 shows the relationship between the memory map definition during the boot processing and the default memory map of Cortex-R4.



Note 1. Because SCTLR[17] BR is set to 1, the default memory map is applied to the areas for which no region is set.

Figure 3.4 Relationship Between the Memory Map Definition During the Boot Processing and the Default Memory Map of Cortex-R4

3.5.6 Boot-Related Information and Error Processing

The boot function determines whether the boot processing finishes normally, and retains the result in a specific address. If the processing is determined to be an error, the boot processing is aborted, and an infinite loop is executed.
If the debugger is connected, reading the result of the boot processing that was stored in a specific address at the point of break can determine the error source.
Table 3.7 describes the error sources and the results of the boot processing.

Table 3.7 Error Sources and Results of Boot Processing

Storage Address*1	Stored Value*1	Error Sources and Results of Boot Processing
0080 09C4h	0	The boot processing finished normally.
	-1	A mode error occurred. When the setting is prohibited in the read value of the mode monitor register (MDMONR)
	-2	Checksum error of the parameters for the loader When the checksum (CHECK_SUM) of the parameters for the loader does not match
	-3	Error in a parameter for the loader When one of the following is satisfied: - The size of the loader program is smaller than 512 bytes. - The size of the loader program is larger than 24 Kbytes. - The size of the loader program is not a multiple of 512 bytes. - The destination address of the loader program is outside of the TCM area.

Note 1. The access size is 32 bits.

3.5.7 Note

3.5.7.1 Exception Processing

Only the reset exception due to the RES#-pin reset can be accepted during the boot processing. When a reset exception occurs, this LSI is reset, and the boot processing restarts. If an exception processing other than reset exception occurs, the jump instruction to the relevant exception handler address repeats an infinite loop.

Table 3.8 Exception Processing During the Boot Processing

Exception	Handler Address	Operation During the Boot Processing
Reset exception	FFFF 0000h	Branched to the reset exception handler
Undefined instruction exception	FFFF 0004h	Branched to the undefined instruction exception handler (Infinite loop)
Software interrupt exception	FFFF 0008h	Branched to the software interrupt exception handler (Infinite loop)
Prefetch abort exception	FFFF 000Ch	Branched to the prefetch abort exception handler (Infinite loop)
Data abort exception	FFFF 0010h	Branched to the data abort exception handler (Infinite loop)
IRQ exception	FFFF 0018h	Branched to the IRQ exception handler (Infinite loop)
FIQ exception	FFFF 001Ch	Branched to the FIQ exception handler (Infinite loop)

Note: Before the boot processing finishes, the register is in high vector status, where SCTLR V[13] = 1 (FFFF 0000h). Use the loader program to write appropriate processing in the low vectors (0000 0000h), and then to change the register to low vector status, where V[13] = 0 (0000 0000h).

3.5.7.2 Serial Flash Memory in SPI Boot Mode

In SPI boot mode, after release from the reset state, boot processing starts by reading from the serial flash memory via the SPI multi-I/O bus controller (SPIBSC) with the initial settings given in section 3.5.4.1, Operation Settings in SPI Boot Mode.

The setting of serial flash memory can be changed via the SPIBSC after the processing to boot up is finished. Depending on the settings, however, reading from the serial flash memory may not be possible when boot processing needs to be started again following a reset. Therefore, caution is required on this point.

When the active level of the signal on the RES# pin is applied to reset this LSI chip, the serial flash memory can be simultaneously initialized by input of the same reset signal to the reset pin of the serial flash memory. Therefore, we recommend using serial flash memory that includes a reset pin. As the reset signal of a serial flash memory in a small package may be multiplexed with another pin function, make sure that the reset function is selected.

In addition, when an internal reset such as a software reset or ECM reset is to be generated, initialize the serial flash memory by software in advance so that it can be connected in boot processing.

4. Address Space

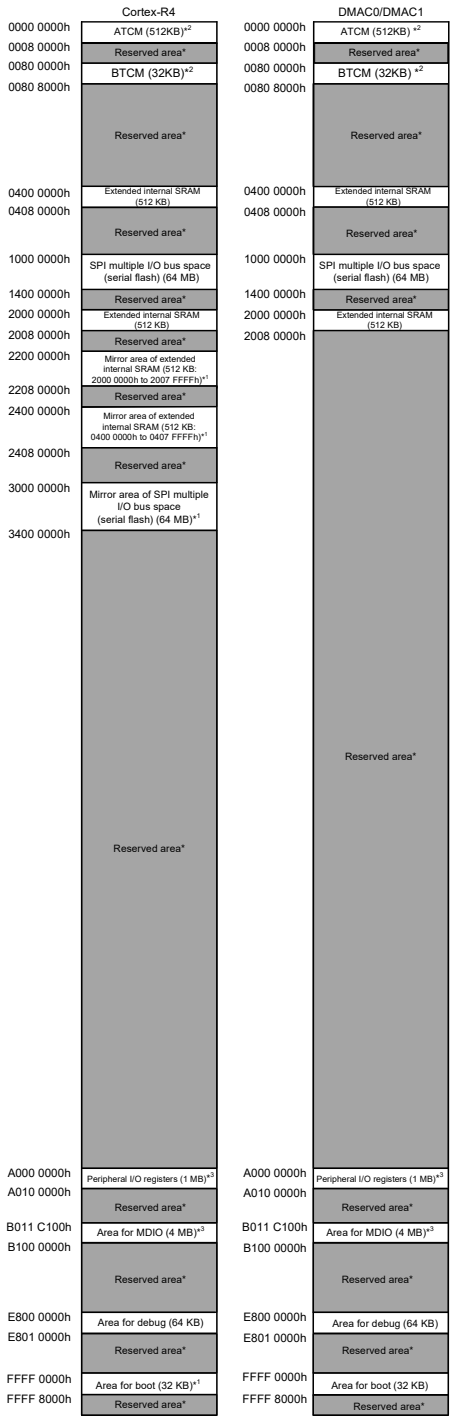
4.1 Address Space

This LSI has a 4-Gbyte address space ranging from 0000 0000h to FFFF FFFFh. That is, up to total 4 Gbytes of program and data areas can be accessed linearly.

Figure 4.1 and Figure 4.2 show the memory maps for respective products.

Accessible areas will differ depending on the operating mode and the states of control bits.

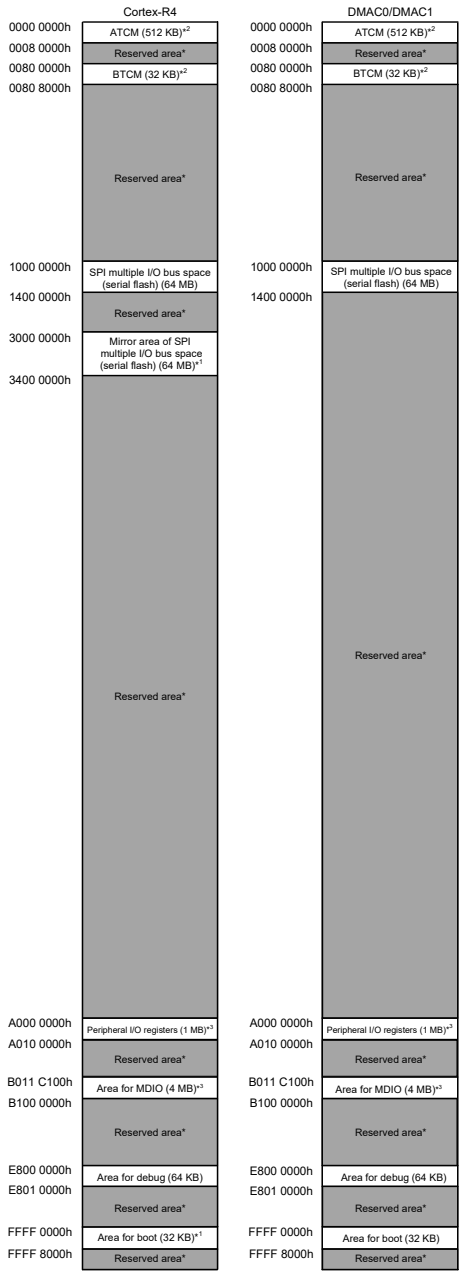
In addition, since a non-cached access area from each bus master is assigned to the same area in this product, in access from the Cortex-R4, a mirror area is set as a cache-enabled area. The MPU should not enable caching of areas other than the mirror area.



Note 1. Can be cached.
(For other areas, caching should not be enabled by the MPU.)
Note 2. Cortex-R4 tightly coupled memory space
Note 3. Peripheral space

Note: Reserved areas should not be accessed.

Figure 4.1 Memory Map (1-Mbyte Extended Internal SRAM)



Note 1. Can be cached.
(For other areas, caching should not be enabled by the MPU.)
Note 2. Cortex-R4 tightly coupled memory space
Note 3. Peripheral space

Note: Reserved areas should not be accessed.

Figure 4.2 Memory Map (0-Kbyte Extended Internal SRAM)

5. I/O Registers

This section gives information on the on-chip I/O register addresses. The information is given as shown below.

(1) I/O register addresses (address order)

- Registers are listed from the lower allocation addresses.
- Registers are classified according to module symbols.
- Among the internal I/O register area, addresses not listed in the list of registers are reserved. Reserved addresses must not be accessed. Do not access these addresses; otherwise, the operation when accessing these bits and subsequent operations cannot be guaranteed.

5.1 I/O Register Addresses (Address Order)

Table 5.1 List of I/O Registers (Address Order) (1 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A000 0000h	PORT0	Port direction register	PDR	16	16
A000 0002h	PORT1	Port direction register	PDR	16	16
A000 0004h	PORT2	Port direction register	PDR	16	16
A000 0006h	PORT3	Port direction register	PDR	16	16
A000 0008h	PORT4	Port direction register	PDR	16	16
A000 000Ah	PORT5	Port direction register	PDR	16	16
A000 000Ch	PORT6	Port direction register	PDR	16	16
A000 000Eh	PORT7	Port direction register	PDR	16	16
A000 0012h	PORT9	Port direction register	PDR	16	16
A000 0014h	PORTA	Port direction register	PDR	16	16
A000 0018h	PORTC	Port direction register	PDR	16	16
A000 001Ch	PORTE	Port direction register	PDR	16	16
A000 0020h	PORTG	Port direction register	PDR	16	16
A000 0040h	PORT0	Port output data register	PODR	8	8
A000 0041h	PORT1	Port output data register	PODR	8	8
A000 0042h	PORT2	Port output data register	PODR	8	8
A000 0043h	PORT3	Port output data register	PODR	8	8
A000 0044h	PORT4	Port output data register	PODR	8	8
A000 0046h	PORT6	Port output data register	PODR	8	8
A000 0047h	PORT7	Port output data register	PODR	8	8
A000 0049h	PORT9	Port output data register	PODR	8	8
A000 004Ah	PORTA	Port output data register	PODR	8	8
A000 004Ch	PORTC	Port output data register	PODR	8	8
A000 004Eh	PORTE	Port output data register	PODR	8	8
A000 0050h	PORTG	Port output data register	PODR	8	8
A000 0060h	PORT0	Port input data register	PIDR	8	8
A000 0061h	PORT1	Port input data register	PIDR	8	8
A000 0062h	PORT2	Port input data register	PIDR	8	8
A000 0063h	PORT3	Port input data register	PIDR	8	8
A000 0064h	PORT4	Port input data register	PIDR	8	8
A000 0066h	PORT6	Port input data register	PIDR	8	8
A000 0067h	PORT7	Port input data register	PIDR	8	8
A000 0069h	PORT9	Port input data register	PIDR	8	8
A000 006Ah	PORTA	Port input data register	PIDR	8	8
A000 006Ch	PORTC	Port input data register	PIDR	8	8
A000 006Eh	PORTE	Port input data register	PIDR	8	8
A000 0070h	PORTG	Port input data register	PIDR	8	8
A000 0080h	PORT0	Port mode register	PMR	8	8
A000 0081h	PORT1	Port mode register	PMR	8	8
A000 0082h	PORT2	Port mode register	PMR	8	8
A000 0083h	PORT3	Port mode register	PMR	8	8
A000 0084h	PORT4	Port mode register	PMR	8	8
A000 0085h	PORT5	Port mode register	PMR	8	8
A000 0086h	PORT6	Port mode register	PMR	8	8
A000 0087h	PORT7	Port mode register	PMR	8	8
A000 0089h	PORT9	Port mode register	PMR	8	8

Table 5.1 List of I/O Registers (Address Order) (2 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A000 008Ah	PORTA	Port mode register	PMR	8	8
A000 008Ch	PORTC	Port mode register	PMR	8	8
A000 008Eh	PORTE	Port mode register	PMR	8	8
A000 0090h	PORTG	Port mode register	PMR	8	8
A000 0100h	PORT0	Pull-up/pull-down control register	PCR	16	16
A000 0102h	PORT1	Pull-up/pull-down control register	PCR	16	16
A000 0104h	PORT2	Pull-up/pull-down control register	PCR	16	16
A000 0106h	PORT3	Pull-up/pull-down control register	PCR	16	16
A000 0108h	PORT4	Pull-up/pull-down control register	PCR	16	16
A000 010Ah	PORT5	Pull-up/pull-down control register	PCR	16	16
A000 010Ch	PORT6	Pull-up/pull-down control register	PCR	16	16
A000 010Eh	PORT7	Pull-up/pull-down control register	PCR	16	16
A000 0112h	PORT9	Pull-up/pull-down control register	PCR	16	16
A000 0114h	PORTA	Pull-up/pull-down control register	PCR	16	16
A000 011Ch	PORTE	Pull-up/pull-down control register	PCR	16	16
A000 0120h	PORTG	Pull-up/pull-down control register	PCR	16	16
A000 014Ah	PORT5	Driving ability control register	DSCR	16	16
A000 0200h	MPC	Port 00 pin function control register	P00PFS	8	8
A000 0208h	MPC	Port 10 pin function control register	P10PFS	8	8
A000 0211h	MPC	Port 21 pin function control register	P21PFS	8	8
A000 0212h	MPC	Port 22 pin function control register	P22PFS	8	8
A000 0217h	MPC	Port 27 pin function control register	P27PFS	8	8
A000 021Bh	MPC	Port 33 pin function control register	P33PFS	8	8
A000 021Ch	MPC	Port 34 pin function control register	P34PFS	8	8
A000 021Dh	MPC	Port 35 pin function control register	P35PFS	8	8
A000 0220h	MPC	Port 40 pin function control register	P40PFS	8	8
A000 0222h	MPC	Port 42 pin function control register	P42PFS	8	8
A000 0224h	MPC	Port 44 pin function control register	P44PFS	8	8
A000 0228h	MPC	Port 50 pin function control register	P50PFS	8	8
A000 0229h	MPC	Port 51 pin function control register	P51PFS	8	8
A000 022Ah	MPC	Port 52 pin function control register	P52PFS	8	8
A000 022Bh	MPC	Port 53 pin function control register	P53PFS	8	8
A000 022Ch	MPC	Port 54 pin function control register	P54PFS	8	8
A000 022Dh	MPC	Port 55 pin function control register	P55PFS	8	8
A000 022Eh	MPC	Port 56 pin function control register	P56PFS	8	8
A000 0230h	MPC	Port 60 pin function control register	P60PFS	8	8
A000 0231h	MPC	Port 61 pin function control register	P61PFS	8	8
A000 0232h	MPC	Port 62 pin function control register	P62PFS	8	8
A000 0233h	MPC	Port 63 pin function control register	P63PFS	8	8
A000 0234h	MPC	Port 64 pin function control register	P64PFS	8	8
A000 0235h	MPC	Port 65 pin function control register	P65PFS	8	8
A000 0239h	MPC	Port 71 pin function control register	P71PFS	8	8
A000 023Ah	MPC	Port 72 pin function control register	P72PFS	8	8
A000 023Bh	MPC	Port 73 pin function control register	P73PFS	8	8
A000 0248h	MPC	Port 90 pin function control register	P90PFS	8	8
A000 0249h	MPC	Port 91 pin function control register	P91PFS	8	8
A000 024Ah	MPC	Port 92 pin function control register	P92PFS	8	8
A000 024Bh	MPC	Port 93 pin function control register	P93PFS	8	8

Table 5.1 List of I/O Registers (Address Order) (3 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A000 024Ch	MPC	Port 94 pin function control register	P94PFS	8	8
A000 024Dh	MPC	Port 95 pin function control register	P95PFS	8	8
A000 024Eh	MPC	Port 96 pin function control register	P96PFS	8	8
A000 024Fh	MPC	Port 97 pin function control register	P97PFS	8	8
A000 0253h	MPC	Port A3 pin function control register	PA3PFS	8	8
A000 0254h	MPC	Port A4 pin function control register	PA4PFS	8	8
A000 0255h	MPC	Port A5 pin function control register	PA5PFS	8	8
A000 0262h	MPC	Port C2 pin function control register	PC2PFS	8	8
A000 0263h	MPC	Port C3 pin function control register	PC3PFS	8	8
A000 0266h	MPC	Port C6 pin function control register	PC6PFS	8	8
A000 0267h	MPC	Port C7 pin function control register	PC7PFS	8	8
A000 0270h	MPC	Port E0 pin function control register	PE0PFS	8	8
A000 0271h	MPC	Port E1 pin function control register	PE1PFS	8	8
A000 0272h	MPC	Port E2 pin function control register	PE2PFS	8	8
A000 0273h	MPC	Port E3 pin function control register	PE3PFS	8	8
A000 0274h	MPC	Port E4 pin function control register	PE4PFS	8	8
A000 0275h	MPC	Port E5 pin function control register	PE5PFS	8	8
A000 0276h	MPC	Port E6 pin function control register	PE6PFS	8	8
A000 0277h	MPC	Port E7 pin function control register	PE7PFS	8	8
A000 0282h	MPC	Port G2 pin function control register	PG2PFS	8	8
A000 0283h	MPC	Port G3 pin function control register	PG3PFS	8	8
A000 0284h	MPC	Port G4 pin function control register	PG4PFS	8	8
A000 0285h	MPC	Port G5 pin function control register	PG5PFS	8	8
A000 0286h	MPC	Port G6 pin function control register	PG6PFS	8	8
A000 02FFh	MPC	Write protection register	PWPR	8	8
A000 5000h	SPIBSC	Common control register	CMNCR	32	32
A000 5004h	SPIBSC	SSL delay register	SSLDR	32	32
A000 5008h	SPIBSC	Bit rate register	SPBCR	32	32
A000 500Ch	SPIBSC	Data read control register	DRCR	32	32
A000 5010h	SPIBSC	Data read command setting register	DRCMR	32	32
A000 5014h	SPIBSC	Data read extended address setting register	DREAR	32	32
A000 5018h	SPIBSC	Data read option setting register	DROPR	32	32
A000 501Ch	SPIBSC	Data read enable setting register	DRENR	32	32
A000 5020h	SPIBSC	SPI mode control register	SMCR	32	32
A000 5024h	SPIBSC	SPI mode command setting register	SMCMR	32	32
A000 5028h	SPIBSC	SPI mode address setting register	SMADR	32	32
A000 502Ch	SPIBSC	SPI mode option setting register	SMOPR	32	32
A000 5030h	SPIBSC	SPI mode enable setting register	SMENR	32	32
A000 5038h	SPIBSC	SPI mode read data register 0	SMRDR0	32	8, 16, 32
A000 5040h	SPIBSC	SPI mode write data register 0	SMWDR0	32	8, 16, 32
A000 5048h	SPIBSC	Common status register	CMNSR	32	32
A000 5058h	SPIBSC	Data read dummy cycle setting register	DRDMCR	32	32
A000 5060h	SPIBSC	SPI mode dummy cycle setting register	SMDMCR	32	32
A001 0000h	VIC	IRQ status register 0	IRQS0	32	32
A001 0004h	VIC	IRQ status register 1	IRQS1	32	32
A001 0008h	VIC	IRQ status register 2	IRQS2	32	32
A001 000Ch	VIC	IRQ status register 3	IRQS3	32	32
A001 0010h	VIC	IRQ status register 4	IRQS4	32	32

Table 5.1 List of I/O Registers (Address Order) (4 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A001 0014h	VIC	IRQ status register 5	IRQS5	32	32
A001 0018h	VIC	IRQ status register 6	IRQS6	32	32
A001 001Ch	VIC	IRQ status register 7	IRQS7	32	32
A001 0040h	VIC	Interrupt input status register 0	RAIS0	32	32
A001 0044h	VIC	Interrupt input status register 1	RAIS1	32	32
A001 0048h	VIC	Interrupt input status register 2	RAIS2	32	32
A001 004Ch	VIC	Interrupt input status register 3	RAIS3	32	32
A001 0050h	VIC	Interrupt input status register 4	RAIS4	32	32
A001 0054h	VIC	Interrupt input status register 5	RAIS5	32	32
A001 0058h	VIC	Interrupt input status register 6	RAIS6	32	32
A001 005Ch	VIC	Interrupt input status register 7	RAIS7	32	32
A001 0080h	VIC	Interrupt enable register 0	IEN0	32	32
A001 0084h	VIC	Interrupt enable register 1	IEN1	32	32
A001 0088h	VIC	Interrupt enable register 2	IEN2	32	32
A001 008Ch	VIC	Interrupt enable register 3	IEN3	32	32
A001 0090h	VIC	Interrupt enable register 4	IEN4	32	32
A001 0094h	VIC	Interrupt enable register 5	IEN5	32	32
A001 0098h	VIC	Interrupt enable register 6	IEN6	32	32
A001 009Ch	VIC	Interrupt enable register 7	IEN7	32	32
A001 00A0h	VIC	Interrupt enable clear register 0	IEC0	32	32
A001 00A4h	VIC	Interrupt enable clear register 1	IEC1	32	32
A001 00A8h	VIC	Interrupt enable clear register 2	IEC2	32	32
A001 00ACh	VIC	Interrupt enable clear register 3	IEC3	32	32
A001 00B0h	VIC	Interrupt enable clear register 4	IEC4	32	32
A001 00B4h	VIC	Interrupt enable clear register 5	IEC5	32	32
A001 00B8h	VIC	Interrupt enable clear register 6	IEC6	32	32
A001 00BCh	VIC	Interrupt enable clear register 7	IEC7	32	32
A001 0100h	VIC	Interrupt detection type selection register 0	PLS0	32	32
A001 0104h	VIC	Interrupt detection type selection register 1	PLS1	32	32
A001 0108h	VIC	Interrupt detection type selection register 2	PLS2	32	32
A001 010Ch	VIC	Interrupt detection type selection register 3	PLS3	32	32
A001 0110h	VIC	Interrupt detection type selection register 4	PLS4	32	32
A001 0114h	VIC	Interrupt detection type selection register 5	PLS5	32	32
A001 0118h	VIC	Interrupt detection type selection register 6	PLS6	32	32
A001 011Ch	VIC	Interrupt detection type selection register 7	PLS7	32	32
A001 0120h	VIC	Edge detection bit clear register 0	PIC0	32	32
A001 0124h	VIC	Edge detection bit clear register 1	PIC1	32	32
A001 0128h	VIC	Edge detection bit clear register 2	PIC2	32	32
A001 012Ch	VIC	Edge detection bit clear register 3	PIC3	32	32
A001 0130h	VIC	Edge detection bit clear register 4	PIC4	32	32
A001 0134h	VIC	Edge detection bit clear register 5	PIC5	32	32
A001 0138h	VIC	Edge detection bit clear register 6	PIC6	32	32
A001 013Ch	VIC	Edge detection bit clear register 7	PIC7	32	32
A001 01C0h	VIC	Interrupt priority level mask register 0	PRLM0	32	32
A001 01C4h	VIC	Interrupt priority level mask clear register 0	PRLC0	32	32
A001 01C8h	VIC	User mode enable register 0	UEN0	32	32
A001 0200h	VIC	Interrupt address register 0	HVA0	32	32
A001 0210h	VIC	Interrupt service status register 0	ISS0	32	32

Table 5.1 List of I/O Registers (Address Order) (5 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A001 0214h	VIC	Interrupt service status register 1	ISS1	32	32
A001 0218h	VIC	Interrupt service status register 2	ISS2	32	32
A001 021Ch	VIC	Interrupt service status register 3	ISS3	32	32
A001 0220h	VIC	Interrupt service status register 4	ISS4	32	32
A001 0224h	VIC	Interrupt service status register 5	ISS5	32	32
A001 0228h	VIC	Interrupt service status register 6	ISS6	32	32
A001 022Ch	VIC	Interrupt service status register 7	ISS7	32	32
A001 0230h	VIC	Interrupt service current register 0	ISC0	32	32
A001 0234h	VIC	Interrupt service current register 1	ISC1	32	32
A001 0238h	VIC	Interrupt service current register 2	ISC2	32	32
A001 023Ch	VIC	Interrupt service current register 3	ISC3	32	32
A001 0240h	VIC	Interrupt service current register 4	ISC4	32	32
A001 0244h	VIC	Interrupt service current register 5	ISC5	32	32
A001 0248h	VIC	Interrupt service current register 6	ISC6	32	32
A001 024Ch	VIC	Interrupt service current register 7	ISC7	32	32
A001 0404h	VIC	Interrupt address storage register 1	VAD1	32	32
A001 0408h	VIC	Interrupt address storage register 2	VAD2	32	32
A001 040Ch	VIC	Interrupt address storage register 3	VAD3	32	32
A001 0410h	VIC	Interrupt address storage register 4	VAD4	32	32
A001 0414h	VIC	Interrupt address storage register 5	VAD5	32	32
A001 0418h	VIC	Interrupt address storage register 6	VAD6	32	32
A001 041Ch	VIC	Interrupt address storage register 7	VAD7	32	32
A001 0420h	VIC	Interrupt address storage register 8	VAD8	32	32
A001 0424h	VIC	Interrupt address storage register 9	VAD9	32	32
A001 0428h	VIC	Interrupt address storage register 10	VAD10	32	32
A001 042Ch	VIC	Interrupt address storage register 11	VAD11	32	32
A001 0430h	VIC	Interrupt address storage register 12	VAD12	32	32
A001 0434h	VIC	Interrupt address storage register 13	VAD13	32	32
A001 0438h	VIC	Interrupt address storage register 14	VAD14	32	32
A001 043Ch	VIC	Interrupt address storage register 15	VAD15	32	32
A001 0440h	VIC	Interrupt address storage register 16	VAD16	32	32
A001 0444h	VIC	Interrupt address storage register 17	VAD17	32	32
A001 0448h	VIC	Interrupt address storage register 18	VAD18	32	32
A001 044Ch	VIC	Interrupt address storage register 19	VAD19	32	32
A001 0450h	VIC	Interrupt address storage register 20	VAD20	32	32
A001 0454h	VIC	Interrupt address storage register 21	VAD21	32	32
A001 0458h	VIC	Interrupt address storage register 22	VAD22	32	32
A001 045Ch	VIC	Interrupt address storage register 23	VAD23	32	32
A001 0460h	VIC	Interrupt address storage register 24	VAD24	32	32
A001 0464h	VIC	Interrupt address storage register 25	VAD25	32	32
A001 0468h	VIC	Interrupt address storage register 26	VAD26	32	32
A001 046Ch	VIC	Interrupt address storage register 27	VAD27	32	32
A001 0470h	VIC	Interrupt address storage register 28	VAD28	32	32
A001 0474h	VIC	Interrupt address storage register 29	VAD29	32	32
A001 0478h	VIC	Interrupt address storage register 30	VAD30	32	32
A001 047Ch	VIC	Interrupt address storage register 31	VAD31	32	32
A001 0480h	VIC	Interrupt address storage register 32	VAD32	32	32
A001 0484h	VIC	Interrupt address storage register 33	VAD33	32	32

Table 5.1 List of I/O Registers (Address Order) (6 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A001 0488h	VIC	Interrupt address storage register 34	VAD34	32	32
A001 048Ch	VIC	Interrupt address storage register 35	VAD35	32	32
A001 0490h	VIC	Interrupt address storage register 36	VAD36	32	32
A001 0494h	VIC	Interrupt address storage register 37	VAD37	32	32
A001 0498h	VIC	Interrupt address storage register 38	VAD38	32	32
A001 049Ch	VIC	Interrupt address storage register 39	VAD39	32	32
A001 04A0h	VIC	Interrupt address storage register 40	VAD40	32	32
A001 04A4h	VIC	Interrupt address storage register 41	VAD41	32	32
A001 04A8h	VIC	Interrupt address storage register 42	VAD42	32	32
A001 04ACh	VIC	Interrupt address storage register 43	VAD43	32	32
A001 04B0h	VIC	Interrupt address storage register 44	VAD44	32	32
A001 04B4h	VIC	Interrupt address storage register 45	VAD45	32	32
A001 04B8h	VIC	Interrupt address storage register 46	VAD46	32	32
A001 04BCh	VIC	Interrupt address storage register 47	VAD47	32	32
A001 04C0h	VIC	Interrupt address storage register 48	VAD48	32	32
A001 04C4h	VIC	Interrupt address storage register 49	VAD49	32	32
A001 04C8h	VIC	Interrupt address storage register 50	VAD50	32	32
A001 04CCh	VIC	Interrupt address storage register 51	VAD51	32	32
A001 04D0h	VIC	Interrupt address storage register 52	VAD52	32	32
A001 04D4h	VIC	Interrupt address storage register 53	VAD53	32	32
A001 04D8h	VIC	Interrupt address storage register 54	VAD54	32	32
A001 04DCh	VIC	Interrupt address storage register 55	VAD55	32	32
A001 04E0h	VIC	Interrupt address storage register 56	VAD56	32	32
A001 04E4h	VIC	Interrupt address storage register 57	VAD57	32	32
A001 04E8h	VIC	Interrupt address storage register 58	VAD58	32	32
A001 04ECh	VIC	Interrupt address storage register 59	VAD59	32	32
A001 04F0h	VIC	Interrupt address storage register 60	VAD60	32	32
A001 04F4h	VIC	Interrupt address storage register 61	VAD61	32	32
A001 04F8h	VIC	Interrupt address storage register 62	VAD62	32	32
A001 04FCh	VIC	Interrupt address storage register 63	VAD63	32	32
A001 0500h	VIC	Interrupt address storage register 64	VAD64	32	32
A001 0504h	VIC	Interrupt address storage register 65	VAD65	32	32
A001 0508h	VIC	Interrupt address storage register 66	VAD66	32	32
A001 050Ch	VIC	Interrupt address storage register 67	VAD67	32	32
A001 0510h	VIC	Interrupt address storage register 68	VAD68	32	32
A001 0514h	VIC	Interrupt address storage register 69	VAD69	32	32
A001 0518h	VIC	Interrupt address storage register 70	VAD70	32	32
A001 051Ch	VIC	Interrupt address storage register 71	VAD71	32	32
A001 0520h	VIC	Interrupt address storage register 72	VAD72	32	32
A001 0524h	VIC	Interrupt address storage register 73	VAD73	32	32
A001 0528h	VIC	Interrupt address storage register 74	VAD74	32	32
A001 052Ch	VIC	Interrupt address storage register 75	VAD75	32	32
A001 0530h	VIC	Interrupt address storage register 76	VAD76	32	32
A001 0534h	VIC	Interrupt address storage register 77	VAD77	32	32
A001 0538h	VIC	Interrupt address storage register 78	VAD78	32	32
A001 053Ch	VIC	Interrupt address storage register 79	VAD79	32	32
A001 0540h	VIC	Interrupt address storage register 80	VAD80	32	32
A001 0544h	VIC	Interrupt address storage register 81	VAD81	32	32

Table 5.1 List of I/O Registers (Address Order) (7 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A001 0548h	VIC	Interrupt address storage register 82	VAD82	32	32
A001 054Ch	VIC	Interrupt address storage register 83	VAD83	32	32
A001 0550h	VIC	Interrupt address storage register 84	VAD84	32	32
A001 0554h	VIC	Interrupt address storage register 85	VAD85	32	32
A001 0558h	VIC	Interrupt address storage register 86	VAD86	32	32
A001 055Ch	VIC	Interrupt address storage register 87	VAD87	32	32
A001 0560h	VIC	Interrupt address storage register 88	VAD88	32	32
A001 0564h	VIC	Interrupt address storage register 89	VAD89	32	32
A001 0568h	VIC	Interrupt address storage register 90	VAD90	32	32
A001 056Ch	VIC	Interrupt address storage register 91	VAD91	32	32
A001 0570h	VIC	Interrupt address storage register 92	VAD92	32	32
A001 0574h	VIC	Interrupt address storage register 93	VAD93	32	32
A001 0578h	VIC	Interrupt address storage register 94	VAD94	32	32
A001 057Ch	VIC	Interrupt address storage register 95	VAD95	32	32
A001 0580h	VIC	Interrupt address storage register 96	VAD96	32	32
A001 0584h	VIC	Interrupt address storage register 97	VAD97	32	32
A001 0588h	VIC	Interrupt address storage register 98	VAD98	32	32
A001 058Ch	VIC	Interrupt address storage register 99	VAD99	32	32
A001 0590h	VIC	Interrupt address storage register 100	VAD100	32	32
A001 0594h	VIC	Interrupt address storage register 101	VAD101	32	32
A001 0598h	VIC	Interrupt address storage register 102	VAD102	32	32
A001 059Ch	VIC	Interrupt address storage register 103	VAD103	32	32
A001 05A0h	VIC	Interrupt address storage register 104	VAD104	32	32
A001 05A4h	VIC	Interrupt address storage register 105	VAD105	32	32
A001 05A8h	VIC	Interrupt address storage register 106	VAD106	32	32
A001 05ACh	VIC	Interrupt address storage register 107	VAD107	32	32
A001 05B0h	VIC	Interrupt address storage register 108	VAD108	32	32
A001 05B4h	VIC	Interrupt address storage register 109	VAD109	32	32
A001 05B8h	VIC	Interrupt address storage register 110	VAD110	32	32
A001 05BCh	VIC	Interrupt address storage register 111	VAD111	32	32
A001 05C0h	VIC	Interrupt address storage register 112	VAD112	32	32
A001 05C4h	VIC	Interrupt address storage register 113	VAD113	32	32
A001 05C8h	VIC	Interrupt address storage register 114	VAD114	32	32
A001 05CCh	VIC	Interrupt address storage register 115	VAD115	32	32
A001 05D0h	VIC	Interrupt address storage register 116	VAD116	32	32
A001 05D4h	VIC	Interrupt address storage register 117	VAD117	32	32
A001 05D8h	VIC	Interrupt address storage register 118	VAD118	32	32
A001 05DCh	VIC	Interrupt address storage register 119	VAD119	32	32
A001 05E0h	VIC	Interrupt address storage register 120	VAD120	32	32
A001 05E4h	VIC	Interrupt address storage register 121	VAD121	32	32
A001 05E8h	VIC	Interrupt address storage register 122	VAD122	32	32
A001 05ECh	VIC	Interrupt address storage register 123	VAD123	32	32
A001 05F0h	VIC	Interrupt address storage register 124	VAD124	32	32
A001 05F4h	VIC	Interrupt address storage register 125	VAD125	32	32
A001 05F8h	VIC	Interrupt address storage register 126	VAD126	32	32
A001 05FCh	VIC	Interrupt address storage register 127	VAD127	32	32
A001 0600h	VIC	Interrupt address storage register 128	VAD128	32	32
A001 0604h	VIC	Interrupt address storage register 129	VAD129	32	32

Table 5.1 List of I/O Registers (Address Order) (8 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A001 0608h	VIC	Interrupt address storage register 130	VAD130	32	32
A001 060Ch	VIC	Interrupt address storage register 131	VAD131	32	32
A001 0610h	VIC	Interrupt address storage register 132	VAD132	32	32
A001 0614h	VIC	Interrupt address storage register 133	VAD133	32	32
A001 0618h	VIC	Interrupt address storage register 134	VAD134	32	32
A001 061Ch	VIC	Interrupt address storage register 135	VAD135	32	32
A001 0620h	VIC	Interrupt address storage register 136	VAD136	32	32
A001 0624h	VIC	Interrupt address storage register 137	VAD137	32	32
A001 0628h	VIC	Interrupt address storage register 138	VAD138	32	32
A001 062Ch	VIC	Interrupt address storage register 139	VAD139	32	32
A001 0630h	VIC	Interrupt address storage register 140	VAD140	32	32
A001 0634h	VIC	Interrupt address storage register 141	VAD141	32	32
A001 0638h	VIC	Interrupt address storage register 142	VAD142	32	32
A001 063Ch	VIC	Interrupt address storage register 143	VAD143	32	32
A001 0640h	VIC	Interrupt address storage register 144	VAD144	32	32
A001 0644h	VIC	Interrupt address storage register 145	VAD145	32	32
A001 0648h	VIC	Interrupt address storage register 146	VAD146	32	32
A001 064Ch	VIC	Interrupt address storage register 147	VAD147	32	32
A001 0650h	VIC	Interrupt address storage register 148	VAD148	32	32
A001 0654h	VIC	Interrupt address storage register 149	VAD149	32	32
A001 0658h	VIC	Interrupt address storage register 150	VAD150	32	32
A001 065Ch	VIC	Interrupt address storage register 151	VAD151	32	32
A001 0660h	VIC	Interrupt address storage register 152	VAD152	32	32
A001 0664h	VIC	Interrupt address storage register 153	VAD153	32	32
A001 0668h	VIC	Interrupt address storage register 154	VAD154	32	32
A001 066Ch	VIC	Interrupt address storage register 155	VAD155	32	32
A001 0670h	VIC	Interrupt address storage register 156	VAD156	32	32
A001 0674h	VIC	Interrupt address storage register 157	VAD157	32	32
A001 0678h	VIC	Interrupt address storage register 158	VAD158	32	32
A001 067Ch	VIC	Interrupt address storage register 159	VAD159	32	32
A001 0680h	VIC	Interrupt address storage register 160	VAD160	32	32
A001 0684h	VIC	Interrupt address storage register 161	VAD161	32	32
A001 0688h	VIC	Interrupt address storage register 162	VAD162	32	32
A001 068Ch	VIC	Interrupt address storage register 163	VAD163	32	32
A001 0690h	VIC	Interrupt address storage register 164	VAD164	32	32
A001 0694h	VIC	Interrupt address storage register 165	VAD165	32	32
A001 0698h	VIC	Interrupt address storage register 166	VAD166	32	32
A001 069Ch	VIC	Interrupt address storage register 167	VAD167	32	32
A001 06A0h	VIC	Interrupt address storage register 168	VAD168	32	32
A001 06A4h	VIC	Interrupt address storage register 169	VAD169	32	32
A001 06A8h	VIC	Interrupt address storage register 170	VAD170	32	32
A001 06ACh	VIC	Interrupt address storage register 171	VAD171	32	32
A001 06B0h	VIC	Interrupt address storage register 172	VAD172	32	32
A001 06B4h	VIC	Interrupt address storage register 173	VAD173	32	32
A001 06B8h	VIC	Interrupt address storage register 174	VAD174	32	32
A001 06BCh	VIC	Interrupt address storage register 175	VAD175	32	32
A001 06C0h	VIC	Interrupt address storage register 176	VAD176	32	32
A001 06C4h	VIC	Interrupt address storage register 177	VAD177	32	32

Table 5.1 List of I/O Registers (Address Order) (9 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A001 06C8h	VIC	Interrupt address storage register 178	VAD178	32	32
A001 06CCh	VIC	Interrupt address storage register 179	VAD179	32	32
A001 06D0h	VIC	Interrupt address storage register 180	VAD180	32	32
A001 06D4h	VIC	Interrupt address storage register 181	VAD181	32	32
A001 06D8h	VIC	Interrupt address storage register 182	VAD182	32	32
A001 06DCh	VIC	Interrupt address storage register 183	VAD183	32	32
A001 06E0h	VIC	Interrupt address storage register 184	VAD184	32	32
A001 06E4h	VIC	Interrupt address storage register 185	VAD185	32	32
A001 06E8h	VIC	Interrupt address storage register 186	VAD186	32	32
A001 06ECh	VIC	Interrupt address storage register 187	VAD187	32	32
A001 06F0h	VIC	Interrupt address storage register 188	VAD188	32	32
A001 06F4h	VIC	Interrupt address storage register 189	VAD189	32	32
A001 06F8h	VIC	Interrupt address storage register 190	VAD190	32	32
A001 06FCh	VIC	Interrupt address storage register 191	VAD191	32	32
A001 0700h	VIC	Interrupt address storage register 192	VAD192	32	32
A001 0704h	VIC	Interrupt address storage register 193	VAD193	32	32
A001 0708h	VIC	Interrupt address storage register 194	VAD194	32	32
A001 070Ch	VIC	Interrupt address storage register 195	VAD195	32	32
A001 0710h	VIC	Interrupt address storage register 196	VAD196	32	32
A001 0714h	VIC	Interrupt address storage register 197	VAD197	32	32
A001 0718h	VIC	Interrupt address storage register 198	VAD198	32	32
A001 071Ch	VIC	Interrupt address storage register 199	VAD199	32	32
A001 0720h	VIC	Interrupt address storage register 200	VAD200	32	32
A001 0724h	VIC	Interrupt address storage register 201	VAD201	32	32
A001 0728h	VIC	Interrupt address storage register 202	VAD202	32	32
A001 072Ch	VIC	Interrupt address storage register 203	VAD203	32	32
A001 0730h	VIC	Interrupt address storage register 204	VAD204	32	32
A001 0734h	VIC	Interrupt address storage register 205	VAD205	32	32
A001 0738h	VIC	Interrupt address storage register 206	VAD206	32	32
A001 073Ch	VIC	Interrupt address storage register 207	VAD207	32	32
A001 0740h	VIC	Interrupt address storage register 208	VAD208	32	32
A001 0744h	VIC	Interrupt address storage register 209	VAD209	32	32
A001 0748h	VIC	Interrupt address storage register 210	VAD210	32	32
A001 074Ch	VIC	Interrupt address storage register 211	VAD211	32	32
A001 0750h	VIC	Interrupt address storage register 212	VAD212	32	32
A001 0754h	VIC	Interrupt address storage register 213	VAD213	32	32
A001 0758h	VIC	Interrupt address storage register 214	VAD214	32	32
A001 075Ch	VIC	Interrupt address storage register 215	VAD215	32	32
A001 0760h	VIC	Interrupt address storage register 216	VAD216	32	32
A001 0764h	VIC	Interrupt address storage register 217	VAD217	32	32
A001 0768h	VIC	Interrupt address storage register 218	VAD218	32	32
A001 076Ch	VIC	Interrupt address storage register 219	VAD219	32	32
A001 0770h	VIC	Interrupt address storage register 220	VAD220	32	32
A001 0774h	VIC	Interrupt address storage register 221	VAD221	32	32
A001 0778h	VIC	Interrupt address storage register 222	VAD222	32	32
A001 077Ch	VIC	Interrupt address storage register 223	VAD223	32	32
A001 0780h	VIC	Interrupt address storage register 224	VAD224	32	32
A001 0784h	VIC	Interrupt address storage register 225	VAD225	32	32

Table 5.1 List of I/O Registers (Address Order) (10 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A001 0788h	VIC	Interrupt address storage register 226	VAD226	32	32
A001 078Ch	VIC	Interrupt address storage register 227	VAD227	32	32
A001 0790h	VIC	Interrupt address storage register 228	VAD228	32	32
A001 0794h	VIC	Interrupt address storage register 229	VAD229	32	32
A001 0798h	VIC	Interrupt address storage register 230	VAD230	32	32
A001 079Ch	VIC	Interrupt address storage register 231	VAD231	32	32
A001 07A0h	VIC	Interrupt address storage register 232	VAD232	32	32
A001 07A4h	VIC	Interrupt address storage register 233	VAD233	32	32
A001 07A8h	VIC	Interrupt address storage register 234	VAD234	32	32
A001 07ACh	VIC	Interrupt address storage register 235	VAD235	32	32
A001 07B0h	VIC	Interrupt address storage register 236	VAD236	32	32
A001 07B4h	VIC	Interrupt address storage register 237	VAD237	32	32
A001 07B8h	VIC	Interrupt address storage register 238	VAD238	32	32
A001 07BCh	VIC	Interrupt address storage register 239	VAD239	32	32
A001 07C0h	VIC	Interrupt address storage register 240	VAD240	32	32
A001 07C4h	VIC	Interrupt address storage register 241	VAD241	32	32
A001 07C8h	VIC	Interrupt address storage register 242	VAD242	32	32
A001 07CCh	VIC	Interrupt address storage register 243	VAD243	32	32
A001 07D0h	VIC	Interrupt address storage register 244	VAD244	32	32
A001 07D4h	VIC	Interrupt address storage register 245	VAD245	32	32
A001 07D8h	VIC	Interrupt address storage register 246	VAD246	32	32
A001 07DCh	VIC	Interrupt address storage register 247	VAD247	32	32
A001 07E0h	VIC	Interrupt address storage register 248	VAD248	32	32
A001 07E4h	VIC	Interrupt address storage register 249	VAD249	32	32
A001 07E8h	VIC	Interrupt address storage register 250	VAD250	32	32
A001 07ECh	VIC	Interrupt address storage register 251	VAD251	32	32
A001 07F0h	VIC	Interrupt address storage register 252	VAD252	32	32
A001 07F4h	VIC	Interrupt address storage register 253	VAD253	32	32
A001 07F8h	VIC	Interrupt address storage register 254	VAD254	32	32
A001 07FCh	VIC	Interrupt address storage register 255	VAD255	32	32
A001 0804h	VIC	Interrupt priority level storage register 1	PRL1	32	32
A001 0808h	VIC	Interrupt priority level storage register 2	PRL2	32	32
A001 080Ch	VIC	Interrupt priority level storage register 3	PRL3	32	32
A001 0810h	VIC	Interrupt priority level storage register 4	PRL4	32	32
A001 0814h	VIC	Interrupt priority level storage register 5	PRL5	32	32
A001 0818h	VIC	Interrupt priority level storage register 6	PRL6	32	32
A001 081Ch	VIC	Interrupt priority level storage register 7	PRL7	32	32
A001 0820h	VIC	Interrupt priority level storage register 8	PRL8	32	32
A001 0824h	VIC	Interrupt priority level storage register 9	PRL9	32	32
A001 0828h	VIC	Interrupt priority level storage register 10	PRL10	32	32
A001 082Ch	VIC	Interrupt priority level storage register 11	PRL11	32	32
A001 0830h	VIC	Interrupt priority level storage register 12	PRL12	32	32
A001 0834h	VIC	Interrupt priority level storage register 13	PRL13	32	32
A001 0838h	VIC	Interrupt priority level storage register 14	PRL14	32	32
A001 083Ch	VIC	Interrupt priority level storage register 15	PRL15	32	32
A001 0840h	VIC	Interrupt priority level storage register 16	PRL16	32	32
A001 0844h	VIC	Interrupt priority level storage register 17	PRL17	32	32
A001 0848h	VIC	Interrupt priority level storage register 18	PRL18	32	32

Table 5.1 List of I/O Registers (Address Order) (11 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A001 084Ch	VIC	Interrupt priority level storage register 19	PRL19	32	32
A001 0850h	VIC	Interrupt priority level storage register 20	PRL20	32	32
A001 0854h	VIC	Interrupt priority level storage register 21	PRL21	32	32
A001 0858h	VIC	Interrupt priority level storage register 22	PRL22	32	32
A001 085Ch	VIC	Interrupt priority level storage register 23	PRL23	32	32
A001 0860h	VIC	Interrupt priority level storage register 24	PRL24	32	32
A001 0864h	VIC	Interrupt priority level storage register 25	PRL25	32	32
A001 0868h	VIC	Interrupt priority level storage register 26	PRL26	32	32
A001 086Ch	VIC	Interrupt priority level storage register 27	PRL27	32	32
A001 0870h	VIC	Interrupt priority level storage register 28	PRL28	32	32
A001 0874h	VIC	Interrupt priority level storage register 29	PRL29	32	32
A001 0878h	VIC	Interrupt priority level storage register 30	PRL30	32	32
A001 087Ch	VIC	Interrupt priority level storage register 31	PRL31	32	32
A001 0880h	VIC	Interrupt priority level storage register 32	PRL32	32	32
A001 0884h	VIC	Interrupt priority level storage register 33	PRL33	32	32
A001 0888h	VIC	Interrupt priority level storage register 34	PRL34	32	32
A001 088Ch	VIC	Interrupt priority level storage register 35	PRL35	32	32
A001 0890h	VIC	Interrupt priority level storage register 36	PRL36	32	32
A001 0894h	VIC	Interrupt priority level storage register 37	PRL37	32	32
A001 0898h	VIC	Interrupt priority level storage register 38	PRL38	32	32
A001 089Ch	VIC	Interrupt priority level storage register 39	PRL39	32	32
A001 08A0h	VIC	Interrupt priority level storage register 40	PRL40	32	32
A001 08A4h	VIC	Interrupt priority level storage register 41	PRL41	32	32
A001 08A8h	VIC	Interrupt priority level storage register 42	PRL42	32	32
A001 08ACh	VIC	Interrupt priority level storage register 43	PRL43	32	32
A001 08B0h	VIC	Interrupt priority level storage register 44	PRL44	32	32
A001 08B4h	VIC	Interrupt priority level storage register 45	PRL45	32	32
A001 08B8h	VIC	Interrupt priority level storage register 46	PRL46	32	32
A001 08BCh	VIC	Interrupt priority level storage register 47	PRL47	32	32
A001 08C0h	VIC	Interrupt priority level storage register 48	PRL48	32	32
A001 08C4h	VIC	Interrupt priority level storage register 49	PRL49	32	32
A001 08C8h	VIC	Interrupt priority level storage register 50	PRL50	32	32
A001 08CCh	VIC	Interrupt priority level storage register 51	PRL51	32	32
A001 08D0h	VIC	Interrupt priority level storage register 52	PRL52	32	32
A001 08D4h	VIC	Interrupt priority level storage register 53	PRL53	32	32
A001 08D8h	VIC	Interrupt priority level storage register 54	PRL54	32	32
A001 08DCh	VIC	Interrupt priority level storage register 55	PRL55	32	32
A001 08E0h	VIC	Interrupt priority level storage register 56	PRL56	32	32
A001 08E4h	VIC	Interrupt priority level storage register 57	PRL57	32	32
A001 08E8h	VIC	Interrupt priority level storage register 58	PRL58	32	32
A001 08ECh	VIC	Interrupt priority level storage register 59	PRL59	32	32
A001 08F0h	VIC	Interrupt priority level storage register 60	PRL60	32	32
A001 08F4h	VIC	Interrupt priority level storage register 61	PRL61	32	32
A001 08F8h	VIC	Interrupt priority level storage register 62	PRL62	32	32
A001 08FCh	VIC	Interrupt priority level storage register 63	PRL63	32	32
A001 0900h	VIC	Interrupt priority level storage register 64	PRL64	32	32
A001 0904h	VIC	Interrupt priority level storage register 65	PRL65	32	32
A001 0908h	VIC	Interrupt priority level storage register 66	PRL66	32	32

Table 5.1 List of I/O Registers (Address Order) (12 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A001 090Ch	VIC	Interrupt priority level storage register 67	PRL67	32	32
A001 0910h	VIC	Interrupt priority level storage register 68	PRL68	32	32
A001 0914h	VIC	Interrupt priority level storage register 69	PRL69	32	32
A001 0918h	VIC	Interrupt priority level storage register 70	PRL70	32	32
A001 091Ch	VIC	Interrupt priority level storage register 71	PRL71	32	32
A001 0920h	VIC	Interrupt priority level storage register 72	PRL72	32	32
A001 0924h	VIC	Interrupt priority level storage register 73	PRL73	32	32
A001 0928h	VIC	Interrupt priority level storage register 74	PRL74	32	32
A001 092Ch	VIC	Interrupt priority level storage register 75	PRL75	32	32
A001 0930h	VIC	Interrupt priority level storage register 76	PRL76	32	32
A001 0934h	VIC	Interrupt priority level storage register 77	PRL77	32	32
A001 0938h	VIC	Interrupt priority level storage register 78	PRL78	32	32
A001 093Ch	VIC	Interrupt priority level storage register 79	PRL79	32	32
A001 0940h	VIC	Interrupt priority level storage register 80	PRL80	32	32
A001 0944h	VIC	Interrupt priority level storage register 81	PRL81	32	32
A001 0948h	VIC	Interrupt priority level storage register 82	PRL82	32	32
A001 094Ch	VIC	Interrupt priority level storage register 83	PRL83	32	32
A001 0950h	VIC	Interrupt priority level storage register 84	PRL84	32	32
A001 0954h	VIC	Interrupt priority level storage register 85	PRL85	32	32
A001 0958h	VIC	Interrupt priority level storage register 86	PRL86	32	32
A001 095Ch	VIC	Interrupt priority level storage register 87	PRL87	32	32
A001 0960h	VIC	Interrupt priority level storage register 88	PRL88	32	32
A001 0964h	VIC	Interrupt priority level storage register 89	PRL89	32	32
A001 0968h	VIC	Interrupt priority level storage register 90	PRL90	32	32
A001 096Ch	VIC	Interrupt priority level storage register 91	PRL91	32	32
A001 0970h	VIC	Interrupt priority level storage register 92	PRL92	32	32
A001 0974h	VIC	Interrupt priority level storage register 93	PRL93	32	32
A001 0978h	VIC	Interrupt priority level storage register 94	PRL94	32	32
A001 097Ch	VIC	Interrupt priority level storage register 95	PRL95	32	32
A001 0980h	VIC	Interrupt priority level storage register 96	PRL96	32	32
A001 0984h	VIC	Interrupt priority level storage register 97	PRL97	32	32
A001 0988h	VIC	Interrupt priority level storage register 98	PRL98	32	32
A001 098Ch	VIC	Interrupt priority level storage register 99	PRL99	32	32
A001 0990h	VIC	Interrupt priority level storage register 100	PRL100	32	32
A001 0994h	VIC	Interrupt priority level storage register 101	PRL101	32	32
A001 0998h	VIC	Interrupt priority level storage register 102	PRL102	32	32
A001 099Ch	VIC	Interrupt priority level storage register 103	PRL103	32	32
A001 09A0h	VIC	Interrupt priority level storage register 104	PRL104	32	32
A001 09A4h	VIC	Interrupt priority level storage register 105	PRL105	32	32
A001 09A8h	VIC	Interrupt priority level storage register 106	PRL106	32	32
A001 09ACh	VIC	Interrupt priority level storage register 107	PRL107	32	32
A001 09B0h	VIC	Interrupt priority level storage register 108	PRL108	32	32
A001 09B4h	VIC	Interrupt priority level storage register 109	PRL109	32	32
A001 09B8h	VIC	Interrupt priority level storage register 110	PRL110	32	32
A001 09BCh	VIC	Interrupt priority level storage register 111	PRL111	32	32
A001 09C0h	VIC	Interrupt priority level storage register 112	PRL112	32	32
A001 09C4h	VIC	Interrupt priority level storage register 113	PRL113	32	32
A001 09C8h	VIC	Interrupt priority level storage register 114	PRL114	32	32

Table 5.1 List of I/O Registers (Address Order) (13 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A001 09CCh	VIC	Interrupt priority level storage register 115	PRL115	32	32
A001 09D0h	VIC	Interrupt priority level storage register 116	PRL116	32	32
A001 09D4h	VIC	Interrupt priority level storage register 117	PRL117	32	32
A001 09D8h	VIC	Interrupt priority level storage register 118	PRL118	32	32
A001 09DCh	VIC	Interrupt priority level storage register 119	PRL119	32	32
A001 09E0h	VIC	Interrupt priority level storage register 120	PRL120	32	32
A001 09E4h	VIC	Interrupt priority level storage register 121	PRL121	32	32
A001 09E8h	VIC	Interrupt priority level storage register 122	PRL122	32	32
A001 09ECh	VIC	Interrupt priority level storage register 123	PRL123	32	32
A001 09F0h	VIC	Interrupt priority level storage register 124	PRL124	32	32
A001 09F4h	VIC	Interrupt priority level storage register 125	PRL125	32	32
A001 09F8h	VIC	Interrupt priority level storage register 126	PRL126	32	32
A001 09FCh	VIC	Interrupt priority level storage register 127	PRL127	32	32
A001 0A00h	VIC	Interrupt priority level storage register 128	PRL128	32	32
A001 0A04h	VIC	Interrupt priority level storage register 129	PRL129	32	32
A001 0A08h	VIC	Interrupt priority level storage register 130	PRL130	32	32
A001 0A0Ch	VIC	Interrupt priority level storage register 131	PRL131	32	32
A001 0A10h	VIC	Interrupt priority level storage register 132	PRL132	32	32
A001 0A14h	VIC	Interrupt priority level storage register 133	PRL133	32	32
A001 0A18h	VIC	Interrupt priority level storage register 134	PRL134	32	32
A001 0A1Ch	VIC	Interrupt priority level storage register 135	PRL135	32	32
A001 0A20h	VIC	Interrupt priority level storage register 136	PRL136	32	32
A001 0A24h	VIC	Interrupt priority level storage register 137	PRL137	32	32
A001 0A28h	VIC	Interrupt priority level storage register 138	PRL138	32	32
A001 0A2Ch	VIC	Interrupt priority level storage register 139	PRL139	32	32
A001 0A30h	VIC	Interrupt priority level storage register 140	PRL140	32	32
A001 0A34h	VIC	Interrupt priority level storage register 141	PRL141	32	32
A001 0A38h	VIC	Interrupt priority level storage register 142	PRL142	32	32
A001 0A3Ch	VIC	Interrupt priority level storage register 143	PRL143	32	32
A001 0A40h	VIC	Interrupt priority level storage register 144	PRL144	32	32
A001 0A44h	VIC	Interrupt priority level storage register 145	PRL145	32	32
A001 0A48h	VIC	Interrupt priority level storage register 146	PRL146	32	32
A001 0A4Ch	VIC	Interrupt priority level storage register 147	PRL147	32	32
A001 0A50h	VIC	Interrupt priority level storage register 148	PRL148	32	32
A001 0A54h	VIC	Interrupt priority level storage register 149	PRL149	32	32
A001 0A58h	VIC	Interrupt priority level storage register 150	PRL150	32	32
A001 0A5Ch	VIC	Interrupt priority level storage register 151	PRL151	32	32
A001 0A60h	VIC	Interrupt priority level storage register 152	PRL152	32	32
A001 0A64h	VIC	Interrupt priority level storage register 153	PRL153	32	32
A001 0A68h	VIC	Interrupt priority level storage register 154	PRL154	32	32
A001 0A6Ch	VIC	Interrupt priority level storage register 155	PRL155	32	32
A001 0A70h	VIC	Interrupt priority level storage register 156	PRL156	32	32
A001 0A74h	VIC	Interrupt priority level storage register 157	PRL157	32	32
A001 0A78h	VIC	Interrupt priority level storage register 158	PRL158	32	32
A001 0A7Ch	VIC	Interrupt priority level storage register 159	PRL159	32	32
A001 0A80h	VIC	Interrupt priority level storage register 160	PRL160	32	32
A001 0A84h	VIC	Interrupt priority level storage register 161	PRL161	32	32
A001 0A88h	VIC	Interrupt priority level storage register 162	PRL162	32	32

Table 5.1 List of I/O Registers (Address Order) (14 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A001 0A8Ch	VIC	Interrupt priority level storage register 163	PRL163	32	32
A001 0A90h	VIC	Interrupt priority level storage register 164	PRL164	32	32
A001 0A94h	VIC	Interrupt priority level storage register 165	PRL165	32	32
A001 0A98h	VIC	Interrupt priority level storage register 166	PRL166	32	32
A001 0A9Ch	VIC	Interrupt priority level storage register 167	PRL167	32	32
A001 0AA0h	VIC	Interrupt priority level storage register 168	PRL168	32	32
A001 0AA4h	VIC	Interrupt priority level storage register 169	PRL169	32	32
A001 0AA8h	VIC	Interrupt priority level storage register 170	PRL170	32	32
A001 0AACh	VIC	Interrupt priority level storage register 171	PRL171	32	32
A001 0AB0h	VIC	Interrupt priority level storage register 172	PRL172	32	32
A001 0AB4h	VIC	Interrupt priority level storage register 173	PRL173	32	32
A001 0AB8h	VIC	Interrupt priority level storage register 174	PRL174	32	32
A001 0ABCh	VIC	Interrupt priority level storage register 175	PRL175	32	32
A001 0AC0h	VIC	Interrupt priority level storage register 176	PRL176	32	32
A001 0AC4h	VIC	Interrupt priority level storage register 177	PRL177	32	32
A001 0AC8h	VIC	Interrupt priority level storage register 178	PRL178	32	32
A001 0ACCh	VIC	Interrupt priority level storage register 179	PRL179	32	32
A001 0AD0h	VIC	Interrupt priority level storage register 180	PRL180	32	32
A001 0AD4h	VIC	Interrupt priority level storage register 181	PRL181	32	32
A001 0AD8h	VIC	Interrupt priority level storage register 182	PRL182	32	32
A001 0ADCh	VIC	Interrupt priority level storage register 183	PRL183	32	32
A001 0AE0h	VIC	Interrupt priority level storage register 184	PRL184	32	32
A001 0AE4h	VIC	Interrupt priority level storage register 185	PRL185	32	32
A001 0AE8h	VIC	Interrupt priority level storage register 186	PRL186	32	32
A001 0AECh	VIC	Interrupt priority level storage register 187	PRL187	32	32
A001 0AF0h	VIC	Interrupt priority level storage register 188	PRL188	32	32
A001 0AF4h	VIC	Interrupt priority level storage register 189	PRL189	32	32
A001 0AF8h	VIC	Interrupt priority level storage register 190	PRL190	32	32
A001 0AFCh	VIC	Interrupt priority level storage register 191	PRL191	32	32
A001 0B00h	VIC	Interrupt priority level storage register 192	PRL192	32	32
A001 0B04h	VIC	Interrupt priority level storage register 193	PRL193	32	32
A001 0B08h	VIC	Interrupt priority level storage register 194	PRL194	32	32
A001 0B0Ch	VIC	Interrupt priority level storage register 195	PRL195	32	32
A001 0B10h	VIC	Interrupt priority level storage register 196	PRL196	32	32
A001 0B14h	VIC	Interrupt priority level storage register 197	PRL197	32	32
A001 0B18h	VIC	Interrupt priority level storage register 198	PRL198	32	32
A001 0B1Ch	VIC	Interrupt priority level storage register 199	PRL199	32	32
A001 0B20h	VIC	Interrupt priority level storage register 200	PRL200	32	32
A001 0B24h	VIC	Interrupt priority level storage register 201	PRL201	32	32
A001 0B28h	VIC	Interrupt priority level storage register 202	PRL202	32	32
A001 0B2Ch	VIC	Interrupt priority level storage register 203	PRL203	32	32
A001 0B30h	VIC	Interrupt priority level storage register 204	PRL204	32	32
A001 0B34h	VIC	Interrupt priority level storage register 205	PRL205	32	32
A001 0B38h	VIC	Interrupt priority level storage register 206	PRL206	32	32
A001 0B3Ch	VIC	Interrupt priority level storage register 207	PRL207	32	32
A001 0B40h	VIC	Interrupt priority level storage register 208	PRL208	32	32
A001 0B44h	VIC	Interrupt priority level storage register 209	PRL209	32	32
A001 0B48h	VIC	Interrupt priority level storage register 210	PRL210	32	32

Table 5.1 List of I/O Registers (Address Order) (15 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A001 0B4Ch	VIC	Interrupt priority level storage register 211	PRL211	32	32
A001 0B50h	VIC	Interrupt priority level storage register 212	PRL212	32	32
A001 0B54h	VIC	Interrupt priority level storage register 213	PRL213	32	32
A001 0B58h	VIC	Interrupt priority level storage register 214	PRL214	32	32
A001 0B5Ch	VIC	Interrupt priority level storage register 215	PRL215	32	32
A001 0B60h	VIC	Interrupt priority level storage register 216	PRL216	32	32
A001 0B64h	VIC	Interrupt priority level storage register 217	PRL217	32	32
A001 0B68h	VIC	Interrupt priority level storage register 218	PRL218	32	32
A001 0B6Ch	VIC	Interrupt priority level storage register 219	PRL219	32	32
A001 0B70h	VIC	Interrupt priority level storage register 220	PRL220	32	32
A001 0B74h	VIC	Interrupt priority level storage register 221	PRL221	32	32
A001 0B78h	VIC	Interrupt priority level storage register 222	PRL222	32	32
A001 0B7Ch	VIC	Interrupt priority level storage register 223	PRL223	32	32
A001 0B80h	VIC	Interrupt priority level storage register 224	PRL224	32	32
A001 0B84h	VIC	Interrupt priority level storage register 225	PRL225	32	32
A001 0B88h	VIC	Interrupt priority level storage register 226	PRL226	32	32
A001 0B8Ch	VIC	Interrupt priority level storage register 227	PRL227	32	32
A001 0B90h	VIC	Interrupt priority level storage register 228	PRL228	32	32
A001 0B94h	VIC	Interrupt priority level storage register 229	PRL229	32	32
A001 0B98h	VIC	Interrupt priority level storage register 230	PRL230	32	32
A001 0B9Ch	VIC	Interrupt priority level storage register 231	PRL231	32	32
A001 0BA0h	VIC	Interrupt priority level storage register 232	PRL232	32	32
A001 0BA4h	VIC	Interrupt priority level storage register 233	PRL233	32	32
A001 0BA8h	VIC	Interrupt priority level storage register 234	PRL234	32	32
A001 0BACh	VIC	Interrupt priority level storage register 235	PRL235	32	32
A001 0BB0h	VIC	Interrupt priority level storage register 236	PRL236	32	32
A001 0BB4h	VIC	Interrupt priority level storage register 237	PRL237	32	32
A001 0BB8h	VIC	Interrupt priority level storage register 238	PRL238	32	32
A001 0BBCh	VIC	Interrupt priority level storage register 239	PRL239	32	32
A001 0BC0h	VIC	Interrupt priority level storage register 240	PRL240	32	32
A001 0BC4h	VIC	Interrupt priority level storage register 241	PRL241	32	32
A001 0BC8h	VIC	Interrupt priority level storage register 242	PRL242	32	32
A001 0BCCh	VIC	Interrupt priority level storage register 243	PRL243	32	32
A001 0BD0h	VIC	Interrupt priority level storage register 244	PRL244	32	32
A001 0BD4h	VIC	Interrupt priority level storage register 245	PRL245	32	32
A001 0BD8h	VIC	Interrupt priority level storage register 246	PRL246	32	32
A001 0BDCh	VIC	Interrupt priority level storage register 247	PRL247	32	32
A001 0BE0h	VIC	Interrupt priority level storage register 248	PRL248	32	32
A001 0BE4h	VIC	Interrupt priority level storage register 249	PRL249	32	32
A001 0BE8h	VIC	Interrupt priority level storage register 250	PRL250	32	32
A001 0BECh	VIC	Interrupt priority level storage register 251	PRL251	32	32
A001 0BF0h	VIC	Interrupt priority level storage register 252	PRL252	32	32
A001 0BF4h	VIC	Interrupt priority level storage register 253	PRL253	32	32
A001 0BF8h	VIC	Interrupt priority level storage register 254	PRL254	32	32
A001 0BFCh	VIC	Interrupt priority level storage register 255	PRL255	32	32
A001 1000h	VIC	IRQ status register 8	IRQS8	32	32
A001 1004h	VIC	IRQ status register 9	IRQS9	32	32
A001 1040h	VIC	Interrupt input status register 8	RAIS8	32	32

Table 5.1 List of I/O Registers (Address Order) (16 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A001 1044h	VIC	Interrupt input status register 9	RAIS9	32	32
A001 1080h	VIC	Interrupt enable register 8	IEN8	32	32
A001 1084h	VIC	Interrupt enable register 9	IEN9	32	32
A001 10A0h	VIC	Interrupt enable clear register 8	IEC8	32	32
A001 10A4h	VIC	Interrupt enable clear register 9	IEC9	32	32
A001 1100h	VIC	Interrupt detection type selection register 8	PLS8	32	32
A001 1104h	VIC	Interrupt detection type selection register 9	PLS9	32	32
A001 1120h	VIC	Edge detection bit clear register 8	PIC8	32	32
A001 1124h	VIC	Edge detection bit clear register 9	PIC9	32	32
A001 11C0h	VIC	Interrupt priority level mask register 1	PRLM1	32	32
A001 11C4h	VIC	Interrupt priority level mask clear register 1	PRLC1	32	32
A001 11C8h	VIC	User mode enable register 1	UEN1	32	32
A001 1210h	VIC	Interrupt service status register 8	ISS8	32	32
A001 1214h	VIC	Interrupt service status register 9	ISS9	32	32
A001 1230h	VIC	Interrupt service current register 8	ISC8	32	32
A001 1234h	VIC	Interrupt service current register 9	ISC9	32	32
A001 1400h	VIC	Interrupt address storage register 256	VAD256	32	32
A001 1404h	VIC	Interrupt address storage register 257	VAD257	32	32
A001 1408h	VIC	Interrupt address storage register 258	VAD258	32	32
A001 140Ch	VIC	Interrupt address storage register 259	VAD259	32	32
A001 1410h	VIC	Interrupt address storage register 260	VAD260	32	32
A001 1414h	VIC	Interrupt address storage register 261	VAD261	32	32
A001 1418h	VIC	Interrupt address storage register 262	VAD262	32	32
A001 141Ch	VIC	Interrupt address storage register 263	VAD263	32	32
A001 1420h	VIC	Interrupt address storage register 264	VAD264	32	32
A001 1424h	VIC	Interrupt address storage register 265	VAD265	32	32
A001 1428h	VIC	Interrupt address storage register 266	VAD266	32	32
A001 142Ch	VIC	Interrupt address storage register 267	VAD267	32	32
A001 1430h	VIC	Interrupt address storage register 268	VAD268	32	32
A001 1434h	VIC	Interrupt address storage register 269	VAD269	32	32
A001 1438h	VIC	Interrupt address storage register 270	VAD270	32	32
A001 143Ch	VIC	Interrupt address storage register 271	VAD271	32	32
A001 1440h	VIC	Interrupt address storage register 272	VAD272	32	32
A001 1444h	VIC	Interrupt address storage register 273	VAD273	32	32
A001 1448h	VIC	Interrupt address storage register 274	VAD274	32	32
A001 144Ch	VIC	Interrupt address storage register 275	VAD275	32	32
A001 1450h	VIC	Interrupt address storage register 276	VAD276	32	32
A001 1454h	VIC	Interrupt address storage register 277	VAD277	32	32
A001 1458h	VIC	Interrupt address storage register 278	VAD278	32	32
A001 145Ch	VIC	Interrupt address storage register 279	VAD279	32	32
A001 1460h	VIC	Interrupt address storage register 280	VAD280	32	32
A001 1464h	VIC	Interrupt address storage register 281	VAD281	32	32
A001 1468h	VIC	Interrupt address storage register 282	VAD282	32	32
A001 146Ch	VIC	Interrupt address storage register 283	VAD283	32	32
A001 1470h	VIC	Interrupt address storage register 284	VAD284	32	32
A001 1474h	VIC	Interrupt address storage register 285	VAD285	32	32
A001 1478h	VIC	Interrupt address storage register 286	VAD286	32	32
A001 147Ch	VIC	Interrupt address storage register 287	VAD287	32	32

Table 5.1 List of I/O Registers (Address Order) (17 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A001 1480h	VIC	Interrupt address storage register 288	VAD288	32	32
A001 1484h	VIC	Interrupt address storage register 289	VAD289	32	32
A001 1488h	VIC	Interrupt address storage register 290	VAD290	32	32
A001 148Ch	VIC	Interrupt address storage register 291	VAD291	32	32
A001 1490h	VIC	Interrupt address storage register 292	VAD292	32	32
A001 1494h	VIC	Interrupt address storage register 293	VAD293	32	32
A001 1498h	VIC	Interrupt address storage register 294	VAD294	32	32
A001 1800h	VIC	Interrupt priority level storage register 256	PRL256	32	32
A001 1804h	VIC	Interrupt priority level storage register 257	PRL257	32	32
A001 1808h	VIC	Interrupt priority level storage register 258	PRL258	32	32
A001 180Ch	VIC	Interrupt priority level storage register 259	PRL259	32	32
A001 1810h	VIC	Interrupt priority level storage register 260	PRL260	32	32
A001 1814h	VIC	Interrupt priority level storage register 261	PRL261	32	32
A001 1818h	VIC	Interrupt priority level storage register 262	PRL262	32	32
A001 181Ch	VIC	Interrupt priority level storage register 263	PRL263	32	32
A001 1820h	VIC	Interrupt priority level storage register 264	PRL264	32	32
A001 1824h	VIC	Interrupt priority level storage register 265	PRL265	32	32
A001 1828h	VIC	Interrupt priority level storage register 266	PRL266	32	32
A001 182Ch	VIC	Interrupt priority level storage register 267	PRL267	32	32
A001 1830h	VIC	Interrupt priority level storage register 268	PRL268	32	32
A001 1834h	VIC	Interrupt priority level storage register 269	PRL269	32	32
A001 1838h	VIC	Interrupt priority level storage register 270	PRL270	32	32
A001 183Ch	VIC	Interrupt priority level storage register 271	PRL271	32	32
A001 1840h	VIC	Interrupt priority level storage register 272	PRL272	32	32
A001 1844h	VIC	Interrupt priority level storage register 273	PRL273	32	32
A001 1848h	VIC	Interrupt priority level storage register 274	PRL274	32	32
A001 184Ch	VIC	Interrupt priority level storage register 275	PRL275	32	32
A001 1850h	VIC	Interrupt priority level storage register 276	PRL276	32	32
A001 1854h	VIC	Interrupt priority level storage register 277	PRL277	32	32
A001 1858h	VIC	Interrupt priority level storage register 278	PRL278	32	32
A001 185Ch	VIC	Interrupt priority level storage register 279	PRL279	32	32
A001 1860h	VIC	Interrupt priority level storage register 280	PRL280	32	32
A001 1864h	VIC	Interrupt priority level storage register 281	PRL281	32	32
A001 1868h	VIC	Interrupt priority level storage register 282	PRL282	32	32
A001 186Ch	VIC	Interrupt priority level storage register 283	PRL283	32	32
A001 1870h	VIC	Interrupt priority level storage register 284	PRL284	32	32
A001 1874h	VIC	Interrupt priority level storage register 285	PRL285	32	32
A001 1878h	VIC	Interrupt priority level storage register 286	PRL286	32	32
A001 187Ch	VIC	Interrupt priority level storage register 287	PRL287	32	32
A001 1880h	VIC	Interrupt priority level storage register 288	PRL288	32	32
A001 1884h	VIC	Interrupt priority level storage register 289	PRL289	32	32
A001 1888h	VIC	Interrupt priority level storage register 290	PRL290	32	32
A001 188Ch	VIC	Interrupt priority level storage register 291	PRL291	32	32
A001 1890h	VIC	Interrupt priority level storage register 292	PRL292	32	32
A001 1894h	VIC	Interrupt priority level storage register 293	PRL293	32	32
A001 1898h	VIC	Interrupt priority level storage register 294	PRL294	32	32
A006 2000h	DMA0	Next 0 source address register 0	DMAC0_N0SA_0_N	32	32
A006 2000h	DMA0	Next 0 source address register 0	DMAC0_N0SA_0_W	32	32

Table 5.1 List of I/O Registers (Address Order) (18 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 2004h	DMA0	Next 0 destination address register 0	DMAC0_N0DA_0	32	32
A006 2008h	DMA0	Next 0 transaction byte register 0	DMAC0_N0TB_0	32	32
A006 200Ch	DMA0	Next 1 source address register 0	DMAC0_N1SA_0_N	32	32
A006 200Ch	DMA0	Next 1 source address register 0	DMAC0_N1SA_0_W	32	32
A006 2010h	DMA0	Next 1 destination address register 0	DMAC0_N1DA_0	32	32
A006 2014h	DMA0	Next 1 transaction byte register 0	DMAC0_N1TB_0	32	32
A006 2018h	DMA0	Current source address register 0	DMAC0_CRSA_0	32	32
A006 201Ch	DMA0	Current destination address register 0	DMAC0_CRDA_0	32	32
A006 2020h	DMA0	Current transaction byte register 0	DMAC0_CRTB_0	32	32
A006 2024h	DMA0	Channel status register 0	DMAC0_CHSTAT_0	32	32
A006 2028h	DMA0	Channel control register 0	DMAC0_CHCTRL_0	32	32
A006 202Ch	DMA0	Channel configuration register 0	DMAC0_CHCFG_0	32	32
A006 2030h	DMA0	Channel interval register 0	DMAC0_CHITVL_0	32	32
A006 2038h	DMA0	Next link address register 0	DMAC0_NXLA_0	32	32
A006 203Ch	DMA0	Current link address register 0	DMAC0_CRLA_0	32	32
A006 2040h	DMA0	Next 0 source address register 1	DMAC0_N0SA_1_N	32	32
A006 2040h	DMA0	Next 0 source address register 1	DMAC0_N0SA_1_W	32	32
A006 2044h	DMA0	Next 0 destination address register 1	DMAC0_N0DA_1	32	32
A006 2048h	DMA0	Next 0 transaction byte register 1	DMAC0_N0TB_1	32	32
A006 204Ch	DMA0	Next 1 source address register 1	DMAC0_N1SA_1_N	32	32
A006 204Ch	DMA0	Next 1 source address register 1	DMAC0_N1SA_1_W	32	32
A006 2050h	DMA0	Next 1 destination address register 1	DMAC0_N1DA_1	32	32
A006 2054h	DMA0	Next 1 transaction byte register 1	DMAC0_N1TB_1	32	32
A006 2058h	DMA0	Current source address register 1	DMAC0_CRSA_1	32	32
A006 205Ch	DMA0	Current destination address register 1	DMAC0_CRDA_1	32	32
A006 2060h	DMA0	Current transaction byte register 1	DMAC0_CRTB_1	32	32
A006 2064h	DMA0	Channel status register 1	DMAC0_CHSTAT_1	32	32
A006 2068h	DMA0	Channel control register 1	DMAC0_CHCTRL_1	32	32
A006 206Ch	DMA0	Channel configuration register 1	DMAC0_CHCFG_1	32	32
A006 2070h	DMA0	Channel interval register 1	DMAC0_CHITVL_1	32	32
A006 2078h	DMA0	Current link address register 1	DMAC0_NXLA_1	32	32
A006 207Ch	DMA0	Next link address register 1	DMAC0_CRLA_1	32	32
A006 2080h	DMA0	Next 0 source address register 2	DMAC0_N0SA_2_N	32	32
A006 2080h	DMA0	Next 0 source address register 2	DMAC0_N0SA_2_W	32	32
A006 2084h	DMA0	Next 0 destination address register 2	DMAC0_N0DA_2	32	32
A006 2088h	DMA0	Next 0 transaction byte register 2	DMAC0_N0TB_2	32	32
A006 208Ch	DMA0	Next 1 source address register 2	DMAC0_N1SA_2_N	32	32
A006 208Ch	DMA0	Next 1 source address register 2	DMAC0_N1SA_2_W	32	32
A006 2090h	DMA0	Next 1 destination address register 2	DMAC0_N1DA_2	32	32
A006 2094h	DMA0	Next 1 transaction byte register 2	DMAC0_N1TB_2	32	32
A006 2098h	DMA0	Current source address register 2	DMAC0_CRSA_2	32	32
A006 209Ch	DMA0	Current destination address register 2	DMAC0_CRDA_2	32	32
A006 20A0h	DMA0	Current transaction byte register 2	DMAC0_CRTB_2	32	32
A006 20A4h	DMA0	Channel status register 2	DMAC0_CHSTAT_2	32	32
A006 20A8h	DMA0	Channel control register 2	DMAC0_CHCTRL_2	32	32
A006 20ACh	DMA0	Channel configuration register 2	DMAC0_CHCFG_2	32	32
A006 20B0h	DMA0	Channel interval register 2	DMAC0_CHITVL_2	32	32
A006 20B8h	DMA0	Next link address register 2	DMAC0_NXLA_2	32	32

Table 5.1 List of I/O Registers (Address Order) (19 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 20BCh	DMA0	Current link address register 2	DMAC0_CRLA_2	32	32
A006 20C0h	DMA0	Next 0 source address register 3	DMAC0_N0SA_3_N	32	32
A006 20C0h	DMA0	Next 0 source address register 3	DMAC0_N0SA_3_W	32	32
A006 20C4h	DMA0	Next 0 destination address register 3	DMAC0_N0DA_3	32	32
A006 20C8h	DMA0	Next 0 transaction byte register 3	DMAC0_N0TB_3	32	32
A006 20CCh	DMA0	Next 1 source address register 3	DMAC0_N1SA_3_N	32	32
A006 20CCh	DMA0	Next 1 source address register 3	DMAC0_N1SA_3_W	32	32
A006 20D0h	DMA0	Next 1 destination address register 3	DMAC0_N1DA_3	32	32
A006 20D4h	DMA0	Next 1 transaction byte register 3	DMAC0_N1TB_3	32	32
A006 20D8h	DMA0	Current source address register 3	DMAC0_CRSA_3	32	32
A006 20DCh	DMA0	Current destination address register 3	DMAC0_CRDA_3	32	32
A006 20E0h	DMA0	Current transaction byte register 3	DMAC0_CRTB_3	32	32
A006 20E4h	DMA0	Channel status register 3	DMAC0_CHSTAT_3	32	32
A006 20E8h	DMA0	Channel control register 3	DMAC0_CHCTRL_3	32	32
A006 20ECh	DMA0	Channel configuration register 3	DMAC0_CHCFG_3	32	32
A006 20F0h	DMA0	Channel interval register 3	DMAC0_CHITVL_3	32	32
A006 20F8h	DMA0	Next link address register 3	DMAC0_NXLA_3	32	32
A006 20FCh	DMA0	Current link address register 3	DMAC0_CRLA_3	32	32
A006 2100h	DMA0	Next 0 source address register 4	DMAC0_N0SA_4_N	32	32
A006 2100h	DMA0	Next 0 source address register 4	DMAC0_N0SA_4_W	32	32
A006 2104h	DMA0	Next 0 destination address register 4	DMAC0_N0DA_4	32	32
A006 2108h	DMA0	Next 0 transaction byte register 4	DMAC0_N0TB_4	32	32
A006 210Ch	DMA0	Next 1 source address register 4	DMAC0_N1SA_4_N	32	32
A006 210Ch	DMA0	Next 1 source address register 4	DMAC0_N1SA_4_W	32	32
A006 2110h	DMA0	Next 1 destination address register 4	DMAC0_N1DA_4	32	32
A006 2114h	DMA0	Next 1 transaction byte register 4	DMAC0_N1TB_4	32	32
A006 2118h	DMA0	Current source address register 4	DMAC0_CRSA_4	32	32
A006 211Ch	DMA0	Current destination address register 4	DMAC0_CRDA_4	32	32
A006 2120h	DMA0	Current transaction byte register 4	DMAC0_CRTB_4	32	32
A006 2124h	DMA0	Channel status register 4	DMAC0_CHSTAT_4	32	32
A006 2128h	DMA0	Channel control register 4	DMAC0_CHCTRL_4	32	32
A006 212Ch	DMA0	Channel configuration register 4	DMAC0_CHCFG_4	32	32
A006 2130h	DMA0	Channel interval register 4	DMAC0_CHITVL_4	32	32
A006 2138h	DMA0	Next link address register 4	DMAC0_NXLA_4	32	32
A006 213Ch	DMA0	Current link address register 4	DMAC0_CRLA_4	32	32
A006 2140h	DMA0	Next 0 source address register 5	DMAC0_N0SA_5_N	32	32
A006 2140h	DMA0	Next 0 source address register 5	DMAC0_N0SA_5_W	32	32
A006 2144h	DMA0	Next 0 destination address register 5	DMAC0_N0DA_5	32	32
A006 2148h	DMA0	Next 0 transaction byte register 5	DMAC0_N0TB_5	32	32
A006 214Ch	DMA0	Next 1 source address register 5	DMAC0_N1SA_5_N	32	32
A006 214Ch	DMA0	Next 1 source address register 5	DMAC0_N1SA_5_W	32	32
A006 2150h	DMA0	Next 1 destination address register 5	DMAC0_N1DA_5	32	32
A006 2154h	DMA0	Next 1 transaction byte register 5	DMAC0_N1TB_5	32	32
A006 2158h	DMA0	Current source address register 5	DMAC0_CRSA_5	32	32
A006 215Ch	DMA0	Current destination address register 5	DMAC0_CRDA_5	32	32
A006 2160h	DMA0	Current transaction byte register 5	DMAC0_CRTB_5	32	32
A006 2164h	DMA0	Channel status register 5	DMAC0_CHSTAT_5	32	32
A006 2168h	DMA0	Channel control register 5	DMAC0_CHCTRL_5	32	32

Table 5.1 List of I/O Registers (Address Order) (20 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 216Ch	DMA0	Channel configuration register 5	DMAC0_CHCFG_5	32	32
A006 2170h	DMA0	Channel interval register 5	DMAC0_CHITVL_5	32	32
A006 2178h	DMA0	Next link address register 5	DMAC0_NXLA_5	32	32
A006 217Ch	DMA0	Current link address register 5	DMAC0_CRLA_5	32	32
A006 2180h	DMA0	Next 0 source address register 6	DMAC0_N0SA_6_N	32	32
A006 2180h	DMA0	Next 0 source address register 6	DMAC0_N0SA_6_W	32	32
A006 2184h	DMA0	Next 0 destination address register 6	DMAC0_N0DA_6	32	32
A006 2188h	DMA0	Next 0 transaction byte register 6	DMAC0_N0TB_6	32	32
A006 218Ch	DMA0	Next 1 source address register 6	DMAC0_N1SA_6_N	32	32
A006 218Ch	DMA0	Next 1 source address register 6	DMAC0_N1SA_6_W	32	32
A006 2190h	DMA0	Next 1 destination address register 6	DMAC0_N1DA_6	32	32
A006 2194h	DMA0	Next 1 transaction byte register 6	DMAC0_N1TB_6	32	32
A006 2198h	DMA0	Current source address register 6	DMAC0_CRSA_6	32	32
A006 219Ch	DMA0	Current destination address register 6	DMAC0_CRDA_6	32	32
A006 21A0h	DMA0	Current transaction byte register 6	DMAC0_CRTB_6	32	32
A006 21A4h	DMA0	Channel status register 6	DMAC0_CHSTAT_6	32	32
A006 21A8h	DMA0	Channel control register 6	DMAC0_CHCTRL_6	32	32
A006 21ACh	DMA0	Channel configuration register 6	DMAC0_CHCFG_6	32	32
A006 21B0h	DMA0	Channel interval register 6	DMAC0_CHITVL_6	32	32
A006 21B8h	DMA0	Next link address register 6	DMAC0_NXLA_6	32	32
A006 21BCh	DMA0	Current link address register 6	DMAC0_CRLA_6	32	32
A006 21C0h	DMA0	Next 0 source address register 7	DMAC0_N0SA_7_N	32	32
A006 21C0h	DMA0	Next 0 source address register 7	DMAC0_N0SA_7_W	32	32
A006 21C4h	DMA0	Next 0 destination address register 7	DMAC0_N0DA_7	32	32
A006 21C8h	DMA0	Next 0 transaction byte register 7	DMAC0_N0TB_7	32	32
A006 21CCh	DMA0	Next 1 source address register 7	DMAC0_N1SA_7_N	32	32
A006 21CCh	DMA0	Next 1 source address register 7	DMAC0_N1SA_7_W	32	32
A006 21D0h	DMA0	Next 1 destination address register 7	DMAC0_N1DA_7	32	32
A006 21D4h	DMA0	Next 1 transaction byte register 7	DMAC0_N1TB_7	32	32
A006 21D8h	DMA0	Current source address register 7	DMAC0_CRSA_7	32	32
A006 21DCh	DMA0	Current destination address register 7	DMAC0_CRDA_7	32	32
A006 21E0h	DMA0	Current transaction byte register 7	DMAC0_CRTB_7	32	32
A006 21E4h	DMA0	Channel status register 7	DMAC0_CHSTAT_7	32	32
A006 21E8h	DMA0	Channel control register 7	DMAC0_CHCTRL_7	32	32
A006 21ECh	DMA0	Channel configuration register 7	DMAC0_CHCFG_7	32	32
A006 21F0h	DMA0	Channel interval register 7	DMAC0_CHITVL_7	32	32
A006 21F8h	DMA0	Next link address register 7	DMAC0_NXLA_7	32	32
A006 21FCh	DMA0	Current link address register 7	DMAC0_CRLA_7	32	32
A006 2200h	DMA0	Source continuous register 0	DMAC0_SCNT_0	32	32
A006 2204h	DMA0	Source skip register 0	DMAC0_SSKP_0	32	32
A006 2208h	DMA0	Destination continuous register 0	DMAC0_DCNT_0	32	32
A006 220Ch	DMA0	Destination skip register 0	DMAC0_DSKP_0	32	32
A006 2220h	DMA0	Source continuous register 1	DMAC0_SCNT_1	32	32
A006 2224h	DMA0	Source skip register 1	DMAC0_SSKP_1	32	32
A006 2228h	DMA0	Destination continuous register 1	DMAC0_DCNT_1	32	32
A006 222Ch	DMA0	Destination skip register 1	DMAC0_DSKP_1	32	32
A006 2240h	DMA0	Source continuous register 2	DMAC0_SCNT_2	32	32
A006 2244h	DMA0	Source skip register 2	DMAC0_SSKP_2	32	32

Table 5.1 List of I/O Registers (Address Order) (21 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 2248h	DMA0	Destination continuous register 2	DMAC0_DCNT_2	32	32
A006 224Ch	DMA0	Destination skip register 2	DMAC0_DSKP_2	32	32
A006 2260h	DMA0	Source continuous register 3	DMAC0_SCNT_3	32	32
A006 2264h	DMA0	Source skip register 3	DMAC0_SSKP_3	32	32
A006 2268h	DMA0	Destination continuous register 3	DMAC0_DCNT_3	32	32
A006 226Ch	DMA0	Destination skip register 3	DMAC0_DSKP_3	32	32
A006 2280h	DMA0	Source continuous register 4	DMAC0_SCNT_4	32	32
A006 2284h	DMA0	Source skip register 4	DMAC0_SSKP_4	32	32
A006 2288h	DMA0	Destination continuous register 4	DMAC0_DCNT_4	32	32
A006 228Ch	DMA0	Destination skip register 4	DMAC0_DSKP_4	32	32
A006 22A0h	DMA0	Source continuous register 5	DMAC0_SCNT_5	32	32
A006 22A4h	DMA0	Source skip register 5	DMAC0_SSKP_5	32	32
A006 22A8h	DMA0	Destination continuous register 5	DMAC0_DCNT_5	32	32
A006 22ACh	DMA0	Destination skip register 5	DMAC0_DSKP_5	32	32
A006 22C0h	DMA0	Source continuous register 6	DMAC0_SCNT_6	32	32
A006 22C4h	DMA0	Source skip register 6	DMAC0_SSKP_6	32	32
A006 22C8h	DMA0	Destination continuous register 6	DMAC0_DCNT_6	32	32
A006 22CCh	DMA0	Destination skip register 6	DMAC0_DSKP_6	32	32
A006 22E0h	DMA0	Source continuous register 7	DMAC0_SCNT_7	32	32
A006 22E4h	DMA0	Source skip register 7	DMAC0_SSKP_7	32	32
A006 22E8h	DMA0	Destination continuous register 7	DMAC0_DCNT_7	32	32
A006 22ECh	DMA0	Destination skip register 7	DMAC0_DSKP_7	32	32
A006 2300h	DMA0	DMA control register A	DMAC0_DCTRL_A	32	32
A006 2304h	DMA0	Descriptor interval register A	DMAC0_DSCITVL_A	32	32
A006 2310h	DMA0	DMA status EN register A	DMAC0_DST_EN_A	32	32
A006 2314h	DMA0	DMA status ER register A	DMAC0_DST_ER_A	32	32
A006 2318h	DMA0	DMA status END register A	DMAC0_DST_END_A	32	32
A006 2320h	DMA0	DMA status SUS register A	DMAC0_DST_SUS_A	32	32
A006 2400h	DMA0	Next 0 source address register 8	DMAC0_N0SA_8_N	32	32
A006 2400h	DMA0	Next 0 source address register 8	DMAC0_N0SA_8_W	32	32
A006 2404h	DMA0	Next 0 destination address register 8	DMAC0_N0DA_8	32	32
A006 2408h	DMA0	Next 0 transaction byte register 8	DMAC0_N0TB_8	32	32
A006 240Ch	DMA0	Next 1 source address register 8	DMAC0_N1SA_8_N	32	32
A006 240Ch	DMA0	Next 1 source address register 8	DMAC0_N1SA_8_W	32	32
A006 2410h	DMA0	Next 1 destination address register 8	DMAC0_N1DA_8	32	32
A006 2414h	DMA0	Next 1 transaction byte register 8	DMAC0_N1TB_8	32	32
A006 2418h	DMA0	Current source address register 8	DMAC0_CRSA_8	32	32
A006 241Ch	DMA0	Current destination address register 8	DMAC0_CRDA_8	32	32
A006 2420h	DMA0	Current transaction byte register 8	DMAC0_CRTB_8	32	32
A006 2424h	DMA0	Channel status register 8	DMAC0_CHSTAT_8	32	32
A006 2428h	DMA0	Channel control register 8	DMAC0_CHCTRL_8	32	32
A006 242Ch	DMA0	Channel configuration register 8	DMAC0_CHCFG_8	32	32
A006 2430h	DMA0	Channel interval register 8	DMAC0_CHITVL_8	32	32
A006 2438h	DMA0	Next link address register 8	DMAC0_NXLA_8	32	32
A006 243Ch	DMA0	Current link address register 8	DMAC0_CRLA_8	32	32
A006 2440h	DMA0	Next 0 source address register 9	DMAC0_N0SA_9_N	32	32
A006 2440h	DMA0	Next 0 source address register 9	DMAC0_N0SA_9_W	32	32
A006 2444h	DMA0	Next 0 destination address register 9	DMAC0_N0DA_9	32	32

Table 5.1 List of I/O Registers (Address Order) (22 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 2448h	DMA0	Next 0 transaction byte register 9	DMAC0_N0TB_9	32	32
A006 244Ch	DMA0	Next 1 source address register 9	DMAC0_N1SA_9_N	32	32
A006 244Ch	DMA0	Next 1 source address register 9	DMAC0_N1SA_9_W	32	32
A006 2450h	DMA0	Next 1 destination address register 9	DMAC0_N1DA_9	32	32
A006 2454h	DMA0	Next 1 transaction byte register 9	DMAC0_N1TB_9	32	32
A006 2458h	DMA0	Current source address register 9	DMAC0_CRSA_9	32	32
A006 245Ch	DMA0	Current destination address register 9	DMAC0_CRDA_9	32	32
A006 2460h	DMA0	Current transaction byte register 9	DMAC0_CRTB_9	32	32
A006 2464h	DMA0	Channel status register 9	DMAC0_CHSTAT_9	32	32
A006 2468h	DMA0	Channel control register 9	DMAC0_CHCTRL_9	32	32
A006 246Ch	DMA0	Channel configuration register 9	DMAC0_CHCFG_9	32	32
A006 2470h	DMA0	Channel interval register 9	DMAC0_CHITVL_9	32	32
A006 2478h	DMA0	Next link address register 9	DMAC0_NXLA_9	32	32
A006 247Ch	DMA0	Current link address register 9	DMAC0_CRLA_9	32	32
A006 2480h	DMA0	Next 0 source address register 10	DMAC0_N0SA_10_N	32	32
A006 2480h	DMA0	Next 0 source address register 10	DMAC0_N0SA_10_W	32	32
A006 2484h	DMA0	Next 0 destination address register 10	DMAC0_N0DA_10	32	32
A006 2488h	DMA0	Next 0 transaction byte register 10	DMAC0_N0TB_10	32	32
A006 248Ch	DMA0	Next 1 source address register 10	DMAC0_N1SA_10_N	32	32
A006 248Ch	DMA0	Next 1 source address register 10	DMAC0_N1SA_10_W	32	32
A006 2490h	DMA0	Next 1 destination address register 10	DMAC0_N1DA_10	32	32
A006 2494h	DMA0	Next 1 transaction byte register 10	DMAC0_N1TB_10	32	32
A006 2498h	DMA0	Current source address register 10	DMAC0_CRSA_10	32	32
A006 249Ch	DMA0	Current destination address register 10	DMAC0_CRDA_10	32	32
A006 24A0h	DMA0	Current transaction byte register 10	DMAC0_CRTB_10	32	32
A006 24A4h	DMA0	Channel status register 10	DMAC0_CHSTAT_10	32	32
A006 24A8h	DMA0	Channel control register 10	DMAC0_CHCTRL_10	32	32
A006 24ACh	DMA0	Channel configuration register 10	DMAC0_CHCFG_10	32	32
A006 24B0h	DMA0	Channel interval register 10	DMAC0_CHITVL_10	32	32
A006 24B8h	DMA0	Next link address register 10	DMAC0_NXLA_10	32	32
A006 24BCh	DMA0	Current link address register 10	DMAC0_CRLA_10	32	32
A006 24C0h	DMA0	Next 0 source address register 11	DMAC0_N0SA_11_N	32	32
A006 24C0h	DMA0	Next 0 source address register 11	DMAC0_N0SA_11_W	32	32
A006 24C4h	DMA0	Next 0 destination address register 11	DMAC0_N0DA_11	32	32
A006 24C8h	DMA0	Next 0 transaction byte register 11	DMAC0_N0TB_11	32	32
A006 24CCh	DMA0	Next 1 source address register 11	DMAC0_N1SA_11_N	32	32
A006 24CCh	DMA0	Next 1 source address register 11	DMAC0_N1SA_11_W	32	32
A006 24D0h	DMA0	Next 1 destination address register 11	DMAC0_N1DA_11	32	32
A006 24D4h	DMA0	Next 1 transaction byte register 11	DMAC0_N1TB_11	32	32
A006 24D8h	DMA0	Current source address register 11	DMAC0_CRSA_11	32	32
A006 24DCh	DMA0	Current destination address register 11	DMAC0_CRDA_11	32	32
A006 24E0h	DMA0	Current transaction byte register 11	DMAC0_CRTB_11	32	32
A006 24E4h	DMA0	Channel status register 11	DMAC0_CHSTAT_11	32	32
A006 24E8h	DMA0	Channel control register 11	DMAC0_CHCTRL_11	32	32
A006 24ECh	DMA0	Channel configuration register 11	DMAC0_CHCFG_11	32	32
A006 24F0h	DMA0	Channel interval register 11	DMAC0_CHITVL_11	32	32
A006 24F8h	DMA0	Next link address register 11	DMAC0_NXLA_11	32	32
A006 24FCh	DMA0	Current link address register 11	DMAC0_CRLA_11	32	32

Table 5.1 List of I/O Registers (Address Order) (23 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 2500h	DMA0	Next 0 source address register 12	DMAC0_N0SA_12_N	32	32
A006 2500h	DMA0	Next 0 source address register 12	DMAC0_N0SA_12_W	32	32
A006 2504h	DMA0	Next 0 destination address register 12	DMAC0_N0DA_12	32	32
A006 2508h	DMA0	Next 0 transaction byte register 12	DMAC0_N0TB_12	32	32
A006 250Ch	DMA0	Next 1 source address register 12	DMAC0_N1SA_12_N	32	32
A006 250Ch	DMA0	Next 1 source address register 12	DMAC0_N1SA_12_W	32	32
A006 2510h	DMA0	Next 1 destination address register 12	DMAC0_N1DA_12	32	32
A006 2514h	DMA0	Next 1 transaction byte register 12	DMAC0_N1TB_12	32	32
A006 2518h	DMA0	Current source address register 12	DMAC0_CRSA_12	32	32
A006 251Ch	DMA0	Current destination address register 12	DMAC0_CRDA_12	32	32
A006 2520h	DMA0	Current transaction byte register 12	DMAC0_CRTB_12	32	32
A006 2524h	DMA0	Channel status register 12	DMAC0_CHSTAT_12	32	32
A006 2528h	DMA0	Channel control register 12	DMAC0_CHCTRL_12	32	32
A006 252Ch	DMA0	Channel configuration register 12	DMAC0_CHCFG_12	32	32
A006 2530h	DMA0	Channel interval register 12	DMAC0_CHITVL_12	32	32
A006 2538h	DMA0	Next link address register 12	DMAC0_NXLA_12	32	32
A006 253Ch	DMA0	Current link address register 12	DMAC0_CRLA_12	32	32
A006 2540h	DMA0	Next 0 source address register 13	DMAC0_N0SA_13_N	32	32
A006 2540h	DMA0	Next 0 source address register 13	DMAC0_N0SA_13_W	32	32
A006 2544h	DMA0	Next 0 destination address register 13	DMAC0_N0DA_13	32	32
A006 2548h	DMA0	Next 0 transaction byte register 13	DMAC0_N0TB_13	32	32
A006 254Ch	DMA0	Next 1 source address register 13	DMAC0_N1SA_13_N	32	32
A006 254Ch	DMA0	Next 1 source address register 13	DMAC0_N1SA_13_W	32	32
A006 2550h	DMA0	Next 1 destination address register 13	DMAC0_N1DA_13	32	32
A006 2554h	DMA0	Next 1 transaction byte register 13	DMAC0_N1TB_13	32	32
A006 2558h	DMA0	Current source address register 13	DMAC0_CRSA_13	32	32
A006 255Ch	DMA0	Current destination address register 13	DMAC0_CRDA_13	32	32
A006 2560h	DMA0	Current transaction byte register 13	DMAC0_CRTB_13	32	32
A006 2564h	DMA0	Channel status register 13	DMAC0_CHSTAT_13	32	32
A006 2568h	DMA0	Channel control register 13	DMAC0_CHCTRL_13	32	32
A006 256Ch	DMA0	Channel configuration register 13	DMAC0_CHCFG_13	32	32
A006 2570h	DMA0	Channel interval register 13	DMAC0_CHITVL_13	32	32
A006 2578h	DMA0	Next link address register 13	DMAC0_NXLA_13	32	32
A006 257Ch	DMA0	Current link address register 13	DMAC0_CRLA_13	32	32
A006 2580h	DMA0	Next 0 source address register 14	DMAC0_N0SA_14_N	32	32
A006 2580h	DMA0	Next 0 source address register 14	DMAC0_N0SA_14_W	32	32
A006 2584h	DMA0	Next 0 destination address register 14	DMAC0_N0DA_14	32	32
A006 2588h	DMA0	Next 0 transaction byte register 14	DMAC0_N0TB_14	32	32
A006 258Ch	DMA0	Next 1 source address register 14	DMAC0_N1SA_14_N	32	32
A006 258Ch	DMA0	Next 1 source address register 14	DMAC0_N1SA_14_W	32	32
A006 2590h	DMA0	Next 1 destination address register 14	DMAC0_N1DA_14	32	32
A006 2594h	DMA0	Next 1 transaction byte register 14	DMAC0_N1TB_14	32	32
A006 2598h	DMA0	Current source address register 14	DMAC0_CRSA_14	32	32
A006 259Ch	DMA0	Current destination address register 14	DMAC0_CRDA_14	32	32
A006 25A0h	DMA0	Current transaction byte register 14	DMAC0_CRTB_14	32	32
A006 25A4h	DMA0	Channel status register 14	DMAC0_CHSTAT_14	32	32
A006 25A8h	DMA0	Channel control register 14	DMAC0_CHCTRL_14	32	32
A006 25ACh	DMA0	Channel configuration register 14	DMAC0_CHCFG_14	32	32

Table 5.1 List of I/O Registers (Address Order) (24 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 25B0h	DMA0	Channel interval register 14	DMAC0_CHITVL_14	32	32
A006 25B8h	DMA0	Next link address register 14	DMAC0_NXLA_14	32	32
A006 25BCh	DMA0	Current link address register 14	DMAC0_CRLA_14	32	32
A006 25C0h	DMA0	Next 0 source address register 15	DMAC0_N0SA_15_N	32	32
A006 25C0h	DMA0	Next 0 source address register 15	DMAC0_N0SA_15_W	32	32
A006 25C4h	DMA0	Next 0 destination address register 15	DMAC0_N0DA_15	32	32
A006 25C8h	DMA0	Next 0 transaction byte register 15	DMAC0_N0TB_15	32	32
A006 25CCh	DMA0	Next 1 source address register 15	DMAC0_N1SA_15_N	32	32
A006 25CCh	DMA0	Next 1 source address register 15	DMAC0_N1SA_15_W	32	32
A006 25D0h	DMA0	Next 1 destination address register 15	DMAC0_N1DA_15	32	32
A006 25D4h	DMA0	Next 1 transaction byte register 15	DMAC0_N1TB_15	32	32
A006 25D8h	DMA0	Current source address register 15	DMAC0_CRSA_15	32	32
A006 25DCh	DMA0	Current destination address register 15	DMAC0_CRDA_15	32	32
A006 25E0h	DMA0	Current transaction byte register 15	DMAC0_CRTB_15	32	32
A006 25E4h	DMA0	Channel status register 15	DMAC0_CHSTAT_15	32	32
A006 25E8h	DMA0	Channel control register 15	DMAC0_CHCTRL_15	32	32
A006 25ECh	DMA0	Channel configuration register 15	DMAC0_CHCFG_15	32	32
A006 25F0h	DMA0	Channel interval register 15	DMAC0_CHITVL_15	32	32
A006 25F8h	DMA0	Next link address register 15	DMAC0_NXLA_15	32	32
A006 25FCh	DMA0	Current link address register 15	DMAC0_CRLA_15	32	32
A006 2600h	DMA0	Source continuous register 8	DMAC0_SCNT_8	32	32
A006 2604h	DMA0	Source skip register 8	DMAC0_SSKP_8	32	32
A006 2608h	DMA0	Destination continuous register 8	DMAC0_DCNT_8	32	32
A006 260Ch	DMA0	Destination skip register 8	DMAC0_DSKP_8	32	32
A006 2620h	DMA0	Source continuous register 9	DMAC0_SCNT_9	32	32
A006 2624h	DMA0	Source skip register 9	DMAC0_SSKP_9	32	32
A006 2628h	DMA0	Destination continuous register 9	DMAC0_DCNT_9	32	32
A006 262Ch	DMA0	Destination skip register 9	DMAC0_DSKP_9	32	32
A006 2640h	DMA0	Source continuous register 10	DMAC0_SCNT_10	32	32
A006 2644h	DMA0	Source skip register 10	DMAC0_SSKP_10	32	32
A006 2648h	DMA0	Destination continuous register 10	DMAC0_DCNT_10	32	32
A006 264Ch	DMA0	Destination skip register 10	DMAC0_DSKP_10	32	32
A006 2660h	DMA0	Source continuous register 11	DMAC0_SCNT_11	32	32
A006 2664h	DMA0	Source skip register 11	DMAC0_SSKP_11	32	32
A006 2668h	DMA0	Destination continuous register 11	DMAC0_DCNT_11	32	32
A006 266Ch	DMA0	Destination skip register 11	DMAC0_DSKP_11	32	32
A006 2680h	DMA0	Source continuous register 12	DMAC0_SCNT_12	32	32
A006 2684h	DMA0	Source skip register 12	DMAC0_SSKP_12	32	32
A006 2688h	DMA0	Destination continuous register 12	DMAC0_DCNT_12	32	32
A006 268Ch	DMA0	Destination skip register 12	DMAC0_DSKP_12	32	32
A006 26A0h	DMA0	Source continuous register 13	DMAC0_SCNT_13	32	32
A006 26A4h	DMA0	Source skip register 13	DMAC0_SSKP_13	32	32
A006 26A8h	DMA0	Destination continuous register 13	DMAC0_DCNT_13	32	32
A006 26ACh	DMA0	Destination skip register 13	DMAC0_DSKP_13	32	32
A006 26C0h	DMA0	Source continuous register 14	DMAC0_SCNT_14	32	32
A006 26C4h	DMA0	Source skip register 14	DMAC0_SSKP_14	32	32
A006 26C8h	DMA0	Destination continuous register 14	DMAC0_DCNT_14	32	32
A006 26CCh	DMA0	Destination skip register 14	DMAC0_DSKP_14	32	32

Table 5.1 List of I/O Registers (Address Order) (25 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 26E0h	DMA0	Source continuous register 15	DMAC0_SCNT_15	32	32
A006 26E4h	DMA0	Source skip register 15	DMAC0_SSKP_15	32	32
A006 26E8h	DMA0	Destination continuous register 15	DMAC0_DCNT_15	32	32
A006 26ECh	DMA0	Destination skip register 15	DMAC0_DSKP_15	32	32
A006 2700h	DMA0	DMA control register B	DMAC0_DCTRL_B	32	32
A006 2704h	DMA0	Descriptor interval register B	DMAC0_DSCITVL_B	32	32
A006 2710h	DMA0	DMA status EN register B	DMAC0_DST_EN_B	32	32
A006 2714h	DMA0	DMA status ER register B	DMAC0_DST_ER_B	32	32
A006 2718h	DMA0	DMA status END register B	DMAC0_DST_END_B	32	32
A006 2720h	DMA0	DMA status SUS register B	DMAC0_DST_SUS_B	32	32
A006 3000h	DMA1	Next 0 source address register 0	DMAC1_N0SA_0_N	32	32
A006 3000h	DMA1	Next 0 source address register 0	DMAC1_N0SA_0_W	32	32
A006 3004h	DMA1	Next 0 destination address register 0	DMAC1_N0DA_0	32	32
A006 3008h	DMA1	Next 0 transaction byte register 0	DMAC1_N0TB_0	32	32
A006 300Ch	DMA1	Next 1 source address register 0	DMAC1_N1SA_0_N	32	32
A006 300Ch	DMA1	Next 1 source address register 0	DMAC1_N1SA_0_W	32	32
A006 3010h	DMA1	Next 1 destination address register 0	DMAC1_N1DA_0	32	32
A006 3014h	DMA1	Next 1 transaction byte register 0	DMAC1_N1TB_0	32	32
A006 3018h	DMA1	Current source address register 0	DMAC1_CRSA_0	32	32
A006 301Ch	DMA1	Current destination address register 0	DMAC1_CRDA_0	32	32
A006 3020h	DMA1	Current transaction byte register 0	DMAC1_CRTB_0	32	32
A006 3024h	DMA1	Channel status register 0	DMAC1_CHSTAT_0	32	32
A006 3028h	DMA1	Channel control register 0	DMAC1_CHCTRL_0	32	32
A006 302Ch	DMA1	Channel configuration register 0	DMAC1_CHCFG_0	32	32
A006 3030h	DMA1	Channel interval register 0	DMAC1_CHITVL_0	32	32
A006 3038h	DMA1	Next link address register 0	DMAC1_NXLA_0	32	32
A006 303Ch	DMA1	Current link address register 0	DMAC1_CRLA_0	32	32
A006 3040h	DMA1	Next 0 source address register 1	DMAC1_N0SA_1_N	32	32
A006 3040h	DMA1	Next 0 source address register 1	DMAC1_N0SA_1_W	32	32
A006 3044h	DMA1	Next 0 destination address register 1	DMAC1_N0DA_1	32	32
A006 3048h	DMA1	Next 0 transaction byte register 1	DMAC1_N0TB_1	32	32
A006 304Ch	DMA1	Next 1 source address register 1	DMAC1_N1SA_1_N	32	32
A006 304Ch	DMA1	Next 1 source address register 1	DMAC1_N1SA_1_W	32	32
A006 3050h	DMA1	Next 1 destination address register 1	DMAC1_N1DA_1	32	32
A006 3054h	DMA1	Next 1 transaction byte register 1	DMAC1_N1TB_1	32	32
A006 3058h	DMA1	Current source address register 1	DMAC1_CRSA_1	32	32
A006 305Ch	DMA1	Current destination address register 1	DMAC1_CRDA_1	32	32
A006 3060h	DMA1	Current transaction byte register 1	DMAC1_CRTB_1	32	32
A006 3064h	DMA1	Channel status register 1	DMAC1_CHSTAT_1	32	32
A006 3068h	DMA1	Channel control register 1	DMAC1_CHCTRL_1	32	32
A006 306Ch	DMA1	Channel configuration register 1	DMAC1_CHCFG_1	32	32
A006 3070h	DMA1	Channel interval register 1	DMAC1_CHITVL_1	32	32
A006 3078h	DMA1	Next link address register 1	DMAC1_NXLA_1	32	32
A006 307Ch	DMA1	Current link address register 1	DMAC1_CRLA_1	32	32
A006 3080h	DMA1	Next 0 source address register 2	DMAC1_N0SA_2_N	32	32
A006 3080h	DMA1	Next 0 source address register 2	DMAC1_N0SA_2_W	32	32
A006 3084h	DMA1	Next 0 destination address register 2	DMAC1_N0DA_2	32	32
A006 3088h	DMA1	Next 0 transaction byte register 2	DMAC1_N0TB_2	32	32

Table 5.1 List of I/O Registers (Address Order) (26 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 308Ch	DMA1	Next 1 source address register 2	DMAC1_N1SA_2_N	32	32
A006 308Ch	DMA1	Next 1 source address register 2	DMAC1_N1SA_2_W	32	32
A006 3090h	DMA1	Next 1 destination address register 2	DMAC1_N1DA_2	32	32
A006 3094h	DMA1	Next 1 transaction byte register 2	DMAC1_N1TB_2	32	32
A006 3098h	DMA1	Current source address register 2	DMAC1_CRSA_2	32	32
A006 309Ch	DMA1	Current destination address register 2	DMAC1_CRDA_2	32	32
A006 30A0h	DMA1	Current transaction byte register 2	DMAC1_CRTB_2	32	32
A006 30A4h	DMA1	Channel status register 2	DMAC1_CHSTAT_2	32	32
A006 30A8h	DMA1	Channel control register 2	DMAC1_CHCTRL_2	32	32
A006 30ACh	DMA1	Channel configuration register 2	DMAC1_CHCFG_2	32	32
A006 30B0h	DMA1	Channel interval register 2	DMAC1_CHITVL_2	32	32
A006 30B8h	DMA1	Next link address register 2	DMAC1_NXLA_2	32	32
A006 30BCh	DMA1	Current link address register 2	DMAC1_CRLA_2	32	32
A006 30C0h	DMA1	Next 0 source address register 3	DMAC1_N0SA_3_N	32	32
A006 30C0h	DMA1	Next 0 source address register 3	DMAC1_N0SA_3_W	32	32
A006 30C4h	DMA1	Next 0 destination address register 3	DMAC1_N0DA_3	32	32
A006 30C8h	DMA1	Next 0 transaction byte register 3	DMAC1_N0TB_3	32	32
A006 30CCh	DMA1	Next 1 source address register 3	DMAC1_N1SA_3_N	32	32
A006 30CCh	DMA1	Next 1 source address register 3	DMAC1_N1SA_3_W	32	32
A006 30D0h	DMA1	Next 1 destination address register 3	DMAC1_N1DA_3	32	32
A006 30D4h	DMA1	Next 1 transaction byte register 3	DMAC1_N1TB_3	32	32
A006 30D8h	DMA1	Current source address register 3	DMAC1_CRSA_3	32	32
A006 30DCh	DMA1	Current destination address register 3	DMAC1_CRDA_3	32	32
A006 30E0h	DMA1	Current transaction byte register 3	DMAC1_CRTB_3	32	32
A006 30E4h	DMA1	Channel status register 3	DMAC1_CHSTAT_3	32	32
A006 30E8h	DMA1	Channel control register 3	DMAC1_CHCTRL_3	32	32
A006 30ECh	DMA1	Channel configuration register 3	DMAC1_CHCFG_3	32	32
A006 30F0h	DMA1	Channel interval register 3	DMAC1_CHITVL_3	32	32
A006 30F8h	DMA1	Next link address register 3	DMAC1_NXLA_3	32	32
A006 30FCh	DMA1	Current link address register 3	DMAC1_CRLA_3	32	32
A006 3100h	DMA1	Next 0 source address register 4	DMAC1_N0SA_4_N	32	32
A006 3100h	DMA1	Next 0 source address register 4	DMAC1_N0SA_4_W	32	32
A006 3104h	DMA1	Next 0 destination address register 4	DMAC1_N0DA_4	32	32
A006 3108h	DMA1	Next 0 transaction byte register 4	DMAC1_N0TB_4	32	32
A006 310Ch	DMA1	Next 1 source address register 4	DMAC1_N1SA_4_N	32	32
A006 310Ch	DMA1	Next 1 source address register 4	DMAC1_N1SA_4_W	32	32
A006 3110h	DMA1	Next 1 destination address register 4	DMAC1_N1DA_4	32	32
A006 3114h	DMA1	Next 1 transaction byte register 4	DMAC1_N1TB_4	32	32
A006 3118h	DMA1	Current source address register 4	DMAC1_CRSA_4	32	32
A006 311Ch	DMA1	Current destination address register 4	DMAC1_CRDA_4	32	32
A006 3120h	DMA1	Current transaction byte register 4	DMAC1_CRTB_4	32	32
A006 3124h	DMA1	Channel status register 4	DMAC1_CHSTAT_4	32	32
A006 3128h	DMA1	Channel control register 4	DMAC1_CHCTRL_4	32	32
A006 312Ch	DMA1	Channel configuration register 4	DMAC1_CHCFG_4	32	32
A006 3130h	DMA1	Channel interval register 4	DMAC1_CHITVL_4	32	32
A006 3138h	DMA1	Next link address register 4	DMAC1_NXLA_4	32	32
A006 313Ch	DMA1	Current link address register 4	DMAC1_CRLA_4	32	32
A006 3140h	DMA1	Next 0 source address register 5	DMAC1_N0SA_5_N	32	32

Table 5.1 List of I/O Registers (Address Order) (27 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 3140h	DMA1	Next 0 source address register 5	DMAC1_N0SA_5_W	32	32
A006 3144h	DMA1	Next 0 destination address register 5	DMAC1_N0DA_5	32	32
A006 3148h	DMA1	Next 0 transaction byte register 5	DMAC1_N0TB_5	32	32
A006 314Ch	DMA1	Next 1 source address register 5	DMAC1_N1SA_5_N	32	32
A006 314Ch	DMA1	Next 1 source address register 5	DMAC1_N1SA_5_W	32	32
A006 3150h	DMA1	Next 1 destination address register 5	DMAC1_N1DA_5	32	32
A006 3154h	DMA1	Next 1 transaction byte register 5	DMAC1_N1TB_5	32	32
A006 3158h	DMA1	Current source address register 5	DMAC1_CRSA_5	32	32
A006 315Ch	DMA1	Current destination address register 5	DMAC1_CRDA_5	32	32
A006 3160h	DMA1	Current transaction byte register 5	DMAC1_CRTB_5	32	32
A006 3164h	DMA1	Channel status register 5	DMAC1_CHSTAT_5	32	32
A006 3168h	DMA1	Channel control register 5	DMAC1_CHCTRL_5	32	32
A006 316Ch	DMA1	Channel configuration register 5	DMAC1_CHCFG_5	32	32
A006 3170h	DMA1	Channel interval register 5	DMAC1_CHITVL_5	32	32
A006 3178h	DMA1	Next link address register 5	DMAC1_NXLA_5	32	32
A006 317Ch	DMA1	Current link address register 5	DMAC1_CRLA_5	32	32
A006 3180h	DMA1	Next 0 source address register 6	DMAC1_N0SA_6_N	32	32
A006 3180h	DMA1	Next 0 source address register 6	DMAC1_N0SA_6_W	32	32
A006 3184h	DMA1	Next 0 destination address register 6	DMAC1_N0DA_6	32	32
A006 3188h	DMA1	Next 0 transaction byte register 6	DMAC1_N0TB_6	32	32
A006 318Ch	DMA1	Next 1 source address register 6	DMAC1_N1SA_6_N	32	32
A006 318Ch	DMA1	Next 1 source address register 6	DMAC1_N1SA_6_W	32	32
A006 3190h	DMA1	Next 1 destination address register 6	DMAC1_N1DA_6	32	32
A006 3194h	DMA1	Next 1 transaction byte register 6	DMAC1_N1TB_6	32	32
A006 3198h	DMA1	Current source address register 6	DMAC1_CRSA_6	32	32
A006 319Ch	DMA1	Current destination address register 6	DMAC1_CRDA_6	32	32
A006 31A0h	DMA1	Current transaction byte register 6	DMAC1_CRTB_6	32	32
A006 31A4h	DMA1	Channel status register 6	DMAC1_CHSTAT_6	32	32
A006 31A8h	DMA1	Channel control register 6	DMAC1_CHCTRL_6	32	32
A006 31ACh	DMA1	Channel configuration register 6	DMAC1_CHCFG_6	32	32
A006 31B0h	DMA1	Channel interval register 6	DMAC1_CHITVL_6	32	32
A006 31B8h	DMA1	Next link address register 6	DMAC1_NXLA_6	32	32
A006 31BCh	DMA1	Current link address register 6	DMAC1_CRLA_6	32	32
A006 31C0h	DMA1	Next 0 source address register 7	DMAC1_N0SA_7_N	32	32
A006 31C0h	DMA1	Next 0 source address register 7	DMAC1_N0SA_7_W	32	32
A006 31C4h	DMA1	Next 0 destination address register 7	DMAC1_N0DA_7	32	32
A006 31C8h	DMA1	Next 0 transaction byte register 7	DMAC1_N0TB_7	32	32
A006 31CCh	DMA1	Next 1 source address register 7	DMAC1_N1SA_7_N	32	32
A006 31CCh	DMA1	Next 1 source address register 7	DMAC1_N1SA_7_W	32	32
A006 31D0h	DMA1	Next 1 destination address register 7	DMAC1_N1DA_7	32	32
A006 31D4h	DMA1	Next 1 transaction byte register 7	DMAC1_N1TB_7	32	32
A006 31D8h	DMA1	Current source address register 7	DMAC1_CRSA_7	32	32
A006 31DCh	DMA1	Current destination address register 7	DMAC1_CRDA_7	32	32
A006 31E0h	DMA1	Current transaction byte register 7	DMAC1_CRTB_7	32	32
A006 31E4h	DMA1	Channel status register 7	DMAC1_CHSTAT_7	32	32
A006 31E8h	DMA1	Channel control register 7	DMAC1_CHCTRL_7	32	32
A006 31ECh	DMA1	Channel configuration register 7	DMAC1_CHCFG_7	32	32
A006 31F0h	DMA1	Channel interval register 7	DMAC1_CHITVL_7	32	32

Table 5.1 List of I/O Registers (Address Order) (28 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 31F8h	DMA1	Next link address register 7	DMAC1_NXLA_7	32	32
A006 31FCh	DMA1	Current link address register 7	DMAC1_CRLA_7	32	32
A006 3200h	DMA1	Source continuous register 0	DMAC1_SCNT_0	32	32
A006 3204h	DMA1	Source skip register 0	DMAC1_SSKP_0	32	32
A006 3208h	DMA1	Destination continuous register 0	DMAC1_DCNT_0	32	32
A006 320Ch	DMA1	Destination skip register 0	DMAC1_DSKP_0	32	32
A006 3220h	DMA1	Source continuous register 1	DMAC1_SCNT_1	32	32
A006 3224h	DMA1	Source skip register 1	DMAC1_SSKP_1	32	32
A006 3228h	DMA1	Destination continuous register 1	DMAC1_DCNT_1	32	32
A006 322Ch	DMA1	Destination skip register 1	DMAC1_DSKP_1	32	32
A006 3240h	DMA1	Source continuous register 2	DMAC1_SCNT_2	32	32
A006 3244h	DMA1	Source skip register 2	DMAC1_SSKP_2	32	32
A006 3248h	DMA1	Destination continuous register 2	DMAC1_DCNT_2	32	32
A006 324Ch	DMA1	Destination skip register 2	DMAC1_DSKP_2	32	32
A006 3260h	DMA1	Source continuous register 3	DMAC1_SCNT_3	32	32
A006 3264h	DMA1	Source skip register 3	DMAC1_SSKP_3	32	32
A006 3268h	DMA1	Destination continuous register 3	DMAC1_DCNT_3	32	32
A006 326Ch	DMA1	Destination skip register 3	DMAC1_DSKP_3	32	32
A006 3280h	DMA1	Source continuous register 4	DMAC1_SCNT_4	32	32
A006 3284h	DMA1	Source skip register 4	DMAC1_SSKP_4	32	32
A006 3288h	DMA1	Destination continuous register 4	DMAC1_DCNT_4	32	32
A006 328Ch	DMA1	Destination skip register 4	DMAC1_DSKP_4	32	32
A006 32A0h	DMA1	Source continuous register 5	DMAC1_SCNT_5	32	32
A006 32A4h	DMA1	Source skip register 5	DMAC1_SSKP_5	32	32
A006 32A8h	DMA1	Destination continuous register 5	DMAC1_DCNT_5	32	32
A006 32ACh	DMA1	Destination skip register 5	DMAC1_DSKP_5	32	32
A006 32C0h	DMA1	Source continuous register 6	DMAC1_SCNT_6	32	32
A006 32C4h	DMA1	Source skip register 6	DMAC1_SSKP_6	32	32
A006 32C8h	DMA1	Destination continuous register 6	DMAC1_DCNT_6	32	32
A006 32CCh	DMA1	Destination skip register 6	DMAC1_DSKP_6	32	32
A006 32E0h	DMA1	Source continuous register 7	DMAC1_SCNT_7	32	32
A006 32E4h	DMA1	Source skip register 7	DMAC1_SSKP_7	32	32
A006 32E8h	DMA1	Destination continuous register 7	DMAC1_DCNT_7	32	32
A006 32ECh	DMA1	Destination skip register 7	DMAC1_DSKP_7	32	32
A006 3300h	DMA1	DMA control register A	DMAC1_DCTRL_A	32	32
A006 3304h	DMA1	Descriptor interval register A	DMAC1_DSCITVL_A	32	32
A006 3310h	DMA1	DMA status EN register A	DMAC1_DST_EN_A	32	32
A006 3314h	DMA1	DMA status ER register A	DMAC1_DST_ER_A	32	32
A006 3318h	DMA1	DMA status END register A	DMAC1_DST_END_A	32	32
A006 3320h	DMA1	DMA status SUS register A	DMAC1_DST_SUS_A	32	32
A006 3400h	DMA1	Next 0 source address register 8	DMAC1_N0SA_8_N	32	32
A006 3400h	DMA1	Next 0 source address register 8	DMAC1_N0SA_8_W	32	32
A006 3404h	DMA1	Next 0 destination address register 8	DMAC1_N0DA_8	32	32
A006 3408h	DMA1	Next 0 transaction byte register 8	DMAC1_N0TB_8	32	32
A006 340Ch	DMA1	Next 1 source address register 8	DMAC1_N1SA_8_N	32	32
A006 340Ch	DMA1	Next 1 source address register 8	DMAC1_N1SA_8_W	32	32
A006 3410h	DMA1	Next 1 destination address register 8	DMAC1_N1DA_8	32	32
A006 3414h	DMA1	Next 1 transaction byte register 8	DMAC1_N1TB_8	32	32

Table 5.1 List of I/O Registers (Address Order) (29 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 3418h	DMA1	Current source address register 8	DMAC1_CRSA_8	32	32
A006 341Ch	DMA1	Current destination address register 8	DMAC1_CRDA_8	32	32
A006 3420h	DMA1	Current transaction byte register 8	DMAC1_CRTB_8	32	32
A006 3424h	DMA1	Channel status register 8	DMAC1_CHSTAT_8	32	32
A006 3428h	DMA1	Channel control register 8	DMAC1_CHCTRL_8	32	32
A006 342Ch	DMA1	Channel configuration register 8	DMAC1_CHCFG_8	32	32
A006 3430h	DMA1	Channel interval register 8	DMAC1_CHITVL_8	32	32
A006 3438h	DMA1	Next link address register 8	DMAC1_NXLA_8	32	32
A006 343Ch	DMA1	Current link address register 8	DMAC1_CRLA_8	32	32
A006 3440h	DMA1	Next 0 source address register 9	DMAC1_N0SA_9_N	32	32
A006 3440h	DMA1	Next 0 source address register 9	DMAC1_N0SA_9_W	32	32
A006 3444h	DMA1	Next 0 destination address register 9	DMAC1_N0DA_9	32	32
A006 3448h	DMA1	Next 0 transaction byte register 9	DMAC1_N0TB_9	32	32
A006 344Ch	DMA1	Next 1 source address register 9	DMAC1_N1SA_9_N	32	32
A006 344Ch	DMA1	Next 1 source address register 9	DMAC1_N1SA_9_W	32	32
A006 3450h	DMA1	Next 1 destination address register 9	DMAC1_N1DA_9	32	32
A006 3454h	DMA1	Next 1 transaction byte register 9	DMAC1_N1TB_9	32	32
A006 3458h	DMA1	Current source address register 9	DMAC1_CRSA_9	32	32
A006 345Ch	DMA1	Current destination address register 9	DMAC1_CRDA_9	32	32
A006 3460h	DMA1	Current transaction byte register 9	DMAC1_CRTB_9	32	32
A006 3464h	DMA1	Channel status register 9	DMAC1_CHSTAT_9	32	32
A006 3468h	DMA1	Channel control register 9	DMAC1_CHCTRL_9	32	32
A006 346Ch	DMA1	Channel configuration register 9	DMAC1_CHCFG_9	32	32
A006 3470h	DMA1	Channel interval register 9	DMAC1_CHITVL_9	32	32
A006 3478h	DMA1	Next link address register 9	DMAC1_NXLA_9	32	32
A006 347Ch	DMA1	Current link address register 9	DMAC1_CRLA_9	32	32
A006 3480h	DMA1	Next 0 source address register 10	DMAC1_N0SA_10_N	32	32
A006 3480h	DMA1	Next 0 source address register 10	DMAC1_N0SA_10_W	32	32
A006 3484h	DMA1	Next 0 destination address register 10	DMAC1_N0DA_10	32	32
A006 3488h	DMA1	Next 0 transaction byte register 10	DMAC1_N0TB_10	32	32
A006 348Ch	DMA1	Next 1 source address register 10	DMAC1_N1SA_10_N	32	32
A006 348Ch	DMA1	Next 1 source address register 10	DMAC1_N1SA_10_W	32	32
A006 3490h	DMA1	Next 1 destination address register 10	DMAC1_N1DA_10	32	32
A006 3494h	DMA1	Next 1 transaction byte register 10	DMAC1_N1TB_10	32	32
A006 3498h	DMA1	Current source address register 10	DMAC1_CRSA_10	32	32
A006 349Ch	DMA1	Current destination address register 10	DMAC1_CRDA_10	32	32
A006 34A0h	DMA1	Current transaction byte register 10	DMAC1_CRTB_10	32	32
A006 34A4h	DMA1	Channel status register 10	DMAC1_CHSTAT_10	32	32
A006 34A8h	DMA1	Channel control register 10	DMAC1_CHCTRL_10	32	32
A006 34ACh	DMA1	Channel configuration register 10	DMAC1_CHCFG_10	32	32
A006 34B0h	DMA1	Channel interval register 10	DMAC1_CHITVL_10	32	32
A006 34B8h	DMA1	Next link address register 10	DMAC1_NXLA_10	32	32
A006 34BCh	DMA1	Current link address register 10	DMAC1_CRLA_10	32	32
A006 34C0h	DMA1	Next 0 source address register 11	DMAC1_N0SA_11_N	32	32
A006 34C0h	DMA1	Next 0 source address register 11	DMAC1_N0SA_11_W	32	32
A006 34C4h	DMA1	Next 0 destination address register 11	DMAC1_N0DA_11	32	32
A006 34C8h	DMA1	Next 0 transaction byte register 11	DMAC1_N0TB_11	32	32
A006 34CCh	DMA1	Next 1 source address register 11	DMAC1_N1SA_11_N	32	32

Table 5.1 List of I/O Registers (Address Order) (30 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 34CCh	DMA1	Next 1 source address register 11	DMAC1_N1SA_11_W	32	32
A006 34D0h	DMA1	Next 1 destination address register 11	DMAC1_N1DA_11	32	32
A006 34D4h	DMA1	Next 1 transaction byte register 11	DMAC1_N1TB_11	32	32
A006 34D8h	DMA1	Current source address register 11	DMAC1_CRSA_11	32	32
A006 34DCh	DMA1	Current destination address register 11	DMAC1_CRDA_11	32	32
A006 34E0h	DMA1	Current transaction byte register 11	DMAC1_CRTB_11	32	32
A006 34E4h	DMA1	Channel status register 11	DMAC1_CHSTAT_11	32	32
A006 34E8h	DMA1	Channel control register 11	DMAC1_CHCTRL_11	32	32
A006 34ECh	DMA1	Channel configuration register 11	DMAC1_CHCFG_11	32	32
A006 34F0h	DMA1	Channel interval register 11	DMAC1_CHITVL_11	32	32
A006 34F8h	DMA1	Next link address register 11	DMAC1_NXLA_11	32	32
A006 34FCh	DMA1	Current link address register 11	DMAC1_CRLA_11	32	32
A006 3500h	DMA1	Next 0 source address register 12	DMAC1_N0SA_12_N	32	32
A006 3500h	DMA1	Next 0 source address register 12	DMAC1_N0SA_12_W	32	32
A006 3504h	DMA1	Next 0 destination address register 12	DMAC1_N0DA_12	32	32
A006 3508h	DMA1	Next 0 transaction byte register 12	DMAC1_N0TB_12	32	32
A006 350Ch	DMA1	Next 1 source address register 12	DMAC1_N1SA_12_N	32	32
A006 350Ch	DMA1	Next 1 source address register 12	DMAC1_N1SA_12_W	32	32
A006 3510h	DMA1	Next 1 destination address register 12	DMAC1_N1DA_12	32	32
A006 3514h	DMA1	Next 1 transaction byte register 12	DMAC1_N1TB_12	32	32
A006 3518h	DMA1	Current source address register 12	DMAC1_CRSA_12	32	32
A006 351Ch	DMA1	Current destination address register 12	DMAC1_CRDA_12	32	32
A006 3520h	DMA1	Current transaction byte register 12	DMAC1_CRTB_12	32	32
A006 3524h	DMA1	Channel status register 12	DMAC1_CHSTAT_12	32	32
A006 3528h	DMA1	Channel control register 12	DMAC1_CHCTRL_12	32	32
A006 352Ch	DMA1	Channel configuration register 12	DMAC1_CHCFG_12	32	32
A006 3530h	DMA1	Channel interval register 12	DMAC1_CHITVL_12	32	32
A006 3538h	DMA1	Next link address register 12	DMAC1_NXLA_12	32	32
A006 353Ch	DMA1	Current link address register 12	DMAC1_CRLA_12	32	32
A006 3540h	DMA1	Next 0 source address register 13	DMAC1_N0SA_13_N	32	32
A006 3540h	DMA1	Next 0 source address register 13	DMAC1_N0SA_13_W	32	32
A006 3544h	DMA1	Next 0 destination address register 13	DMAC1_N0DA_13	32	32
A006 3548h	DMA1	Next 0 transaction byte register 13	DMAC1_N0TB_13	32	32
A006 354Ch	DMA1	Next 1 source address register 13	DMAC1_N1SA_13_N	32	32
A006 354Ch	DMA1	Next 1 source address register 13	DMAC1_N1SA_13_W	32	32
A006 3550h	DMA1	Next 1 destination address register 13	DMAC1_N1DA_13	32	32
A006 3554h	DMA1	Next 1 transaction byte register 13	DMAC1_N1TB_13	32	32
A006 3558h	DMA1	Current source address register 13	DMAC1_CRSA_13	32	32
A006 355Ch	DMA1	Current destination address register 13	DMAC1_CRDA_13	32	32
A006 3560h	DMA1	Current transaction byte register 13	DMAC1_CRTB_13	32	32
A006 3564h	DMA1	Channel status register 13	DMAC1_CHSTAT_13	32	32
A006 3568h	DMA1	Channel control register 13	DMAC1_CHCTRL_13	32	32
A006 356Ch	DMA1	Channel configuration register 13	DMAC1_CHCFG_13	32	32
A006 3570h	DMA1	Channel interval register 13	DMAC1_CHITVL_13	32	32
A006 3578h	DMA1	Next link address register 13	DMAC1_NXLA_13	32	32
A006 357Ch	DMA1	Current link address register 13	DMAC1_CRLA_13	32	32
A006 3580h	DMA1	Next 0 source address register 14	DMAC1_N0SA_14_N	32	32
A006 3580h	DMA1	Next 0 source address register 14	DMAC1_N0SA_14_W	32	32

Table 5.1 List of I/O Registers (Address Order) (31 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 3584h	DMA1	Next 0 destination address register 14	DMAC1_N0DA_14	32	32
A006 3588h	DMA1	Next 0 transaction byte register 14	DMAC1_N0TB_14	32	32
A006 358Ch	DMA1	Next 1 source address register 14	DMAC1_N1SA_14_N	32	32
A006 358Ch	DMA1	Next 1 source address register 14	DMAC1_N1SA_14_W	32	32
A006 3590h	DMA1	Next 1 destination address register 14	DMAC1_N1DA_14	32	32
A006 3594h	DMA1	Next 1 transaction byte register 14	DMAC1_N1TB_14	32	32
A006 3598h	DMA1	Current source address register 14	DMAC1_CRSA_14	32	32
A006 359Ch	DMA1	Current destination address register 14	DMAC1_CRDA_14	32	32
A006 35A0h	DMA1	Current transaction byte register 14	DMAC1_CRTB_14	32	32
A006 35A4h	DMA1	Channel status register 14	DMAC1_CHSTAT_14	32	32
A006 35A8h	DMA1	Channel control register 14	DMAC1_CHCTRL_14	32	32
A006 35ACh	DMA1	Channel configuration register 14	DMAC1_CHCFG_14	32	32
A006 35B0h	DMA1	Channel interval register 14	DMAC1_CHITVL_14	32	32
A006 35B8h	DMA1	Next link address register 14	DMAC1_NXLA_14	32	32
A006 35BCh	DMA1	Current link address register 14	DMAC1_CRLA_14	32	32
A006 35C0h	DMA1	Next 0 source address register 15	DMAC1_N0SA_15_N	32	32
A006 35C0h	DMA1	Next 0 source address register 15	DMAC1_N0SA_15_W	32	32
A006 35C4h	DMA1	Next 0 destination address register 15	DMAC1_N0DA_15	32	32
A006 35C8h	DMA1	Next 0 transaction byte register 15	DMAC1_N0TB_15	32	32
A006 35CCh	DMA1	Next 1 source address register 15	DMAC1_N1SA_15_N	32	32
A006 35CCh	DMA1	Next 1 source address register 15	DMAC1_N1SA_15_W	32	32
A006 35D0h	DMA1	Next 1 destination address register 15	DMAC1_N1DA_15	32	32
A006 35D4h	DMA1	Next 1 transaction byte register 15	DMAC1_N1TB_15	32	32
A006 35D8h	DMA1	Current source address register 15	DMAC1_CRSA_15	32	32
A006 35DCh	DMA1	Current destination address register 15	DMAC1_CRDA_15	32	32
A006 35E0h	DMA1	Current transaction byte register 15	DMAC1_CRTB_15	32	32
A006 35E4h	DMA1	Channel status register 15	DMAC1_CHSTAT_15	32	32
A006 35E8h	DMA1	Channel control register 15	DMAC1_CHCTRL_15	32	32
A006 35ECh	DMA1	Channel configuration register 15	DMAC1_CHCFG_15	32	32
A006 35F0h	DMA1	Channel interval register 15	DMAC1_CHITVL_15	32	32
A006 35F8h	DMA1	Next link address register 15	DMAC1_NXLA_15	32	32
A006 35FCh	DMA1	Current link address register 15	DMAC1_CRLA_15	32	32
A006 3600h	DMA1	Source continuous register 8	DMAC1_SCNT_8	32	32
A006 3604h	DMA1	Source skip register 8	DMAC1_SSKP_8	32	32
A006 3608h	DMA1	Destination continuous register 8	DMAC1_DCNT_8	32	32
A006 360Ch	DMA1	Destination skip register 8	DMAC1_DSKP_8	32	32
A006 3620h	DMA1	Source continuous register 9	DMAC1_SCNT_9	32	32
A006 3624h	DMA1	Source skip register 9	DMAC1_SSKP_9	32	32
A006 3628h	DMA1	Destination continuous register 9	DMAC1_DCNT_9	32	32
A006 362Ch	DMA1	Destination skip register 9	DMAC1_DSKP_9	32	32
A006 3640h	DMA1	Source continuous register 10	DMAC1_SCNT_10	32	32
A006 3644h	DMA1	Source skip register 10	DMAC1_SSKP_10	32	32
A006 3648h	DMA1	Destination continuous register 10	DMAC1_DCNT_10	32	32
A006 364Ch	DMA1	Destination skip register 10	DMAC1_DSKP_10	32	32
A006 3660h	DMA1	Source continuous register 11	DMAC1_SCNT_11	32	32
A006 3664h	DMA1	Source skip register 11	DMAC1_SSKP_11	32	32
A006 3668h	DMA1	Destination continuous register 11	DMAC1_DCNT_11	32	32
A006 366Ch	DMA1	Destination skip register 11	DMAC1_DSKP_11	32	32

Table 5.1 List of I/O Registers (Address Order) (32 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 3680h	DMA1	Source continuous register 12	DMAC1_SCNT_12	32	32
A006 3684h	DMA1	Source skip register 12	DMAC1_SSKP_12	32	32
A006 3688h	DMA1	Destination continuous register 12	DMAC1_DCNT_12	32	32
A006 368Ch	DMA1	Destination skip register 12	DMAC1_DSKP_12	32	32
A006 36A0h	DMA1	Source continuous register 13	DMAC1_SCNT_13	32	32
A006 36A4h	DMA1	Source skip register 13	DMAC1_SSKP_13	32	32
A006 36A8h	DMA1	Destination continuous register 13	DMAC1_DCNT_13	32	32
A006 36ACh	DMA1	Destination skip register 13	DMAC1_DSKP_13	32	32
A006 36C0h	DMA1	Source continuous register 14	DMAC1_SCNT_14	32	32
A006 36C4h	DMA1	Source skip register 14	DMAC1_SSKP_14	32	32
A006 36C8h	DMA1	Destination continuous register 14	DMAC1_DCNT_14	32	32
A006 36CCh	DMA1	Destination skip register 14	DMAC1_DSKP_14	32	32
A006 36E0h	DMA1	Source continuous register 15	DMAC1_SCNT_15	32	32
A006 36E4h	DMA1	Source skip register 15	DMAC1_SSKP_15	32	32
A006 36E8h	DMA1	Destination continuous register 15	DMAC1_DCNT_15	32	32
A006 36ECh	DMA1	Destination skip register 15	DMAC1_DSKP_15	32	32
A006 3700h	DMA1	DMA control register B	DMAC1_DCTRL_B	32	32
A006 3704h	DMA1	Descriptor interval register B	DMAC1_DSCITVL_B	32	32
A006 3710h	DMA1	DMA status EN register B	DMAC1_DST_EN_B	32	32
A006 3714h	DMA1	DMA status ER register B	DMAC1_DST_ER_B	32	32
A006 3718h	DMA1	DMA status END register B	DMAC1_DST_END_B	32	32
A006 3720h	DMA1	DMA status SUS register B	DMAC1_DST_SUS_B	32	32
A006 5000h	SCIFA0	Serial mode register	SMR	16	16
A006 5002h	SCIFA0	Bit rate register	BRR	8	8
A006 5002h	SCIFA0	Modulation duty register	MDDR	8	8
A006 5004h	SCIFA0	Serial control register	SCR	16	16
A006 5006h	SCIFA0	Transmit FIFO data register	FTDR	8	8
A006 5008h	SCIFA0	Serial status register	FSR	16	16
A006 500Ah	SCIFA0	Receive FIFO data register	FRDR	8	8
A006 500Ch	SCIFA0	FIFO control register	FCR	16	16
A006 500Eh	SCIFA0	FIFO data count register	FDR	16	16
A006 5010h	SCIFA0	Serial port register	SPTR	16	16
A006 5012h	SCIFA0	Line status register	LSR	16	16
A006 5014h	SCIFA0	Serial extended mode register	SEMR	8	8
A006 5016h	SCIFA0	FIFO trigger control register	FTCR	16	16
A006 5400h	SCIFA1	Serial mode register	SMR	16	16
A006 5402h	SCIFA1	Bit rate register	BRR	8	8
A006 5402h	SCIFA1	Modulation duty register	MDDR	8	8
A006 5404h	SCIFA1	Serial control register	SCR	16	16
A006 5406h	SCIFA1	Transmit FIFO data register	FTDR	8	8
A006 5408h	SCIFA1	Serial status register	FSR	16	16
A006 540Ah	SCIFA1	Receive FIFO data register	FRDR	8	8
A006 540Ch	SCIFA1	FIFO control register	FCR	16	16
A006 540Eh	SCIFA1	FIFO data count register	FDR	16	16
A006 5410h	SCIFA1	Serial port register	SPTR	16	16
A006 5412h	SCIFA1	Line status register	LSR	16	16
A006 5414h	SCIFA1	Serial extended mode register	SEMR	8	8
A006 5416h	SCIFA1	FIFO trigger control register	FTCR	16	16

Table 5.1 List of I/O Registers (Address Order) (33 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 5800h	SCIFA2	Serial mode register	SMR	16	16
A006 5802h	SCIFA2	Bit rate register	BRR	8	8
A006 5802h	SCIFA2	Modulation duty register	MDDR	8	8
A006 5804h	SCIFA2	Serial control register	SCR	16	16
A006 5806h	SCIFA2	Transmit FIFO data register	FTDR	8	8
A006 5808h	SCIFA2	Serial status register	FSR	16	16
A006 580Ah	SCIFA2	Receive FIFO data register	FRDR	8	8
A006 580Ch	SCIFA2	FIFO control register	FCR	16	16
A006 580Eh	SCIFA2	FIFO data count register	FDR	16	16
A006 5810h	SCIFA2	Serial port register	SPTR	16	16
A006 5812h	SCIFA2	Line status register	LSR	16	16
A006 5814h	SCIFA2	Serial extended mode register	SEMR	8	8
A006 5816h	SCIFA2	FIFO trigger control register	FTCR	16	16
A006 6000h	SCIFA4	Serial mode register	SMR	16	16
A006 6002h	SCIFA4	Bit rate register	BRR	8	8
A006 6002h	SCIFA4	Modulation duty register	MDDR	8	8
A006 6004h	SCIFA4	Serial control register	SCR	16	16
A006 6006h	SCIFA4	Transmit FIFO data register	FTDR	8	8
A006 6008h	SCIFA4	Serial status register	FSR	16	16
A006 600Ah	SCIFA4	Receive FIFO data register	FRDR	8	8
A006 600Ch	SCIFA4	FIFO control register	FCR	16	16
A006 600Eh	SCIFA4	FIFO data count register	FDR	16	16
A006 6010h	SCIFA4	Serial port register	SPTR	16	16
A006 6012h	SCIFA4	Line status register	LSR	16	16
A006 6014h	SCIFA4	Serial extended mode register	SEMR	8	8
A006 6016h	SCIFA4	FIFO trigger control register	FTCR	16	16
A006 8000h	RSPI0	RSPI control register	SPCR	8	8
A006 8001h	RSPI0	RSPI slave select polarity register	SSLP	8	8
A006 8002h	RSPI0	RSPI pin control register	SPPCR	8	8
A006 8003h	RSPI0	RSPI status register	SPSR	8	8
A006 8004h	RSPI0	RSPI data register	SPDR	32	16, 32
A006 8008h	RSPI0	RSPI sequence control register	SPSCR	8	8
A006 8009h	RSPI0	RSPI sequence status register	SPSSR	8	8
A006 800Ah	RSPI0	RSPI bit rate register	SPBR	8	8
A006 800Bh	RSPI0	RSPI data control register	SPDCR	8	8
A006 800Ch	RSPI0	RSPI clock delay register	SPCKD	8	8
A006 800Dh	RSPI0	RSPI slave select negation delay register	SSLND	8	8
A006 800Eh	RSPI0	RSPI next-access delay register	SPND	8	8
A006 800Fh	RSPI0	RSPI control register 2	SPCR2	8	8
A006 8010h	RSPI0	RSPI command register 0	SPCMD0	16	16
A006 8012h	RSPI0	RSPI command register 1	SPCMD1	16	16
A006 8014h	RSPI0	RSPI command register 2	SPCMD2	16	16
A006 8016h	RSPI0	RSPI command register 3	SPCMD3	16	16
A006 8018h	RSPI0	RSPI command register 4	SPCMD4	16	16
A006 801Ah	RSPI0	RSPI command register 5	SPCMD5	16	16
A006 801Ch	RSPI0	RSPI command register 6	SPCMD6	16	16
A006 801Eh	RSPI0	RSPI command register 7	SPCMD7	16	16
A006 8400h	RSPI1	RSPI control register	SPCR	8	8

Table 5.1 List of I/O Registers (Address Order) (34 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 8401h	RSPI1	RSPI slave select polarity register	SSLP	8	8
A006 8402h	RSPI1	RSPI pin control register	SPPCR	8	8
A006 8403h	RSPI1	RSPI status register	SPSR	8	8
A006 8404h	RSPI1	RSPI data register	SPDR	32	16, 32
A006 8408h	RSPI1	RSPI sequence control register	SPSCR	8	8
A006 8409h	RSPI1	RSPI sequence status register	SPSSR	8	8
A006 840Ah	RSPI1	RSPI bit rate register	SPBR	8	8
A006 840Bh	RSPI1	RSPI data control register	SPDCR	8	8
A006 840Ch	RSPI1	RSPI clock delay register	SPCKD	8	8
A006 840Dh	RSPI1	RSPI slave select negation delay register	SSLND	8	8
A006 840Eh	RSPI1	RSPI next-access delay register	SPND	8	8
A006 840Fh	RSPI1	RSPI control register 2	SPCR2	8	8
A006 8410h	RSPI1	RSPI command register 0	SPCMD0	16	16
A006 8412h	RSPI1	RSPI command register 1	SPCMD1	16	16
A006 8414h	RSPI1	RSPI command register 2	SPCMD2	16	16
A006 8416h	RSPI1	RSPI command register 3	SPCMD3	16	16
A006 8418h	RSPI1	RSPI command register 4	SPCMD4	16	16
A006 841Ah	RSPI1	RSPI command register 5	SPCMD5	16	16
A006 841Ch	RSPI1	RSPI command register 6	SPCMD6	16	16
A006 841Eh	RSPI1	RSPI command register 7	SPCMD7	16	16
A007 C000h	CRC	CRC data input register	CRCDIR	32	32
A007 C004h	CRC	CRC data output register	CRCDOR	32	32
A007 C020h	CRC	CRC control register	CRCCR	8	8
A007 D008h	ECMM	ECM master error source status register 0	ECMMESSTR0	32	32
A007 D00Ch	ECMM	ECM master error source status register 1	ECMMESSTR1	32	32
A007 D010h	ECMM	ECM master error source status register 2	ECMMESSTR2	32	32
A007 D014h	ECMM	ECM master protection command register	ECMMPCMD0	32	32
A007 D048h	ECMC	ECM checker error source status register 0	ECMCESSTR0	32	32
A007 D04Ch	ECMC	ECM checker error source status register 1	ECMCESSTR1	32	32
A007 D050h	ECMC	ECM checker error source status register 2	ECMCESSTR2	32	32
A007 D054h	ECMC	ECM checker protection command register	ECMCPCMD0	32	32
A007 D084h	ECM	ECM maskable interrupt configuration register 0	ECMMICFG0	32	32
A007 D088h	ECM	ECM maskable interrupt configuration register 1	ECMMICFG1	32	32
A007 D08Ch	ECM	ECM maskable interrupt configuration register 2	ECMMICFG2	32	32
A007 D090h	ECM	ECM non-maskable interrupt configuration register 0	ECMNMICFG0	32	32
A007 D094h	ECM	ECM non-maskable interrupt configuration register 1	ECMNMICFG1	32	32
A007 D098h	ECM	ECM non-maskable interrupt configuration register 2	ECMNMICFG2	32	32
A007 D09Ch	ECM	ECM internal reset configuration register 0	ECMIRCFG0	32	32
A007 D0A0h	ECM	ECM internal reset configuration register 1	ECMIRCFG1	32	32
A007 D0A4h	ECM	ECM internal reset configuration register 2	ECMIRCFG2	32	32
A007 D0B4h	ECM	ECM Error Source Status Clear Trigger Register 0	ECMESSTC0	32	32
A007 D0B8h	ECM	ECM Error Source Status Clear Trigger Register 1	ECMESSTC1	32	32
A007 D0BCh	ECM	ECM Error Source Status Clear Trigger Register 2	ECMESSTC2	32	32
A007 D0C0h	ECM	ECM protection command register	ECMPCMD1	32	32
A007 D0C4h	ECM	ECM protection status register	ECMPS	8	8
A007 D0C8h	ECM	ECM pseudo error trigger register 0	ECMPE0	32	32
A007 D0CCh	ECM	ECM pseudo error trigger register 1	ECMPE1	32	32
A007 D0D0h	ECM	ECM pseudo error trigger register 2	ECMPE2	32	32

Table 5.1 List of I/O Registers (Address Order) (35 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A007 D0D4h	ECM	ECM delay timer control register	ECMDTMCTL	8	8
A007 D0D8h	ECM	ECM delay timer register	ECMDTMR	16	16
A007 D0DCh	ECM	ECM delay timer compare register	ECMDTMCMP	32	32
A007 D0E0h	ECM	ECM delay timer configuration register 0	ECMDTMCFG0	32	32
A007 D0E4h	ECM	ECM delay timer configuration register 1	ECMDTMCFG1	32	32
A007 D0E8h	ECM	ECM delay timer configuration register 2	ECMDTMCFG2	32	32
A007 D0ECh	ECM	ECM delay timer configuration register 3	ECMDTMCFG3	32	32
A007 D0F0h	ECM	ECM delay timer configuration register 4	ECMDTMCFG4	32	32
A007 D0F4h	ECM	ECM delay timer configuration register 5	ECMDTMCFG5	32	32
A008 0000h	CMT	Compare match timer start register 0	CMSTR0	16	16
A008 0002h	CMT0	Compare match timer control register	CMCR	16	16
A008 0004h	CMT0	Compare match timer counter	CMCNT	16	16
A008 0006h	CMT0	Compare match timer constant register	CMCOR	16	16
A008 0008h	CMT1	Compare match timer control register	CMCR	16	16
A008 000Ah	CMT1	Compare match timer counter	CMCNT	16	16
A008 000Ch	CMT1	Compare match timer constant register	CMCOR	16	16
A008 0020h	CMT	Compare match timer start register 1	CMSTR1	16	16
A008 0022h	CMT2	Compare match timer control register	CMCR	16	16
A008 0024h	CMT2	Compare match timer counter	CMCNT	16	16
A008 0026h	CMT2	Compare match timer constant register	CMCOR	16	16
A008 0028h	CMT3	Compare match timer control register	CMCR	16	16
A008 002Ah	CMT3	Compare match timer counter	CMCNT	16	16
A008 002Ch	CMT3	Compare match timer constant register	CMCOR	16	16
A008 0100h	TPUA	Timer start register	TSTRA	8	8
A008 0101h	TPUA	Timer synchronous register	TSYRA	8	8
A008 0108h	TPU0	Noise filter control register	NFCR	8	8
A008 0109h	TPU1	Noise filter control register	NFCR	8	8
A008 010Ah	TPU2	Noise filter control register	NFCR	8	8
A008 010Bh	TPU3	Noise filter control register	NFCR	8	8
A008 010Ch	TPU4	Noise filter control register	NFCR	8	8
A008 010Dh	TPU5	Noise filter control register	NFCR	8	8
A008 0110h	TPU0	Timer control register	TCR	8	8
A008 0111h	TPU0	Timer mode register	TMDR	8	8
A008 0112h	TPU0	Timer I/O control register	TIORH	8	8
A008 0113h	TPU0	Timer I/O control register	TIORL	8	8
A008 0114h	TPU0	Timer interrupt enable register	TIER	8	8
A008 0115h	TPU0	Timer status register	TSR	8	8
A008 0116h	TPU0	Timer counter	TCNT	16	16
A008 0118h	TPU0	Timer general register A	TGRA	16	16
A008 011Ah	TPU0	Timer general register B	TGRB	16	16
A008 011Ch	TPU0	Timer general register C	TGRC	16	16
A008 011Eh	TPU0	Timer general register D	TGRD	16	16
A008 0120h	TPU1	Timer control register	TCR	8	8
A008 0121h	TPU1	Timer mode register	TMDR	8	8
A008 0122h	TPU1	Timer I/O control register	TIOR	8	8
A008 0124h	TPU1	Timer interrupt enable register	TIER	8	8
A008 0125h	TPU1	Timer status register	TSR	8	8
A008 0126h	TPU1	Timer counter	TCNT	16	16

Table 5.1 List of I/O Registers (Address Order) (36 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A008 0128h	TPU1	Timer general register A	TGRA	16	16
A008 012Ah	TPU1	Timer general register B	TGRB	16	16
A008 0130h	TPU2	Timer control register	TCR	8	8
A008 0131h	TPU2	Timer mode register	TMDR	8	8
A008 0132h	TPU2	Timer I/O control register	TIOR	8	8
A008 0134h	TPU2	Timer interrupt enable register	TIER	8	8
A008 0135h	TPU2	Timer status register	TSR	8	8
A008 0136h	TPU2	Timer counter	TCNT	16	16
A008 0138h	TPU2	Timer general register A	TGRA	16	16
A008 013Ah	TPU2	Timer general register B	TGRB	16	16
A008 0140h	TPU3	Timer control register	TCR	8	8
A008 0141h	TPU3	Timer mode register	TMDR	8	8
A008 0142h	TPU3	Timer I/O control register	TIORH	8	8
A008 0143h	TPU3	Timer I/O control register	TIORL	8	8
A008 0144h	TPU3	Timer interrupt enable register	TIER	8	8
A008 0145h	TPU3	Timer status register	TSR	8	8
A008 0146h	TPU3	Timer counter	TCNT	16	16
A008 0148h	TPU3	Timer general register A	TGRA	16	16
A008 014Ah	TPU3	Timer general register B	TGRB	16	16
A008 014Ch	TPU3	Timer general register C	TGRC	16	16
A008 014Eh	TPU3	Timer general register D	TGRD	16	16
A008 0150h	TPU4	Timer control register	TCR	8	8
A008 0151h	TPU4	Timer mode register	TMDR	8	8
A008 0152h	TPU4	Timer I/O control register	TIOR	8	8
A008 0154h	TPU4	Timer interrupt enable register	TIER	8	8
A008 0155h	TPU4	Timer status register	TSR	8	8
A008 0156h	TPU4	Timer counter	TCNT	16	16
A008 0158h	TPU4	Timer general register A	TGRA	16	16
A008 015Ah	TPU4	Timer general register B	TGRB	16	16
A008 0160h	TPU5	Timer control register	TCR	8	8
A008 0161h	TPU5	Timer mode register	TMDR	8	8
A008 0162h	TPU5	Timer I/O control register	TIOR	8	8
A008 0164h	TPU5	Timer interrupt enable register	TIER	8	8
A008 0165h	TPU5	Timer status register	TSR	8	8
A008 0166h	TPU5	Timer counter	TCNT	16	16
A008 0168h	TPU5	Timer general register A	TGRA	16	16
A008 016Ah	TPU5	Timer general register B	TGRB	16	16
A008 0300h	CMTW0	Timer start register	CMWSTR	16	16
A008 0304h	CMTW0	Timer control register	CMWCR	16	16
A008 0308h	CMTW0	Timer I/O control register	CMWIOR	16	16
A008 0310h	CMTW0	Timer counter	CMWCNT	32	32
A008 0314h	CMTW0	Compare match constant register	CMWCOR	32	32
A008 0318h	CMTW0	Input capture register 0	CMWICR0	32	32
A008 031Ch	CMTW0	Input capture register 1	CMWICR1	32	32
A008 0320h	CMTW0	Output compare register 0	CMWOCR0	32	32
A008 0324h	CMTW0	Output compare register 1	CMWOCR1	32	32
A008 0380h	CMTW1	Timer start register	CMWSTR	16	16
A008 0384h	CMTW1	Timer control register	CMWCR	16	16

Table 5.1 List of I/O Registers (Address Order) (37 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A008 0388h	CMTW1	Timer I/O control register	CMWIOR	16	16
A008 0390h	CMTW1	Timer counter	CMWCNT	32	32
A008 0394h	CMTW1	Compare match constant register	CMWCOR	32	32
A008 0398h	CMTW1	Input capture register 0	CMWICR0	32	32
A008 039Ch	CMTW1	Input capture register 1	CMWICR1	32	32
A008 03A0h	CMTW1	Output compare register 0	CMWOCR0	32	32
A008 03A4h	CMTW1	Output compare register 1	CMWOCR1	32	32
A008 0400h	CMTW	Digital noise filter control register 0	NFCR0	32	32
A008 0404h	CMTW	Digital noise filter control register 1	NFCR1	32	32
A008 0600h	WDT0	WDT refresh register	WDTRR	8	8
A008 0602h	WDT0	WDT control register	WDTCR	16	16
A008 0604h	WDT0	WDT status register	WDTSR	16	16
A008 0606h	WDT0	WDT reset control register	WDTRCR	8	8
A008 0700h	IWDT	IWDT refresh register	IWDTRR	8	8
A008 0702h	IWDT	IWDT control register	IWDTCR	16	16
A008 0704h	IWDT	IWDT status register	IWDTSR	16	16
A008 0706h	IWDT	IWDT reset control register	IWDTRCR	8	8
A008 0900h	RIIC0	I2C bus control register 1	ICCR1	8	8
A008 0901h	RIIC0	I2C bus control register 2	ICCR2	8	8
A008 0902h	RIIC0	I2C bus mode register 1	ICMR1	8	8
A008 0903h	RIIC0	I2C bus mode register 2	ICMR2	8	8
A008 0904h	RIIC0	I2C bus mode register 3	ICMR3	8	8
A008 0905h	RIIC0	I2C bus function enable register	ICFER	8	8
A008 0906h	RIIC0	I2C bus status enable register	ICSER	8	8
A008 0907h	RIIC0	I2C bus interrupt enable register	ICIER	8	8
A008 0908h	RIIC0	I2C bus status register 1	ICSR1	8	8
A008 0909h	RIIC0	I2C bus status register 2	ICSR2	8	8
A008 090Ah	RIIC0	Slave address register L0	ICSARL0	8	8
A008 090Bh	RIIC0	Slave address register U0	ICSARU0	8	8
A008 090Ch	RIIC0	Slave address register L1	ICSARL1	8	8
A008 090Dh	RIIC0	Slave address register U1	ICSARU1	8	8
A008 090Eh	RIIC0	Slave address register L2	ICSARL2	8	8
A008 090Fh	RIIC0	Slave address register U2	ICSARU2	8	8
A008 0910h	RIIC0	I2C bus bitrate low register	ICBRL	8	8
A008 0911h	RIIC0	I2C bus bitrate high register	ICBRH	8	8
A008 0912h	RIIC0	I2C bus transmit data register	ICDRT	8	8
A008 0913h	RIIC0	I2C bus receive data register	ICDRR	8	8
A008 0940h	RIIC1	I2C bus control register 1	ICCR1	8	8
A008 0941h	RIIC1	I2C bus control register 2	ICCR2	8	8
A008 0942h	RIIC1	I2C bus mode register 1	ICMR1	8	8
A008 0943h	RIIC1	I2C bus mode register 2	ICMR2	8	8
A008 0944h	RIIC1	I2C bus mode register 3	ICMR3	8	8
A008 0945h	RIIC1	I2C bus function enable register	ICFER	8	8
A008 0946h	RIIC1	I2C bus status enable register	ICSER	8	8
A008 0947h	RIIC1	I2C bus interrupt enable register	ICIER	8	8
A008 0948h	RIIC1	I2C bus status register 1	ICSR1	8	8
A008 0949h	RIIC1	I2C bus status register 2	ICSR2	8	8
A008 094Ah	RIIC1	Slave address register L0	ICSARL0	8	8

Table 5.1 List of I/O Registers (Address Order) (38 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A008 094Bh	RIIC1	Slave address register U0	ICSARU0	8	8
A008 094Ch	RIIC1	Slave address register L1	ICSARL1	8	8
A008 094Dh	RIIC1	Slave address register U1	ICSARU1	8	8
A008 094Eh	RIIC1	Slave address register L2	ICSARL2	8	8
A008 094Fh	RIIC1	Slave address register U2	ICSARU2	8	8
A008 0950h	RIIC1	I^2C bus bitrate low register	ICBRL	8	8
A008 0951h	RIIC1	I^2C bus bitrate high register	ICBRH	8	8
A008 0952h	RIIC1	I^2C bus transmit data register	ICDRT	8	8
A008 0953h	RIIC1	I^2C bus receive data register	ICDRR	8	8
A008 0A00h	TSN	Temperature sensor control register	TSCR	8	8
A008 0B00h	ELC	Event link control register	ELCR	8	8
A008 0B08h	ELC	Event link setting register 7	ELSR7	8	8
A008 0B10h	ELC	Event link setting register 15	ELSR15	8	8
A008 0B13h	ELC	Event link setting register 18	ELSR18	8	8
A008 0B14h	ELC	Event link setting register 19	ELSR19	8	8
A008 0B15h	ELC	Event link setting register 20	ELSR20	8	8
A008 0B16h	ELC	Event link setting register 21	ELSR21	8	8
A008 0B17h	ELC	Event link setting register 22	ELSR22	8	8
A008 0B18h	ELC	Event link setting register 23	ELSR23	8	8
A008 0B19h	ELC	Event link setting register 24	ELSR24	8	8
A008 0B1Ah	ELC	Event link setting register 25	ELSR25	8	8
A008 0B1Bh	ELC	Event link setting register 26	ELSR26	8	8
A008 0B1Ch	ELC	Event link setting register 27	ELSR27	8	8
A008 0B21h	ELC	Event link option setting register C	ELOPC	8	8
A008 0B24h	ELC	Port group setting register 2	PGR2	8	8
A008 0B26h	ELC	Port group control register 2	PGC2	8	8
A008 0B28h	ELC	Port buffer register 2	PDBF2	8	8
A008 0B29h	ELC	Event link port setting register 0	PEL0	8	8
A008 0B2Ah	ELC	Event link port setting register 1	PEL1	8	8
A008 0B2Bh	ELC	Event link port setting register 2	PEL2	8	8
A008 0B2Ch	ELC	Event link port setting register 3	PEL3	8	8
A008 0B2Dh	ELC	Event link software event generation register	ELSEGR	8	8
A008 0B31h	ELC	Event link setting register 33	ELSR33	8	8
A008 0B33h	ELC	Event link setting register 35	ELSR35	8	8
A008 0B34h	ELC	Event link setting register 36	ELSR36	8	8
A008 0B35h	ELC	Event link setting register 37	ELSR37	8	8
A008 0B36h	ELC	Event link setting register 38	ELSR38	8	8
A008 0B3Dh	ELC	Event link setting register 45	ELSR45	8	8
A008 0B3Fh	ELC	Event link option setting register F	ELOPF	8	8
A008 0B41h	ELC	Event link option setting register H	ELOPH	8	8
A008 1200h	DOC	DOC control register	DOCR	8	8
A008 1202h	DOC	DOC data input register	DODIR	16	16
A008 1204h	DOC	DOC data setting register	DODSR	16	16
A008 C000h	S12ADC0	A/D control register	ADCSR	16	16
A008 C004h	S12ADC0	A/D channel select register A	ADANSA	16	16
A008 C008h	S12ADC0	A/D conversion value addition/average mode select register	ADADS	16	16
A008 C00Ch	S12ADC0	A/D conversion value addition/average number select register	ADADC	8	8
A008 C00Eh	S12ADC0	A/D control extended register	ADCER	16	16

Table 5.1 List of I/O Registers (Address Order) (39 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A008 C010h	S12ADC0	A/D start trigger select register	ADSTRGR	16	16
A008 C014h	S12ADC0	A/D channel select register B	ADANSB	16	16
A008 C018h	S12ADC0	A/D data duplicate register	ADDBLDR	16	16
A008 C01Ah	S12ADC0	A/D temperature sensor data register	ADTSDR	16	16
A008 C01Eh	S12ADC0	A/D self-diagnostic data register	ADRD	16	16
A008 C020h	S12ADC0	A/D data register 0	ADDR0	16	16
A008 C022h	S12ADC0	A/D data register 1	ADDR1	16	16
A008 C024h	S12ADC0	A/D data register 2	ADDR2	16	16
A008 C026h	S12ADC0	A/D data register 3	ADDR3	16	16
A008 C028h	S12ADC0	A/D data register 4	ADDR4	16	16
A008 C02Ah	S12ADC0	A/D data register 5	ADDR5	16	16
A008 C02Ch	S12ADC0	A/D data register 6	ADDR6	16	16
A008 C02Eh	S12ADC0	A/D data register 7	ADDR7	16	16
A008 C060h	S12ADC0	A/D sampling state register 0	ADSSTR0	8	8
A008 C066h	S12ADC0	A/D sampling state register control register	ADSHCR	16	16
A008 C070h	S12ADC0	A/D sampling state register T	ADSSTRT	8	8
A008 C073h	S12ADC0	A/D sampling state register 1	ADSSTR1	8	8
A008 C074h	S12ADC0	A/D sampling state register 2	ADSSTR2	8	8
A008 C075h	S12ADC0	A/D sampling state register 3	ADSSTR3	8	8
A008 C076h	S12ADC0	A/D sampling state register 4	ADSSTR4	8	8
A008 C077h	S12ADC0	A/D sampling state register 5	ADSSTR5	8	8
A008 C078h	S12ADC0	A/D sampling state register 6	ADSSTR6	8	8
A008 C079h	S12ADC0	A/D sampling state register 7	ADSSTR7	8	8
A008 C07Ah	S12ADC0	A/D disconnection detect control register	ADDISCR	8	8
A008 C080h	S12ADC0	A/D group scan priority control register	ADGSPCR	16	16
A008 C090h	S12ADC0	A/D compare control register	ADCMPCR	8	8
A008 C092h	S12ADC0	A/D compare channel select extended register	ADCMPANSER	8	8
A008 C093h	S12ADC0	A/D compare level extended select register	ADCMPLER	8	8
A008 C094h	S12ADC0	A/D compare channel select register	ADCMPANSR	16	16
A008 C098h	S12ADC0	A/D compare level register	ADCMPLR	16	16
A008 C09Ch	S12ADC0	A/D compare data register 0	ADCMPDR0	16	16
A008 C09Eh	S12ADC0	A/D compare data register 1	ADCMPDR1	16	16
A008 C0A0h	S12ADC0	A/D compare status register	ADCMPSR	16	16
A008 C0A4h	S12ADC0	A/D compare status extended register	ADCMPSER	8	8
A008 C0C8h	S12ADC0	A/D pin level self-diagnosis control register	ADTDCR	8	8
A008 C0CAh	S12ADC0	A/D error control register	ADERCR	8	8
A008 C0CBh	S12ADC0	A/D error clear register	ADERCLR	8	8
A008 C0D2h	S12ADC0	A/D overwrite error register	ADOWER	16	16
A008 C0D6h	S12ADC0	A/D overwrite error extended register	ADOWEER	16	16
A008 C400h	S12ADC1	A/D control register	ADCSR	16	16
A008 C404h	S12ADC1	A/D channel select register A	ADANSA	16	16
A008 C408h	S12ADC1	A/D conversion value addition/average mode select register	ADADS	16	16
A008 C40Ch	S12ADC1	A/D conversion value addition/average number select register	ADADC	8	8
A008 C40Eh	S12ADC1	A/D control extended register	ADCER	16	16
A008 C410h	S12ADC1	A/D start trigger select register	ADSTRGR	16	16
A008 C412h	S12ADC1	A/D conversion extended input control register	ADEXICR	16	16
A008 C414h	S12ADC1	A/D channel select register B	ADANSB	16	16
A008 C418h	S12ADC1	A/D data duplicate register	ADDBLDR	16	16

Table 5.1 List of I/O Registers (Address Order) (40 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A008 C41Eh	S12ADC1	A/D self-diagnostic data register	ADRD	16	16
A008 C420h	S12ADC1	A/D data register 0	ADDR0	16	16
A008 C422h	S12ADC1	A/D data register 1	ADDR1	16	16
A008 C424h	S12ADC1	A/D data register 2	ADDR2	16	16
A008 C426h	S12ADC1	A/D data register 3	ADDR3	16	16
A008 C428h	S12ADC1	A/D data register 4	ADDR4	16	16
A008 C42Ah	S12ADC1	A/D data register 5	ADDR5	16	16
A008 C42Ch	S12ADC1	A/D data register 6	ADDR6	16	16
A008 C42Eh	S12ADC1	A/D data register 7	ADDR7	16	16
A008 C460h	S12ADC1	A/D sampling state register 0	ADSSTR0	8	8
A008 C473h	S12ADC1	A/D sampling state register 1	ADSSTR1	8	8
A008 C474h	S12ADC1	A/D sampling state register 2	ADSSTR2	8	8
A008 C475h	S12ADC1	A/D sampling state register 3	ADSSTR3	8	8
A008 C476h	S12ADC1	A/D sampling state register 4	ADSSTR4	8	8
A008 C477h	S12ADC1	A/D sampling state register 5	ADSSTR5	8	8
A008 C478h	S12ADC1	A/D sampling state register 6	ADSSTR6	8	8
A008 C479h	S12ADC1	A/D sampling state register 7	ADSSTR7	8	8
A008 C47Ah	S12ADC1	A/D disconnection detect control register	ADDISCR	8	8
A008 C480h	S12ADC1	A/D group scan priority control register	ADGSPCR	16	16
A008 C490h	S12ADC1	A/D compare control register	ADCMPCR	8	8
A008 C494h	S12ADC1	A/D compare channel select register	ADCMPANSR	16	16
A008 C498h	S12ADC1	A/D compare level register	ADCMPLR	16	16
A008 C49Ch	S12ADC1	A/D compare data register 0	ADCMPDR0	16	16
A008 C49Eh	S12ADC1	A/D compare data register 1	ADCMPDR1	16	16
A008 C4A0h	S12ADC1	A/D compare status register	ADCMPSR	16	16
A008 C4C8h	S12ADC1	A/D pin level self-diagnosis control register	ADTDCR	8	8
A008 C4CAh	S12ADC1	A/D error control register	ADERCR	8	8
A008 C4CBh	S12ADC1	A/D error clear register	ADERCLR	8	8
A008 C4D2h	S12ADC1	A/D overwrite error register	ADOWER	16	16
A008 C4D6h	S12ADC1	A/D overwrite error extended register	ADOWEER	16	16
A009 0000h	CLMA0	CLMA0 control register 0	CLMA0CTL0	8	8
A009 0008h	CLMA0	CLMA0 compare register L	CLMA0CMPL	16	16
A009 000Ch	CLMA0	CLMA0 compare register H	CLMA0CMPH	16	16
A009 0010h	CLMA0	CLMA0 command register	CLMA0PCMD	8	8
A009 0014h	CLMA0	CLMA0 protection status register	CLMA0PS	8	8
A009 0020h	CLMA1	CLMA1 control register 0	CLMA1CTL0	8	8
A009 0028h	CLMA1	CLMA1 compare register L	CLMA1CMPL	16	16
A009 002Ch	CLMA1	CLMA1 compare register H	CLMA1CMPH	16	16
A009 0030h	CLMA1	CLMA1 command register	CLMA1PCMD	8	8
A009 0034h	CLMA1	CLMA1 protection status register	CLMA1PS	8	8
A009 0040h	CLMA2	CLMA2 control register 0	CLMA2CTL0	8	8
A009 0048h	CLMA2	CLMA2 compare register L	CLMA2CMPL	16	16
A009 004Ch	CLMA2	CLMA2 compare register H	CLMA2CMPH	16	16
A009 0050h	CLMA2	CLMA2 command register	CLMA2PCMD	8	8
A009 0054h	CLMA2	CLMA2 protection status register	CLMA2PS	8	8
A009 4000h	DMA0	DMAC unit 0 source select register 0	DMA0SEL0	32	32
A009 4004h	DMA0	DMAC unit 0 source select register 1	DMA0SEL1	32	32
A009 4008h	DMA0	DMAC unit 0 source select register 2	DMA0SEL2	32	32

Table 5.1 List of I/O Registers (Address Order) (41 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A009 400Ch	DMA0	DMAC unit 0 source select register 3	DMA0SEL3	32	32
A009 4010h	DMA0	DMAC unit 0 source select register 4	DMA0SEL4	32	32
A009 4014h	DMA0	DMAC unit 0 source select register 5	DMA0SEL5	32	32
A009 4018h	DMA0	DMAC unit 0 source select register 6	DMA0SEL6	32	32
A009 401Ch	DMA0	DMAC unit 0 source select register 7	DMA0SEL7	32	32
A009 4020h	DMA0	DMAC unit 0 source select register 8	DMA0SEL8	32	32
A009 4024h	DMA0	DMAC unit 0 source select register 9	DMA0SEL9	32	32
A009 4028h	DMA0	DMAC unit 0 source select register 10	DMA0SEL10	32	32
A009 402Ch	DMA0	DMAC unit 0 source select register 11	DMA0SEL11	32	32
A009 4030h	DMA0	DMAC unit 0 source select register 12	DMA0SEL12	32	32
A009 4034h	DMA0	DMAC unit 0 source select register 13	DMA0SEL13	32	32
A009 4038h	DMA0	DMAC unit 0 source select register 14	DMA0SEL14	32	32
A009 403Ch	DMA0	DMAC unit 0 source select register 15	DMA0SEL15	32	32
A009 4040h	DMA1	DMAC unit 1 source select register 0	DMA1SEL0	32	32
A009 4044h	DMA1	DMAC unit 1 source select register 1	DMA1SEL1	32	32
A009 4048h	DMA1	DMAC unit 1 source select register 2	DMA1SEL2	32	32
A009 404Ch	DMA1	DMAC unit 1 source select register 3	DMA1SEL3	32	32
A009 4050h	DMA1	DMAC unit 1 source select register 4	DMA1SEL4	32	32
A009 4054h	DMA1	DMAC unit 1 source select register 5	DMA1SEL5	32	32
A009 4058h	DMA1	DMAC unit 1 source select register 6	DMA1SEL6	32	32
A009 405Ch	DMA1	DMAC unit 1 source select register 7	DMA1SEL7	32	32
A009 4060h	DMA1	DMAC unit 1 source select register 8	DMA1SEL8	32	32
A009 4064h	DMA1	DMAC unit 1 source select register 9	DMA1SEL9	32	32
A009 4068h	DMA1	DMAC unit 1 source select register 10	DMA1SEL10	32	32
A009 406Ch	DMA1	DMAC unit 1 source select register 11	DMA1SEL11	32	32
A009 4070h	DMA1	DMAC unit 1 source select register 12	DMA1SEL12	32	32
A009 4074h	DMA1	DMAC unit 1 source select register 13	DMA1SEL13	32	32
A009 4078h	DMA1	DMAC unit 1 source select register 14	DMA1SEL14	32	32
A009 407Ch	DMA1	DMAC unit 1 source select register 15	DMA1SEL15	32	32
A009 4080h	DMAC	DMAC software start register	DMASTG	32	32
A009 4200h	ICU	IRQ control register 0	IRQCR0	32	32
A009 4204h	ICU	IRQ control register 1	IRQCR1	32	32
A009 4208h	ICU	IRQ control register 2	IRQCR2	32	32
A009 420Ch	ICU	IRQ control register 3	IRQCR3	32	32
A009 4210h	ICU	IRQ control register 4	IRQCR4	32	32
A009 4218h	ICU	IRQ control register 6	IRQCR6	32	32
A009 421Ch	ICU	IRQ control register 7	IRQCR7	32	32
A009 4240h	ICU	IRQ pin digital noise filter enable register	IRQFLTE	32	32
A009 4244h	ICU	IRQ pin digital noise filter setting register	IRQFLTC	32	32
A009 4248h	ICU	Non-maskable interrupt status register	NMISR	32	32
A009 424Ch	ICU	Non-maskable interrupt status clear register	NMICLR	32	32
A009 4250h	ICU	NMI pin interrupt control register	NMICR	32	32
A009 4254h	ICU	NMI pin digital noise filter enable register	NMIFLTE	32	32
A009 4258h	ICU	NMI pin digital noise filter setting register	NMIFLTC	32	32
A00B 0020h	SYSTEM	System clock control register	SCKCR	32	32
A00B 0024h	SYSTEM	System clock control register 2	SCKCR2	32	32
A00B 0034h	SYSTEM	PLL1 control register	PLL1CR	32	32
A00B 0038h	SYSTEM	PLL1 control register 2	PLL1CR2	32	32

Table 5.1 List of I/O Registers (Address Order) (42 / 42)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A00B 0040h	SYSTEM	Low-speed on-chip oscillator control register	LOCOCR	32	32
A00B 004Ch	SYSTEM	Oscillation stop detect control register	OSTDCR	32	32
A00B 0200h	SYSTEM	Reset status register 0	RSTSR0	32	32
A00B 0210h	SYSTEM	Software reset register	SWRR1	32	32
A00B 0300h	SYSTEM	Module stop control register A	MSTPCRA	32	32
A00B 0304h	SYSTEM	Module stop control register B	MSTPCRB	32	32
A00B 0308h	SYSTEM	Module stop control register C	MSTPCRC	32	32
A00B 0310h	SYSTEM	Module stop control register E	MSTPCRE	32	32
A00B 0314h	SYSTEM	Module stop control register F	MSTPCRF	32	32
A00B 0800h	SYSTEM	ATCM wait control register	SYTATCMWAIT	32	32
A00B 0A00h	SYSTEM	Debug interface control register	DBGIFCNT	32	32
A00B 0A60h	SYSTEM	Mode monitor register	MDMONR	32	32
A00B 0A80h	SYSTEM	ECM mask control register	ECMMCNT	32	32
A00B 0B00h	SYSTEM	Protect register	PRCR	32	32
A00F 3000h	ECCRAM	Protection command register	RAMPCMD	32	32
A00F 3100h	ECCRAM	ECC decoder configuration register	RAMEDC	32	32
A00F 3104h	ECCRAM	ECC encoder configuration register	RAMEEC	32	32
A00F 3108h	ECCRAM	2-bit ECC error status register	RAMDBEST	32	32
A00F 310Ch	ECCRAM	2-bit ECC error address register	RAMDBEAD	32	32
A00F 3110h	ECCRAM	2-bit ECC error counter register	RAMDBECNT	32	32
B011 C100h	MDIO	Transmission register	TX	16	16
B011 C102h	MDIO	Interrupt clear register	CL	8	8
B011 C104h	MDIO	Control register	CTL	8	8
B011 C105h	MDIO	Mode register	MODE	8	8
B011 C106h	MDIO	Interrupt enable register	INTE	8	8
B011 C108h	MDIO	Physical port address register	PADR	16	16
B011 C10Ah	MDIO	Device address register	DADR	8	8
B011 C10Ch	MDIO	Enable address register	ENADR	32	32
B051 C700h	MDIO	Reception register	RX	32	32
B051 C704h	MDIO	Reception register 2	RX2	16	16
B051 C70Eh	MDIO	Version register	VER	16	16
B0A3 0012h	MDIOM1*1	Bit rate register	BR	16	16
B0B1 C100h	MDIOM1*1	Transmission register	TX	32	32
B0B1 C500h	MDIOM1*1	Version register	VER	16	16
B0B1 CD04h	MDIOM1*1	Reception register	RX	32	32

Note 1. The MDIO master is optional. For the products which have it, see Table 1.3, List of Products.

6. Reset

6.1 Overview

Available reset types are RES# pin reset, error control module (ECM) reset, and software reset.
Table 6.1 lists the reset names and sources.

Table 6.1 Reset Names and Sources

Reset Name	Source
RES# pin reset	Voltage input to the RES# pin is low.
ECM reset	Reset request from the error control module (ECM)
Software reset	SWRR1 register setting

The internal states and pins are initialized by reset.
Table 6.2 lists the reset targets to be initialized for each reset type. For details on reset control during debugging, see section 10.3.5, Reset Configuration and the Method of Connecting with the Emulator.

Table 6.2 Targets to be Initialized for Each Reset Type (√: To be initialized, —: No change)

	Reset Source		
Reset Target	**RES# Pin Reset**	**ECM Reset**	**Software Reset**
RES# pin reset flag (RSTSR0.TRF)	—	√	√
ECM reset detect flag (RSTSR0.ECMRF)	√	—	√
Software reset detect flag (RSTSR0.SWRF1)	√	√	—
Pin state	√	√	√
Operation mode	√*1	—*2	—*2
ECM ECM master error source status registers 0 to 2 ECM checker error source status registers 0 to 2	√	—	—
Registers other than the above, and internal state	√	√	√
RSTOUT# pin output	√ (Low)*3	√ (Low)*3	√ (Low)*3

Note 1. An operation mode is selected according to the input level of the mode setting pin (MD1 or MD0) when the pin reset state (RES# and TRST# pins are both low) is released. For details on the operation mode, see section 3.2, Types and Selection of Operating Modes.
Note 2. The operating mode is not initialized in response to this type of reset and the chip will be in the operating mode that was selected following the previous release from the reset state applied by the low level on the RES#. For details, see section 3, Operating Modes.
Note 3. For the low output period, see section 6.3.5, Reset Output Pin (RSTOUT#).

Table 6.3 lists the input and output pins related to the reset.

Table 6.3 Input and Output Pins Related to Reset

Pin Name	I/O	Function
RES#	Input	Reset pin. Use this pin to reset the entire LSI except the debugging circuit and TAP (Test Access Port). Since the power-on reset circuit is not incorporated in RZ/T1-M, a reset circuit must be implemented outside this LSI. For an example of configuration of the external reset circuit, see section 10.3.5, Reset Configuration and the Method of Connecting with the Emulator.
TRST#	Input	Test reset pin. Use this pin to reset TAP. If you design a board which enables an emulator, set the TRST# pin to Low level during the same period as the RES# pin at the time of power-on. The TRST# pin should also be controllable independently. When unused, this pin must be set to low level or connected to the same signal as the RES# pin. For details, see section 10.3.5, Reset Configuration and the Method of Connecting with the Emulator.
RSTOUT#	Output	Reset output pin. This pin outputs low-level signal upon occurrence of reset. For details, see section 6.3.5, Reset Output Pin (RSTOUT#). This pin can be used for resetting the external device.

Note: For details on resetting the debugging circuit, see section 10, Debugging Interface.

6.2 Register Descriptions

The reset status register 0 contains bits assigned to respective reset types to indicate reset generation sources. RSTSR0 and SWRR1 are protected by the register write protection function. To write these registers, clear the write protection bit 1 in the protect register (PRCR). For details on the register write protection, see section 11, Register Write Protection Function.

6.2.1 Reset Status Register 0 (RSTSR0)

RSTSR0 is the register that indicates reset generation sources.

Address(es): RSTSR0: A00B 0200h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	SWR1F	ECMRF	TRF	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	*1	*1	*1	0

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/(W)
b1	TRF	RES# Pin Reset Detect Flag	0: RES# pin reset not detected 1: RES# pin reset detected [Setting condition] • When Low is input to the RES# pin. [Clearing conditions] • When a ECM reset or software reset occurs. • When “0000 0000h” is written to RSTSR0 after RSTSR0 is read.	R/(W)*2
b2	ECMRF	ECM Reset Detect Flag	0: ECM reset not detected 1: ECM reset detected [Setting condition] • When an error predefined as a reset source by ECM setting occurs. [Clearing conditions] • When an RES# pin reset or software reset occurs. • When “0000 0000h” is written to RSTSR0 after RSTSR0 is read.	R/(W)*2
b3	SWR1F	Software Reset Detect Flag	0: Software reset not detected. 1: Software reset detected. [Setting condition] • When a software reset occurs. [Clearing conditions] • When an RES# pin reset or ECM reset occurs. • When “0000 0000h” is written to RSTSR0 after RSTSR0 is read.	R/(W)*2
b31 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/(W)

Note 1. The value after reset depends on the reset source.
Note 2. Only 0 can be written to clear the flag.

6.2.2 Software Reset Register (SWRR1)

SWRR1 is a register that controls the software reset.

Address(es): SWRR1: A00B 0210h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	SWRR1[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SWRR1[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	SWRR1[31:0]	Software Reset	When “4321 A501h” is written, a software reset occurs. These bits are read as 0000 0000h.	R/W

6.3 Operation

6.3.1 RES# Pin Reset

This reset occurs when a signal arrives at the RES# pin from the externally connected reset circuit. When the signal becomes Low at the RES# pin, all the ongoing processes are aborted and the LSI enters the reset state. In order to reset the LSI without fail, the RES# pin should be held at Low level for the specified time after power-on. For details on the reset configuration, see section 10.3.5, Reset Configuration and the Method of Connecting with the Emulator.
After the reset is released, the CPU (Cortex-R4) starts reset exception handling.
When an RES# pin reset occurs, the RSTSR0.TRF flag is set to 1.

6.3.2 ECM Reset

This reset is generated in response to a reset request from ECM (Error Control Module).
ECM receives serious errors, such as oscillation stop detection, from individual modules in the LSI, and generates reset requests corresponding to respective errors. For details on the ECM operation, see section 29, Error Control Module (ECM). When an ECM reset occurs, all the ongoing processes are aborted and the LSI enters the reset state.
After the reset is released, the CPU (Cortex-R4) starts reset exception handling.
When an ECM reset occurs, the RSTSR0.ECMRF flag is set to 1.

6.3.3 Software Reset

The software reset occurs when “4321 A501h” is written to the SWRR1 register. When a software reset occurs, all the ongoing processes are aborted and the LSI enters the reset state.
After the reset is released, the CPU (Cortex-R4) starts reset exception handling.
When the software reset occurs, the RSTSR0.SWR1F flag is set to 1.

6.3.4 Determination of Reset Generation Source

Reading the RSTSR0 register determines which reset source was used for reset execution. Figure 6.1 shows an example of the flow to identify a reset generation source.

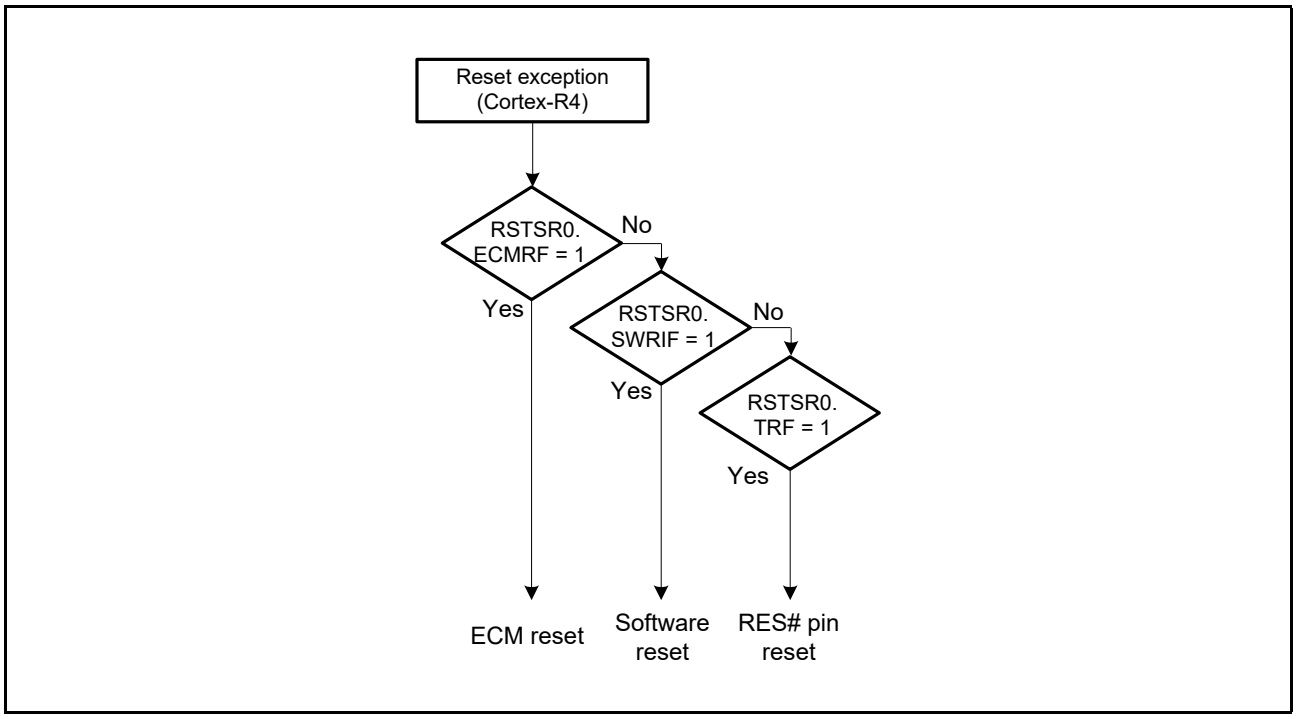


Figure 6.1 Example of Reset Generation Source Determination Flow

6.3.5 Reset Output Pin (RSTOUT#)

The reset output pin (RSTOUT#) outputs Low level upon occurrence of reset when the input level of the RES# pin is Low. It also outputs Low level upon occurrence of an ECM reset or software reset.
If the RES# pin remains Low for a specified time period and then changes to high, the reset output remains Low for 500 us (Typ.) and then changes to High. Similarly, when an ECM reset or software reset occurs, the reset output remains Low for 500 us (Typ.) and then changes to High.

6.3.6 Noise Cancellation for Reset Input

Noise cancellation using analog delay is applied to the RES# pin. This can eliminate noises within 100 ns (Min.).

6.4 Usage Note

6.4.1 Connection of Reset Output Pin (RSTOUT#)

The low level is output for a specified period of time on RSTOUT# after de-assertion of the signal on the RES# pin. This means that RSTOUT# should not be directly connected as a reset signal for a flash memory to be used in booting the LSI chip. Otherwise, release of the LSI chip itself from the reset state may precede that for the flash memory.
However, using RSTOUT# as the reset signal for external device is still possible as long as the timing requirement imposed by the output on the RSTOUT# pin is satisfied.
For details, see section 6.3.5, Reset Output Pin (RSTOUT#).

7. Clock Generation Circuit

7.1 Overview

This LSI incorporates a clock generation circuit.
Table 7.1 lists the specifications of the clock generation circuit. Figure 7.1 shows a block diagram of the clock generation circuit.

Table 7.1 Specifications of Clock Generation Circuit

Item	Specifications
Main clock oscillator	Resonator frequency: 25 MHz
	Connectable resonator or additional circuit: Ceramic resonator, crystal resonator Connection pin: EXTAL, XTAL
	Oscillation stop detection function: When an oscillation stop is detected with the main clock, the system clock source is switched to LOCO.
PLL0 circuit	Input clock source: Main clock oscillator
	Input frequency: 25 MHz
	Frequency multiplication ratio: 48 multiplication
	Output clock frequency of the PLL0 circuit: 1200 MHz
	Oscillation abnormality detection function: When the CLMA0 detects an abnormality in oscillation of the PLL0, the system clock source is switched to main clock.
PLL1 circuit	Input clock source: PLL0 clock divided by 80
	Input frequency: 15 MHz
	Frequency multiplication ratio: Selectable from 60 and 80
	Output clock frequency of the PLL1 circuit: 900 MHz, 1200 MHz
	Oscillation abnormality detection function: When the CLMA1 detects an abnormality in oscillation of the PLL1, the system clock source is switched to main clock.
Low-speed on-chip oscillator (LOCO)	Oscillation frequency: 240 kHz
	Oscillation abnormality detection function: The CLMA2 can detect an abnormality in oscillation of the LOCO.
External clock input (TCK) for JTAG	Input frequency: 50 MHz (max.)

Table 7.2 Specifications of Clock Generation Circuit (Internal Clock)

Item	Clock Source	Supplied to	Frequency
CPU clock (CPUCLK)	Selected from frequency-dividing clocks for PLL0 or PLL1	CPU (Cortex-R4)	150 MHz 300 MHz 450 MHz
System clock (ICLK)	Selected from frequency-dividing clocks for PLL0 or PLL1	DMAC, interrupt controller, on-chip extended SRAM	150 MHz
High-speed peripheral module clock (PCLKA)	Selected from frequency-dividing clocks for PLL0 or PLL1	Peripheral module	150 MHz
Low-speed peripheral module clock (PCLKB)	Selected from frequency-dividing clocks for PLL0 or PLL1	Peripheral module	75 MHz
Low-speed peripheral module clock (PCLKD)	Frequency-dividing clock for PLL0	Peripheral modules (CRC, DOC, ECM, ELC, TPU, CMT, CMTW, and RIIC)	75 MHz
Low-speed peripheral module clock (PCLKE)	Frequency-dividing clock for PLL0	Peripheral module (WDTA)	Up to 75 MHz
Low-speed peripheral module clock (PCLKF)	Frequency-dividing clock for PLL0	Peripheral module (12-bit A/D converter Unit 0)	Up to 60 MHz
Low-speed peripheral module clock (PCLKG)	Frequency-dividing clock for PLL0	Peripheral module (12-bit A/D converter Unit 1)	Up to 60 MHz
Low-speed peripheral module clock (PCLKH)	Frequency-dividing clock for PLL0	Peripheral module (12-bit A/D converter, bus clock for units 0 and 1)	60 MHz
High-speed serial clock (SERICLK)	Frequency-dividing clock for PLL0	RSPIa, SCIFA	150 MHz, 120 MHz
CLMAn sampling clock (CLMAMCLKA) (n = 1 or 0)	Main clock divided by 2	CLMA0, CLMA1	12.5 MHz
CLMA2 sampling clock (CLMAMCLKB)	Main clock divided by 256	CLMA2	97.6 kHz
CLMA2 monitor clock (CLMALCLK)	LOCO	CLMA2	240 kHz
CLMA0 monitor clock (CLMAPLCLK0)	PLL0 clock divided by 16	CLMA0	75 MHz
CLMA1 monitor clock (CLMAPLCLK1)	PLL1 clock divided by 16	CLMA1	75 MHz, 56.25 MHz*1
IWDT clock (IWDTCLK)	LOCO clock divided by 2	IWDT	120 kHz
ECM clock (ECMCKL)	LOCO	ECM	240 kHz
JTAG clock (JTAGTCK)	TCK	JTAG	Up to 50 MHz
Trace interface clock (TCLK)	Selected from frequency-dividing clocks for PLL0 or PLL1	CoreSight TPIU	75 MHz

Note 1. Selecting 10b in the PLL1CR.CPUCKSEL[1:0] bits outputs 56.25 MHz. Selecting values other than 10b outputs 75 MHz.

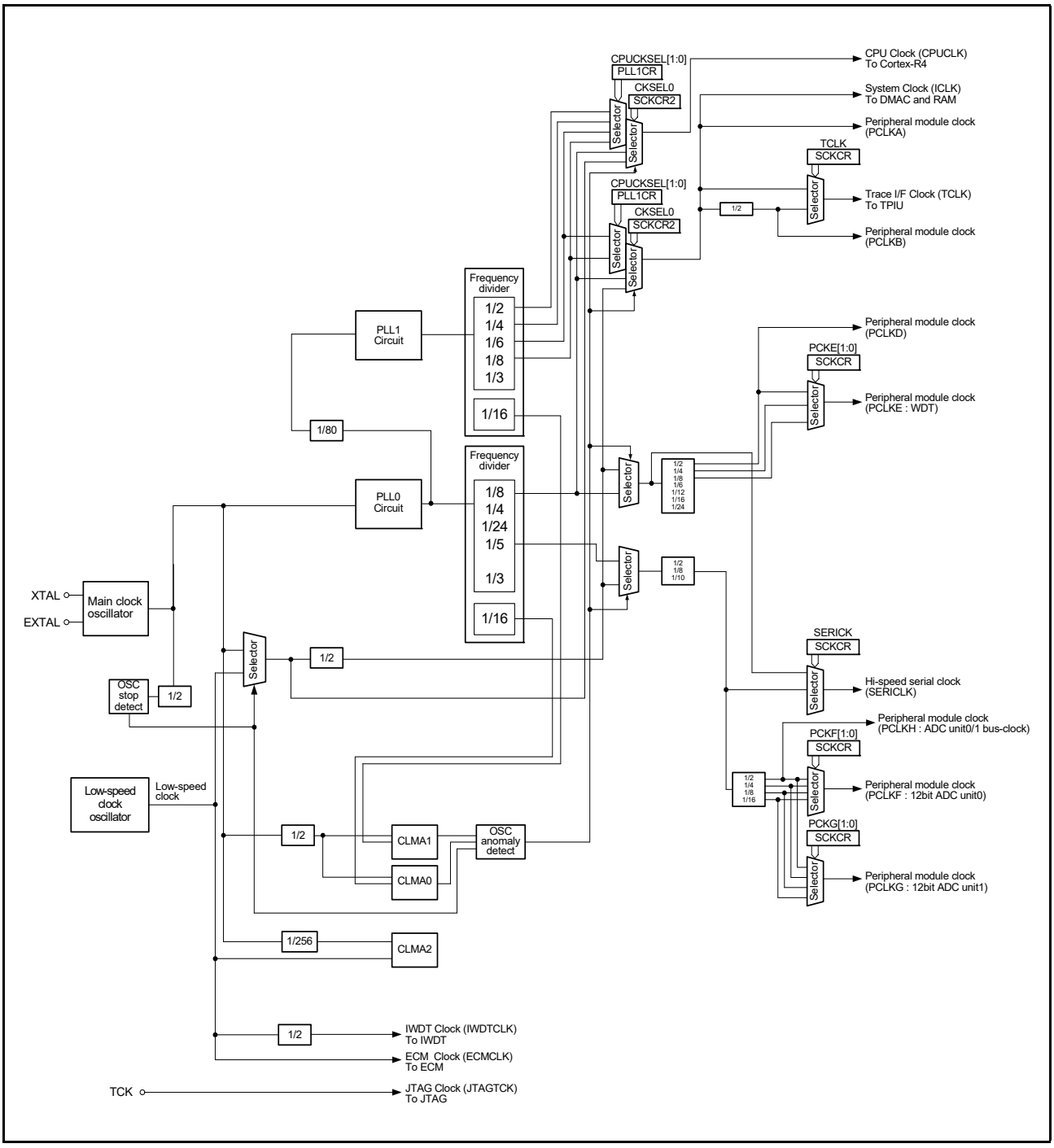


Figure 7.1 Block Diagram of Clock Generation Circuit

Table 7.3 shows the input/output pins of the clock generation circuit.

Table 7.3 Pin Configuration of Clock Generation Circuit

Pin Name	I/O	Description
XTAL	Output	These pins are used to connect a crystal resonator.
EXTAL	Input	
TCK	Input	This pin is used to input the clock for the JTAG.

7.2 Register Descriptions

The registers related to the clock generation circuit can be write-protected. To write to the registers, specify bit 0 of the Protect Register (PRCR) to cancel the write protection. For details, see section 11, Register Write Protection Function.

7.2.1 System Clock Control Register (SCKCR)

The SCKCR register is used to select the frequency for each of trace interface clock (TCLK), high-speed serial clock (SERICLK), and peripheral module clocks (PCLKE, PCLKF, and PCLKG).

Address(es): A00B 0020h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	TCLK	—	—	—	SERICK
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	PCKE[1:0]		PCKF[1:0]		PCKG[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	PCKG[1:0]	Peripheral Module Clock G (PCLKG) Select	Select the low-speed peripheral module clock PCLKG supplied to the 12-bit A/D converter unit 1. 00: 60 MHz 01: 30 MHz 10: 15 MHz 11: 7.5 MHz	R/W
b3, b2	PCKF[1:0]	Peripheral Module Clock F (PCLKF) Select	Select the low-speed peripheral module clock PCLKF supplied to the 12-bit A/D converter unit 0. 00: 60 MHz 01: 30 MHz 10: 15 MHz 11: 7.5 MHz	R/W
b5, b4	PCKE[1:0]	Peripheral Module Clock E (PCLKE) Select	Select the low-speed peripheral module clock PCLKE supplied to the WDTA. 00: 75 MHz 01: 37.5 MHz 10: 18.75 MHz Settings other than above are prohibited.	R/W
b15 to b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b16	SERICK	High-Speed Serial Clock (SERICLK) Select	Selects the high-speed serial clock SERICLK supplied to RSPIa and SCIFA. 0: 150 MHz 1: 120 MHz	R/W
b19 to b17	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b20	TCLK	Trace interface clock (TCLK)	Selects the clock supplied to the trace I/F clock TCLK (CoreSight TPIU). 0: Setting prohibited 1: 75 MHz	R/W
b31 to b21	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

7.2.2 System Clock Control Register 2 (SCKCR2)

The SCKCR2 register is used to select PLL0 or PLL1 as the source of clocks supplied to the system clock.

Address(es): A00B 0024h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CKSEL
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CKSEL0	System Clock Source Select	Selects the source of the CPU clock (CPUCLK), system clock (ICLK), high-speed peripheral module clock (PCLKA), and low-speed peripheral module clock (PCLKB). Switching to a clock source which is not in operation is prohibited. 0: PLL0 is selected. 1: PLL1 is selected.	R/W
b31 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

7.2.3 PLL1 Control Register (PLL1CR)

The PLL1CR register is used to set the CPU clock frequency.

Address(es): A00B 0034h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CPUCKSEL [1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	CPUCKSEL [1:0]	CPU Operating Frequency Select	Select the CPU clock frequency. After setting this bit to select the CPU operating frequency, use the PLL1CR2 register to control operation of the PLL1 circuit. For the procedure for changing the CPU frequency, see Figure 7.2. b1 b0 00: 150 MHz 01: 300 MHz 10: 450 MHz	R/W
b31 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Set the frequency within the allowable range for the electrical characteristics of the CPU frequency clock.
Note 2. The relationship between the PLL1 frequency multiplication ratio and clock division ratio specified with CPUCKSEL[1:0] is shown in Table 7.4.

Table 7.4 Relationship Between PLL1 Frequency Multiplication Ratio and Clock Division Ratio Specified with CPUCKSEL[1:0]

CPUCKSEL[1:0]	PLL1 Frequency Multiplication Ratio	Clock Division Ratio	CPU Operating Frequency
00	80	1/8	150 MHz
01	80	1/4	300 MHz
10	60	1/2	450 MHz

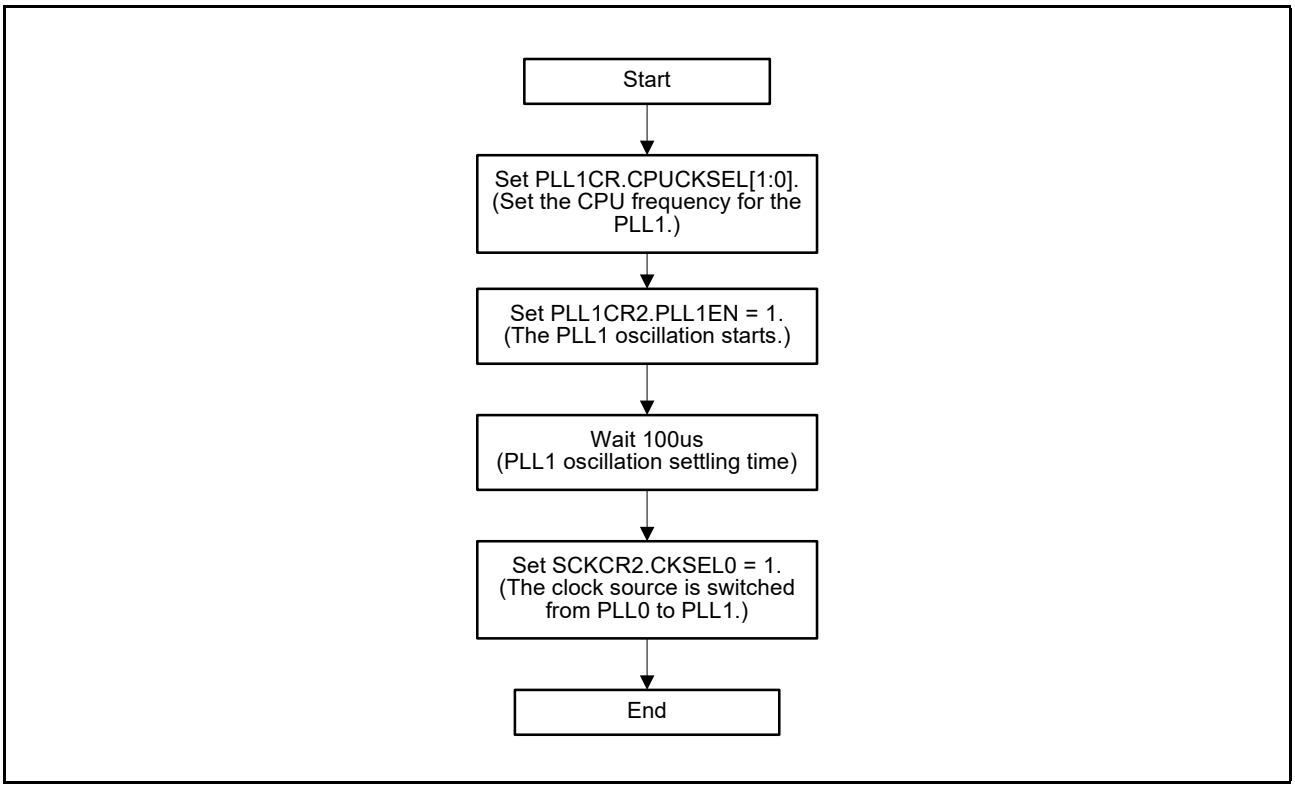


Figure 7.2 Changing the CPU Frequency

7.2.4 PLL1 Control Register 2 (PLL1CR2)

The PLL1CR2 register is used to control operation of the PLL1 circuit.

Address(es): A00B 0038h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PLL1EN
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	PLL1EN	PLL1 Operation Control	Specifies whether to run or stop PLL1. 0: PLL1 stops 1: PLL1 runs If the PLL1EN bit is set to 1, 100 us of the PLL oscillation settling time must be counted by using loop processing in the CPU or by a timer. For details about how to change the CPU frequency, see Figure 7.2.	R/W
b31 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note: Before you attempt to set the CPU clock frequency by using the PLL1CR register, stop the PLL by setting the PLL1EN bit to 0.

7.2.5 Low-Speed On-Chip Oscillator Control Register (LOCOCR)

LOCOCR is used to control operation of a low-speed on-chip oscillator.

Address(es): A00B 0040h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LCSTP
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b0	LCSTP	LOCO Stop	Specify whether to run or stop the low-speed on-chip oscillator (LOCO). 0: Run 1: Stop After setting this bit to run the LOCO, start using the LOCO clock after the LOCO oscillation stabilization time (t_{LOCOWT}) has elapsed.	R/W
b31 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. To stop the LOCO and then run it again, wait at least the LOCO oscillation stabilization time (t_{LOCOWT}) is elapsed after setting the LCSTP bit to stop the LOCO, and then set the LCSTP bit to run the LOCO again.
To set the bit to stop the LOCO, make sure that the LOCO oscillation is stable.

Note 2. Writing of 1 to the LCSTP bit (stop the LOCO) is prohibited if the oscillation stop detection function is enabled by setting the OSTDCR.OSTDE bit.

7.2.6 Oscillation Stop Detection Control Register (OSTDCR)

The OSTDCR register is used to control the oscillation stop detection function of the main clock.

Address(es): A00B 004Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	OSTDE	—	—	—	—	—	—	OSTDIE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	OSTDIE	Oscillation Stop Detection Interrupt Enable	Enable or disable the oscillation stop detection interrupt (OSTDI). 0: Oscillation stop detection interrupt is disabled. 1: Oscillation stop detection interrupt is enabled.	R/W
b6 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b7	OSTDE	Oscillation Stop Detection Function Enable	Enable or disable the oscillation stop detection function. If the oscillation stop detection function is enabled, the LCSTP bit of the low-speed on-chip oscillator control register (LOCOCR) is also set to 0, and the low-speed on-chip oscillator operation starts.*1 0: The oscillation stop detection function is disabled. 1: The oscillation stop detection function is enabled.	R/W
b31 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. The low-speed on-chip oscillator cannot be stopped while the oscillation stop detection function is enabled (OSTDE bit = 1). Writing 1 to the LOCOCR.LCSTP bit is ignored.

7.3 Input to Main Clock Oscillator

The clock signal is supplied to the main clock oscillator by connecting a resonator.

7.3.1 Connecting a Crystal Resonator

Figure 7.3 shows an example of connecting a crystal resonator.
A damping resistor Rd should be added, if necessary. Because the resistor values vary depending on the resonator and the oscillation drive capability, use values recommended by the resonator manufacturer. If use of an external feedback resistor (Rf) is directed by the resonator manufacturer, insert an Rf between EXTAL and XTAL by following the instruction.
When connecting a resonator to supply the clock, the frequency of the resonator should be in the frequency range of the resonator for the main clock oscillator described in Table 7.1.

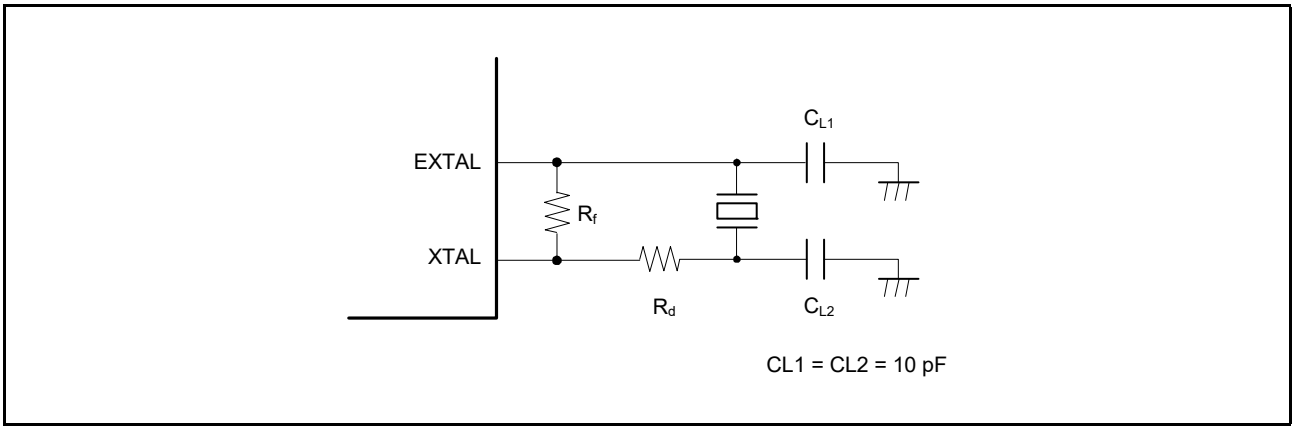


Figure 7.3 Example of Crystal Resonator Connection

Table 7.5 Damping Resistance (Reference Values)

Frequency (MHz)	25
Rd (Ω)	2.2K

Figure 7.4 shows an equivalent circuit of the crystal resonator. Use a crystal resonator that has the characteristics shown in Table 7.6.

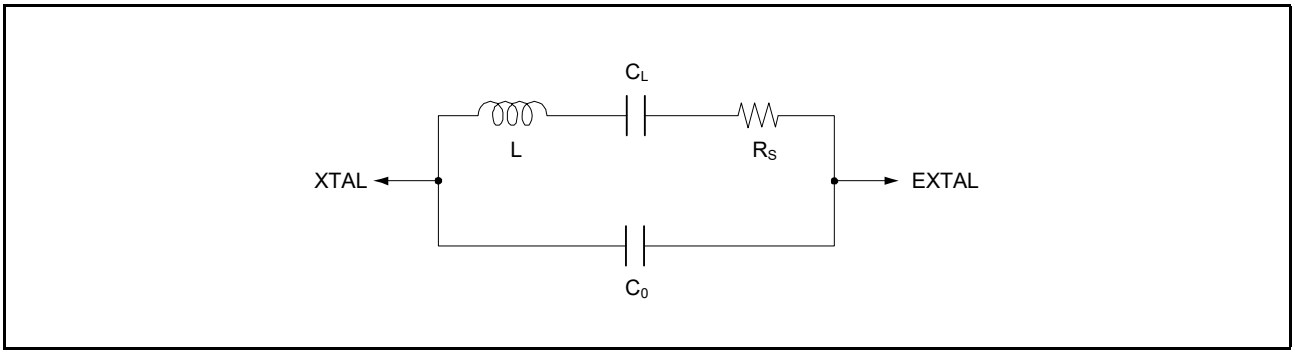


Figure 7.4 Equivalent Circuit of Crystal Resonator

Table 7.6 Crystal Resonator Characteristics (Reference Values)

Frequency (MHz)	25
R_S max (Ω)	100

7.4 Oscillation Stop Detection Function

7.4.1 Oscillation Stop Detection and Operation after Detection

The oscillation stop detection function detects the main clock oscillator stop and to supply LOCO clock pulses from the low-speed on-chip oscillator as the system clock source instead of the main clock, PLL0 clock, and PLL1 clock.
An oscillation stop detection interrupt request can be generated when an oscillation stop is detected.
This LSI detects a main clock oscillation stop when the input clock remains to be 0 or 1 for a certain period, for example, due to a malfunction of the main clock oscillator (see Table 34.27, Oscillation Stop Detection Circuit Characteristics, in section 34.7, Oscillation Stop Detection Timing).
When an oscillation stop is detected, clocks are switched to the LOCO clock.

After a reset is released, the oscillation stop detection function is disabled. To enable the oscillation stop detection function, set the OSTDCR.OSTDE bit to 1.
The clocks that are switched to the LOCO clock by the oscillation stop detection are the PLL0 clock and PLL1 clock.

7.4.2 Oscillation Stop Detection Interrupt

An oscillation-stop detection interrupt (OSTDI) will be generated if the oscillation-stop detection interrupt enable bit (OSTDCR.OSTDIE) is 1 (enabling interrupt generation on oscillation stop detection).
The oscillation stop detection interrupt is connected to the Error Control Module (ECM) as an error interrupt. Interrupts are disabled in the initial state after a reset release. To use oscillation stop detection interrupts, select maskable or non-maskable interrupt as the operation to be performed when the ECN detects an error interrupt. For details, see section 29, Error Control Module (ECM).

7.5 PLL Oscillation Abnormality Detection Function

The PLL oscillation abnormality detection function detects an abnormality in oscillation of the PLL0 or PLL1 by using the clock monitor circuits (CLMA0 and CLMA1) to monitor the frequency, and supplies the main clock instead of the PLL0 clock or PLL1 clock. For details about the CLMA, see section 8, Clock Monitor Circuit (CLMA).

7.6 Low-Speed On-Chip Oscillator Oscillation Abnormality Detection Function

The clock monitor circuit (CLMA2) can be used to detect an abnormality in oscillation of the low-speed on-chip oscillator. For details about the CLMA, see section 8, Clock Monitor Circuit (CLMA).

7.7 PLL Circuit

The PLL circuit has a function to multiply the frequency from the oscillator.

7.8 Internal Clock

Clock sources of internal clock signals are the main clock, LOCO clock, PLL0 clock, PLL1 clock, and the external clock for JTAG. The 8 types of internal clocks listed below are produced from these sources.

(1) Operating clock of the CPU: CPU clock (CPUCLK)
(2) Operating clock of DMAC, interrupt controller, and RAM with ECC: System clock (ICLK)
(3) Operating clocks of peripheral modules: Peripheral module clocks (PCLKA, PCLKB, PCLKD, PCLKE, PCLKF, PCLKG, and PCLKH)
(4) Operating clock for high-speed serial clock: High-speed serial clock (SERICLK)
(5) Operating clock for the CLMA module: CLMA clocks (CLMAMCLKA, CLMAMCLKB, CLMALCLK, CLMAPLCLK0, and CLMAPLCLK1)
(6) Operating clock for the IWDT module: IWDT-dedicated clock (IWDTCLK)
(7) Operating clock for the JTAG module: JTAG clock (JTAGTCK)
(8) Operating clock for the trace interface: Trace interface clock (TCLK)

7.8.1 CPU Clock (CPUCLK)

The CPU clock (CPUCLK) is used as the operating clock of the CPU.
The CPU operating frequency is specified by the PLL1CR.CPUCKSEL[1:0] bits.

7.8.2 System Clock (ICLK)

The system clock (ICLK) is used as the operating clock of DMAC, interrupt controller, and on-chip extended SRAM with ECC.
The frequency of ICLK is fixed to 150 MHz. It cannot be specified by the user.

7.8.3 High-Speed Peripheral Module Clock (PCLKA)

The high-speed peripheral module clock (PCLKA) is used as the operating clock for high-speed peripheral modules.
The frequency of PCLKA is fixed to 150 MHz. It cannot be specified by the user.

7.8.4 Low-Speed Peripheral Module Clock (PCLKB)

The low-speed peripheral module clock (PCLKB) is used as the operating clock for low-speed peripheral modules.
The frequency of PCLKB is fixed to 75 MHz. It cannot be specified by the user.

7.8.5 Low-Speed Peripheral Module Clocks (PCLKD, PCLKE, PCLKF, PCLKG, and PCLKH)

The unmodulated low-speed peripheral module clocks (PCLKD, PCLKE, PCLKF, PCLKG, and PCLKH) are operating clocks for use by low-speed peripheral modules. The frequencies of PCLKD and PCLKH are respectively fixed to 75 MHz and 60 MHz. They cannot be specified by the user. The frequency of PCLKE, PCLKF, and PCLKG is set by the SCKCR.PCKE[1:0], SCKCR.PCKF[1:0], and SCKCR.PCKG[1:0] bits, respectively.

7.8.6 High-Speed Serial Clock (SERICLK)

The high-speed serial clock (SERICLK) is used as the operating clock for SCIFA and RSPIa.
The SERICLK operating frequency is specified by the SCKCR.SERICK bit.

7.8.7 CLMA Clocks (CLMAMCLKA, CLMAMCLKB, CLMALCLK, CLMAPLCLK0, and CLMAPLCLK1)

The CLMA clocks (CLMAMCLKA, CLMAMCLKB, CLMALCLK, CLMAPLCLK0, and CLMAPLCLK1) are used as the operating clocks for the CLMA module.
CLMAMCLKA and CLMAMCLKB are obtained by frequency-dividing the main clock.
CMLAMCLK is generated by internal oscillation of the low-speed on-chip oscillator.
CMLAPLCLK0 and CLMAPLCLK1 are obtained by frequency-dividing the clocks which are generated by internal oscillation of PLL0 and PLL1 circuits.

7.8.8 IWDT Clock (IWDTCLK)

The IWDT clock (IWDTCLK) is used as the operating clock for the IWDT module.
IWDTCLK is a 1/2 clock internally generated by the low-speed on-chip oscillator.

7.8.9 ECM Clock (ECMCLK)

The ECM clock (ECMCLK) is used as the delay counter operating clock for the ECM module.
ECMCLK is generated by internal oscillation of the low-speed on-chip oscillator.

7.8.10 JTAG Clock

The JTAG-dedicated clock (JTAGTCK) is the operating clock for the JTAG.
JTAGTCK is generated by the external clock for JTAG (TCK).

7.8.11 Trace Interface Clock (TCLK)

The trace interface clock (TCLK) is used as the operating clock for the trace interface within Coresight.
This clock is obtained by frequency-dividing the clock which is generated by internal oscillation within the PLL0 or PLL1 circuit.
This clock divided by two is output from the LSI as the trace clock for on-chip debugger (TRACECLK).

7.9 Usage Notes

7.9.1 Notes on Clock Generation Circuit

(1) Note that the operating frequency of the high-speed serial clock (SERICLK) that is supplied to the modules based on the SCKCR register setting varies before and after the frequency is changed.

(2) After the clock source of CPUCLK is changed from PLL0 to PLL1, do not change the frequency specified in the PLL1CR.CPUCKSEL bit.

(3) Do not change the PCLKE clock frequency after the WDTA counting started. To start the WDTA counting after the clock frequency is changed, confirm that the change to the frequency has been completed before you start the WDTA counting.

(4) Do not change the SERICLK clock frequency while the RSPIa (channels 0 to 3) or SCIFA(channel 0 to 2, and channel 4) is operating. To cancel the module-stop state after the clock frequency is changed, confirm that the change to the frequency has been completed before you cancel the module-stop state.

(5) In order to secure processing after the clock frequency is changed, dummy read the same register at least three times after writing to change the frequency, and only then proceed with the subsequent processing.

7.9.2 Notes on Resonator

Since various resonator characteristics relate closely to the user's board design, adequate evaluation is required on the user side before use, referencing the resonator connection example shown in this section. The circuit constants for the resonator depend on the resonator to be used and the stray capacitance of the mounting circuit. Therefore, the circuit constants should be determined in full consultation with the resonator manufacturer. The voltage to be applied between the resonator pins must be within the absolute maximum rating.

7.9.3 Notes on Designing the Board

When using a crystal resonator, place the resonator and its load capacitors as close to the XTAL and EXTAL pins as possible. Other signal lines should be routed away from the oscillation circuit as shown in Figure 7.5. This prevents electromagnetic induction from interfering with correct oscillation.

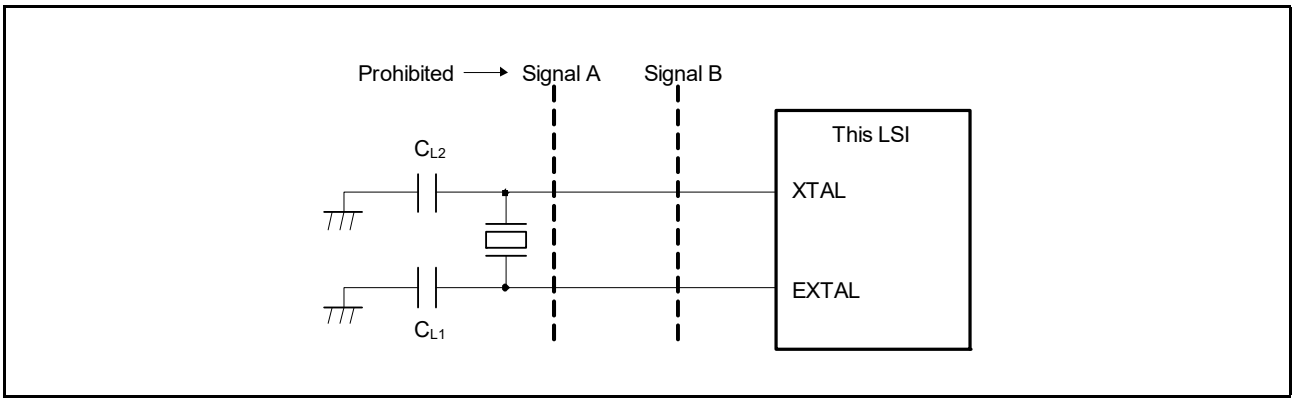


Figure 7.5 Notes on Board Design for Oscillation Circuit

8. Clock Monitor Circuit (CLMA)

A clock monitor circuit (CLMAn) (n = 2 to 0) detects a frequency error in PLL0 output, PLL1 output, or low-speed on-chip oscillator (LOCO) output, and then sends an error signal.

8.1 Overview

CLMAn (n = 2 to 0) can detect a frequency error in PLL0 output, PLL1 output, or on-chip oscillator (LOCO) output. During 16 cycles of the sampling clock, it counts the rising edges of the monitoring clock (frequency-dividing clock from PLL0, PLL1, or LOCO), and then compares the counter value with the compare register.

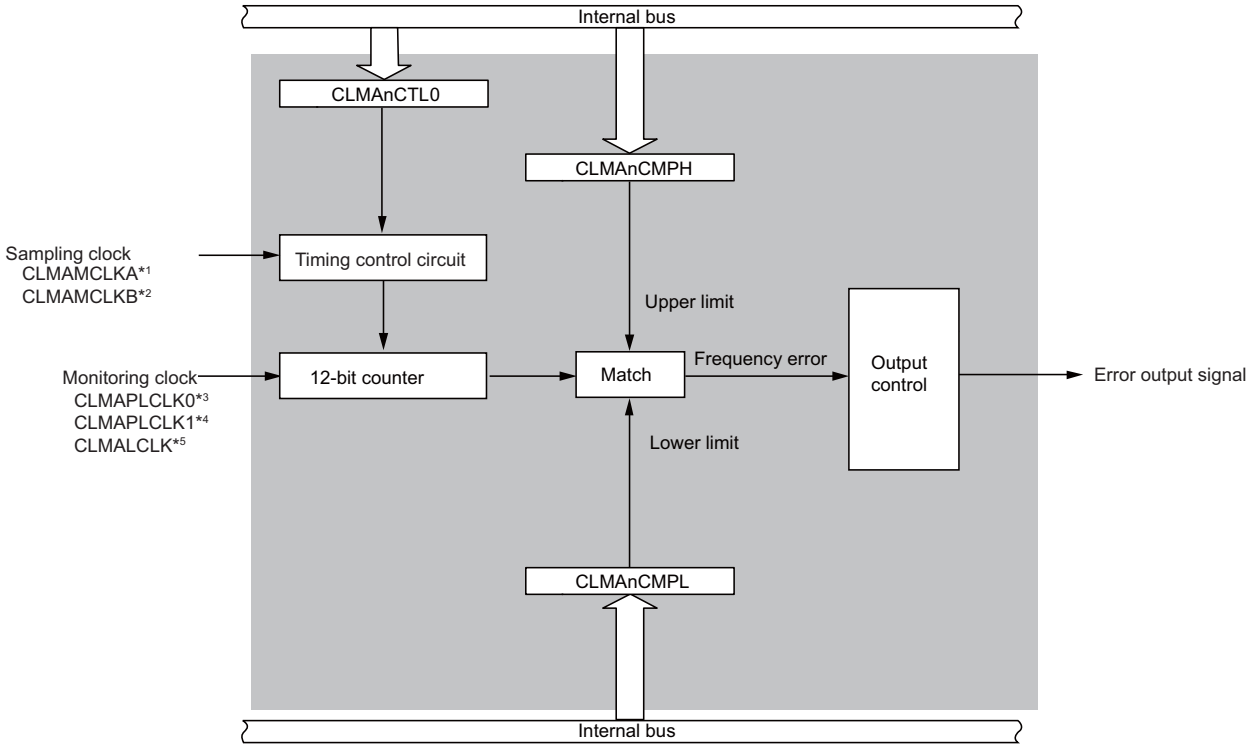
When CLMAn (n = 2 to 0) detects an error, it sends an error signal to the error control module (ECM). Besides, it switches the clock to supply the main clock instead of PLL0 or PLL1 output when it detects an error in PLL0 or PLL1 output.
For details on error signals, see section 29, Error Control Module (ECM).

Table 8.1 Specifications of CLMAn (n = 2 to 0)

Item	Specifications
Monitoring clock	The following monitoring clock frequency errors can be detected: • PLL0 output clock divided by 16 (CLMAPLCLK0, which is to be supplied to CLMA0): 75 MHz • PLL1 output clock divided by 16 (CLMAPLCLK1, which is to be supplied to CLMA1): 75 MHz / 56.25 MHz*1 • Low-speed on-chip oscillator (LOCO) output clock (CLMALCLK, which is to be supplied to CLMA2): 240 kHz
Sampling clock	The following clock frequency errors are monitored as the sampling clock: • Clock which equals to the main clock frequency divided by 2 (CLMAMCLKA, which is to be supplied to CLMA0): 12.5 MHz • Clock which equals to the main clock frequency divided by 2 (CLMAMCLKA, which is to be supplied to CLMA1): 12.5 MHz • Clock which equals to the main clock frequency divided by 256 (CLMAMCLKB, which is to be supplied to CLMA2): 97.66 kHz
Error signal output	When CLMAn detects a frequency error, it sends an error signal to the error control module (ECM). • CLMA0 oscillator-stopped detection error signal • CLMA1 oscillator-stopped detection error signal • CLMA2 oscillator-stopped detection error signal
Clock switching function when an error occurs	When an error is detected in PLL0 or PLL1 output, switches the clock to supply the main clock instead of PLL0 or PLL1 output.

Note 1. 56.25 MHz is selected when 10b is set for CPUCKSEL[1:0] of the PLL1CR register. When a value other than 10b is set, 75 MHz is selected. For details, see section 7, Clock Generation Circuit.

Figure 8.1 is a block diagram of CLMAn (n = 2 to 0).



Note 1. Main clock frequency divided by 2 to be supplied to CLMA0 and CLMA1
Note 2. Main clock frequency divided by 256 to be supplied to CLMA2
Note 3. PLL0 output clock divided by 16 to be supplied to CLMA0
Note 4. PLL1 output clock divided by 16 to be supplied to CLMA1
Note 5. Low-speed on-chip oscillator (LOCO) output clock to be supplied to CLMA2

Figure 8.1 Block Diagram of CLMAn (n = 2 to 0)

8.2 Register Descriptions

8.2.1 CLMAn Control Registers 0 (CLMAnCTL0) (n = 2 to 0)

The CLMAnCTL0 registers control operation of clock monitor circuit CLMAn.
Writing to these registers is protected by a specific command sequence. For details, see section 8.3.1, (1) Enabling operations.

Address(es): CLMA0CTL0: A009 0000h
CLMA1CTL0: A009 0020h
CLMA2CTL0: A009 0040h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	CLMAn CLME
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CLMAnCLME	Clock Monitor Enable n	Enables or disables operation of clock monitor circuit CLMAn (n = 2 to 0). 0: Disables CLMAn operation. 1: Enables CLMAn operation.	R/W
b7 to b1	—	Reserved	These bits are always read as 0. The write value should be 0.	R/(W)

Note 1. Once the CLMAnCLME bit is set to 1, it cannot be cleared by any operation other than reset.

8.2.2 CLMAn Compare Registers L (CLMAnCMPL) (n = 2 to 0)

The CLMAnCMPL registers set the lower limit for comparing frequency domains.
Values can be written to these registers when the CLMAnCLME bit is set to 0. When the CLMAnCLME bit is set to 1, writing to these registers has no effect.

Address(es): CLMA0CMPL: A009 0008h
CLMA1CMPL: A009 0028h
CLMA2CMPL: A009 0048h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	CLMAnCMPL[11:0]											
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b11 to b0	CLMAnCMPL [11:0]	Clock Monitor Compare L	Specify the lower-limit threshold for the frequency domain.*1 • For details, see section 8.3.2, (2) Method of calculating threshold values, CLMAnCMPL.CLMAnCMPL[11:0] and CLMAnCMPH.CLMAnCMPH[11:0]. Recommended value: $f_{CLMATMON\ (min)} / f_{CLMATSMP\ (max)} \times 16 - 1$ $f_{CLMATMON}$: Monitoring clock frequency $f_{CLMATSMP}$: Sampling clock frequency • Minimum value: 0001h	R/W
b15 to b12	—	Reserved	These bits are always read as 0. The write value should be 0.	R/(W)

Note 1. To set the CLMAnCMPL registers, the following conditions must be met:
- 1 ≤ CLMAnCMPL
- CLMAnCMPL + 3 ≤ CLMAnCMPH

8.2.3 CLMAn Compare Registers H (CLMAnCMPH) (n = 2 to 0)

The CLMAnCMPH registers set the upper limit for comparing frequency domains.
Values can be written to the CLMAnCMPH registers when the CLMAnCLME bit is set to 0. When the CLMAnCLME bit is set to 1, writing to these registers has no effect.

Address(es): CLMA0CMPH: A009 000Ch
CLMA1CMPH: A009 002Ch
CLMA2CMPH: A009 004Ch

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	CLMAnCMPH[11:0]											
Value after reset:	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b11 to b0	CLMAnCMPH[11:0]	Clock Monitor Compare H	Specify the upper-limit threshold for the frequency domain*1. • For details, see section 8.3.2, (2) Method of calculating threshold values, CLMAnCMPL.CLMAnCMPL[11:0] and CLMAnCMPH.CLMAnCMPH[11:0]. Recommended value: $f_{CLMATMON\ (max)} / f_{CLMATSMP\ (min)} \times 16 + 1$ $f_{CLMATMON}$: Monitoring clock frequency $f_{CLMATSMP}$: Sampling clock frequency • Minimum value: CLMAnCMPL + 0003h	R/W
b15 to b12	—	Reserved	These bits are always read as 0. The write value should be 0.	R/(W)

Note 1. To set the CLMAnCMPH registers, the following conditions must be met:
- 1 ≤ CLMAnCMPL
- CLMAnCMPL + 3 ≤ CLMAnCMPH

8.2.4 CLMAn Command Registers (CLMAnPCMD) (n = 2 to 0)

The CLMAnPCMD registers control writing to the protected registers.
For details, see section 8.3.1, (1) Enabling operations.

Address(es): CLMA0PCMD: A009 0010h
CLMA1PCMD: A009 0030h
CLMA2PCMD: A009 0050h

	b7	b6	b5	b4	b3	b2	b1	b0
	CLMAnPR[7:0]							
Value after reset:	1	1	1	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	CLMAnPR[7:0]	CLMAn Protect Key Code	Write a specific command sequence.	W

8.2.5 CLMAn Protection Status Registers (CLMAnPS) (n = 2 to 0)

The CLMAnPS registers indicate whether writing to the protected register is performed correctly. If writing is not performed correctly, a protection error occurs, and the CLMAnPS.CLMAnPRERR bit is set to 1.
For details, see section 8.3.1, (1) Enabling operations.

Address(es): CLMA0PS: A009 0014h
CLMA1PS: A009 0034h
CLMA2PS: A009 0054h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	CLMAn PRERR
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CLMAnPRERR	CLMAn Error	0: No protection error occurred. 1: A protection error occurred.	R
b7 to b1	—	Reserved	These bits are always read as 0.	R

8.3 Operation

8.3.1 CLMAn Operation

(1) Enabling operations

Setting the CLMAnCTL0.CLMAnCLME bit to 1 starts monitoring the monitoring clock (PLL0 output divided by 16, PLL1 output divided by 16, and LOCO output) by using clock monitor circuit CLMAn (n = 2 to 0) output. To write 1 in the CLMAnCTL0.CLMAnCLME bit, follow the following command sequence:

1. Write A5h to the CLMAnPCMD register.
2. Writing to CLMAnCTL0 is performed by the following sequence:
   - Write the target setting value (01h).
   - Write the reversed value of the target (FEh).
   - Write the target value (01h) again.
3. Read CLMAnCTL0.
   When the value of CLMAnCTL0 shows 01h, the operation of CLMAn is enabled.
   If it shows another value, check the value of CLMAn protection status register (CLMAnPS).

When CLMAnPS = 01h, the command sequence is not performed correctly. Execute the sequence again from step 1 to perform writing.

(2) Stopping operations

When a monitoring clock stops due to register operation, the corresponding clock monitor circuit (CLMAn) is disabled automatically. After that, when the monitoring clock starts oscillating again and stabilizes, the clock monitor circuit (CLMAn) resumes operation.

8.3.2 Detecting Error Clock Frequency

(1) Detection Method

1. During 16 cycles of the sampling clock, CLMAn counts the rising edge of the monitoring clock (PLL0 output divided by 16, PLL1 output divided by 16, and LOCO output), and compares the counter value with the set threshold (n = 2 to 0).
   - CLMAnCMPL.CLMAnCMPL[11:0] set the lower-limit threshold of the frequency domain.
   - CLMAnCMPH.CLMAnCMPH[11:0] set the upper-limit threshold of the frequency domain.
2. If the monitoring clock stops, or shows a value lower than the expected frequency, the counter value shows a value lower than the setting of CLMAnCMPL.CLMAnCMPL[11:0].
3. If the monitoring clock shows a value higher than the expected frequency, the counter value shows a value higher than the setting of CLMAnCMPH.CLMAnCMPH[11:0].

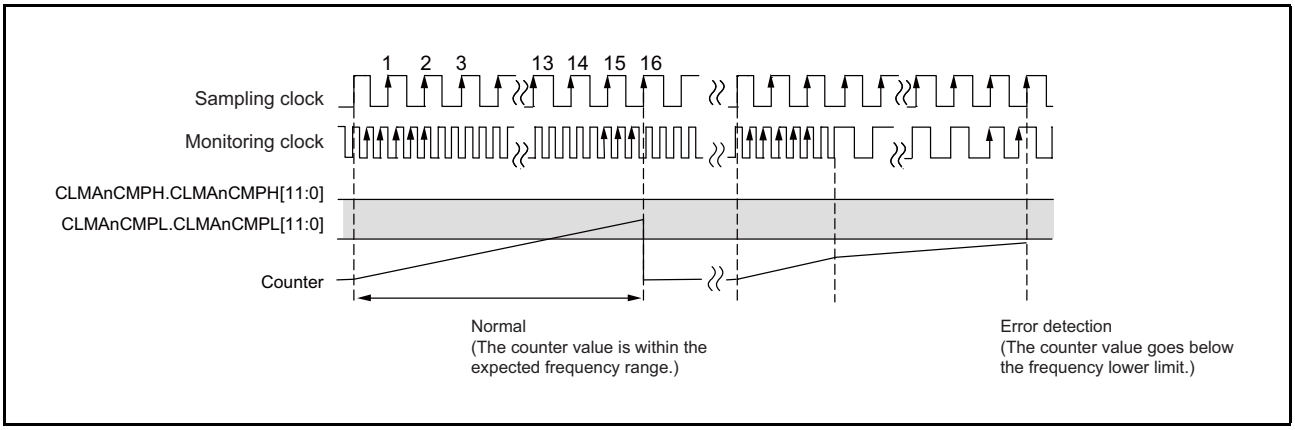


Figure 8.2 Example When the Monitoring Clock Shows a Value Lower Than the Expected Frequency

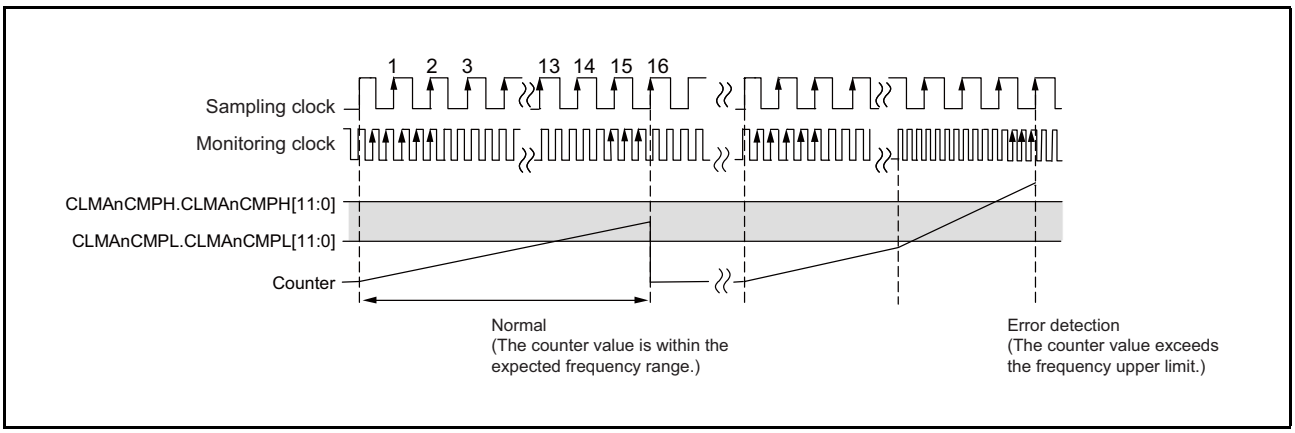


Figure 8.3 Example When the Monitoring Clock Shows a Value Higher Than the Expected Frequency

Note 1. No error is detected if the monitoring clock frequency changes within a sampling interval, and it stays within a valid counter value eventually. A monitoring clock error is detected after a sampling interval (16 cycles of the sampling clock).

(2) Method of calculating threshold values, CLMAnCMPL.CLMAnCMPL[11:0] and CLMAnCMPH.CLMAnCMPH[11:0]

For compare registers CLMAnCMPL and CLMAnCMPH, set the minimum and maximum values of the number of cycles (the number of rising edges) for the monitoring clock which is assumed to be normal within 16 cycles of the sampling clock (main clock frequency dividing clock).

$f_{CLMATSMP}$ indicates the sampling clock frequency, $f_{CLMATMON}$ indicates the monitoring clock frequency, and N indicates the number of cycles (rising edges) of the monitoring clock, which is expected within 16 cycles of the sampling clock.

$$\frac{16}{f_{CLMATSMP}} = \frac{N}{f_{CLMATMON}}$$

$$N = \frac{f_{CLMATMON}}{f_{CLMATSMP}} \times 16$$

Considering allowable frequency deviations for the monitoring clock and the sampling clock, use the following formula to calculate the threshold:

Lower-limit threshold $= N_{min}$

$$= \frac{f_{CLMATMON(min)}}{f_{CLMATSMP(max)}} \times 16 - 1$$

Upper-limit threshold $= N_{max}$

$$= \frac{f_{CLMATMON(max)}}{f_{CLMATSMP(min)}} \times 16 + 1$$

Note: Set the threshold within the following range:
CLMAnCMPL ≥ 0001_H
CLMAnCMPH ≥ CLMAnCMPL + 0003_H

Example: For CLMA0

For example, when the sampling clock equals to the main clock frequency divided by 2, $f_{CLMATSMP}$ = 12.5 MHz (±5%), or the monitoring clock equals to PLL0 output divided by 16, $f_{CLMATMON}$ = 75 MHz (±5%), the recommended threshold is calculated as follows:

$$N_{min} = f_{CLMATMON}(min) / f_{CLMATSMP}(max) = 71.25 / 13.125 \times 16 - 1$$
$$= 85.86$$

CLMAnCMPL = 86 = 0056h

$$N_{max} = f_{CLMATMON}(max) / f_{CLMATSMP}(min) = 78.75 / 11.875 \times 16 + 1$$
$$= 107.11$$

CLMAnCMPH = 107 = 006Bh

8.3.3 Detecting Error Clock Frequency

When the monitoring clock frequency (PLL0 output divided by 16, PLL1 output divided by 16, and LOCO output) is higher than the upper-limit threshold, or lower than the lower-limit threshold, any of the following signals is sent to the error control module (ECM):

- CLMA0 oscillator-stopped detection error signal
- CLMA1 oscillator-stopped detection error signal
- CLMA2 oscillator-stopped detection error signal

For details on error signals, see section 29, Error Control Module (ECM).

8.4 Notes on Using CLMAn

Do not use a clock for which CLMAn detected an error. If you use the clock, operation of the device is not guaranteed.

9. Low-Power Consumption Function

9.1 Overview

This LSI has several functions for reducing power consumption, including the Cortex-R4 standby function and the module stop function that stops functions independently for each peripheral module.
Table 9.1 lists the specifications of low-power consumption functions. Table 9.2 describes how to stop each peripheral module and exit the stop state.

Table 9.1 Specifications of Low-Power Consumption Function

Item	Specifications
Low-power consumption mode	Standby mode (Cortex-R4)
Module-stop function	Functions can be stopped independently for each peripheral module.

Table 9.2 Stopping Peripheral Modules and Exiting Module-Stop State (1 / 2)

Module	How to Stop Operation and Exit the Stop State	Initial State*1
Cortex-R4	Stop condition: Execution of the Wait For Interrupt (WFI) instruction Condition for release from the stop-state: Interrupt	Operating
Internal bus	Always operating	Operating
Tightly Coupled Memory (ATCM or BTCM)	Operate only during access	Operate only during access
On-chip extended SRAM (with ECC)	Operate only during access	Operate only during access
Interrupt controller	Operation is always enabled.	Operating
Error control module (ECM)	Operation is always enabled.	Operating
16-bit timer pulse unit (TPUa)	Set the control register to enter or exit the module-stop state.	Stop
Compare match timer (CMT)	Set the control register to enter or exit the module-stop state.	Stop
Compare match timer W (CMTW)	Set the control register to enter or exit the module-stop state.	Stop
Serial peripheral interface (RSPIa)	Set the control register to enter or exit the module-stop state.	Stop
Serial communication interface with FIFO (SCIFA)	Set the control register to enter or exit the module-stop state.	Stop
I^2C bus interface (RIICa)	Set the control register to enter or exit the module-stop state.	Stop
Clock monitor circuit (CLMA)	Set the control register to enter or exit the module-stop state.	Stop
CRC operation part (CRC)	Set the control register to enter or exit the module-stop state.	Stop
Data operation circuit (DOC)	Set the control register to enter or exit the module-stop state.	Stop
SPI multi I/O bus controller (SPIBSC)	Set the control register to enter or exit the module-stop state.	Stop*2
Event link controller (ELC)	Set the control register to enter or exit the module-stop state.	Stop
12-bit A/D converter (S12ADCa)	Set the control register to enter or exit the module-stop state.	Stop
Temperature sensor	Set the control register to enter or exit the module-stop state.	Stop
Direct memory access controller (DMAC)	Set the control register to stop the module or exit the module-stop state.	Operating

Table 9.2 Stopping Peripheral Modules and Exiting Module-Stop State (2 / 2)

Module	How to Stop Operation and Exit the Stop State	Initial State*1
Management data input/output interface (MDIOM/MDIO)	Set the control register to enter or exit the module-stop state.	Stop
I/O port	Operation is always enabled.	Operating
Coresight	Set the control register to stop the module or exit the module-stop state.	Operating
Watchdog timer (WDTA)	Operation is always enabled.	Operating
Independent watchdog timer (IWDTa)	Operation is always enabled.	Operating

Note 1. Each module is returned to the initial state by means of the RES# pin reset, error control module (ECM) reset, or software reset.
Note 2. The state of the SPI multi I/O bus controller (SPIBSC) after boot processing varies depending on the settings of the parameters for the loader. For details, see section 3, Operating Modes.

9.2 Register Descriptions

The registers are applicable to the register write protection function. For writing these registers, clear write protection for the target register by setting b1 of the Protect Register (PRCR). For details, see section 11, Register Write Protection Function.

9.2.1 Module Stop Control Register A (MSTPCRA)

The MSTPCRA register controls the module-stop state.
For release from the module-stop state, see section 9.3.1, Module-Stop Function.

Address(es): A00B 0300h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	--	—	MSTP CRA8	—	—	—	MSTP CRA4	MSTP CRA3	—	MSTP CRA1	MSTP CRA0
Value after reset:	0	0	0	0	1	0	1	1	1	1	1	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b0	MSTPCRA0	CMTW Unit 1 Module Stop	Target module: CMTW unit 1 0: Release from the module-stop state 1: Transition to the module-stop state is made.	R/W
b1	MSTPCRA1	CMTW Unit 0 Module Stop	Target module: CMTW unit 0 0: Release from the module-stop state 1: Transition to the module-stop state is made	R/W
b2	—	Reserved	This bit is read as 1. The write value should be 1.	R/W
b3	MSTPCRA3	CMT Unit 1 Module Stop	Target module: CMT unit 1 0: Release from the module-stop state 1: Transition to the module-stop state is made	R/W
b4	MSTPCRA4	CMT Unit 0 Module Stop	Target module: CMT unit 0 0: Release from the module-stop state 1: Transition to the module-stop state is made	R/W
b7 to b5	—	Reserved	These bits are read as 1. The write value should be 1.	R/W
b8	MSTPCRA8	TPUa Unit 0 Module Stop	Target module: TPUa unit 0 0: Release from the module-stop state 1: Transition to the module-stop state is made	R/W
b9	—	Reserved	This bit is read as 1. The write value should be 1.	R/W
b10	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b11	—	Reserved	This bit is read as 1. The write value should be 1.	R/W
b31 to b12	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

9.2.2 Module Stop Control Register B (MSTPCRB)

The MSTPCRB register controls the module-stop state.

For release from the module-stop state, see section 9.3.1, Module-Stop Function.

Address(es): A00B 0304h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	MSTP CRB13	MSTPC RB12	—	—	MSTP CRB9	MSTP CRB8	MSTP CRB7	—	MSTP CRB5	—	MSTP CRB3	MSTP CRB2	—	—
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b1	—	Reserved	This bit is read as 1. The write value should be 1.	R/W
b2	MSTPCRB2	RIICa Channel 1 Module Stop	Target module: RIICa channel 1 0: Release from the module-stop state 1: Transition to the module-stop state is made	R/W
b3	MSTPCRB3	RIICa Channel 0 Module Stop	Target module: RIICa channel 0 0: Release from the module-stop state 1: Transition to the module-stop state is made	R/W
b4	—	Reserved	This bit is read as 1. The write value should be 1.	R/W
b5	MSTPCRB5	SCIFA Channel 4 Module Stop	Target module: SCIFA channel 4 0: Release from the module-stop state 1: Transition to the module-stop state is made	R/W
b6	—	Reserved	This bit is read as 1. The write value should be 1.	R/W
b7	MSTPCRB7	SCIFA Channel 2 Module Stop	Target module: SCIFA channel 2 0: Release from the module-stop state 1: Transition to the module-stop state is made	R/W
b8	MSTPCRB8	SCIFA Channel 1 Module Stop	Target module: SCIFA channel 1 0: Release from the module-stop state 1: Transition to the module-stop state is made	R/W
b9	MSTPCRB9	SCIFA Channel 0 Module Stop	Target module: SCIFA channel 0 0: Release from the module-stop state 1: Transition to the module-stop state is made	R/W
b11, b10	—	Reserved	These bits are read as 1. The write value should be 1.	R/W
b12	MSTPCRB12	RSPIa Channel 1 Module Stop	Target module: RSPIa channel 1 0: Release from the module-stop state 1: Transition to the module-stop state is made	R/W
b13	MSTPCRB13	RSPIa Channel 0 Module Stop	Target module: RSPIa channel 0 0: Release from the module-stop state 1: Transition to the module-stop state is made	R/W
b19 to b14	—	Reserved	These bits are read as 1. The write value should be 1.	R/W
b31 to b20	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

9.2.3 Module Stop Control Register C (MSTPCRC)

The MSTPCRC register controls the module-stop state.

For release from the module-stop state, see section 9.3.1, Module-Stop Function.

Address(es): A00B 0308h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	MSTP CRC14	MSTP CRC13	MSTP CRC12	MSTP CRC11	MSTP CRC10	MSTP CRC9	—	—	MSTP CRC6	MSTP CRC5	MSTP CRC4	MSTP CRC3	—	—	—
Value after reset:	0	1	1	1	1	1	1*1	1	0	1	1	1	1	1	1	0

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2, b1	—	Reserved	These bits are read as 1. The write value should be 1.	R/W
b3	MSTPCRC3	Temperature Sensor Module Stop	Target module: Temperature sensor 0: Release from the module-stop state 1: Transition to the module-stop state is made	R/W
b4	MSTPCRC4	ADC Unit 1 Module Stop	Target module: ADC unit 1 0: Release from the module-stop state 1: Transition to the module-stop state is made	R/W
b5	MSTPCRC5	ADC Unit 0 Module Stop	Target module: ADC unit 0 0: Release from the module-stop state 1: Transition to the module-stop state is made	R/W
b6	MSTPCRC6	ELC Module Stop	Target module: ELC 0: Release from the module-stop state 1: Transition to the module-stop state is made	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b8	—	Reserved	This bit is read as 1. The write value should be 1.	R/W
b9	MSTPCRC9	SPIBSC Module Stop	Target module: SPIBSC 0: Release from the module-stop state 1: Transition to the module-stop state is made	R/W
b10	MSTPCRC10	DOC Module Stop	Target module: DOC 0: Release from the module-stop state 1: Transition to the module-stop state is made	R/W
b11	MSTPCRC11	CRC Module Stop	Target module: CRC 0: Release from the module-stop state 1: Transition to the module-stop state is made	R/W
b12	MSTPCRC12	CLMA Unit 2 Module Stop	Target module: CLMA unit 2 0: Release from the module-stop state 1: Transition to the module-stop state is made	R/W
b13	MSTPCRC13	CLMA Unit 1 Module Stop	Target module: CLMA unit 1 0: Release from the module-stop state 1: Transition to the module-stop state is made	R/W
b14	MSTPCRC14	CLMA Unit 0 Module Stop	Target module: CLMA unit 0 0: Release from the module-stop state 1: Transition to the module-stop state is made	R/W
b31 to b15	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. The state of the SPI multi I/O bus controller (SPIBSC) after boot processing varies depending on the settings of the parameters for the loader. For details, see section 3, Operating Modes.

9.2.4 Module Stop Control Register E (MSTPCRE)

The MSTPCRE register controls the module-stop state.

For release from the module-stop state, see section 9.3.1, Module-Stop Function.

Address(es): A00B 0310h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	MSTP CRE5	MSTP CRE4	—	—	—	MSTP CRE0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b0	MSTPCRE0*1	MDIO Module Stop	Target module: MDIOM, MDIO 0: Release from the module-stop state 1: Transition to the module-stop state is made	R/W
b3 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b4	MSTPCRE4	DMAC Unit 1 Module Stop	Target module: DMAC unit 1 0: Release from the module-stop state 1: Transition to the module-stop state is made	R/W
b5	MSTPCRE5	DMAC Unit 0 Module Stop	Target module: DMAC unit 0 0: Release from the module-stop state 1: Transition to the module-stop state is made	R/W
b31 to b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Explicit software-driven release from the module-stop state by the setting of this bit is not required, because the MDIO configuration library handles processing for release.

9.2.5 Module Stop Control Register F (MSTPCRF)

The MSTPCRF register controls the module-stop state.

For release from the module-stop state, see section 9.3.1, Module-Stop Function.

Address(es): A00B 0314h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MSTPCRF0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	MSTPCRF0	Coresight Module Stop	Target module: Coresight 0: Release from the module-stop state 1: Transition to the module-stop state is made	R/W
b31 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

9.3 Operation

9.3.1 Module-Stop Function

The module-stop function can stop operation of each module of on-chip peripheral functions.
When the MSTPmi bit (m = A to C, E, F, i = 31 to 0) in the MSTPCRA to MSTPCRC, MSTPCRE, and MSTPCRF registers is set to 1, the specified module stops operating and enters the module-stop state.
Clearing the MSTPmi bit (m = A to C, E, F, i = 31 to 0) to 0 leads to release from the module-stop state.
Follow the procedure given below when releasing any of the peripheral modules listed in Table 9.3 from the module-stop state. This procedure is not required for peripheral modules not listed in Table 9.3. However, release from the module-stop state should be in accord with the procedure for initialization described in the section for the given peripheral module.

[Procedures]

(1) Clear the corresponding bit in the relevant module stop control register (MSTPCRm; m = A to C, E) then immediately dummy-read the MSTPCRm register once.
(2) Dummy-read any register of the peripheral module which is being released from the module-stop state. After that, all registers of that peripheral module will be accessible.

Remarks: The MPU having set the access control attribute for the peripheral I/O register region to ‘Strongly-ordered’ or ‘Device’ is a prerequisite for the procedure.

<Example code>

```
volatile unsigned long dummy;                 //   Declared volatile to prevent optimization being applied

   SYSTEM.MSTPCRA.BIT.MSTPCRA0 = 0;           //   Setting to release CMTW unit 1 from the module-stop state
   dummy = SYSTEM.MSTPCRA.BIT.MSTPCRA0;       //   Step 1: Dummy-read the MSTPCRm register.

   dummy = CMTW1.CMWIOR.WORD;                 //   Step 2: Dummy-read any register of CMTW unit 1.
   CMTW1.CMWIOR.WORD = 0x81;                  //   The first setting for CMTW unit 1 (value is an example)
```

For details about the initial state after a release from the reset state, see Table 9.2, Stopping Peripheral Modules and Exiting Module-Stop State.

Note: Directly after a module is set to the module-stop state, writing might still be possible to the control register of that module.

Table 9.3 Peripheral Functions Requiring the above Procedure for Release from the Module-Stop State (1 / 2)

Peripheral Functions	Corresponding Module-Stop Control Register
CMTW unit 1	MSTPCRA register, MSTPCRA0 bit
CMTW unit 0	MSTPCRA register, MSTPCRA1 bit
CMT unit 1	MSTPCRA register, MSTPCRA3 bit
CMT unit 0	MSTPCRA register, MSTPCRA4 bit
TPUa unit 0	MSTPCRA register, MSTPCRA8 bit
RIICa channel 1	MSTPCRB register, MSTPCRB2 bit
RIICa channel 0	MSTPCRB register, MSTPCRB3 bit
SCIFA channel 4	MSTPCRB register, MSTPCRB5 bit
SCIFA channel 2	MSTPCRB register, MSTPCRB7 bit
SCIFA channel 1	MSTPCRB register, MSTPCRB8 bit

Table 9.3 Peripheral Functions Requiring the above Procedure for Release from the Module-Stop State (2 / 2)

Peripheral Functions	Corresponding Module-Stop Control Register
SCIFA channel 0	MSTPCRB register, MSTPCRB9 bit
RSPIa channel 1	MSTPCRB register, MSTPCRB12 bit
RSPIa channel 0	MSTPCRB register, MSTPCRB13 bit
Temperature sensor	MSTPCRC register, MSTPCRC3 bit
ADC unit 1	MSTPCRC register, MSTPCRC4 bit
ADC unit 0	MSTPCRC register, MSTPCRC5 bit
ELC	MSTPCRC register, MSTPCRC6 bit
SPIBSC	MSTPCRC register, MSTPCRC9 bit
DOC	MSTPCRC register, MSTPCRC10 bit
CRC	MSTPCRC register, MSTPCRC11 bit
CLMA unit 2	MSTPCRC register, MSTPCRC12 bit
CLMA unit 1	MSTPCRC register, MSTPCRC13 bit
CLMA unit 0	MSTPCRC register, MSTPCRC14 bit
MDIOM/MDIO*1	MSTPCRE register, MSTPCRE0 bit
DMAC unit 1	MSTPCRE register, MSTPCRE4 bit
DMAC unit 0	MSTPCRE register, MSTPCRE5 bit

Note 1. Explicit software-driven release from the module-stop state by the setting of this bit is not required, because the MDIO configuration library handles processing for release.

9.3.2 Cortex-R4 Standby Mode

9.3.2.1 Transition to Cortex-R4 Standby Mode

Cortex-R4 enters standby mode by execution of a WFI instruction. Cortex-R4 enters standby mode by execution of a WFI instruction. In standby mode, Cortex-R4 stops operating, thus reducing power consumption. For details, see the technical reference manual provided by Arm.

9.3.2.2 Release from Cortex-R4 Standby Mode

Release from Cortex-R4 standby mode is initiated by any interrupt, the RES# pin reset, an ECM reset, or a software reset.

- Release triggered by an interrupt signal
  Generation of an interrupt in a CPU triggers release from standby mode or sleep mode of the CPU and the interrupt exception handling starts. Release is triggered by a non-maskable interrupt or a maskable interrupt that meets the following conditions:

  (1) The interrupt request is permitted by using the interrupt enable register.
  (2) Nothing has been assigned to DMAC by using the DMAC source selection register.

- Release by a reset
  After the RES# pin reset, ECM reset, or software reset is cleared, Cortex-R4 starts the reset exception handling. For details about resets, see section 6, Reset.

9.4 Usage Notes

9.4.1 I/O Port State

To reduce I/O power consumption, pin processing based on I/O control is required. For details, see section 16, I/O Ports.

9.4.2 Module-Stop State of DMAC

Do not set a module-stop state while DMAC is operating. Make sure that the DMAC is inactive before you attempt to set the module-stop state.
For details, see section 14, DMA Controller (DMACAa).

9.4.3 On-Chip Peripheral Module Interrupts in Module-Stop State

Peripheral modules cannot interrupt in a module-stop state. Therefore, if the module-stop state is set during interrupt processing of the module or during DMA transfer by the DMAC, a CPU interrupt source or a DMAC or DTC startup source cannot be cleared. For this reason, disable the relevant interrupts before setting the module-stop state.

9.4.4 Write Protection Function

The module stop control registers (MSTPCRA to MSTPCRC, MSTPCRE, and MSTPCRF) can be write-protected. To write to the MSTPCRA to MSTPCRC, MSTPCRE, and MSTPCRF registers, specify bit 1 of the Protect Register (PRCR) to unlock the write protection. For details, see section 11, Register Write Protection Function.

10. Debugging Interface

This LSI has an internal debugging interface, which adopts an architecture in which Cortex-R4 is integrated by CoreSight. This LSI supports debugging functions, such as downloading, running, and breaking a program, and trace function, which outputs execution history of programs.

10.1 Overview

This LSI supports JTAG and SWD interfaces as interfaces for debugging, and trace port and SWV interfaces as interfaces for trace.
This LSI has TAP controllers for CoreSight debugging.

Table 10.1 lists the specifications of CoreSight, and Figure 10.1 is a block diagram of CoreSight. Moreover, Table 10.4 and Table 10.5 indicate the CoreSight address maps. For details on CoreSight, see Arm's technical reference manual.

Table 10.1 CoreSight Specifications

Item	Specifications
Debugging function	• JTAG interface • SWD (Serial Wire Debug) interface
Trace function	• Trace port interface 8 bits × 75 Mbps (37.5 MHz, DDR) trace data pin output Embedded Trace Buffer (ETB) 4 Kbytes • SWV (Serial Wire Viewer) interface

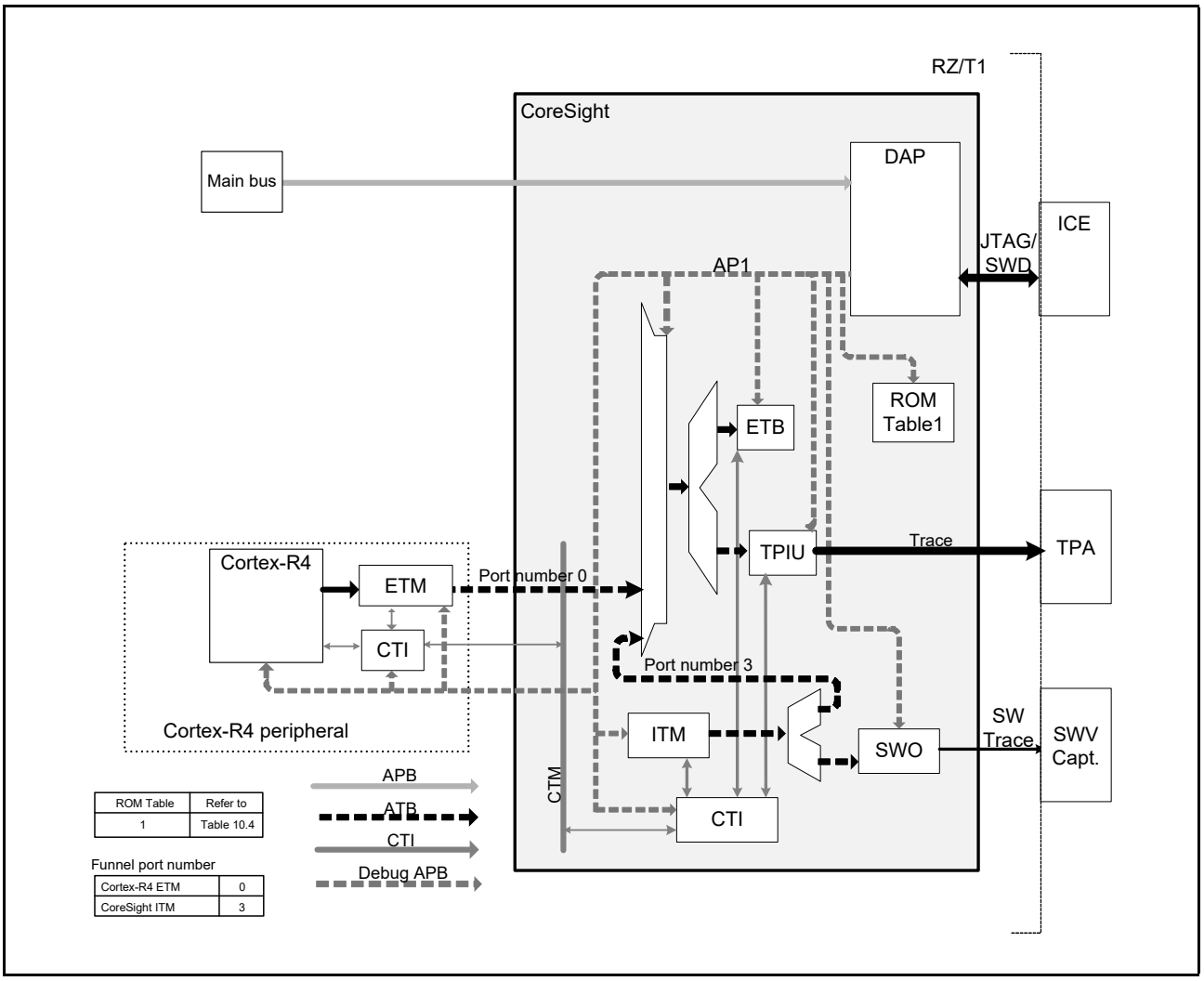


Figure 10.1 Block Diagram of CoreSight

Table 10.2 CTI Trigger Input and Output (CoreSight)

CTI Input Pin	Source Device	Signal	CTI Output Pin	Destination Device	Signal
CTITRIGIN[0]	—	—	CTITRIGOUT[0]	ETB	FLUSHIN
CTITRIGIN[1]	—	—	CTITRIGOUT[1]	ETB	TRIGIN
CTITRIGIN[2]	ETB	FULL	CTITRIGOUT[2]	TPIU	FLUSHIN
CTITRIGIN[3]	ETB	ACQCOMP	CTITRIGOUT[3]	TPIU	TRIGIN
CTITRIGIN[4]	ITM	TRIGOUT	CTITRIGOUT[4]	—	—
CTITRIGIN[5]	—	—	CTITRIGOUT[5]	—	—
CTITRIGIN[6]	—	—	CTITRIGOUT[6]	—	—
CTITRIGIN[7]	—	—	CTITRIGOUT[7]	—	—

Table 10.3 CTI Trigger Input and Output (Cortex-R4)

CTI Input Pin	Source Device	Signal	CTI Output Pin	Destination Device	Signal
CTITRIGIN[0]	Cortex-R4	DBGTRIGGER	CTITRIGOUT[0]	Cortex-R4	EDBGRQ
CTITRIGIN[1]	Cortex-R4	nPMUIRQ	CTITRIGOUT[1]	ETM-R4	EXTIN[0]
CTITRIGIN[2]	ETM-R4	EXOUT[0]	CTITRIGOUT[2]	ETM-R4	EXTIN[1]
CTITRIGIN[3]	ETM-R4	EXOUT[1]	CTITRIGOUT[3]	VIC	TRIGINT
CTITRIGIN[4]	Cortex-R4	COMMRX	CTITRIGOUT[4]	—	—
CTITRIGIN[5]	Cortex-R4	COMMTX	CTITRIGOUT[5]	—	—
CTITRIGIN[6]	ETM-R4	TRIGGER	CTITRIGOUT[6]	—	—
CTITRIGIN[7]	—	—	CTITRIGOUT[7]	Cortex-R4	DBGRESTART

Table 10.4 CoreSight Address Map (Debug-APB)

Cortex-R4 CPU View	Debugger View*1 (AP = 1)	Module
H'E8000000 to H'E8000FFF	H'00000000 to H'00000FFF	CoreSight / DAP ROM
H'E8001000 to H'E8001FFF	H'00001000 to H'00001FFF	CoreSight / ETB
H'E8002000 to H'E8002FFF	H'00002000 to H'00002FFF	CoreSight / CTI
H'E8003000 to H'E8003FFF	H'00003000 to H'00003FFF	CoreSight / TPIU
H'E8004000 to H'E8004FFF	H'00004000 to H'00004FFF	CoreSight / Funnel
H'E8005000 to H'E8005FFF	H'00005000 to H'00005FFF	CoreSight / ITM
H'E8006000 to H'E8006FFF	H'00006000 to H'00006FFF	CoreSight / SWO
H'E8007000 to H'E8007FFF	H'00007000 to H'00007FFF	—
H'E8008000 to H'E8008FFF	H'00008000 to H'00008FFF	Cortex-R4 / CPU
H'E8009000 to H'E8009FFF	H'00009000 to H'00009FFF	Cortex-R4 / CTI
H'E800A000 to H'E800AFFF	H'0000A000 to H'0000AFFF	Cortex-R4 / ETM-R4
H'E800B000 to H'E800BFFF	H'0000B000 to H'0000BFFF	—
H'E800C000 to H'E800CFFF	H'0000C000 to H'0000CFFF	—
H'E800D000 to H'E800DFFF	H'0000D000 to H'0000DFFF	—
H'E800E000 to H'E800EFFF	H'0000E000 to H'0000EFFF	—
H'E800F000 to H'E800FFFF	H'0000F000 to H'0000FFFF	—

Note 1. When A31 (the most significant bit of the address) is set to 1, access without releasing the access lock becomes possible.

Table 10.5 CoreSight Address Map

Cortex-R4 CPU View	Module
E800 0000h to E800 0FFFh	CoreSight / DAP ROM
E800 1000h to E800 1FFFh	CoreSight / ETB
E800 2000h to E800 2FFFh	CoreSight / CTI
E800 3000h to E800 3FFFh	CoreSight / TPIU
E800 4000h to E800 4FFFh	CoreSight / Funnel
E800 5000h to E800 5FFFh	CoreSight / ITM
E800 6000h to E800 6FFFh	CoreSight / SWO
E800 7000h to E800 7FFFh	—
E800 8000h to E800 8FFFh	Cortex-R4 / CPU
E800 9000h to E800 9FFFh	Cortex-R4 / CTI
E800 A000h to E800 AFFFh	Cortex-R4 / ETM-R4
E800 B000h to E800 BFFFh	—
E800 C000h to E800 CFFFh	—
E800 D000h to E800 DFFFh	—
E800 E000h to E800 EFFFh	—
E800 F000h to E800 FFFFh	—

Table 10.6 lists the input/output pins of the debugging interface.

Table 10.6 Configuration of Pins for the Debugging Interface

Name	Pin Name	I/O	Functions
Test Clock	TCK	Input	Data is serially supplied from the Test Data Input (TDI) pin to this module, synchronized with this clock, and output from the Test Data Output (TDO) pin. In SWD mode, this pin works as the SWDCLK pin.
Test Mode Select	TMS	Input/Output	Changing the level of this signal, by synchronizing with TCK, will determine the status of the TAP (Test Access Port) control circuit. The protocol conforms to the JTAG standard (IEEE Std.1149.1). In SWD mode, this pin works as the SWDIO pin.
Test Reset	TRST#*1	Input	This pin accepts input asynchronously with TCK. When the level of this pin is Low, TAP (Test Access Port) is reset. When the levels of the TRST# pin and RES# pin are both Low, TAP and the debugging circuit are reset.
Test Data Input	TDI	Input	Changing the level of this pin, by synchronizing with TCK, will send data to this module. This pin can also be used as a general-purpose port. The initial function is TDI.
Test Data Output	TDO	Output	Reading the level of this pin, by synchronizing with TCK, will read data from this module. This pin can also be selected as the output pin of SWV. This pin can also be used as a general-purpose port. The initial function is TDO.
Trace Clock Output	TRACECLK	Output	This pin is an output pin of the clock used for synchronizing trace data.
Trace Enable Output	TRACECTL	Output	This pin is an output pin of the enable signal for trace control. This pin can also be selected as the output pin of SWV.
Trace Data Output	TRACEDATA7 to TRACEDATA0	Output	This pin is an output pin of trace data. TRACEDATA0 can be selected as an output pin of SWV.

Note 1. When you design a board on which an emulator can be used, set the TRST# pin to Low while the RES# pin is asserted at power up, and configure the board so that control is available by the TRST# pin only. When this pin is not used, fix it to Low, or connect it so that the same signal as that on the RES# pin is to be input. For details, see section 10.3.5, Reset Configuration and the Method of Connecting with the Emulator.

10.2 Register Descriptions

10.2.1 Debugging Interface Control Register (DBGIFCNT)

The DBGIFCNT register controls the pins used by the debugging interface.

Address(es): A00B 0A00h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SWVSEL[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	SWVSEL[1:0]	SWV Output Select	Selects the pin for SWV (Serial Wire Viewer) output. b1 b0 0 0: SWV output is not output. 0 1: SWV output is output from the TDO pin. 1 0: SWV output is output from the TRACEDATA0 pin. 1 1: SWV output is output from the TRACECTL pin.	R/W
b31 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/(W)

10.3 Operation

10.3.1 JTAG Interface

The JTAG interface uses five signals (TCK, TMS, TDO, TDI, and TRST#) to communicate with the host machine (PC) via the emulator. Figure 10.2 shows an example connection, which includes the RES# pin connection.

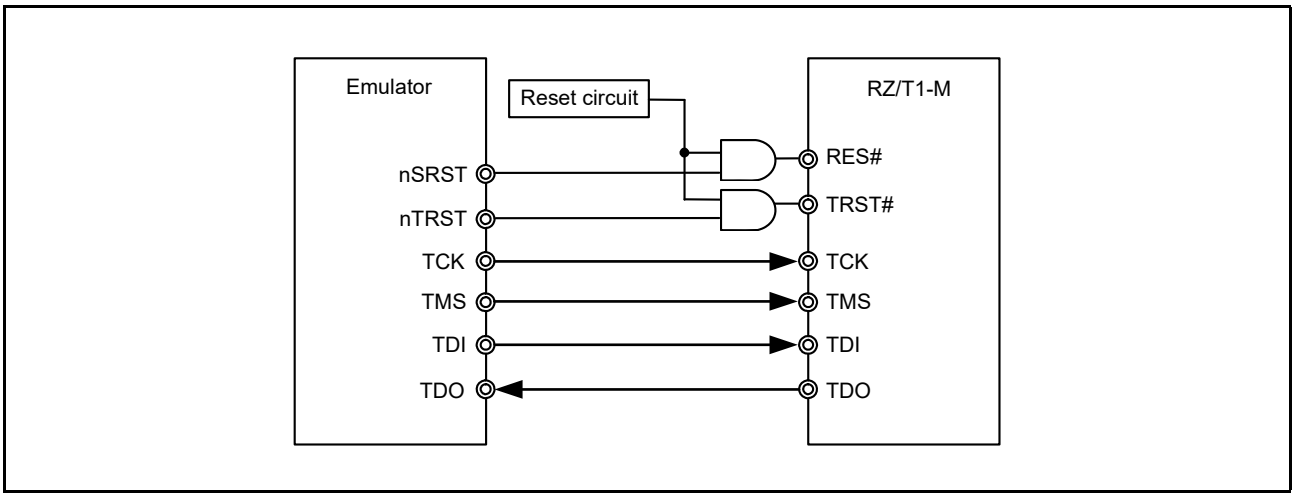


Figure 10.2 Example Connection of the JTAG Interface

10.3.2 SWD Interface

The SWD (Serial Wire Debug) interface uses two signals (SWCLK (TCK) and SWDIO (TMS)) to communicate with the host machine (PC) via the emulator. Figure 10.3 shows an example connection, which includes the RES# pin connection.

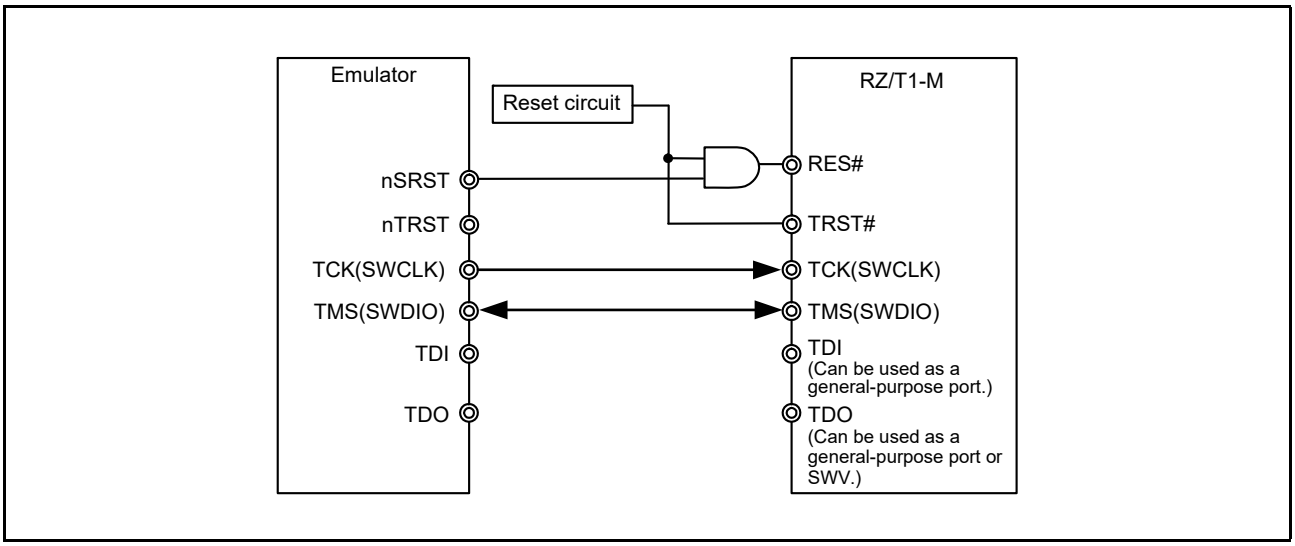


Figure 10.3 Example Connection of the SWD Interface

When the SWD interface is used for debugging, the TDI and TDO pins can be used as general-purpose ports. When you use these pins as general-purpose ports, perform pin settings by referring to section 17, Multi-Function Pin Controller (MPC).

Note: In the initial status of this LSI, the debugging interface is in JTAG mode. If you use the TDI and TDO pins as general-purpose ports and the emulator connection for debugging, switch the mode to SWD (Serial Wire Debug) mode by the control from the debugger, and then start debugging.

10.3.3 Trace Port Interface

The trace port interface uses ten signals (TRACECLK, TRACECTL, and TRACEDATA7 to TRACEDATA0) to output trace information. Information about branch instructions of the executed program (obtained by the ETM (Embedded Trace Macrocell) trace) is output from the trace port interface. After the debugger complements the information, you can know the branch source and destination at the time a branch occurred. For details on trace information, see the manual of each emulator vendor.

Only DDR clocking mode is supported for the synchronization relationship between the TRACECLK pin and TRACEDATA pin.

The maximum number of available TRACEDATA pins is 8. If the number of TRACEDATA pins is smaller than 8, the pins with smaller numbers are used (from TRACEDATA0). Set whether to connect the TRACECTL pin to TPA (Trace Port Analyzer), according to the specifications of the trace data transfer format of the TPA.

As the output frequency of the TRACECLK pin, 37.5 MHz (obtained by dividing the trace I/F clock (TCLK) by 2) can be set. For details, see section 7, Clock Generation Circuit.

In the initial status, different functions are assigned to the TRACECLK, TRACEDATA0 to TRACEDATA7, and TRACECTL pins. Perform pin settings by referring to section 17, Multi-Function Pin Controller (MPC).

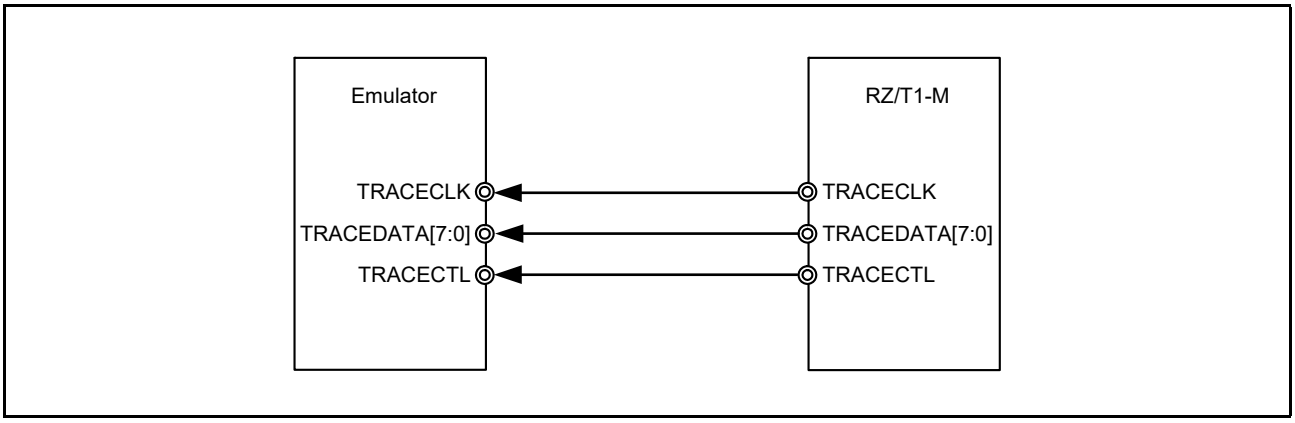


Figure 10.4 Example Connection of the Trace Port Interface

10.3.4 SWV Interface

The SWV (Serial Wire Viewer) interface is used to output trace information from the pin (TDO (SWV), TRACEDATA0 (SWV), or TRACECTL (SWV)) set by the DBGIFCNT register. When the JTAG interface is used, TDO (SWV) cannot be used. The SWV trace is a function that samples specified data at the specified sampling-cycle interval. For details on trace information, see the manual of each emulator vendor.

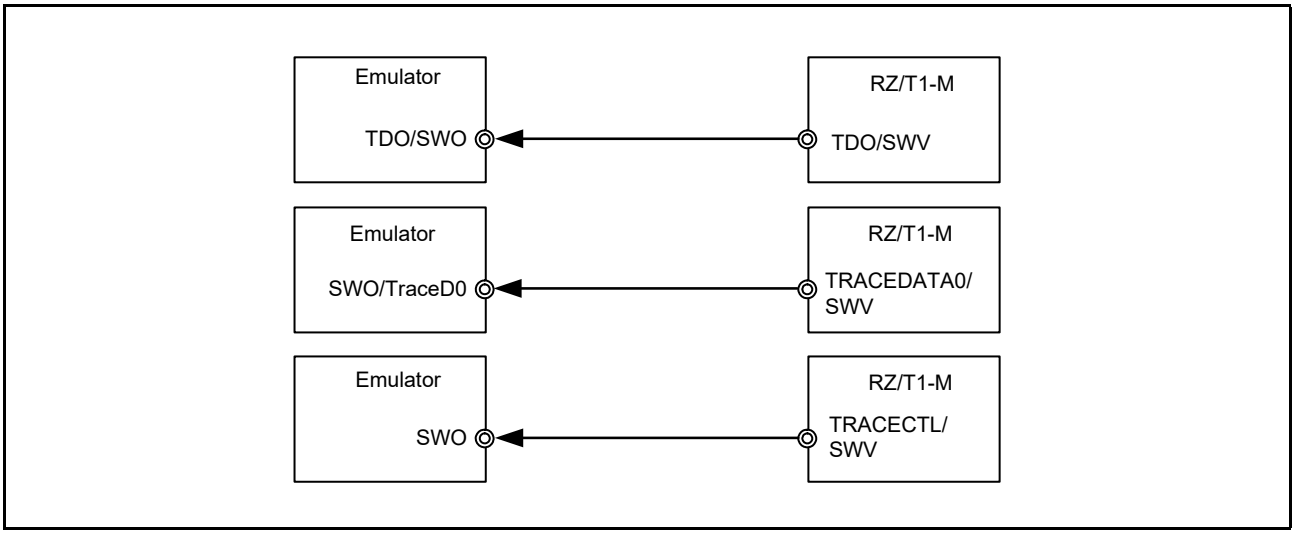


Figure 10.5 Example Connection of the SWV Interface

10.3.5 Reset Configuration and the Method of Connecting with the Emulator

When you design a board on which the emulator can be used, set the TRST# pin to Low while the RES# pin is asserted at power up. Also, configure the board so that control is available by the TRST# pin only.
When debugging is performed, if the initial level of the RES# and TRST# pins are both Low, the CPU and debugging section become the reset status. Then, setting the TRST# pin to High while the RES# pin is kept to Low will enable the debugging setting before CPU startup.
When the emulator is not connected, fix the TRST# pin to Low, or let the signal same as that on the RES# pin input to the TRST# pin.

10.3.5.1 Example Connection of the Emulator That Cannot Drive the nTRST Output to High

Figure 10.6 shows an example of connection circuit when an emulator that cannot drive the nTRST output to High is used. The TRST# pin is pulled up, and is asserted to Low by the emulator. To perform debugging settings before CPU startup, follow the timing chart for when the emulator is connected (see Figure 10.6).

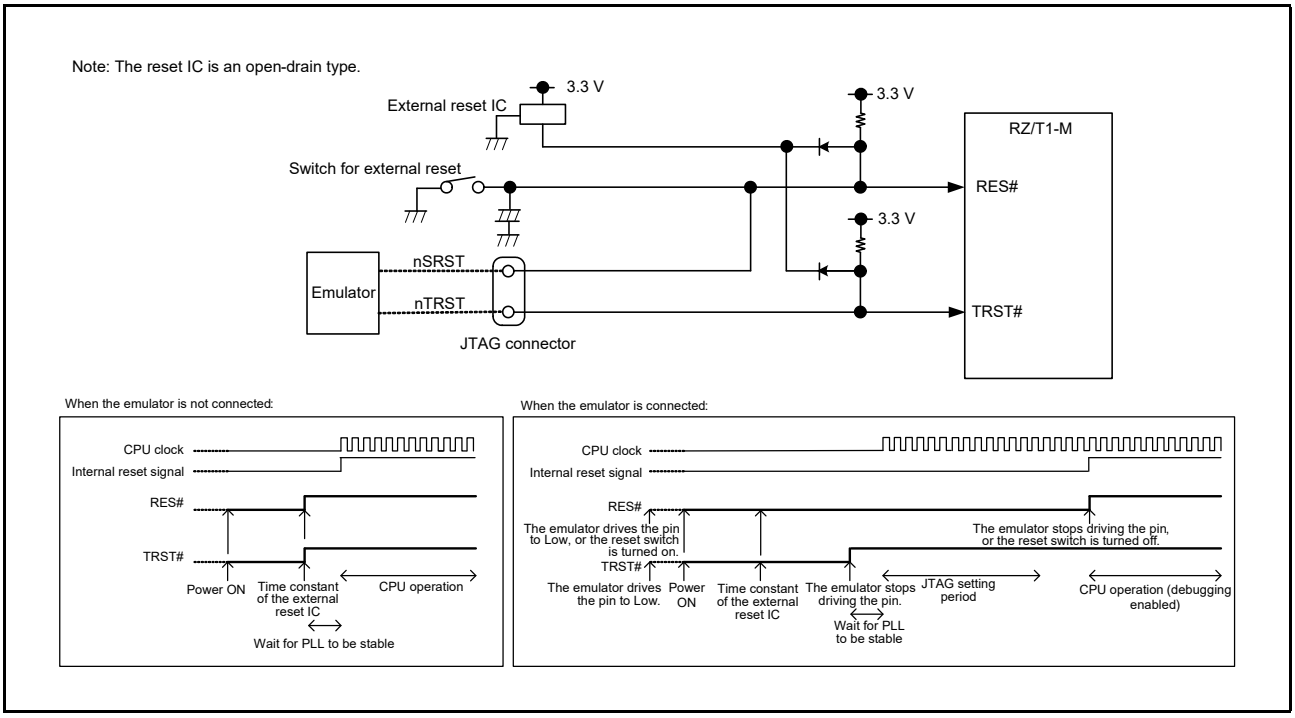


Figure 10.6 Example of Connection Circuit of an Emulator That Cannot Drive the nTRST Output to High

10.3.5.2 Example Connection of the Emulator That Can Drive the nTRST Output to High

Figure 10.7 shows an example of connection circuit when an emulator that can drive the nTRST output to High is used. The TRST# pin (High or Low) is controlled by the emulator. To perform debugging settings before CPU startup, follow the timing chart when the emulator is connected (see Figure 10.7).

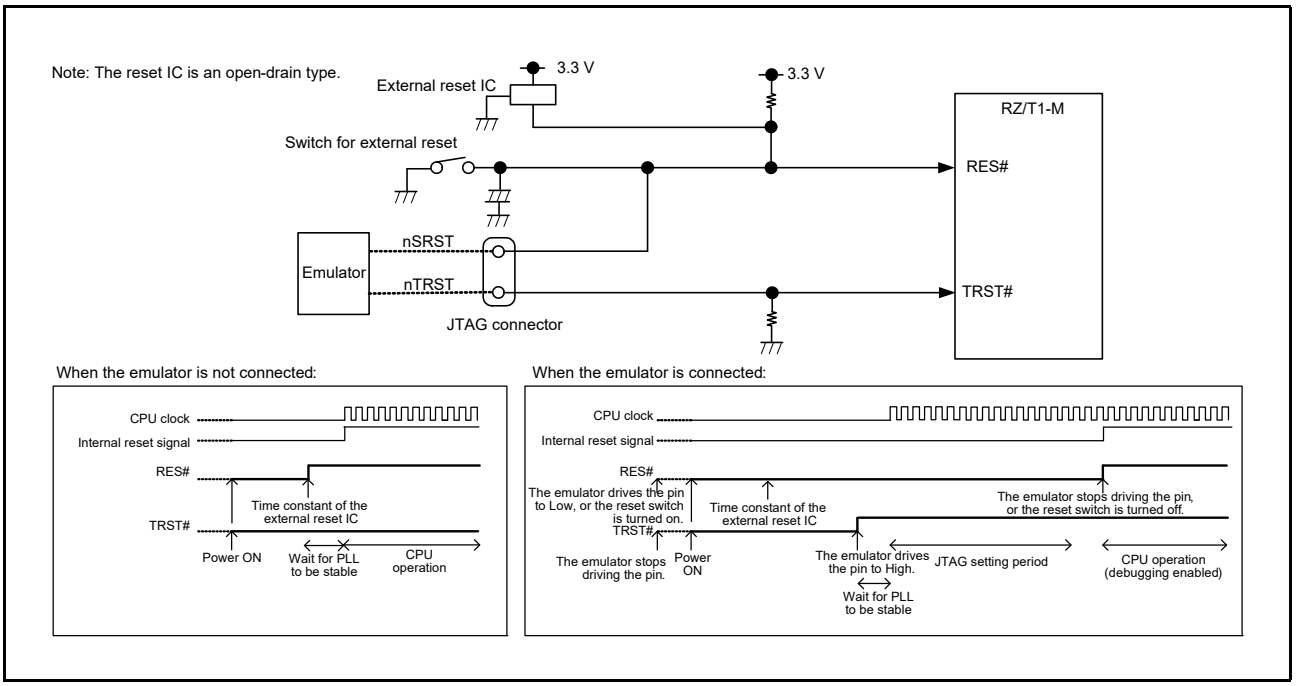


Figure 10.7 Example of Connection Circuit of an Emulator That Can Drive the nTRST Output to High

10.3.6 Handling of JTAG Pins When No Emulator Is Connected

If no emulator is connected, pin handling is required according to Table 10.7.

Table 10.7 Handling of JTAG Pins When No Emulator Is Connected

Pin Name	Handling
TCK	Pull down the pin.
TMS	Pull up the pin.
TDI	Pull up the pin (except when the port is used as a general-purpose port).
TDO	Open the pin (except when the port is used as a general-purpose port).
TRST#	Pull down the pin, or let the signal same as that on the RES# pin input.

10.3.7 Noise Reduction of the TRST# Pin

Analog delay noise reduction is performed for the TRST# pin. This measure against noise can remove noise that is within 100 ns at minimum.

10.3.8 Available Trace Functions

Table 10.8 lists the trace functions that are available via the respective debugging ports (trace port interface, SWV, and SWD or JTAG).

Table 10.8 Available Trace Functions

CPU Core	Debugging Port	Trace Functions
Cortex-R4 (CR4)	Trace Port interface	Full instruction tracing through the ETM of the Cortex-R4 Software tracing through the ITM among the CoreSight macrocells
	SWV	Only software tracing through the ITM among the CoreSight macrocells
	SWD/JTAG	The same information as for the trace port interface can be acquired via the ETB.

When using software tracing through the ITM among the CoreSight macrocells, access the ITM by software from the CPU. For the address range of the ITM within the CoreSight registers, see Table 10.5.

10.3.9 Access to the Main Bus

Access to the main bus by the DAP is via the Cortex-R4.

11. Register Write Protection Function

11.1 Overview

The register write protection function protects important registers from being overwritten in case a program runs out of control. The registers to be protected are set with the protect register (PRCR).

Table 11.1 shows the correspondence between the PRCR register bits and the registers to be protected.

Table 11.1 Correspondence between PRCR Register Bits and Registers to be Protected

PRCR Register	Registers to be Protected
PRC0 bit	• Registers related to the clock generation circuit: SCKCR, SCKCR2, PLL1CR, PLL1CR2, LOCOCR, OSTDCR
PRC1 bit	• Registers related to the low-power consumption functions: MSTPCRA, MSTPCRB, MSTPCRC, MSTPCRE, MSTPCRF • Reset-related registers: RSTSR0, SWRR1
PRC3 bit	• ATCM wait control register SYTATCMWAIT

11.2 Register Descriptions

11.2.1 Protect Register (PRCR)

The PRCR register controls writing to the protected registers.

Address(es): A00B 0B00h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PRKEY[7:0]								—	—	—	—	PRC3	—	PRC1	PRC0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	PRC0	Protect 0	Enables writing to the registers related to the clock generation circuit. 0: Write disabled 1: Write enabled	R/W
b1	PRC1	Protect 1	Enables writing to the registers related to low-power consumption functions and reset. 0: Write disabled 1: Write enabled	R/W
b2	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b3	PRC3	Protect 3	Enables writing to the ATCM wait control register. 0: Write disabled 1: Write enabled	R/W
b7 to 4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15 to b8	PRKEY[7:0]	PRC Key Code	These bits control permission and prohibition of writing to the PRCR register. To modify the PRCR register, write A5h to PRKEY[7:0]. When a value other than A5h is written to these bits, writing to the PRCR register has no effect.	R/(W)*1
b31 to b16	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. The value written to these bits is not retained. These bits are always read as 00h.

PRCi Bits (Protect i) (i = 0, 1, 3)

These bits enable or disable writing to the corresponding registers to be protected.

Setting the PRCi bits to 1 and 0 enables and disables writing to the corresponding registers to be protected, respectively.

12. Interrupt Controller (ICUA)

12.1 Overview

As an interrupt controller, the vector interrupt controller (VIC) for the Cortex-R4 is provided. The interrupt controller accepts interrupt requests from peripheral modules and external pins. An interrupt accepted by the interrupt controller is set either as an interrupt for conveying to the CPU (the Cortex-R4) or as an activating trigger signal for the DMAC. Table 12.1 lists interrupt specifications, and Figure 12.1 is a block diagram of the interrupt controller.

Table 12.1 Specifications of Interrupt Controller

Item		Description
Interrupts	Interrupt contact destinations	• Cortex-R4 • Two DMAC units (unit 0: 16ch., unit 1: 16ch.)
	Peripheral function interrupts	Interrupts from peripheral modules*1 Interrupt detection: Edge detection/level detection
	External pin interrupts	Interrupts from pins IRQ0 to IRQ4, IRQ6, IRQ7 Number of sources: 7 Interrupt detection: Low level/falling edge/rising edge/rising and falling edges. One of these detection methods can be set for each source. Digital noise filter function: Supported
	Interrupt priority level	For interrupts to the CPU, the priority level is set in 16 levels by the register.*2
	DMAC control	According to the interrupt source, DMAC can be activated. Switches interrupts from peripheral modules to DMA transfer completion interrupts.*3
Non-maskable interrupts	NMI pin interrupts	Interrupts from the NMI pin Interrupt detection: Falling edge/rising edge Digital noise filter function: Supported
	For Cortex-R4	The following sources can be allocated as the non-maskable high-speed interrupt (FIQ) source. • Non-maskable interrupts from ECM (Error Control Module) • Non-maskable interrupts from the NMI pin
Restoration from the sleep status		Restoration due to non-maskable interrupt and all unmasked interrupt sources

Note 1. According to interrupt contact destinations, interrupt sources differ. For details on activation sources, see Table 12.3, Cortex-R4/DMAC Interrupt Vector Table .

Note 2. The 16 priority levels are valid for all sources with CR4 (VIC) vector numbers 1 to 255. Interrupt sources with CR4 (VIC) vector numbers 256 and later have priority lower than sources for vector numbers 1 to 255. For details, see section 12.4.6.1, Restrictions on VIC Priority Levels.

Note 3. When an interrupt signal is selected as the source for activating the DMAC, generation of the interrupt signal activates the DMAC but branching to interrupt handling does not proceed at this time. The DMAC generates a transfer completion interrupt when it completes the data transfer. For details, see section 12.3.1, Selecting Interrupt Request Destinations.

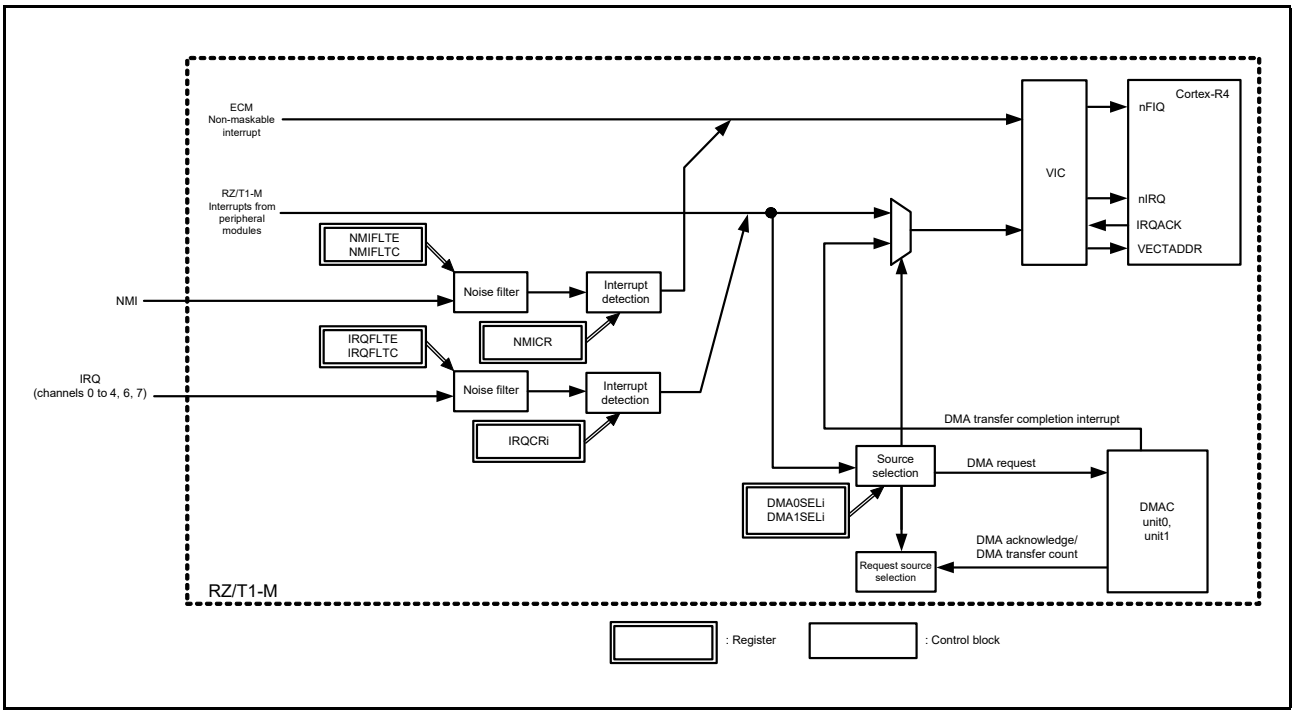


Figure 12.1 Block Diagram of Interrupt Controller

Table 12.2 lists input/output pins that are used by interrupt controllers.

Table 12.2 Input/output Pins for Interrupt Controllers

Pin Name	I/O	Description
NMI	Input	Non-maskable interrupt request pin
IRQ0 to IRQ4, IRQ6, IRQ7	Input	Maskable interrupt request pin

12.2 Register Descriptions

12.2.1 IRQ Control Register i (IRQCRi) (i = 0 to 4, 6, 7)

The IRQCRi register sets the method for detecting the external pin interrupt source (IRQ0 to IRQ4, IRQ6, IRQ7).

Address(es): ICU.IRQCR0 A009 4200h, ICU.IRQCR1 A009 4204h, ICU.IRQCR2 A009 4208h, ICU.IRQCR3 A009 420Ch, ICU.IRQCR4 A009 4210h, ICU.IRQCR6 A009 4218h, ICU.IRQCR7 A009 421Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	IRQMD[1:0]		—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b3, b2	IRQMD[1:0]	IRQ Detection Sense Select	b3 b2 0 0: Low level 0 1: Falling edge 1 0: Rising edge 1 1: Rising and falling edges	R/W
b31 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

IRQMD[1:0] Bits (IRQ Detection Sense Select)

These bits set the method for detecting the external pin interrupt source (IRQ0 to IRQ4, IRQ6, IRQ7).

For details on the methods for detecting external pin interrupt sources, see section 12.3.3, External Pin Interrupts.

Note: The same detection method should be set to the PLSn register of the VIC.

12.2.2 IRQ Pin Digital Noise Filter Enable Register (IRQFLTE)

The IRQFLTE register sets whether to use the digital noise filter for the external pin interrupt source (IRQ0 to IRQ4, IRQ6, IRQ7).

Address(es): ICU.IRQFLTE A009 4240h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	FLTEN7	FLTEN6	—	FLTEN4	FLTEN3	FLTEN2	FLTEN1	FLTEN0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	FLTEN0	IRQ0 Digital Noise Filter Enable	0: Digital noise filter is disabled. 1: Digital noise filter is enabled.	R/W
b1	FLTEN1	IRQ1 Digital Noise Filter Enable		R/W
b2	FLTEN2	IRQ2 Digital Noise Filter Enable		R/W
b3	FLTEN3	IRQ3 Digital Noise Filter Enable		R/W
b4	FLTEN4	IRQ4 Digital Noise Filter Enable		R/W
b5	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b6	FLTEN6	IRQ6 Digital Noise Filter Enable	0: Digital noise filter is disabled. 1: Digital noise filter is enabled.	R/W
b7	FLTEN7	IRQ7 Digital Noise Filter Enable		R/W
b31 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

FLTENi Bit (IRQi Digital Noise Filter Enable) (i = 0 to 4, 6, 7)

This bit enables the digital noise filter used for the external pin interrupt source (IRQ0 to IRQ4, IRQ6, IRQ7).

When this bit is set to 1, the digital noise filter is enabled. When it is cleared to 0, the digital noise filter function is disabled.

The IRQi pin level is sampled at the sampling clock cycle specified with the IRQFLTC.FCLKSELi[1:0] bits. When the sampled level matches three times, the output level from the digital filter changes.

For details on the digital noise filter, see section 12.3.2, Digital Noise Filter.

12.2.3 IRQ Pin Digital Noise Filter Setting Register (IRQFLTC)

The IRQFLTC register sets the digital noise filter sampling clock for external pin interrupt request pins (IRQ0 to IRQ4, IRQ6, IRQ7).

Address(es): ICU.IRQFLTC A009 4244h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	FCLKSEL7 [1:0]		FCLKSEL6 [1:0]		—	—	FCLKSEL4 [1:0]		FCLKSEL3 [1:0]		FCLKSEL2 [1:0]		FCLKSEL1 [1:0]		FCLKSEL0 [1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	FCLKSEL0[1:0]	IRQ0 Digital Noise Filter Sampling Clock Setting	Odd b Even b 0 0: PCLKB 0 1: PCLKB/8 1 0: PCLKB/32 1 1: PCLKB/64	R/W
b3, b2	FCLKSEL1[1:0]	IRQ1 Digital Noise Filter Sampling Clock Setting		R/W
b5, b4	FCLKSEL2[1:0]	IRQ2 Digital Noise Filter Sampling Clock Setting		R/W
b7, b6	FCLKSEL3[1:0]	IRQ3 Digital Noise Filter Sampling Clock Setting		R/W
b9, b8	FCLKSEL4[1:0]	IRQ4 Digital Noise Filter Sampling Clock Setting		R/W
b11, b10	—	Reserved		R/W
b13, b12	FCLKSEL6[1:0]	IRQ6 Digital Noise Filter Sampling Clock Setting		R/W
b15, b14	FCLKSEL7[1:0]	IRQ7 Digital Noise Filter Sampling Clock Setting		R/W
b31 to b16	—	Reserved		R/W

FCLKSELi[1:0] Bits (IRQi Digital Noise Filter Sampling Clock Setting) (i = 0 to 4, 6, 7)

These bits select the digital noise filter sampling clock for external pin interrupt request pins (IRQ0 to IRQ4, IRQ6, IRQ7).

The sampling clock cycle can be selected from the PCLKB (every cycle), PCKLB/8 (once every eight cycles), PCKLB/32 (once every 32 cycles), and PCKLB/64 (once every 64 cycles).

For details on the digital noise filter, see section 12.3.2, Digital Noise Filter. Note that even if the digital noise filter is disabled, PCLKB of which interrupts are to be sampled does not stop.

12.2.4 Non-maskable Interrupt Status Register (NMISR)

The NMISR register monitors the status of non-maskable interrupt sources. Writing to this register is ignored.
For information on non-maskable interrupt requests from ECM, read ECMm error source status register m (ECMmESSTRm, m = 0 to 2) for ECM, and check the error source.
Before ending non-maskable interrupt handler processing, read the NMISR register, and check the occurrence status of other non-maskable interrupts. Make sure all bits of the NMISR register are cleared to 0 before ending the interrupt handler processing.

Address(es): ICU.NMISR A009 4248h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ECMST	NMIST
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	NMIST	NMI Status Flag	0: NMI pin interrupt is not requested. 1: NMI pin interrupt is requested.	R
b1	ECMST	ECM Error Status Flag	0: ECM non-maskable interrupt is not requested. 1: ECM non-maskable interrupt is requested.	R
b31 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R

NMIST Flag (NMI Status Flag)

This flag indicates whether NMI pin interrupts are requested.
The NMIST flag is read only, and it can be cleared to 0 with the NMICLR.NMICLR bit.
[Setting condition]

- When the edgeset for the NMICR.NMIMD bit is input for the NMI pin

[Clearing condition]

- When 1 is written to the NMICLR.NMICLR bit

ECMST Flag (ECM Error Status Flag)

This flag indicates whether ECM non-maskable interrupts are requested.
The ECMST flag is read only, and it can be cleared to 0 by the NMICLR.ECMCLR bit.
[Setting condition]

- When ECM non-maskable interrupts are generated

[Clearing condition]

- When 1 is written to the NMICLR.ECMCLR bit

12.2.5 Non-maskable Interrupt Status Clear Register (NMICLR)

The NMICLR register clears NMI or ECM non-maskable interrupt requests.

Address(es): ICU.NMICLR A009 424Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ECMCLR	NMICLR
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	NMICLR	NMI Clear	This bit is read as 0. Writing 1 clears the NMISR.NMIST flag. Writing 0 to this bit is disabled.	R/(W)*1
b1	ECMCLR	ECM Clear	This bit is read as 0. Writing 1 clears the NMISR.ECMST flag. Writing 0 to this bit is disabled.	R/(W)*1
b31 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Only 1 can be written.

12.2.6 NMI Pin Interrupt Control Register (NMICR)

The NMICR register sets the method for detecting the NMI pin interrupt.

Address(es): ICU.NMICR A009 4250h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	NMIMD	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b3	NMIMD	NMI Detection Setting	0: Falling edge 1: Rising edge	R/W
b31 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

NMIMD Bit (NMI Detection Setting)

This bit sets the method for detecting the NMI pin interrupt.

12.2.7 NMI Pin Digital Noise Filter Enable Register (NMIFLTE)

The NMIFLTE register sets whether to use the digital noise filter for NMI pin interrupts.

Address(es): ICU.NMIFLTE A009 4254h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	NFLTEN
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	NFLTEN	NMI Digital Noise Filter Enable	0: Digital noise filter is disabled. 1: Digital noise filter is enabled.	R/W
b31 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

NFLTEN Bit (NMI Digital Noise Filter Enable)

This bit enables the digital noise filter used for NMI pin interrupts.

When this bit is set to 1, the digital noise filter is enabled. When it is cleared to 0, the digital noise filter function is disabled.

The NMI pin level is sampled at the sampling clock cycle specified with the NMIFLTC.NFCLKSEL[1:0] bits. When the sampled level matches three times, the output level from the digital noise filter changes.

For details on the digital noise filter, see section 12.3.2, Digital Noise Filter.

12.2.8 NMI Pin Digital Noise Filter Setting Register (NMIFLTC)

The NMIFLTC register sets the digital noise filter sampling clock for the NMI pin interrupt.

Address(es): ICU.NMIFLTC A009 4258h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	NFCLKSEL [1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	NFCLKSEL[1:0]	NMI Digital Noise Filter Sampling Clock Setting	b1 b0 0 0: PCLKB 0 1: PCLKB/8 1 0: PCLKB/32 1 1: PCLKB/64	R/W
b31 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

NFCLKSEL[1:0] Bits (NMI Digital Noise Filter Sampling Clock Setting)

These bits set the digital noise filter sampling clock of the NMI pin interrupt.

The sampling clock cycle can be selected from the PCLKB (every cycle), PCKLB/8 (once every eight cycles), PCKLB/32 (once every 32 cycles), and PCKLB/64 (once every 64 cycles).

For details on the digital noise filter, see section 12.3.2, Digital Noise Filter. Note that if the digital noise filter is disabled, PCLKB of which interrupts are to be sampled does not stop.

12.3 Operation

12.3.1 Selecting Interrupt Request Destinations

Table 12.3, Cortex-R4/DMAC Interrupt Vector Table, is a list of the requesting sources and indicates the source for which the CPU or DMAC is selectable as the destination. When the CPU is selected as the destination, the processing currently in progress branches to the interrupt handling routine in response to the interrupt request. When the DMAC is selected as the destination, DMA transfer starts in response to the interrupt request signal. In this case, a DMAC transfer completion interrupt is generated on completion of the transfer. Do not select interrupt request destinations that do not have the letter Y in the given request destination column in Table 12.3, Cortex-R4/DMAC Interrupt Vector Table.
Figure 12.2 shows the flow of selecting an interrupt source when vector number m is allocated to channel N of unit 0 as the DMA source. Vector numbers selected in the DMA source select register are not connected to an interrupt controller (VIC), but they are connected as DMA transfer requests to the corresponding channels of one of the DMACs. On completion of the DMA transfer, the transfer completion interrupt signal for the given channel of the DMAC is connected as the trigger for interrupt handling by the routine indicated by vector number m for the VIC.
For example, when interrupt vector number 21 (compare match interrupt_ch.0 of CMT unit 0) is selected for IFC[7:0] of the DMAC unit 0 source select register 0 (DMA0SEL0), if this interrupt occurs, DMA transfer is requested on channel 0 of DMAC unit 0. After DMA transfer, if a DMA transfer completion interrupt is generated, the DMA transfer completion request for channel 0 of DMAC unit 0 is connected to the same interrupt vector number (21) for VIC.
If vector number m is not selected by the source select register, interrupts from external pins and peripheral modules are connected to the interrupt controller VIC for the CPU (Figure 12.3).

Note: When the DMAC is selected as the destination for an interrupt request with vector number m, the DMA transfer completion interrupt signal is conveyed to the interrupt controller as the interrupt for vector number m on completion of the DMA transfer. This means that the detection type for an interrupt with vector number m whose destination is set as the DMAC should always be edge-sensing regardless of the vector number.

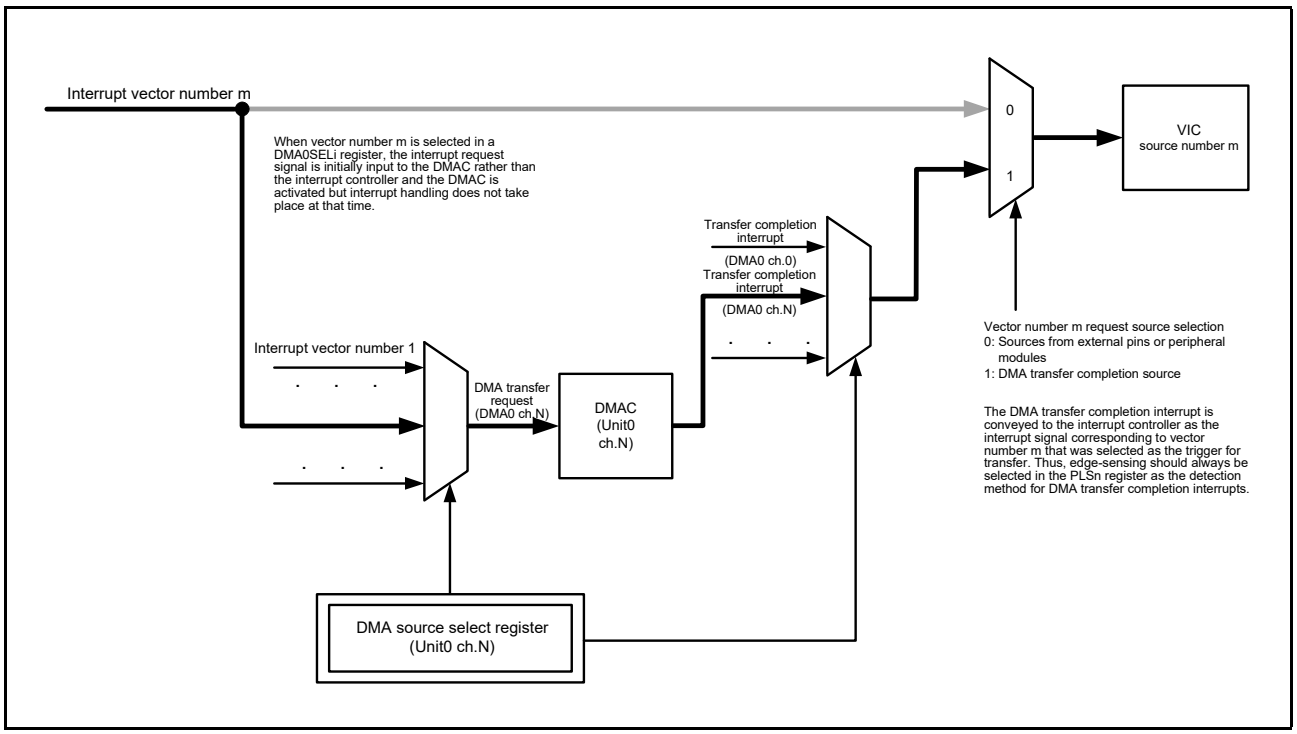


Figure 12.2 DMAC as the Interrupt Request Destination

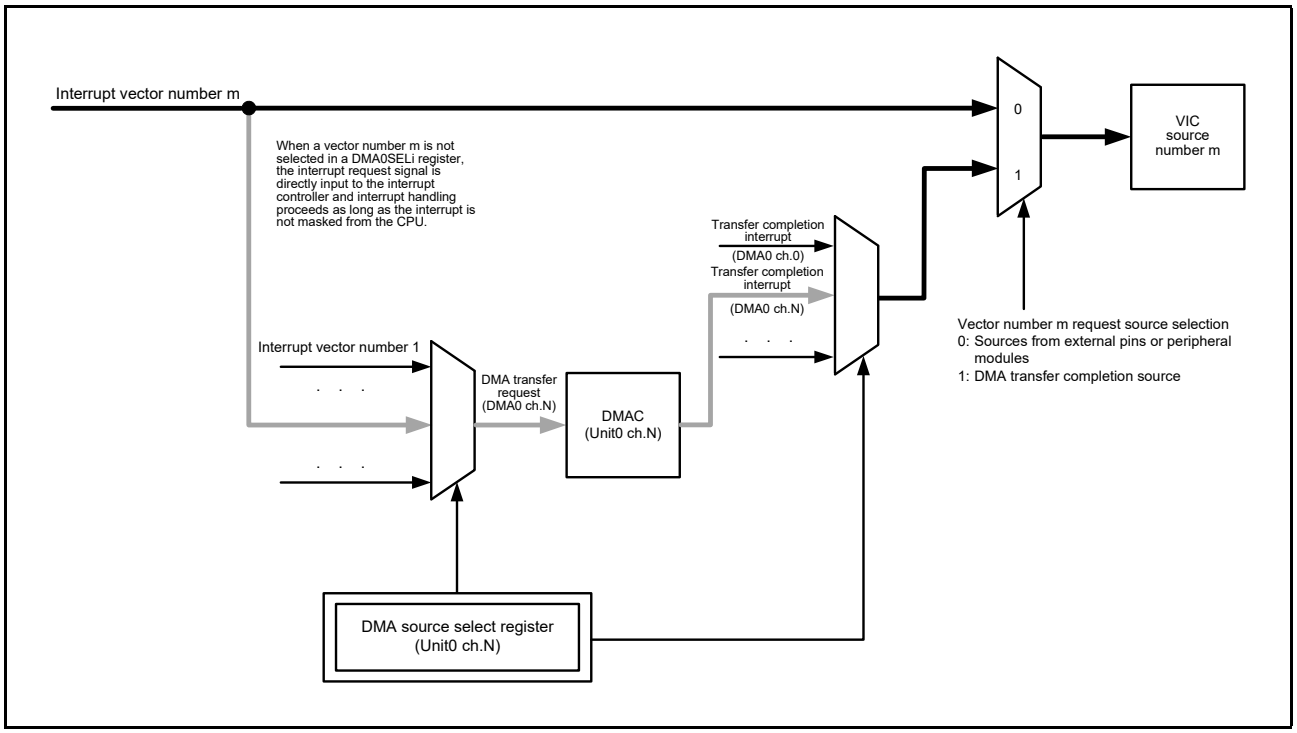


Figure 12.3 CPU (Interrupt Controller) as the Interrupt Request Destination

12.3.2 Digital Noise Filter

The digital noise filter function is provided for the external interrupt request IRQi pins (i = 0 to 4, 6, and 7) and NMI pin interrupts.

The digital noise filter samples input signals at the filter sampling clock (PCLKB) and pulses with levels that only match once or twice are removed.

To use the digital noise filter for the IRQi pins (IRQ0 to IRQ4, IRQ6, IRQ7), set the sampling clock cycle (PCLKB, PCLKB/8, PCLKB/32, or PCLKB/64) with the IRQFLTC.FCLKSELi[1:0] bits (IRQ0 to IRQ4, IRQ6, IRQ7), and set the IRQFLTE.FLTENi bits (IRQ0 to IRQ4, IRQ6, IRQ7) to 1.

To use the digital noise filter for the NMI pin interrupt, set the sampling clock cycle (PCLKB, PCLKB/8, PCLKB/32, or PCLKB/64) with the NMICR.NFCLKSEL[1:0] bits, and set the NMICR.NFLTEN bit to 1.

Figure 12.4 shows an example of digital noise filter operation.

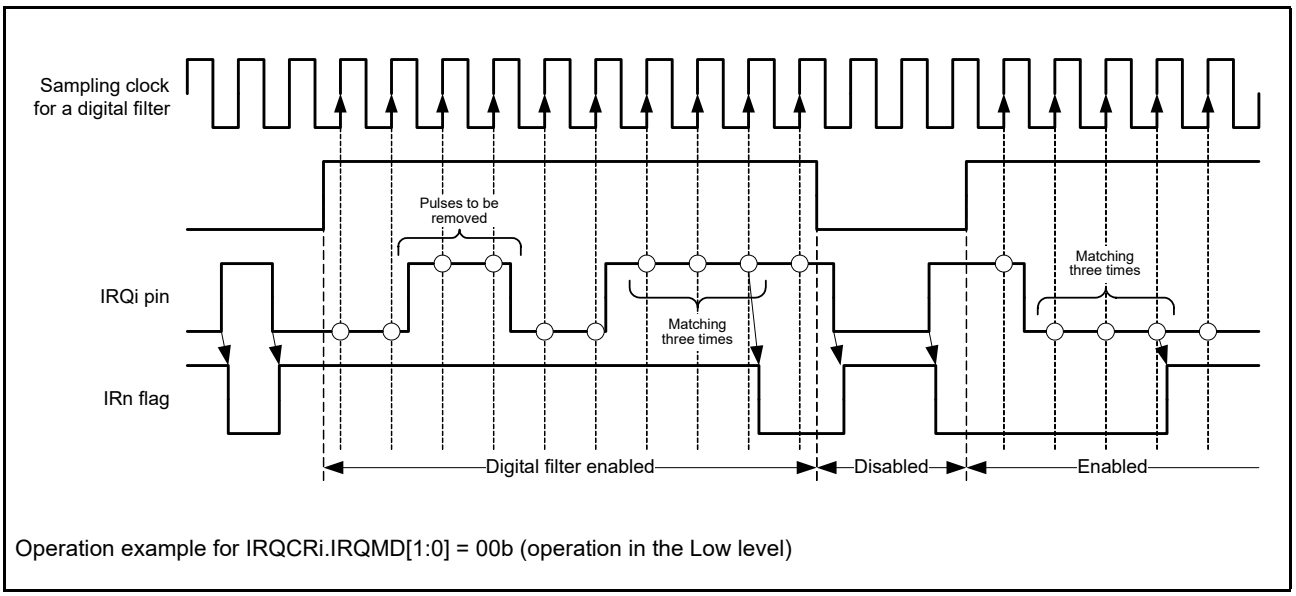


Figure 12.4 Digital Noise Filter Operation Example

12.3.3 External Pin Interrupts

The procedure for connecting an external pin interrupt to Cortex-R4 is shown below. For details on VIC, see section 12.4, Cortex-R4 Vector Interrupt Controller (VIC). To use the external pins at their falling edges or rising and falling edges, see section 12.5.1, Using “Falling-Edge” or “Rising and Falling Edges” Detection with the External Pin Interrupts.

[For IRQ pins]

1. Clear the applicable IENn bit to 0 (set the IECn bit).
2. Clear the IRQFLTE.FLTENi bit to 0.*1
3. Set the digital noise filter sampling clock with the IRQFLTC.FCLKSEL[1:0] bits.*1
4. Set the Pmn I/O select bit in the port direction register (PDR) of the I/O port to 10b (input).
5. Set the I/O port (PmnPFS.ISEL bit).
6. Set the method of detection with the IRQCRi.IRQMD[1:0] bits.
7. Set the IRQFLTE.FLTENi bit to 1.*1
8. Set the applicable PICn register to 1 (when an edge is detected).
9. Set the applicable IENn bit to 1.

Note 1. This setting is only required when the digital noise filter is to be used.

12.3.4 NMI Pin Interrupts

Pins that can be used as NMI pins serve as general I/O ports after a reset. To use them as NMI pins, the following procedure is required.
Note that setting these pins to serve as general I/O ports after setting them to serve as the NMI pins is prohibited. To use the NMI pins at their falling edges, see section 12.5.2, Using Falling-Edge Detection with the NMI Pin.

1. Set the NMIFLTE.NFLTEN bit to 0.*1
2. Set the sampling clock of the digital noise filter with the NMIFLTC.NFCLKSEL[1:0] bits.*1
3. Set edge detection with the NMICR.NMIMD bit.
4. Set the NMICLR.NMICLR bit to 1 and clear the NMISR.NMIST flag to 0.
5. Set the NMIFLTE.NFLTEN bit to 1.*1
6. Set the P35 I/O select bit in the port direction register (PDR) of the I/O port to 10b (input).
7. Set the I/O port (P35PFS.ISEL bit) and confirm the setting.

Note 1. This setting is only required when the digital noise filter is to be used.

12.4 Cortex-R4 Vector Interrupt Controller (VIC)

12.4.1 Overview

The RZ/T1-M has the vector interrupt controller (VIC) to control interrupts for Cortex-R4. Non-maskable interrupt requests from the NMI pin or ECM are treated as FIQ interrupts and are always accepted at high-speed. Interrupts from external pins other than the NMI pin and those from on-chip peripheral modules are accepted as IRQ interrupts (maskable interrupts). The vector addresses of the individual IRQ interrupt sources are stored in the interrupt address storage registers (VADn). When an IRQ interrupt occurs, the interrupt controller provides the Cortex-R4 with the address in VADn as the destination for branching, so the program counter directly branches to the address set in VADn.

12.4.2 Register Descriptions

12.4.2.1 IRQ Status Register n (IRQSn) (n = 0 to 9)

The IRQSn (n = 0 to 9) register indicates the interrupt status after IRQ interrupt mask. This register is enabled when an interrupt is enabled (IENn = 1). Interrupt status is not reflected when an interrupt is disabled (IENn = 0).
This register can only be read in 32-bit units.
Before completing the level interrupt, use the register to make sure no interrupt is requested. (See section 12.4.4.3, (2) IRQ Interrupt (Level interrupt)).

- IRQS0

Address(es): VIC.IRQS0 A001 0000h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IRQ31	IRQ30	IRQ29	IRQ28	IRQ27	IRQ26	IRQ25	IRQ24	IRQ23	IRQ22	IRQ21	IRQ20	IRQ19	IRQ18	IRQ17	IRQ16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IRQ15	IRQ14	IRQ13	IRQ12	IRQ11	IRQ10	IRQ9	IRQ8	IRQ7	IRQ6	IRQ5	IRQ4	IRQ3	IRQ2	IRQ1	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	This bit is read as 0.	R
b31 to b1	IRQ[31:1]	Interrupt Status Flag	0: IRQ interrupt is not requested. 1: IRQ interrupt is requested.	R

- IRQS1

Address(es): VIC.IRQS1 A001 0004h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IRQ63	IRQ62	IRQ61	IRQ60	IRQ59	IRQ58	IRQ57	IRQ56	IRQ55	IRQ54	IRQ53	IRQ52	IRQ51	IRQ50	IRQ49	IRQ48
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IRQ47	IRQ46	IRQ45	IRQ44	IRQ43	IRQ42	IRQ41	IRQ40	IRQ39	IRQ38	IRQ37	IRQ36	IRQ35	IRQ34	IRQ33	IRQ32
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IRQ[63:32]	Interrupt Status Flag	0: IRQ interrupt is not requested. 1: IRQ interrupt is requested.	R

IRQi Flag (Interrupt Status Flag) (i = 1 to 63)

This flag indicates the interrupt status after interrupt mask by the IEN registers.

- IRQS2

Address(es): VIC.IRQS2 A001 0008h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IRQ95	IRQ94	IRQ93	IRQ92	IRQ91	IRQ90	IRQ89	IRQ88	IRQ87	IRQ86	IRQ85	IRQ84	IRQ83	IRQ82	IRQ81	IRQ80
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IRQ79	IRQ78	IRQ77	IRQ76	IRQ75	IRQ74	IRQ73	IRQ72	IRQ71	IRQ70	IRQ69	IRQ68	IRQ67	IRQ66	IRQ65	IRQ64
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IRQ[95:64]	Interrupt Status Flag	0: IRQ interrupt is not requested. 1: IRQ interrupt is requested.	R

- IRQS3

Address(es): VIC.IRQS3 A001 000Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IRQ127	IRQ126	IRQ125	IRQ124	IRQ123	IRQ122	IRQ121	IRQ120	IRQ119	IRQ118	IRQ117	IRQ116	IRQ115	IRQ114	IRQ113	IRQ112
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IRQ111	IRQ110	IRQ109	IRQ108	IRQ107	IRQ106	IRQ105	IRQ104	IRQ103	IRQ102	IRQ101	IRQ100	IRQ99	IRQ98	IRQ97	IRQ96
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IRQ[127:96]	Interrupt Status Flag	0: IRQ interrupt is not requested. 1: IRQ interrupt is requested.	R

IRQi Flag (Interrupt Status Flag) (i = 64 to 127)

This flag indicates the interrupt status after interrupt mask by the IEN registers.

- IRQS4

Address(es): VIC.IRQS4 A001 0010h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IRQ159	IRQ158	IRQ157	IRQ156	IRQ155	IRQ154	IRQ153	IRQ152	IRQ151	IRQ150	IRQ149	IRQ148	IRQ147	IRQ146	IRQ145	IRQ144
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IRQ143	IRQ142	IRQ141	IRQ140	IRQ139	IRQ138	IRQ137	IRQ136	IRQ135	IRQ134	IRQ133	IRQ132	IRQ131	IRQ130	IRQ129	IRQ128
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IRQ[159:128]	Interrupt Status Flag	0: IRQ interrupt is not requested. 1: IRQ interrupt is requested.	R

- IRQS5

Address(es): VIC.IRQS5 A001 0014h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IRQ191	IRQ190	IRQ189	IRQ188	IRQ187	IRQ186	IRQ185	IRQ184	IRQ183	IRQ182	IRQ181	IRQ180	IRQ179	IRQ178	IRQ177	IRQ176
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IRQ175	IRQ174	IRQ173	IRQ172	IRQ171	IRQ170	IRQ169	IRQ168	IRQ167	IRQ166	IRQ165	IRQ164	IRQ163	IRQ162	IRQ161	IRQ160
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IRQ[191:160]	Interrupt Status Flag	0: IRQ interrupt is not requested. 1: IRQ interrupt is requested.	R

IRQi Flag (Interrupt Status Flag) (i = 128 to 191)

This flag indicates the interrupt status after interrupt mask by the IEN registers.

- IRQS6

Address(es): VIC.IRQS6 A001 0018h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IRQ223	IRQ222	IRQ221	IRQ220	IRQ219	IRQ218	IRQ217	IRQ216	IRQ215	IRQ214	IRQ213	IRQ212	IRQ211	IRQ210	IRQ209	IRQ208
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IRQ207	IRQ206	IRQ205	IRQ204	IRQ203	IRQ202	IRQ201	IRQ200	IRQ199	IRQ198	IRQ197	IRQ196	IRQ195	IRQ194	IRQ193	IRQ192
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IRQ[223:192]	Interrupt Status Flag	0: IRQ interrupt is not requested. 1: IRQ interrupt is requested.	R

- IRQS7

Address(es): VIC.IRQS7 A001 001Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IRQ255	IRQ254	IRQ253	IRQ252	IRQ251	IRQ250	IRQ249	IRQ248	IRQ247	IRQ246	IRQ245	IRQ244	IRQ243	IRQ242	IRQ241	IRQ240
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IRQ239	IRQ238	IRQ237	IRQ236	IRQ235	IRQ234	IRQ233	IRQ232	IRQ231	IRQ230	IRQ229	IRQ228	IRQ227	IRQ226	IRQ225	IRQ224
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IRQ[255:224]	Interrupt Status Flag	0: IRQ interrupt is not requested. 1: IRQ interrupt is requested.	R

IRQi Flag (Interrupt Status Flag) (i = 192 to 255)

This flag indicates the interrupt status after interrupt mask by the IEN registers.

- IRQS8

Address(es): VIC.IRQS8 A001 1000h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IRQ287	IRQ286	IRQ285	IRQ284	IRQ283	IRQ282	IRQ281	IRQ280	IRQ279	IRQ278	IRQ277	IRQ276	IRQ275	IRQ274	IRQ273	IRQ272
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IRQ271	IRQ270	IRQ269	IRQ268	IRQ267	IRQ266	IRQ265	IRQ264	IRQ263	IRQ262	IRQ261	IRQ260	IRQ259	IRQ258	IRQ257	IRQ256
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IRQ[287:256]	Interrupt Status Flag	0: IRQ interrupt is not requested. 1: IRQ interrupt is requested.	R

- IRQS9

Address(es): VIC.IRQS9 A001 1004h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	IRQ294	IRQ293	IRQ292	IRQ291	IRQ290	IRQ289	IRQ288
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b6 to b0	IRQ[294:288]	Interrupt Status Flag	0: IRQ interrupt is not requested. 1: IRQ interrupt is requested.	R
b31 to b7	—	Reserved	These bits are read as 0.	R

IRQi Flag (Interrupt Status Flag) (i = 256 to 294)

This flag indicates the interrupt status after interrupt mask by the IEN registers.

12.4.2.2 Interrupt Input Status Register n (RAISn) (n = 0 to 9)

The RAISn (n = 0 to 9) register indicates the interrupt input status before IRQ (maskable) interrupt mask. The interrupt status is reflected to this register regardless of the IENn register setting (interrupt enabled or disabled).
This register can only be read in 32-bit units. The states of an interrupt source can be confirmed while the interrupt is disabled (the corresponding bit in IENn is 0) by, for example, polling the source bit (see section 12.4.4.6, Handling IRQ Interrupt Source Conditions by Polling).

- RAIS0

Address(es): VIC.RAIS0 A001 0040h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	RAI31	RAI30	RAI29	RAI28	RAI27	RAI26	RAI25	RAI24	RAI23	RAI22	RAI21	RAI20	RAI19	RAI18	RAI17	RAI16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RAI15	RAI14	RAI13	RAI12	RAI11	RAI10	RAI9	RAI8	RAI7	RAI6	RAI5	RAI4	RAI3	RAI2	RAI1	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	This bit is read as 0.	R
b31 to b1	RAI[31:1]	Interrupt Input Status Flag	0: Interrupt is not requested. 1: Interrupt is requested.	R

- RAIS1

Address(es): VIC.RAIS1 A001 0044h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	RAI63	RAI62	RAI61	RAI60	RAI59	RAI58	RAI57	RAI56	RAI55	RAI54	RAI53	RAI52	RAI51	RAI50	RAI49	RAI48
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RAI47	RAI46	RAI45	RAI44	RAI43	RAI42	RAI41	RAI40	RAI39	RAI38	RAI37	RAI36	RAI35	RAI34	RAI33	RAI32
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	RAI[63:32]	Interrupt Input Status Flag	0: Interrupt is not requested. 1: Interrupt is requested.	R

RAIi Flag (Interrupt Input Status Flag) (i = 1 to 63)

This flag indicates the interrupt request input status before interrupt mask.

- RAIS2

Address(es): VIC.RAIS2 A001 0048h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	RAI95	RAI94	RAI93	RAI92	RAI91	RAI90	RAI89	RAI88	RAI87	RAI86	RAI85	RAI84	RAI83	RAI82	RAI81	RAI80
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RAI79	RAI78	RAI77	RAI76	RAI75	RAI74	RAI73	RAI72	RAI71	RAI70	RAI69	RAI68	RAI67	RAI66	RAI65	RAI64
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	RAI[95:64]	Interrupt Input Status Flag	0: Interrupt is not requested. 1: Interrupt is requested.	R

- RAIS3

Address(es): VIC.RAIS3 A001 004Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	RAI127	RAI126	RAI125	RAI124	RAI123	RAI122	RAI121	RAI120	RAI119	RAI118	RAI117	RAI116	RAI115	RAI114	RAI113	RAI112
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RAI111	RAI110	RAI109	RAI108	RAI107	RAI106	RAI105	RAI104	RAI103	RAI102	RAI101	RAI100	RAI99	RAI98	RAI97	RAI96
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	RAI[127:96]	Interrupt Input Status Flag	0: Interrupt is not requested. 1: Interrupt is requested.	R

RAIi Flag (Interrupt Input Status Flag) (i = 64 to 127)

This flag indicates the interrupt request input status before interrupt mask.

- RAIS4

Address(es): VIC.RAIS4 A001 0050h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	RAI159	RAI158	RAI157	RAI156	RAI155	RAI154	RAI153	RAI152	RAI151	RAI150	RAI149	RAI148	RAI147	RAI146	RAI145	RAI144
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RAI143	RAI142	RAI141	RAI140	RAI139	RAI138	RAI137	RAI136	RAI135	RAI134	RAI133	RAI132	RAI131	RAI130	RAI129	RAI128
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	RAI[159:128]	Interrupt Input Status Flag	0: Interrupt is not requested. 1: Interrupt is requested.	R

- RAIS5

Address(es): VIC.RAIS5 A001 0054h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	RAI191	RAI190	RAI189	RAI188	RAI187	RAI186	RAI185	RAI184	RAI183	RAI182	RAI181	RAI180	RAI179	RAI178	RAI177	RAI176
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RAI175	RAI174	RAI173	RAI172	RAI171	RAI170	RAI169	RAI168	RAI167	RAI166	RAI165	RAI164	RAI163	RAI162	RAI161	RAI160
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	RAI[191:160]	Interrupt Input Status Flag	0: Interrupt is not requested. 1: Interrupt is requested.	R

RAIi Flag (Interrupt Input Status Flag) (i = 128 to 191)

This flag indicates the interrupt request input status before interrupt mask.

- RAIS6

Address(es): VIC.RAIS6 A001 0058h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	RAI223	RAI222	RAI221	RAI220	RAI219	RAI218	RAI217	RAI216	RAI215	RAI214	RAI213	RAI212	RAI211	RAI210	RAI209	RAI208
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RAI207	RAI206	RAI205	RAI204	RAI203	RAI202	RAI201	RAI200	RAI199	RAI198	RAI197	RAI196	RAI195	RAI194	RAI193	RAI192
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	RAI[223:192]	Interrupt Input Status Flag	0: Interrupt is not requested. 1: Interrupt is requested.	R

- RAIS7

Address(es): VIC.RAIS7 A001 005Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	RAI255	RAI254	RAI253	RAI252	RAI251	RAI250	RAI249	RAI248	RAI247	RAI246	RAI245	RAI244	RAI243	RAI242	RAI241	RAI240
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RAI239	RAI238	RAI237	RAI236	RAI235	RAI234	RAI233	RAI232	RAI231	RAI230	RAI229	RAI228	RAI227	RAI226	RAI225	RAI224
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	RAI[255:224]	Interrupt Input Status Flag	0: Interrupt is not requested. 1: Interrupt is requested.	R

RAIi Flag (Interrupt Input Status Flag) (i = 192 to 255)

This flag indicates the interrupt request input status before interrupt mask.

- RAIS8

Address(es): VIC.RAIS8 A001 1040h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	RAI287	RAI286	RAI285	RAI284	RAI283	RAI282	RAI281	RAI280	RAI279	RAI278	RAI277	RAI276	RAI275	RAI274	RAI273	RAI272
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RAI271	RAI270	RAI269	RAI268	RAI267	RAI266	RAI265	RAI264	RAI263	RAI262	RAI261	RAI260	RAI259	RAI258	RAI257	RAI256
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	RAI[287:256]	Interrupt Input Status Flag	0: Interrupt is not requested. 1: Interrupt is requested.	R

- RAIS9

Address(es): VIC.RAIS9 A001 1044h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	RAI294	RAI293	RAI292	RAI291	RAI290	RAI289	RAI288
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b6 to b0	RAI[294:288]	Interrupt Input Status Flag	0: Interrupt is not requested. 1: Interrupt is requested.	R
b31 to b7	—	Reserved	These bits are read as 0.	R

RAIi Flag (Interrupt Input Status Flag) (i = 256 to 294)

This flag indicates the interrupt request input status before interrupt mask.

12.4.2.3 Interrupt Enable Register n (IENn) (n = 0 to 9)

The IENn (n = 0 to 9) register enables or masks IRQ interrupts. When it is reset, all interrupt requests are masked. When a bit of this register is set to 1, it cannot be cleared to 0. To clear the bit to 0, use interrupt enable clear register n (IECn).

This register can only be read and written in 32-bit units.

- IEN0

Address(es): VIC.IEN0 A001 0080h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEN31	IEN30	IEN29	IEN28	IEN27	IEN26	IEN25	IEN24	IEN23	IEN22	IEN21	IEN20	IEN19	IEN18	IEN17	IEN16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEN15	IEN14	IEN13	IEN12	IEN11	IEN10	IEN9	IEN8	IEN7	IEN6	IEN5	IEN4	IEN3	IEN2	IEN1	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	This bit is read as 0.	R/W
b31 to b1	IEN[31:1]	Interrupt Request Enable	0: Interrupt is masked (disabled). 1: Interrupt is enabled.	R/W

- IEN1

Address(es): VIC.IEN1 A001 0084h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEN63	IEN62	IEN61	IEN60	IEN59	IEN58	IEN57	IEN56	IEN55	IEN54	IEN53	IEN52	IEN51	IEN50	IEN49	IEN48
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEN47	IEN46	IEN45	IEN44	IEN43	IEN42	IEN41	IEN40	IEN39	IEN38	IEN37	IEN36	IEN35	IEN34	IEN33	IEN32
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IEN[63:32]	Interrupt Request Enable	0: Interrupt is masked (disabled). 1: Interrupt is enabled.	R/W

IENi Bit (Interrupt Request Enable) (i = 1 to 63)

This bit specifies interrupt request enable settings. When the bit is enabled, it cannot be masked with the IENn register. Perform interrupt mask with the IECn register.

- IEN2

Address(es): VIC.IEN2 A001 0088h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEN95	IEN94	IEN93	IEN92	IEN91	IEN90	IEN89	IEN88	IEN87	IEN86	IEN85	IEN84	IEN83	IEN82	IEN81	IEN80
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEN79	IEN78	IEN77	IEN76	IEN75	IEN74	IEN73	IEN72	IEN71	IEN70	IEN69	IEN68	IEN67	IEN66	IEN65	IEN64
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IEN[95:64]	Interrupt Request Enable	0: Interrupt is masked (disabled). 1: Interrupt is enabled.	R/W

- IEN3

Address(es): VIC.IEN3 A001 008Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEN127	IEN126	IEN125	IEN124	IEN123	IEN122	IEN121	IEN120	IEN119	IEN118	IEN117	IEN116	IEN115	IEN114	IEN113	IEN112
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEN111	IEN110	IEN109	IEN108	IEN107	IEN106	IEN105	IEN104	IEN103	IEN102	IEN101	IEN100	IEN99	IEN98	IEN97	IEN96
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IEN[127:96]	Interrupt Request Enable	0: Interrupt is masked (disabled). 1: Interrupt is enabled.	R/W

IENi Bit (Interrupt Request Enable) (i = 64 to 127)

This bit specifies interrupt request enable settings. When the bit is enabled, it cannot be masked with the IENn register. Perform interrupt mask with the IECn register.

- IEN4

Address(es): VIC.IEN4 A001 0090h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEN159	IEN158	IEN157	IEN156	IEN155	IEN154	IEN153	IEN152	IEN151	IEN150	IEN149	IEN148	IEN147	IEN146	IEN145	IEN144
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEN143	IEN142	IEN141	IEN140	IEN139	IEN138	IEN137	IEN136	IEN135	IEN134	IEN133	IEN132	IEN131	IEN130	IEN129	IEN128
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IEN[159:128]	Interrupt Request Enable	0: Interrupt is masked (disabled). 1: Interrupt is enabled.	R/W

- IEN5

Address(es): VIC.IEN5 A001 0094h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEN191	IEN190	IEN189	IEN188	IEN187	IEN186	IEN185	IEN184	IEN183	IEN182	IEN181	IEN180	IEN179	IEN178	IEN177	IEN176
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEN175	IEN174	IEN173	IEN172	IEN171	IEN170	IEN169	IEN168	IEN167	IEN166	IEN165	IEN164	IEN163	IEN162	IEN161	IEN160
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IEN[191:160]	Interrupt Request Enable	0: Interrupt is masked (disabled). 1: Interrupt is enabled.	R/W

IENi Bit (Interrupt Request Enable) (i = 128 to 191)

This bit specifies interrupt request enable settings. When the bit is enabled, it cannot be masked with the IENn register. Perform interrupt mask with the IECn register.

- IEN6

Address(es): VIC.IEN6 A001 0098h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEN223	IEN222	IEN221	IEN220	IEN219	IEN218	IEN217	IEN216	IEN215	IEN214	IEN213	IEN212	IEN211	IEN210	IEN209	IEN208
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEN207	IEN206	IEN205	IEN204	IEN203	IEN202	IEN201	IEN200	IEN199	IEN198	IEN197	IEN196	IEN195	IEN194	IEN193	IEN192
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IEN[223:192]	Interrupt Request Enable	0: Interrupt is masked (disabled). 1: Interrupt is enabled.	R/W

- IEN7

Address(es): VIC.IEN7 A001 009Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEN255	IEN254	IEN253	IEN252	IEN251	IEN250	IEN249	IEN248	IEN247	IEN246	IEN245	IEN244	IEN243	IEN242	IEN241	IEN240
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEN239	IEN238	IEN237	IEN236	IEN235	IEN234	IEN233	IEN232	IEN231	IEN230	IEN229	IEN228	IEN227	IEN226	IEN225	IEN224
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IEN[255:224]	Interrupt Request Enable	0: Interrupt is masked (disabled). 1: Interrupt is enabled.	R/W

IENi Bit (Interrupt Request Enable) (i = 192 to 255)

This bit specifies interrupt request enable settings. When the bit is enabled, it cannot be masked with the IENn register. Perform interrupt mask with the IECn register.

- IEN8

Address(es): VIC.IEN8 A001 1080h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEN287	IEN286	IEN285	IEN284	IEN283	IEN282	IEN281	IEN280	IEN279	IEN278	IEN277	IEN276	IEN275	IEN274	IEN273	IEN272
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEN271	IEN270	IEN269	IEN268	IEN267	IEN266	IEN265	IEN264	IEN263	IEN262	IEN261	IEN260	IEN259	IEN258	IEN257	IEN256
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IEN[287:256]	Interrupt Request Enable	0: Interrupt is masked (disabled). 1: Interrupt is enabled.	R/W

- IEN9

Address(es): VIC.IEN9 A001 1084h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	IEN294	IEN293	IEN292	IEN291	IEN290	IEN289	IEN288
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b6 to b0	IEN[294:288]	Interrupt Request Enable	0: Interrupt is masked (disabled). 1: Interrupt is enabled.	R/W
b31 to b7	—	Reserved	These bits are always read as 0. When written, always write 0.	R/W

IENi Bit (Interrupt Request Enable) (i = 256 to 294)

This bit specifies interrupt request enable settings. When the bit is enabled, it cannot be masked with the IENn register. Perform interrupt mask with the IECn register.

12.4.2.4 Interrupt Enable Clear Register n (IECn) (n = 0 to 9)

The IECn (n = 0 to 9) register clears a bit of the IENn register, and masks (disables) the applicable interrupt request. This register can only be written in 32-bit units.

If the value of the IECn register is to be changed, do so while interrupts are disabled. To disable interrupts, set the I bit in the CPSR register of the Arm CPU to 1.

- IEC0

Address(es): VIC.IEC0 A001 00A0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEC31	IEC30	IEC29	IEC28	IEC27	IEC26	IEC25	IEC24	IEC23	IEC22	IEC21	IEC20	IEC19	IEC18	IEC17	IEC16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEC15	IEC14	IEC13	IEC12	IEC11	IEC10	IEC9	IEC8	IEC7	IEC6	IEC5	IEC4	IEC3	IEC2	IEC1	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	The write value should be 0.	W
b31 to b1	IEC[31:1]	Interrupt Request Clear	0: Nothing is changed. 1: The applicable bit of the IEN register of which interrupt is masked (disabled) is cleared to 0.	W

- IEC1

Address(es): VIC.IEC1 A001 00A4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEC63	IEC62	IEC61	IEC60	IEC59	IEC58	IEC57	IEC56	IEC55	IEC54	IEC53	IEC52	IEC51	IEC50	IEC49	IEC48
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEC47	IEC46	IEC45	IEC44	IEC43	IEC42	IEC41	IEC40	IEC39	IEC38	IEC37	IEC36	IEC35	IEC34	IEC33	IEC32
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IEC[63:32]	Interrupt Request Clear	0: Nothing is changed. 1: The applicable bit of the IEN register of which interrupt is masked (disabled) is cleared to 0.	W

IECi Bit (Interrupt Request Clear) (i = 1 to 63)

This bit specifies settings for masking (disabling) interrupt requests. When 1 is set to a bit, the same bit of the IEN register is cleared to 0, and the interrupt request is masked (disabled).

- IEC2

Address(es): VIC.IEC2 A001 00A8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEC95	IEC94	IEC93	IEC92	IEC91	IEC90	IEC89	IEC88	IEC87	IEC86	IEC85	IEC84	IEC83	IEC82	IEC81	IEC80
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEC79	IEC78	IEC77	IEC76	IEC75	IEC74	IEC73	IEC72	IEC71	IEC70	IEC69	IEC68	IEC67	IEC66	IEC65	IEC64
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IEC[95:64]	Interrupt Request Clear	0: Nothing is changed. 1: The applicable bit of the IEN register of which interrupt is masked (disabled) is cleared to 0.	W

- IEC3

Address(es): VIC.IEC3 A001 00ACh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEC127	IEC126	IEC125	IEC124	IEC123	IEC122	IEC121	IEC120	IEC119	IEC118	IEC117	IEC116	IEC115	IEC114	IEC113	IEC112
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEC111	IEC110	IEC109	IEC108	IEC107	IEC106	IEC105	IEC104	IEC103	IEC102	IEC101	IEC100	IEC99	IEC98	IEC97	IEC96
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IEC[127:96]	Interrupt Request Clear	0: Nothing is changed. 1: The applicable bit of the IEN register of which interrupt is masked (disabled) is cleared to 0.	W

IECi Bit (Interrupt Request Clear) (i = 64 to 127)

This bit specifies settings for masking (disabling) interrupt requests. When 1 is set to a bit, the same bit of the IEN register is cleared to 0, and the interrupt request is masked (disabled).

- IEC4

Address(es): VIC.IEC4 A001 00B0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEC159	IEC158	IEC157	IEC156	IEC155	IEC154	IEC153	IEC152	IEC151	IEC150	IEC149	IEC148	IEC147	IEC146	IEC145	IEC144
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEC143	IEC142	IEC141	IEC140	IEC139	IEC138	IEC137	IEC136	IEC135	IEC134	IEC133	IEC132	IEC131	IEC130	IEC129	IEC128
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IEC[159:128]	Interrupt Request Clear	0: Nothing is changed. 1: The applicable bit of the IEN register of which interrupt is masked (disabled) is cleared to 0.	W

- IEC5

Address(es): VIC.IEC5 A001 00B4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEC191	IEC190	IEC189	IEC188	IEC187	IEC186	IEC185	IEC184	IEC183	IEC182	IEC181	IEC180	IEC179	IEC178	IEC177	IEC176
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEC175	IEC174	IEC173	IEC172	IEC171	IEC170	IEC169	IEC168	IEC167	IEC166	IEC165	IEC164	IEC163	IEC162	IEC161	IEC160
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IEC[191:160]	Interrupt Request Clear	0: Nothing is changed. 1: The applicable bit of the IEN register of which interrupt is masked (disabled) is cleared to 0.	W

IECi Bit (Interrupt Request Clear) (i = 128 to 191)

This bit specifies settings for masking (disabling) interrupt requests. When 1 is set to a bit, the same bit of the IEN register is cleared to 0, and the interrupt request is masked (disabled).

- IEC6

Address(es): VIC.IEC6 A001 00B8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEC223	IEC222	IEC221	IEC220	IEC219	IEC218	IEC217	IEC216	IEC215	IEC214	IEC213	IEC212	IEC211	IEC210	IEC209	IEC208
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEC207	IEC206	IEC205	IEC204	IEC203	IEC202	IEC201	IEC200	IEC199	IEC198	IEC197	IEC196	IEC195	IEC194	IEC193	IEC192
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IEC[223:192]	Interrupt Request Clear	0: Nothing is changed. 1: The applicable bit of the IEN register of which interrupt is masked (disabled) is cleared to 0.	W

- IEC7

Address(es): VIC.IEC7 A001 00BCh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEC255	IEC254	IEC253	IEC252	IEC251	IEC250	IEC249	IEC248	IEC247	IEC246	IEC245	IEC244	IEC243	IEC242	IEC241	IEC240
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEC239	IEC238	IEC237	IEC236	IEC235	IEC234	IEC233	IEC232	IEC231	IEC230	IEC229	IEC228	IEC227	IEC226	IEC225	IEC224
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IEC[255:224]	Interrupt Request Clear	0: Nothing is changed. 1: The applicable bit of the IEN register of which interrupt is masked (disabled) is cleared to 0.	W

IECi Bit (Interrupt Request Clear) (i = 192 to 255)

This bit specifies settings for masking (disabling) interrupt requests. When 1 is set to a bit, the same bit of the IEN register is cleared to 0, and the interrupt request is masked (disabled).

- IEC8

Address(es): VIC.IEC8 A001 10A0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEC287	IEC286	IEC285	IEC284	IEC283	IEC282	IEC281	IEC280	IEC279	IEC278	IEC277	IEC276	IEC275	IEC274	IEC273	IEC272
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEC271	IEC270	IEC269	IEC268	IEC267	IEC266	IEC265	IEC264	IEC263	IEC262	IEC261	IEC260	IEC259	IEC258	IEC257	IEC256
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IEC[287:256]	Interrupt Request Clear	0: Nothing is changed. 1: The applicable bit of the IEN register of which interrupt is masked (disabled) is cleared to 0.	W

- IEC9

Address(es): VIC.IEC9 A001 10A4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	IEC294	IEC293	IEC292	IEC291	IEC290	IEC289	IEC288
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b6 to b0	IEC[294:288]	Interrupt Request Clear	0: Nothing is changed. 1: The applicable bit of the IEN register of which interrupt is masked (disabled) is cleared to 0.	W
b31 to b7	—	Reserved	The write value should be 0.	W

IECi Bit (Interrupt Request Clear) (i = 256 to 294)

This bit specifies settings for masking (disabling) interrupt requests. When 1 is set to a bit, the same bit of the IEN register is cleared to 0, and the interrupt request is masked (disabled).

12.4.2.5 Interrupt Detection Type Selection Register n (PLSn) (n = 0 to 9)

The PLSn (n = 0 to 9) register detects the edge or level for each interrupt input.
This register can only be read and written in 32-bit units.

- PLS0

Address(es): VIC.PLS0 A001 0100h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PLS31	PLS30	PLS29	PLS28	PLS27	PLS26	PLS25	PLS24	PLS23	PLS22	PLS21	PLS20	PLS19	PLS18	PLS17	PLS16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PLS15	PLS14	PLS13	PLS12	PLS11	PLS10	PLS9	PLS8	PLS7	PLS6	PLS5	PLS4	PLS3	PLS2	PLS1	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b31 to b1	PLS[31:1]	Interrupt Input Detection Type Selection	0: Detects the level. 1: Detects the edge.	R/W

- PLS1

Address(es): VIC.PLS1 A001 0104h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PLS63	PLS62	PLS61	PLS60	PLS59	PLS58	PLS57	PLS56	PLS55	PLS54	PLS53	PLS52	PLS51	PLS50	PLS49	PLS48
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PLS47	PLS46	PLS45	PLS44	PLS43	PLS42	PLS41	PLS40	PLS39	PLS38	PLS37	PLS36	PLS35	PLS34	PLS33	PLS32
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	PLS[63:32]	Interrupt Input Detection Type Selection	0: Detects the level. 1: Detects the edge.	R/W

PLSi Bit (Interrupt Input Detection Type Selection) (i = 1 to 63)

This bit selects the interrupt input detection type.
PLS[63:1] corresponds to vector numbers 63 to 1.

- PLS2

Address(es): VIC.PLS2 A001 0108h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PLS95	PLS94	PLS93	PLS92	PLS91	PLS90	PLS89	PLS88	PLS87	PLS86	PLS85	PLS84	PLS83	PLS82	PLS81	PLS80
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PLS79	PLS78	PLS77	PLS76	PLS75	PLS74	PLS73	PLS72	PLS71	PLS70	PLS69	PLS68	PLS67	PLS66	PLS65	PLS64
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	PLS[95:64]	Interrupt Input Detection Type Selection	0: Detects the level. 1: Detects the edge.	R/W

- PLS3

Address(es): VIC.PLS3 A001 010Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PLS127	PLS126	PLS125	PLS124	PLS123	PLS122	PLS121	PLS120	PLS119	PLS118	PLS117	PLS116	PLS115	PLS114	PLS113	PLS112
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PLS111	PLS110	PLS109	PLS108	PLS107	PLS106	PLS105	PLS104	PLS103	PLS102	PLS101	PLS100	PLS99	PLS98	PLS97	PLS96
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	PLS[127:96]	Interrupt Input Detection Type Selection	0: Detects the level. 1: Detects the edge.	R/W

PLSi Bit (Interrupt Input Detection Type Selection) (i = 64 to 127)

This bit selects the interrupt input detection type.

PLS[127:64] corresponds to vector numbers 127 to 64.

- PLS4

Address(es): VIC.PLS4 A001 0110h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PLS159	PLS158	PLS157	PLS156	PLS155	PLS154	PLS153	PLS152	PLS151	PLS150	PLS149	PLS148	PLS147	PLS146	PLS145	PLS144
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PLS143	PLS142	PLS141	PLS140	PLS139	PLS138	PLS137	PLS136	PLS135	PLS134	PLS133	PLS132	PLS131	PLS130	PLS129	PLS128
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	PLS[159:128]	Interrupt Input Detection Type Selection	0: Detects the level. 1: Detects the edge.	R/W

- PLS5

Address(es): VIC.PLS5 A001 0114h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PLS191	PLS190	PLS189	PLS188	PLS187	PLS186	PLS185	PLS184	PLS183	PLS182	PLS181	PLS180	PLS179	PLS178	PLS177	PLS176
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PLS175	PLS174	PLS173	PLS172	PLS171	PLS170	PLS169	PLS168	PLS167	PLS166	PLS165	PLS164	PLS163	PLS162	PLS161	PLS160
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	PLS[191:160]	Interrupt Input Detection Type Selection	0: Detects the level. 1: Detects the edge.	R/W

PLSi Bit (Interrupt Input Detection Type Selection) (i = 128 to 191)

This bit selects the interrupt input detection type.

PLS[191:128] corresponds to vector numbers 191 to 128.

- PLS6

Address(es): VIC.PLS6 A001 0118h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PLS223	PLS222	PLS221	PLS220	PLS219	PLS218	PLS217	PLS216	PLS215	PLS214	PLS213	PLS212	PLS211	PLS210	PLS209	PLS208
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PLS207	PLS206	PLS205	PLS204	PLS203	PLS202	PLS201	PLS200	PLS199	PLS198	PLS197	PLS196	PLS195	PLS194	PLS193	PLS192
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	PLS[223:192]	Interrupt Input Detection Type Selection	0: Detects the level. 1: Detects the edge.	R/W

- PLS7

Address(es): VIC.PLS7 A001 011Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PLS255	PLS254	PLS253	PLS252	PLS251	PLS250	PLS249	PLS248	PLS247	PLS246	PLS245	PLS244	PLS243	PLS242	PLS241	PLS240
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PLS239	PLS238	PLS237	PLS236	PLS235	PLS234	PLS233	PLS232	PLS231	PLS230	PLS229	PLS228	PLS227	PLS226	PLS225	PLS224
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	PLS[255:224]	Interrupt Input Detection Type Selection	0: Detects the level. 1: Detects the edge.	R/W

PLSi Bit (Interrupt Input Detection Type Selection) (i = 192 to 255)

This bit selects the interrupt input detection type.

PLS[255:192] corresponds to vector numbers 255 to 192.

- PLS8

Address(es): VIC.PLS8 A001 1100h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PLS287	PLS286	PLS285	PLS284	PLS283	PLS282	PLS281	PLS280	PLS279	PLS278	PLS277	PLS276	PLS275	PLS274	PLS273	PLS272
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PLS271	PLS270	PLS269	PLS268	PLS267	PLS266	PLS265	PLS264	PLS263	PLS262	PLS261	PLS260	PLS259	PLS258	PLS257	PLS256
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	PLS[287:256]	Interrupt Input Detection Type Selection	0: Detects the level. 1: Detects the edge.	R/W

- PLS9

Address(es): VIC.PLS9 A001 1104h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	PLS294	PLS293	PLS292	PLS291	PLS290	PLS289	PLS288
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b6 to b0	PLS[294:288]	Interrupt Input Detection Type Selection	0: Detects the level. 1: Detects the edge.	R/W
b31 to b7	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

PLSi Bit (Interrupt Input Detection Type Selection) (i = 256 to 294)

This bit selects the interrupt input detection type.

PLS[294:256] corresponds to vector numbers 294 to 256.

12.4.2.6 Edge Detection Bit Clear Register n (PICn) (n = 0 to 9)

If you detect an edge, the interrupt detection status is retained for each interrupt input bit (See section 12.4, Cortex-R4 Vector Interrupt Controller (VIC) and section 12.4.4.3, (3) IRQ Interrupt (Edge Interrupt)).
The PICn (n = 0 to 9) register clears the edge detection circuit for the interrupt input bit of which edge was detected to 0. This register can only be written in 32-bit units.

- PIC0

Address(es): VIC.PIC0 A001 0120h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PIC31	PIC30	PIC29	PIC28	PIC27	PIC26	PIC25	PIC24	PIC23	PIC22	PIC21	PIC20	PIC19	PIC18	PIC17	PIC16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PIC15	PIC14	PIC13	PIC12	PIC11	PIC10	PIC9	PIC8	PIC7	PIC6	PIC5	PIC4	PIC3	PIC2	PIC1	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	The write value should be 0.	W
b31 to b1	PIC[31:1]	Edge Detection Clear	0: Nothing is changed. 1: Clears edge detection.	W

- PIC1

Address(es): VIC.PIC1 A001 0124h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PIC63	PIC62	PIC61	PIC60	PIC59	PIC58	PIC57	PIC56	PIC55	PIC54	PIC53	PIC52	PIC51	PIC50	PIC49	PIC48
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PIC47	PIC46	PIC45	PIC44	PIC43	PIC42	PIC41	PIC40	PIC39	PIC38	PIC37	PIC36	PIC35	PIC34	PIC33	PIC32
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	PIC[63:32]	Edge Detection Clear	0: Nothing is changed. 1: Clears edge detection.	W

PICi Bit (Edge Detection Clear) (i = 1 to 63)

For interrupt requests of which edges were detected, this bit clears the edge detection circuit for each interrupt request.

- PIC2

Address(es): VIC.PIC2 A001 0128h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PIC95	PIC94	PIC93	PIC92	PIC91	PIC90	PIC89	PIC88	PIC87	PIC86	PIC85	PIC84	PIC83	PIC82	PIC81	PIC80
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PIC79	PIC78	PIC77	PIC76	PIC75	PIC74	PIC73	PIC72	PIC71	PIC70	PIC69	PIC68	PIC67	PIC66	PIC65	PIC64
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	PIC[95:64]	Edge Detection Clear	0: Nothing is changed. 1: Clears edge detection.	W

- PIC3

Address(es): VIC.PIC3 A001 012Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PIC127	PIC126	PIC125	PIC124	PIC123	PIC122	PIC121	PIC120	PIC119	PIC118	PIC117	PIC116	PIC115	PIC114	PIC113	PIC112
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PIC111	PIC110	PIC109	PIC108	PIC107	PIC106	PIC105	PIC104	PIC103	PIC102	PIC101	PIC100	PIC99	PIC98	PIC97	PIC96
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	PIC[127:96]	Edge Detection Clear	0: Nothing is changed. 1: Clears edge detection.	W

PICi Bit (Edge Detection Clear) (i = 64 to 127)

For interrupt requests of which edges were detected, this bit clears the edge detection circuit for each interrupt request.

- PIC4

Address(es): VIC.PIC4 A001 0130h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PIC159	PIC158	PIC157	PIC156	PIC155	PIC154	PIC153	PIC152	PIC151	PIC150	PIC149	PIC148	PIC147	PIC146	PIC145	PIC144
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PIC143	PIC142	PIC141	PIC140	PIC139	PIC138	PIC137	PIC136	PIC135	PIC134	PIC133	PIC132	PIC131	PIC130	PIC129	PIC128
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	PIC[159:128]	Edge Detection Clear	0: Nothing is changed. 1: Clears edge detection.	W

- PIC5

Address(es): VIC.PIC5 A001 0134h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PIC191	PIC190	PIC189	PIC188	PIC187	PIC186	PIC185	PIC184	PIC183	PIC182	PIC181	PIC180	PIC179	PIC178	PIC177	PIC176
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PIC175	PIC174	PIC173	PIC172	PIC171	PIC170	PIC169	PIC168	PIC167	PIC166	PIC165	PIC164	PIC163	PIC162	PIC161	PIC160
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	PIC[191:160]	Edge Detection Clear	0: Nothing is changed. 1: Clears edge detection.	W

PICi Bit (Edge Detection Clear) (i = 128 to 191)

For interrupt requests of which edges were detected, this bit clears the edge detection circuit for each interrupt request.

- PIC6

Address(es): VIC.PIC6 A001 0138h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PIC223	PIC222	PIC221	PIC220	PIC219	PIC218	PIC217	PIC216	PIC215	PIC214	PIC213	PIC212	PIC211	PIC210	PIC209	PIC208
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PIC207	PIC206	PIC205	PIC204	PIC203	PIC202	PIC201	PIC200	PIC199	PIC198	PIC197	PIC196	PIC195	PIC194	PIC193	PIC192
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	PIC[223:192]	Edge Detection Clear	0: Nothing is changed. 1: Clears edge detection.	W

- PIC7

Address(es): VIC.PIC7 A001 013Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PIC255	PIC254	PIC253	PIC252	PIC251	PIC250	PIC249	PIC248	PIC247	PIC246	PIC245	PIC244	PIC243	PIC242	PIC241	PIC240
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PIC239	PIC238	PIC237	PIC236	PIC235	PIC234	PIC233	PIC232	PIC231	PIC230	PIC229	PIC228	PIC227	PIC226	PIC225	PIC224
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	PIC[255:224]	Edge Detection Clear	0: Nothing is changed. 1: Clears edge detection.	W

PICi Bit (Edge Detection Clear) (i = 192 to 255)

For interrupt requests of which edges were detected, this bit clears the edge detection circuit for each interrupt request.

- PIC8

Address(es): VIC.PIC8 A001 1120h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PIC287	PIC286	PIC285	PIC284	PIC283	PIC282	PIC281	PIC280	PIC279	PIC278	PIC277	PIC276	PIC275	PIC274	PIC273	PIC272
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PIC271	PIC270	PIC269	PIC268	PIC267	PIC266	PIC265	PIC264	PIC263	PIC262	PIC261	PIC260	PIC259	PIC258	PIC257	PIC256
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	PIC[287:256]	Edge Detection Clear	0: Nothing is changed. 1: Clears edge detection.	W

- PIC9

Address(es): VIC.PIC9 A001 1124h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	PIC294	PIC293	PIC292	PIC291	PIC290	PIC289	PIC288
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b6 to b0	PIC[294:288]	Edge Detection Clear	0: Nothing is changed. 1: Clears edge detection.	W
b31 to b7	—	Reserved	The write value should be 0.	W

PICi Bit (Edge Detection Clear) (i = 256 to 294)

For interrupt requests of which edges were detected, this bit clears the edge detection circuit for each interrupt request.

12.4.2.7 Interrupt Priority Level Mask Register 0 (PRLM0)

The PRLM0 register controls mask for interrupts to the interrupt priority level.
When a bit of this register is set to 1, it cannot be cleared to 0. To clear the bit to 0, use the interrupt priority level mask clear register 0 (PRLC0).
This register can only be read and written in 32-bit units.

Address(es): VIC.PRLM0 A001 01C0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PRLM15	PRLM14	PRLM13	PRLM12	PRLM11	PRLM10	PRLM9	PRLM8	PRLM7	PRLM6	PRLM5	PRLM4	PRLM3	PRLM2	PRLM1	PRLM0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	PRLM[15:0]	Interrupt Priority Level Setting	0: Nothing is changed. 1: Mask the same priority level as the corresponding bit number.	R/W
b31 to b16	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

PRLMi Bit (Interrupt Priority Level Setting) (i = 0 to 15)

This is an interrupt priority level setting bit. For the sources with vector numbers 1 to 255, this bit sets mask of interrupts for the interrupt priority level.
The bit position of the register equals to the applicable priority level. When a bit is set to 1, the same priority level as the corresponding bit number is masked.

12.4.2.8 Interrupt Priority Level Mask Register 1 (PRLM1)

The PRLM1 register controls mask for interrupts to the interrupt priority level.
When a bit of this register is set to 1, it cannot be cleared to 0. To clear the bit to 0, use the interrupt priority level mask clear register 1 (PRLC1).
This register can only be read and written in 32-bit units.

Address(es): VIC.PRLM1 A001 11C0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PRLM15	PRLM14	PRLM13	PRLM12	PRLM11	PRLM10	PRLM9	PRLM8	PRLM7	PRLM6	PRLM5	PRLM4	PRLM3	PRLM2	PRLM1	PRLM0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	PRLM[15:0]	Interrupt Priority Level Setting	0: Nothing is changed. 1: Mask the same priority level as the corresponding bit number (PRLMi) + 16.	R/W
b31 to b16	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

PRLMi Bit (Interrupt Priority Level Setting) (i = 0 to 15)

This is an interrupt priority level setting bit. For the sources with vector numbers 256 to 294, this bit sets mask of interrupts for the interrupt priority level.
When a bit is set to 1, the same priority level as the corresponding bit number (PRLMi) + 16 is masked.

12.4.2.9 Interrupt Priority Level Mask Clear Register 0 (PRLC0)

The PRLC0 register clears each bit of the PRLM0 register to 0.
This register can only be written in 32-bit units.

Address(es): VIC.PRLC0 A001 01C4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PRLC 15	PRLC 14	PRLC 13	PRLC 12	PRLC 11	PRLC 10	PRLC 9	PRLC8	PRLC7	PRLC6	PRLC5	PRLC4	PRLC3	PRLC2	PRLC1	PRLC0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	PRLC[15:0]	Interrupt Priority Level Clear	0: Nothing is changed. 1: The bit corresponding to the bit number is cleared to 0.	W
b31 to b16	—	Reserved	The write value should be 0.	W

PRLCi Bit (Interrupt Priority Level Clear) (i = 0 to 15)

This is an interrupt priority level clear bit. For the sources with vector numbers 1 to 255, this bit clears interrupt priority level mask register 0 (PRLM0) to 0.
Once a bit is set to 1, the bit corresponding to the set bit number is cleared to 0.

12.4.2.10 Interrupt Priority Level Mask Clear Register 1 (PRLC1)

The PRLC1 register clears each bit of the PRLM1 register to 0.
The PRLC1 register can only be written in 32-bit units.

Address(es): VIC.PRLC1 A001 11C4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PRLC 15	PRLC 14	PRLC 13	PRLC 12	PRLC 11	PRLC 10	PRLC 9	PRLC8	PRLC7	PRLC6	PRLC5	PRLC4	PRLC3	PRLC2	PRLC1	PRLC0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	PRLC[15:0]	Interrupt Priority Level Clear	0: Nothing is changed. 1: The bit corresponding to the bit number is cleared to 0.	W
b31 to b16	—	Reserved	The write value should be 0.	W

PRLCi Bit (Interrupt Priority Level Clear) (i = 0 to 15)

This is an interrupt priority level clear bit. For the sources with vector numbers 256 to 294, this bit clears interrupt priority level mask register 1 (PRLM1) to 0.
Once the bit is set to 1, the bit corresponding to the set bit number is cleared to 0.

12.4.2.11 User Mode Enable Register 0 (UEN0)

This register is used to enable or disable access to the interrupt control register (VIC control registers except UEN0 and UEN1 registers) in privilege mode.
When a bus master cannot generate protection information correctly, set the UE bit to 1 to enable access to the interrupt control register in user mode. The initial value of the UE bit is 1 and access to the interrupt control register in user mode is enabled.
The register can be read in 32-bit units.
Unlike other registers, the register can be written in 32-bit units only in privilege mode.

Address VIC.UEN0 A001 01C8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b0	UE*1	Interrupt Control Register Access Selection	0: Disables access in user mode. Enables access only in privilege mode. 1: Enables access in user mode. Access to interrupt control register in both user and privilege mode is possible (initial value).	W
b31 to b1	—	Reserved	The write value should be 0.	W

Note 1. Write access to the register is only possible in privilege mode.

UE Bit (Interrupt Control Register Access Selection)

This bit enables or disables access to the interrupt control register for sources corresponding to vector numbers 1 to 255.

12.4.2.12 User Mode Enable Register 1 (UEN1)

This register is used to enable or disable access to the interrupt control register (VIC control registers except UEN0 and UEN1 registers) in privilege mode.

When a bus master cannot generate protection information correctly, set the UE bit to 1 to enable access to the interrupt control register in user mode. The initial value of the UE bit is 1 and access to the interrupt control register in user mode is enabled.

The register can be read in 32-bit units.

Unlike other registers, the register can be written in 32-bit units only in privilege mode.

Address VIC.UEN1 A001 11C8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b0	UE*1	Interrupt Control Register Access Selection	0: Disables access in user mode. Enables access only in privilege mode. 1: Enables access in user mode. Access to interrupt control register in both user and privilege mode is possible (initial value).	W
b31 to b1	—	Reserved	The write value should be 0.	W

Note 1. Write access to the register is only possible in privilege mode.

UE Bit (Interrupt Control Register Access Selection)

This bit enables or disables access to the interrupt control register for sources corresponding to vector numbers 256 to 294.

12.4.2.13 Interrupt Address Register (HVA0)

An arbitrary value must be written to the HVA0 register after being released from a reset in order to initialize the VIC. Also, an arbitrary value must be written to the HVA0 register at the end of an interrupt service routine (ISR). Writing to the HVA0 register causes the interrupt controller to recognize the completion of interrupt processing and clear the priority level of the stored interrupt. This leads to the processing of interrupts at the next priority level from that of the interrupt for which processing was just completed. The HVA0 register does not reflect values written to it.
Access to the register for any purpose other than initializing the VIC after it is released from a reset or ending interrupt processing is prohibited and attempting such access may result in incorrect interrupt operations.
This register can only be written in 32-bit units.

Address VIC.HVA0 A001 0200h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	HVA[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	HVA[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	HVA[31:0]	Interrupt Processing Ending Notification	These bits notify an end of interrupt processing (by writing an arbitrary value)	W

12.4.2.14 Interrupt Service Status Register n (ISSn) (n = 0 to 9)

The ISSn (n = 0 to 9) register indicates the service status of an IRQ interrupt.
This register stores information for which Cortex-R4 is executing or suspending an interrupt service routine (ISR).
This register can only be read in 32-bit units.

- ISS0

Address(es): VIC.ISS0 A001 0210h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISS31	ISS30	ISS29	ISS28	ISS27	ISS26	ISS25	ISS24	ISS23	ISS22	ISS21	ISS20	ISS19	ISS18	ISS17	ISS16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISS15	ISS14	ISS13	ISS12	ISS11	ISS10	ISS9	ISS8	ISS7	ISS6	ISS5	ISS4	ISS3	ISS2	ISS1	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	This bit is read as 0.	R
b31 to b1	ISS[31:1]	IRQ Interrupt Request Service Flag	0: The service is not executed. 1: The interrupt service routine (ISR) is being executed or suspended.	R

- ISS1

Address(es): VIC.ISS1 A001 0214h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISS63	ISS62	ISS61	ISS60	ISS59	ISS58	ISS57	ISS56	ISS55	ISS54	ISS53	ISS52	ISS51	ISS50	ISS49	ISS48
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISS47	ISS46	ISS45	ISS44	ISS43	ISS42	ISS41	ISS40	ISS39	ISS38	ISS37	ISS36	ISS35	ISS34	ISS33	ISS32
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	ISS[63:32]	IRQ Interrupt Request Service Flag	0: The service is not executed. 1: The interrupt service routine (ISR) is being executed or suspended.	R

ISSi Bit (IRQ Interrupt Request Service) (i = 1 to 63)

This flag indicates the service status of an IRQ interrupt request from vector numbers 1 to 63.

- ISS2

Address(es): VIC.ISS2 A001 0218h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISS95	ISS94	ISS93	ISS92	ISS91	ISS90	ISS89	ISS88	ISS87	ISS86	ISS85	ISS84	ISS83	ISS82	ISS81	ISS80
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISS79	ISS78	ISS77	ISS76	ISS75	ISS74	ISS73	ISS72	ISS71	ISS70	ISS69	ISS68	ISS67	ISS66	ISS65	ISS64
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	ISS[95:64]	IRQ Interrupt Request Service Flag	0: The service is not executed. 1: The interrupt service routine (ISR) is being executed or suspended.	R

- ISS3

Address(es): VIC.ISS3 A001 021Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISS127	ISS126	ISS125	ISS124	ISS123	ISS122	ISS121	ISS120	ISS119	ISS118	ISS117	ISS116	ISS115	ISS114	ISS113	ISS112
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISS111	ISS110	ISS109	ISS108	ISS107	ISS106	ISS105	ISS104	ISS103	ISS102	ISS101	ISS100	ISS99	ISS98	ISS97	ISS96
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	ISS[127:96]	IRQ Interrupt Request Service Flag	0: The service is not executed. 1: The interrupt service routine (ISR) is being executed or suspended.	R

ISSi Bit (IRQ Interrupt Request Service Flag) (i = 64 to 127)

This flag indicates the service status of an IRQ interrupt request from vector numbers 127 to 64.

- ISS4

Address(es): VIC.ISS4 A001 0220h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISS159	ISS158	ISS157	ISS156	ISS155	ISS154	ISS153	ISS152	ISS151	ISS150	ISS149	ISS148	ISS147	ISS146	ISS145	ISS144
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISS143	ISS142	ISS141	ISS140	ISS139	ISS138	ISS137	ISS136	ISS135	ISS134	ISS133	ISS132	ISS131	ISS130	ISS129	ISS128
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	ISS[159:128]	IRQ Interrupt Request Service Flag	0: The service is not executed. 1: The interrupt service routine (ISR) is being executed or suspended.	R

- ISS5

Address(es): VIC.ISS5 A001 0224h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISS191	ISS190	ISS189	ISS188	ISS187	ISS186	ISS185	ISS184	ISS183	ISS182	ISS181	ISS180	ISS179	ISS178	ISS177	ISS176
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISS175	ISS174	ISS173	ISS172	ISS171	ISS170	ISS169	ISS168	ISS167	ISS166	ISS165	ISS164	ISS163	ISS162	ISS161	ISS160
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	ISS[191:160]	IRQ Interrupt Request Service Flag	0: The service is not executed. 1: The interrupt service routine (ISR) is being executed or suspended.	R

ISSi Bit (IRQ Interrupt Request Service Flag) (i = 128 to 191)

This flag indicates the service status of an IRQ interrupt request from vector numbers 191 to 128.

- ISS6

Address(es): VIC.ISS6 A001 0228h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISS223	ISS222	ISS221	ISS220	ISS219	ISS218	ISS217	ISS216	ISS215	ISS214	ISS213	ISS212	ISS211	ISS210	ISS209	ISS208
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISS207	ISS206	ISS205	ISS204	ISS203	ISS202	ISS201	ISS200	ISS199	ISS198	ISS197	ISS196	ISS195	ISS194	ISS193	ISS192
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	ISS[223:192]	IRQ Interrupt Request Service Flag	0: The service is not executed. 1: The interrupt service routine (ISR) is being executed or suspended.	R

- ISS7

Address(es): VIC.ISS7 A001 022Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISS255	ISS254	ISS253	ISS252	ISS251	ISS250	ISS249	ISS248	ISS247	ISS246	ISS245	ISS244	ISS243	ISS242	ISS241	ISS240
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISS239	ISS238	ISS237	ISS236	ISS235	ISS234	ISS233	ISS232	ISS231	ISS230	ISS229	ISS228	ISS227	ISS226	ISS225	ISS224
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	ISS[255:224]	IRQ Interrupt Request Service Flag	0: The service is not executed. 1: The interrupt service routine (ISR) is being executed or suspended.	R

ISSi Bit (IRQ Interrupt Request Service Flag) (i = 192 to 255)

This flag indicates the service status of IRQ interrupt request from vector numbers 255 to 192.

- ISS8

Address(es): VIC.ISS8 A001 1210h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISS287	ISS286	ISS285	ISS284	ISS283	ISS282	ISS281	ISS280	ISS279	ISS278	ISS277	ISS276	ISS275	ISS274	ISS273	ISS272
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISS271	ISS270	ISS269	ISS268	ISS267	ISS266	ISS265	ISS264	ISS263	ISS262	ISS261	ISS260	ISS259	ISS258	ISS257	ISS256
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	ISS[287:256]	IRQ Interrupt Request Service Flag	0: The service is not executed. 1: The interrupt service routine (ISR) is being executed or suspended.	R

- ISS9

Address(es): VIC.ISS9 A001 1214h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

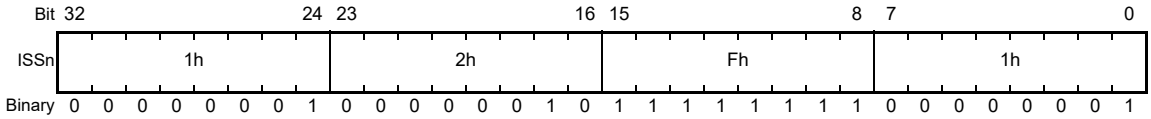
	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	ISS294	ISS293	ISS292	ISS291	ISS290	ISS289	ISS288
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b6 to b0	ISS[294:288]	IRQ Interrupt Request Service Flag	0: The service is not executed. 1: The interrupt service routine (ISR) is being executed or suspended.	R
b31 to b7	—	Reserved	These bits are read as 0.	R

ISSi Bit (IRQ Interrupt Request Service Flag) (i = 256 to 294)

This flag indicates the service status of IRQ interrupt request from vector numbers 294 to 256.

For example, if multiple interrupts are requested to the interrupt controller, the ISSn register shows the following status:



The figure indicates interrupts from vector numbers 24, 17, 15 to 8, and 0. Interrupt service routines (ISR) are serviced in descending order of priority which is set with the PRLm register. When the value of the PRLm is the same, the priority level of an interrupt with a smaller vector number is higher. When ISR finishes, the applicable bit of the ISSn register is cleared to 0, and then ISR that has the next highest priority level starts. In addition, if another interrupt is requested during ISR, the interrupt is also applied to this register.

12.4.2.15 Interrupt Service Current Register n (ISCn) (n = 0 to 9)

The ISCn (n = 0 to 9) register indicates the IRQ interrupt register with the highest priority level among IRQ interrupts which is set to 1 with interrupt service status register n (ISSn).
This register can only be read in 32-bit units.

- ISC0

Address(es): VIC.ISC0 A001 0230h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISC31	ISC30	ISC29	ISC28	ISC27	ISC26	ISC25	ISC24	ISC23	ISC22	ISC21	ISC20	ISC19	ISC18	ISC17	ISC16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISC15	ISC14	ISC13	ISC12	ISC11	ISC10	ISC9	ISC8	ISC7	ISC6	ISC5	ISC4	ISC3	ISC2	ISC1	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	This bit is read as 0.	R
b31 to b1	ISC[31:1]	IRQ Interrupt Request Service Flag	0: The priority level is not the highest, or the interrupt service routine (ISR) is not executed. 1: An interrupt with the highest priority	R

- ISC1

Address(es): VIC.ISC1 A001 0234h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISC63	ISC62	ISC61	ISC60	ISC59	ISC58	ISC57	ISC56	ISC55	ISC54	ISC53	ISC52	ISC51	ISC50	ISC49	ISC48
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISC47	ISC46	ISC45	ISC44	ISC43	ISC42	ISC41	ISC40	ISC39	ISC38	ISC37	ISC36	ISC35	ISC34	ISC33	ISC32
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	ISC[63:32]	IRQ Interrupt Request Service Flag	0: The priority level is not the highest, or the interrupt service routine (ISR) is not executed. 1: An interrupt with the highest priority	R

ISCi Bit (IRQ Interrupt Request Service Flag) (i = 1 to 63)

This flag indicates the service status of the IRQ interrupt request for interrupt service status register n (ISSn, n = 0 to 9).

- ISC2

Address(es): VIC.ISC2 A001 0238h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISC95	ISC94	ISC93	ISC92	ISC91	ISC90	ISC89	ISC88	ISC87	ISC86	ISC85	ISC84	ISC83	ISC82	ISC81	ISC80
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISC79	ISC78	ISC77	ISC76	ISC75	ISC74	ISC73	ISC72	ISC71	ISC70	ISC69	ISC68	ISC67	ISC66	ISC65	ISC64
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	ISC[95:64]	IRQ Interrupt Request Service Flag	0: The priority level is not the highest, or the interrupt service routine (ISR) is not executed. 1: An interrupt with the highest priority	R

- ISC3

Address(es): VIC.ISC3 A001 023Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISC127	ISC126	ISC125	ISC124	ISC123	ISC122	ISC121	ISC120	ISC119	ISC118	ISC117	ISC116	ISC115	ISC114	ISC113	ISC112
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISC111	ISC110	ISC109	ISC108	ISC107	ISC106	ISC105	ISC104	ISC103	ISC102	ISC101	ISC100	ISC99	ISC98	ISC97	ISC96
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	ISC[127:96]	IRQ Interrupt Request Service Flag	0: The priority level is not the highest, or the interrupt service routine (ISR) is not executed. 1: An interrupt with the highest priority	R

ISCi Bit (IRQ Interrupt Request Service Flag) (i = 64 to 127)

This flag indicates the service status of the IRQ interrupt request for interrupt service status register n (ISSn, n = 0 to 9).

- ISC4

Address(es): VIC.ISC4 A001 0240h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISC159	ISC158	ISC157	ISC156	ISC155	ISC154	ISC153	ISC152	ISC151	ISC150	ISC149	ISC148	ISC147	ISC146	ISC145	ISC144
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISC143	ISC142	ISC141	ISC140	ISC139	ISC138	ISC137	ISC136	ISC135	ISC134	ISC133	ISC132	ISC131	ISC130	ISC129	ISC128
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	ISC[159:128]	IRQ Interrupt Request Service Flag	0: The priority level is not the highest, or the interrupt service routine (ISR) is not executed. 1: An interrupt with the highest priority	R

- ISC5

Address(es): VIC.ISC5 A001 0244h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISC191	ISC190	ISC189	ISC188	ISC187	ISC186	ISC185	ISC184	ISC183	ISC182	ISC181	ISC180	ISC179	ISC178	ISC177	ISC176
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISC175	ISC174	ISC173	ISC172	ISC171	ISC170	ISC169	ISC168	ISC167	ISC166	ISC165	ISC164	ISC163	ISC162	ISC161	ISC160
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	ISC[191:160]	IRQ Interrupt Request Service Flag	0: The priority level is not the highest, or the interrupt service routine (ISR) is not executed. 1: An interrupt with the highest priority	R

ISCi Bit (IRQ Interrupt Request Service Flag) (i = 128 to 191)

This flag indicates the service status of the IRQ interrupt request for interrupt service status register n (ISSn, n = 0 to 9).

- ISC6

Address(es): VIC.ISC6 A001 0248h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISC223	ISC222	ISC221	ISC220	ISC219	ISC218	ISC217	ISC216	ISC215	ISC214	ISC213	ISC212	ISC211	ISC210	ISC209	ISC208
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISC207	ISC206	ISC205	ISC204	ISC203	ISC202	ISC201	ISC200	ISC199	ISC198	ISC197	ISC196	ISC195	ISC194	ISC193	ISC192
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	ISC[223:192]	IRQ Interrupt Request Service Flag	0: The priority level is not the highest, or the interrupt service routine (ISR) is not executed. 1: An interrupt with the highest priority	R

- ISC7

Address(es): VIC.ISC7 A001 024Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISC255	ISC254	ISC253	ISC252	ISC251	ISC250	ISC249	ISC248	ISC247	ISC246	ISC245	ISC244	ISC243	ISC242	ISC241	ISC240
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISC239	ISC238	ISC237	ISC236	ISC235	ISC234	ISC233	ISC232	ISC231	ISC230	ISC229	ISC228	ISC227	ISC226	ISC225	ISC224
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	ISC[255:224]	IRQ Interrupt Request Service Flag	0: The priority level is not the highest, or the interrupt service routine (ISR) is not executed. 1: An interrupt with the highest priority	R

ISCi Bit (IRQ Interrupt Request Service Flag) (i = 192 to 255)

This flag indicates the service status of the IRQ interrupt request for interrupt service status register n (ISSn, n = 0 to 9).

- ISC8

Address(es): VIC.ISC8 A001 1230h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISC287	ISC286	ISC285	ISC284	ISC283	ISC282	ISC281	ISC280	ISC279	ISC278	ISC277	ISC276	ISC275	ISC274	ISC273	ISC272
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISC271	ISC270	ISC269	ISC268	ISC267	ISC266	ISC265	ISC264	ISC263	ISC262	ISC261	ISC260	ISC259	ISC258	ISC257	ISC256
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	ISC[287:256]	IRQ Interrupt Request Service Flag	0: The priority level is not the highest, or the interrupt service routine (ISR) is not executed. 1: An interrupt with the highest priority	R

- ISC9

Address(es): VIC.ISC9 A001 1234h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	ISC294	ISC293	ISC292	ISC291	ISC290	ISC289	ISC288
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b6 to b0	ISC[294:288]	IRQ Interrupt Request Service Flag	0: The priority level is not the highest, or the interrupt service routine (ISR) is not executed. 1: An interrupt with the highest priority	R
b31 to b7	—	Reserved	These bits are read as 0.	R

ISCi Bit (IRQ Interrupt Request Service Flag) (i = 256 to 294)

This flag indicates the service status of the IRQ interrupt request for interrupt service status register n (ISSn, n = 0 to 9).

12.4.2.16 Interrupt Address Store Register 0 (VADn) (n = 1 to 255)
Interrupt Address Store Register 1 (VADn) (n = 256 to 294)

The VADn (n = 1 to 294) register stores the vector address for each interrupt input.
This register can only be read and written in 32-bit units.

Address(es): VIC.VAD1 A001 0404h to VIC.VAD255 A001 07FCh
VIC.VAD256 A001 1400h to VIC.VAD294 A001 1498h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	VADn[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	VADn[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	VADn[31:0]	Vector Address Store	VAD1 to VAD294 and vector numbers 1 to 294 are paired.	R/W

VADn[31:0] Bit (Vector Address Store) (n = 1 to 294)

This is a vector address store bit. VAD1 to VAD294 and registers of vector numbers 1 to 294 are paired.

- Connecting the CPU as the destination of interrupt requests
  Set the branch destination address of the interrupt handling routine to be run for interrupt request n in this register.
- Connecting a DMAC as the destination of interrupt requests
  The occurrence of interrupt request n that has been set up to do so starts the DMA transfer. Set the branch destination address of the interrupt handling routine to be run for the DMA transfer completion interrupt in this register. To connect a DMAC, set vector number n in the associated register as described in section 14.2.8, DMAC Unit 0 Source Select Register i (DMA0SELi) (i = 0 to 15), or section 14.2.9, DMAC Unit 1 Source Select Register i (DMA1SELi) (i = 0 to 15).

12.4.2.17 Interrupt Priority Level Store Register 0 (PRLn) (n = 1 to 255)

The PRLn (n = 1 to 255) register stores the interrupt priority level for each interrupt input.
This register can only be read and written in 32-bit units.

Address(es): VIC.PRL1 A001 0804h to VIC.PRL255 A001 0BFCh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	PRL3	PRL2	PRL1	PRL0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b3 to b0	PRL[3:0]	Interrupt Priority Level Store	The highest interrupt priority level is 0, and the lowest is 15.	R/W
b31 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

PRL[3:0] Bits (Interrupt Priority Level Store)

These bits store the interrupt priority level of vector numbers 1 to 255. The highest interrupt priority level is 0, and the lowest is 15.

12.4.2.18 Interrupt Priority Level Store Register 1 (PRLn) (n = 256 to 294)

The PRLn (n = 256 to 294) register stores the interrupt priority level for each interrupt input.
This register can only be read and written in 32-bit units.

Address(es): VIC.PRL256 A001 1800h to VIC.PRL294 A001 1898h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	PRL3	PRL2	PRL1	PRL0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b3 to b0	PRL[3:0]	Interrupt Priority Level Store	The highest interrupt priority level is the value of PRLn+16. The highest is 16, and the lowest is 31.	R/W
b31 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

PRL[3:0] Bits (Interrupt Priority Level Store)

These bits store the interrupt priority level of vector numbers 256 to 294. The highest interrupt priority level is the value of PRLn+16. The highest is 16, and the lowest is 31.

12.4.3 Vector Table

12.4.3.1 Interrupt Vector Table

Table 12.3 describes the vector table for interrupts to Cortex-R4 and DMAC. Instead of an interrupt source from peripheral modules, transfer completion sources of DMAC channels selected by the DMAC source select register are connected to the vector number selected by the DMAC source select register.

The following table explains the items of the Cortex-R4/DMAC interrupt vector table.

Item	Description
Vector number	Indicates the vector number of the IRQ interrupt source in VIC for Cortex-R4.
Request source	Indicates the name of the interrupt request source.
Source	Indicates the interrupt name.
Detection type	Indicates the detection type for interrupts from peripheral modules. To connect a transfer completion interrupt from DMAC, the edge must be selected.
CR4	“Y” indicates the interrupt source for Cortex-R4 (VIC).
DMAC	“Y” indicates the DMAC activation source.

Note: An error signal of each module is not input to the CPU directly, but to the error control module (ECM). The signal is merged into other errors and conveyed as an error detection source to the CPU. For details, see section 29, Error Control Module (ECM).

Table 12.3 Cortex-R4/DMAC Interrupt Vector Table (1 / 8)

Vector Number	Request Source	Source		Detection Type	CR4	DMAC*2
1	System (CR4)	—	Reserved	—	—	—
2		INTCTI	CTI (Cross Trigger Interface) interrupt	Edge	Y	N
3		FPUEX	FPU exception	Level	Y	N
4	External	IRQ0	IRQ pin interrupt 0	Level / Edge	Y	Y
5		IRQ1	IRQ pin interrupt 1	Level / Edge	Y	Y
6		IRQ2	IRQ pin interrupt 2	Level / Edge	Y	Y
7		IRQ3	IRQ pin interrupt 3	Level / Edge	Y	Y
8		IRQ4	IRQ pin interrupt 4	Level / Edge	Y	Y
9	—	—	Reserved	—	—	—
10	External	IRQ6	IRQ pin interrupt 6	Level / Edge	Y	Y
11		IRQ7	IRQ pin interrupt 7	Level / Edge	Y	Y
12	—	—	Reserved	—	—	—
13		—	Reserved	—	—	—
14		—	Reserved	—	—	—
15		—	Reserved	—	—	—
16		—	Reserved	—	—	—
17		—	Reserved	—	—	—
18		—	Reserved	—	—	—
19		—	Reserved	—	—	—
20	ECM	ERRD	Error detection (maskable)	Edge	Y	N
21	CMT unit 0	CMI0	Compare match interrupt_ch0	Edge	Y	Y
22		CMI1	Compare match interrupt_ch1	Edge	Y	Y
23	CMT unit 1	CMI2	Compare match interrupt_ch0	Edge	Y	Y
24		CMI3	Compare match interrupt_ch1	Edge	Y	Y
25	CMTW unit 0	CMWI0	Compare match interrupt	Edge	Y	Y
26		IC0I0	Input capture 0 interrupt	Edge	Y	Y
27		IC1I0	Input capture 1 interrupt	Edge	Y	Y
28		OC0I0	Output compare 0 interrupt	Edge	Y	Y
29		OC1I0	Output compare 1 interrupt	Edge	Y	Y
30	CMTW unit 1	CMWI1	Compare match interrupt	Edge	Y	Y
31		IC0I1	Input capture 0 interrupt	Edge	Y	Y
32		IC1I1	Input capture 1 interrupt	Edge	Y	Y
33		OC0I1	Output compare 0 interrupt	Edge	Y	Y
34		OC1I1	Output compare 1 interrupt	Edge	Y	Y
35	ADC unit 0	S12ADI0	AD conversion completion interrupt	Edge	Y	Y
36		S12GBADI0	Group B AD conversion completion interrupt	Edge	Y	Y
37		S12CMPI0	Compare condition not met	Level	Y	N
38	ADC unit 1	S12ADI1	AD conversion completion interrupt	Edge	Y	Y
39		S12GBADI1	Group B AD conversion completion interrupt	Edge	Y	Y
40		S12CMPI1	Compare condition met	Level	Y	N
41	—	—	Reserved	—	—	—
42		—	Reserved	—	—	—

Table 12.3 Cortex-R4/DMAC Interrupt Vector Table (2 / 8)

Vector Number	Request Source	Source		Detection Type	CR4	DMAC*2
43	—	—	Reserved	—	—	—
44		—	Reserved	—	—	—
45		—	Reserved	—	—	—
46		—	Reserved	—	—	—
47		—	Reserved	—	—	—
48		—	Reserved	—	—	—
49		—	Reserved	—	—	—
50		—	Reserved	—	—	—
51		—	Reserved	—	—	—
52		—	Reserved	—	—	—
53		—	Reserved	—	—	—
54		—	Reserved	—	—	—
55		—	Reserved	—	—	—
56		—	Reserved	—	—	—
57		—	Reserved	—	—	—
58		—	Reserved	—	—	—
59		—	Reserved	—	—	—
60		—	Reserved	—	—	—
61		—	Reserved	—	—	—
62		—	Reserved	—	—	—
63		—	Reserved	—	—	—
64		—	Reserved	—	—	—
65		—	Reserved	—	—	—
66		—	Reserved	—	—	—
67		—	Reserved	—	—	—
68		—	Reserved	—	—	—
69		—	Reserved	—	—	—
70		—	Reserved	—	—	—
71		—	Reserved	—	—	—
72		—	Reserved	—	—	—
73		—	Reserved	—	—	—
74		—	Reserved	—	—	—
75		—	Reserved	—	—	—
76		—	Reserved	—	—	—
77		—	Reserved	—	—	—
78		—	Reserved	—	—	—
79		—	Reserved	—	—	—
80	RSPI ch0	SPRI0	Reception buffer full	Edge	Y	Y
81		SPTI0	Transmission buffer empty	Edge	Y	Y
82		SPEI0	Mode fault error/overrun error/parity error	Level	Y	N
83		SPII0	RSPI idle	Level	Y	N

Table 12.3 Cortex-R4/DMAC Interrupt Vector Table (3 / 8)

Vector Number	Request Source	Source		Detection Type	CR4	DMAC*2
84	RSPI ch1	SPRI1	Reception buffer full	Edge	Y	Y
85		SPTI1	Transmission buffer empty	Edge	Y	Y
86		SPEI1	Mode fault error/overrun error/parity error	Level	Y	N
87		SPII1	RSPI idle	Level	Y	N
88	—	—	Reserved	—	—	—
89		—	Reserved	—	—	—
90		—	Reserved	—	—	—
91		—	Reserved	—	—	—
92	—	—	Reserved	—	—	—
93		—	Reserved	—	—	—
94		—	Reserved	—	—	—
95		—	Reserved	—	—	—
96	SCIFA ch0	BRIF0	Break/overrun/framing error/parity error	Level	Y	N
97		RXIF0	Reception FIFO data full (RDF)	Level	Y	Y
98		TXIF0	Transmission FIFO data empty (TDFE)	Level	Y	Y
99		DRIF0	Transmit end/reception data ready	Level	Y	N
100	SCIFA ch1	BRIF1	Break/overrun/framing error/parity error	Level	Y	N
101		RXIF1	Reception FIFO data full (RDF)	Level	Y	Y
102		TXIF1	Transmission FIFO data empty (TDFE)	Level	Y	Y
103		DRIF1	Transmit end/reception data ready	Level	Y	N
104	—	—	Reserved	—	—	—
105		—	Reserved	—	—	—
106		—	Reserved	—	—	—
107		—	Reserved	—	—	—
108		—	Reserved	—	—	—
109	SCIFA ch2	BRIF2	Break/overrun/framing error/parity error	Level	Y	N
110		RXIF2	Reception FIFO data full (RDF)	Level	Y	Y
111		TXIF2	Transmission FIFO data empty (TDFE)	Level	Y	Y
112		DRIF2	Transmit end/reception data ready	Level	Y	N
113	—	—	Reserved	—	—	—
114		—	Reserved	—	—	—
115		—	Reserved	—	—	—
116		—	Reserved	—	—	—
117	SCIFA ch4	BRIF4	Break/overrun/framing error/parity error	Level	Y	N
118		RXIF4	Reception FIFO data full (RDF)	Level	Y	Y
119		TXIF4	Transmission FIFO data empty (TDFE)	Level	Y	Y
120		DRIF4	Transmit end/reception data ready	Level	Y	N
121	RIIC ch0	TEI0	Data transmission completed (TEND)	Level	Y	N
122		RXI0	Data reception completed (RDRF)	Edge	Y	Y
123		TXI0	Transmission data empty (TDRE)	Edge	Y	Y
124	RIIC ch1	TEI1	Data transmission completed (TEND)	Level	Y	N
125		RXI1	Data reception completed (RDRF)	Edge	Y	Y
126		TXI1	Transmission data empty (TDRE)	Edge	Y	Y

Table 12.3 Cortex-R4/DMAC Interrupt Vector Table (4 / 8)

Vector Number	Request Source	Source		Detection Type	CR4	DMAC*2
127	—	—	Reserved	—	—	—
128		—	Reserved	—	—	—
129		—	Reserved	—	—	—
130		—	Reserved	—	—	—
131	MDIO	INT_REQ_REG_DATA	Data transmission/reception completed	Level	Y	Y
132		INT_ETC	Normal source: Reception of DEVADD being complete and matching the expected value Reception of DEVADD being complete but not matching the expected value Reception of PHYADR being complete and matching the expected value Reception of PHYADR being complete but not matching the expected value Completion of OP reception Error source: Generation of transmission error	Level	Y	Y
133	—	—	Reserved	—	—	—
134		—	Reserved	—	—	—
135	—	—	Reserved	—	—	—
136	—	—	Reserved	—	—	—
137		—	Reserved	—	—	—
138		—	Reserved	—	—	—
139		—	Reserved	—	—	—
140		—	Reserved	—	—	—
141		—	Reserved	—	—	—
142		—	Reserved	—	—	—
143	MDIOM1*3	INT_END1	Completion of transfer through the MDIOM1 master	Level	Y	N
144	—	—	Reserved	—	—	—
145		—	Reserved	—	—	—
146		—	Reserved	—	—	—
147		—	Reserved	—	—	—
148		—	Reserved	—	—	—
149		—	Reserved	—	—	—
150		—	Reserved	—	—	—
151		—	Reserved	—	—	—
152		—	Reserved	—	—	—
153		—	Reserved	—	—	—
154		—	Reserved	—	—	—
155		—	Reserved	—	—	—
156		—	Reserved	—	—	—
157		—	Reserved	—	—	—
158		—	Reserved	—	—	—

Table 12.3 Cortex-R4/DMAC Interrupt Vector Table (5 / 8)

Vector Number	Request Source	Source		Detection Type	CR4	DMAC*2
159	—	—	Reserved	—	—	—
160		—	Reserved	—	—	—
161		—	Reserved	—	—	—
162		—	Reserved	—	—	—
163		—	Reserved	—	—	—
164		—	Reserved	—	—	—
165		—	Reserved	—	—	—
166		—	Reserved	—	—	—
167		—	Reserved	—	—	—
168		—	Reserved	—	—	—
169		—	Reserved	—	—	—
170		—	Reserved	—	—	—
171		—	Reserved	—	—	—
172		—	Reserved	—	—	—
173		—	Reserved	—	—	—
174		—	Reserved	—	—	—
175		—	Reserved	—	—	—
176		—	Reserved	—	—	—
177		—	Reserved	—	—	—
178		—	Reserved	—	—	—
179		—	Reserved	—	—	—
180		—	Reserved	—	—	—
181		—	Reserved	—	—	—
182		—	Reserved	—	—	—
183		—	Reserved	—	—	—
184		—	Reserved	—	—	—
185		—	Reserved	—	—	—
186		—	Reserved	—	—	—
187		—	Reserved	—	—	—
188		—	Reserved	—	—	—
189		—	Reserved	—	—	—
190		—	Reserved	—	—	—
191		—	Reserved	—	—	—
192		—	Reserved	—	—	—
193		—	Reserved	—	—	—
194		—	Reserved	—	—	—
195		—	Reserved	—	—	—
196		—	Reserved	—	—	—
197		—	Reserved	—	—	—
198		—	Reserved	—	—	—
199		—	Reserved	—	—	—
200		—	Reserved	—	—	—
201		—	Reserved	—	—	—
202		—	Reserved	—	—	—
203		—	Reserved	—	—	—

Table 12.3 Cortex-R4/DMAC Interrupt Vector Table (6 / 8)

Vector Number	Request Source	Source		Detection Type	CR4	DMAC*2
204	—	—	Reserved	—	—	—
205		—	Reserved	—	—	—
206		—	Reserved	—	—	—
207		—	Reserved	—	—	—
208		—	Reserved	—	—	—
209		—	Reserved	—	—	—
210		—	Reserved	—	—	—
211		—	Reserved	—	—	—
212		—	Reserved	—	—	—
213		—	Reserved	—	—	—
214		—	Reserved	—	—	—
215		—	Reserved	—	—	—
216	TPUa unit 0	TGI0A	ch0 input capture/compare match A interrupt	Edge	Y	Y
217		TGI0B	ch0 input capture/compare match B interrupt	Edge	Y	Y
218		TGI0C	ch0 input capture/compare match C interrupt	Edge	Y	N
219		TGI0D	ch0 input capture/compare match D interrupt	Edge	Y	N
220		TCI0V	ch0 overflow interrupt	Edge	Y	N
221		TGI1A	ch1 input capture/compare match A interrupt	Edge	Y	Y
222		TGI1B	ch1 input capture/compare match B interrupt	Edge	Y	Y
223		TCI1V	ch1 overflow interrupt	Edge	Y	N
224		TCI1U	ch1 underflow interrupt	Edge	Y	N
225		TGI2A	ch2 input capture/compare match A interrupt	Edge	Y	Y
226		TGI2B	ch2 input capture/compare match B interrupt	Edge	Y	Y
227		TCI2V	ch2 overflow interrupt	Edge	Y	N
228		TCI2U	ch2 underflow interrupt	Edge	Y	N
229		TGI3A	ch3 input capture/compare match A interrupt	Edge	Y	Y
230		TGI3B	ch3 input capture/compare match B interrupt	Edge	Y	Y
231		TGI3C	ch3 input capture/compare match C interrupt	Edge	Y	N
232		TGI3D	ch3 input capture/compare match D interrupt	Edge	Y	N
233		TCI3V	ch3 overflow interrupt	Edge	Y	N
234		TGI4A	ch4 input capture/compare match A interrupt	Edge	Y	Y
235		TGI4B	ch4 input capture/compare match B interrupt	Edge	Y	Y
236		TCI4V	ch4 overflow interrupt	Edge	Y	N
237		TCI4U	ch4 underflow interrupt	Edge	Y	N

Table 12.3 Cortex-R4/DMAC Interrupt Vector Table (7 / 8)

Vector Number	Request Source	Source		Detection Type	CR4	DMAC*2
238	TPUa unit 0	TGI5A	ch5 input capture/compare match A interrupt	Edge	Y	Y
239		TGI5B	ch5 input capture/compare match B interrupt	Edge	Y	Y
240		TCI5V	ch5 overflow interrupt	Edge	Y	N
241		TCI5U	ch5 underflow interrupt	Edge	Y	N
242	ELC	ELCIRQ1	Interrupt 1 (ELSR18)	Edge	Y	Y
243		ELCIRQ2	Interrupt 2 (ELSR19)	Edge	Y	Y
244	—	—	Reserved	—	—	—
245		—	Reserved	—	—	—
246		—	Reserved	—	—	—
247		—	Reserved	—	—	—
248	—	—	Reserved	—	—	—
249		—	Reserved	—	—	—
250		—	Reserved	—	—	—
251	DMAC	DMASRQ0	DMA transfer software activation (unit 0)	Edge	N *1	Y
252		DMASRQ1	DMA transfer software activation (unit 1)	Edge	N *1	Y
253	—	—	Reserved	—	—	—
254		—	Reserved	—	—	—
255		—	Reserved	—	—	—
256		—	Reserved	—	—	—
257		—	Reserved	—	—	—
258		—	Reserved	—	—	—
259		—	Reserved	—	—	—
260	RIIC ch0	EEI0	Stop condition detection/Start condition detection/NACK detection/arbitration lost/time-out occurrence	Level	Y	N
261	RIIC ch1	EEI1	Stop condition detection/Start condition detection/NACK detection/arbitration lost/time-out occurrence	Level	Y	N
262	—	—	Reserved	—	—	—
263		—	Reserved	—	—	—
264		—	Reserved	—	—	—
265		—	Reserved	—	—	—
266		—	Reserved	—	—	—
267	—	—	Reserved	—	—	—
268		—	Reserved	—	—	—
269		—	Reserved	—	—	—
270		—	Reserved	—	—	—
271		—	Reserved	—	—	—
272		—	Reserved	—	—	—
273		—	Reserved	—	—	—
274		—	Reserved	—	—	—
275		—	Reserved	—	—	—
276		—	Reserved	—	—	—

Table 12.3 Cortex-R4/DMAC Interrupt Vector Table (8 / 8)

Vector Number	Request Source	Source		Detection Type	CR4	DMAC*2
277	—	—	Reserved	—	—	—
278		—	Reserved	—	—	—
279		—	Reserved	—	—	—
280		—	Reserved	—	—	—
281		—	Reserved	—	—	—
282		—	Reserved	—	—	—
283		—	Reserved	—	—	—
284		—	Reserved	—	—	—
285		—	Reserved	—	—	—
286		—	Reserved	—	—	—
287		—	Reserved	—	—	—
288		—	Reserved	—	—	—
289		—	Reserved	—	—	—
290		—	Reserved	—	—	—
291		—	Reserved	—	—	—
292		—	Reserved	—	—	—
293	DMAC	DMAERR0	DMA transfer transfer error (unit 0)	Edge	Y	N
294		DMAERR1	DMA transfer transfer error (unit 1)	Edge	Y	N
295	—	—	Reserved	—	—	—
296		—	Reserved	—	—	—
297		—	Reserved	—	—	—
298		—	Reserved	—	—	—
299		—	Reserved	—	—	—
300		—	Reserved	—	—	—

Note: Do not select the interrupt request destinations that do not have Y for the request destination.

Note 1. Though an interrupt servings as a DMA activating source is not generated as an interrupt, the interrupt handling routine with the vector number of the DMA transfer completion interrupt is executed on completion of DMA transfer. For details, see section 12.3.1, Selecting Interrupt Request Destinations, and section 14.4.2, DMA Transfer Completion Interrupts.

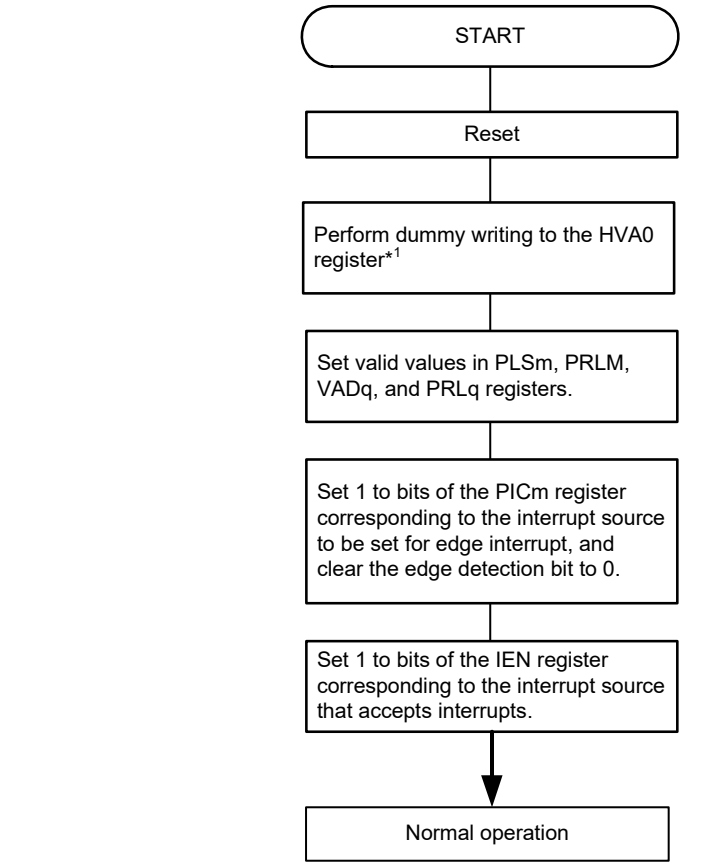
Note 2. When connecting the DMAC transfer completion interrupt, always select the same edge as that selected by the setting of interrupt detection type select register n (PLSn). For details, see section 12.4.4.3 (1), Specifying Interrupt Detection Types.

Note 3. Only for the products with the MDIO master (optional)

12.4.4 Operation

12.4.4.1 Initializing Registers of VIC

Figure 12.5 shows the procedure for initializing registers of VIC.
When you canceled reset, registers are not ready to operate due to interrupt priority level settings of VIC, or other reasons. For this reason, you must initialize the registers after a reset release.



m = 0 to 9, q = 1 to 294
Note 1. For initializing the VIC, write this register once after being released from a reset . Also, write this register at the end of each interrupt handling routine. For details, see section 12.4.2.13, Interrupt Address Register (HVA0).

Figure 12.5 Initializing Registers of VIC

12.4.4.2 Procedure for Rewriting the PLS, PRLM, VAD, and PRL Registers

To rewrite the following registers while VIC is in operation, finish all interrupt processing, and then disable interrupts. To disable interrupts, set 1 to the I bit of the CPSR register for Cortex-R4.

- PLS (Interrupt detection type selection register)
- PRLM (Interrupt priority level mask register)
- VAD (Interrupt address storage register)
- PRL (Interrupt priority level storage register)

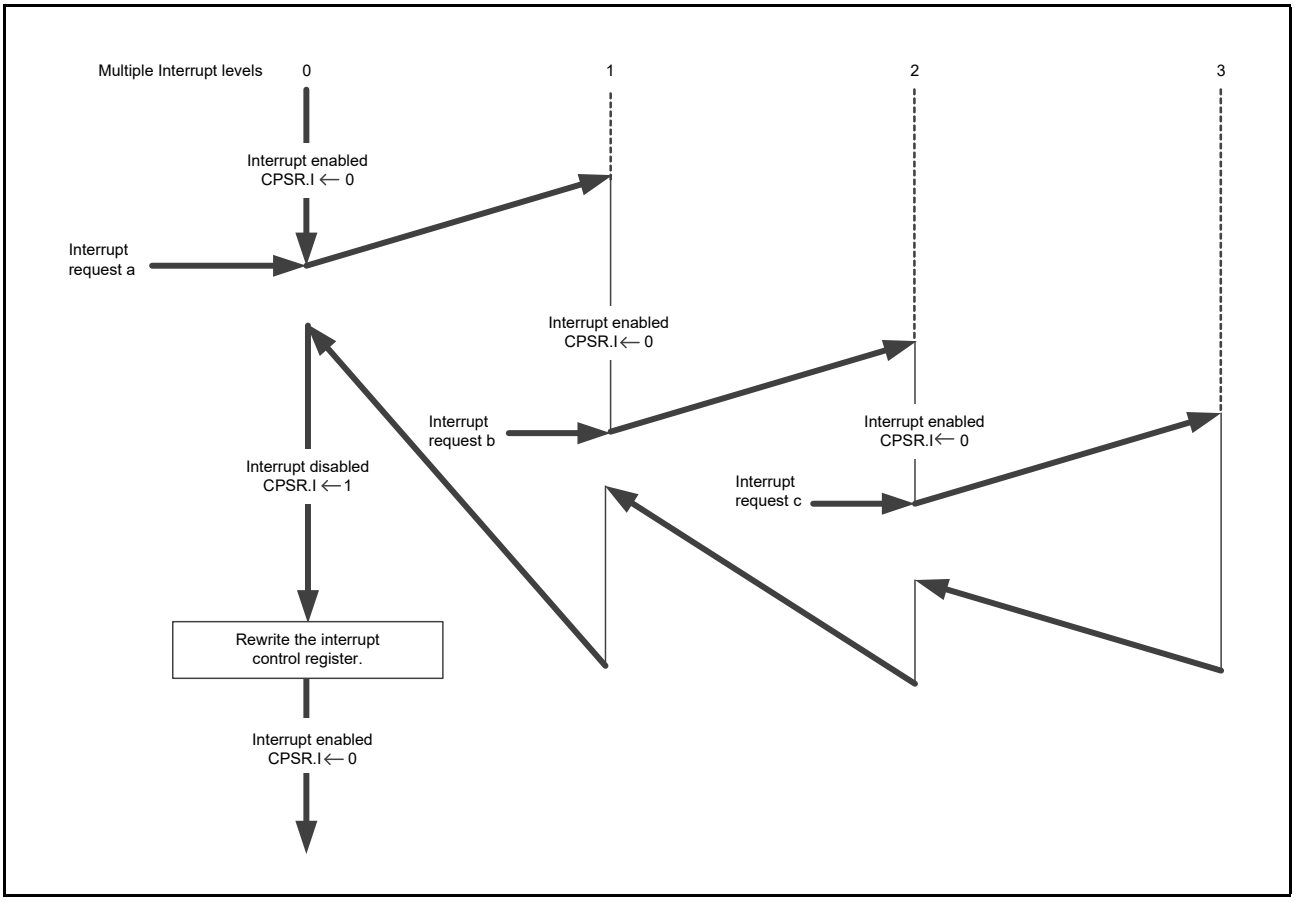


Figure 12.6 Period for Changing Register Settings

Follow the flow below to change register settings by using software.

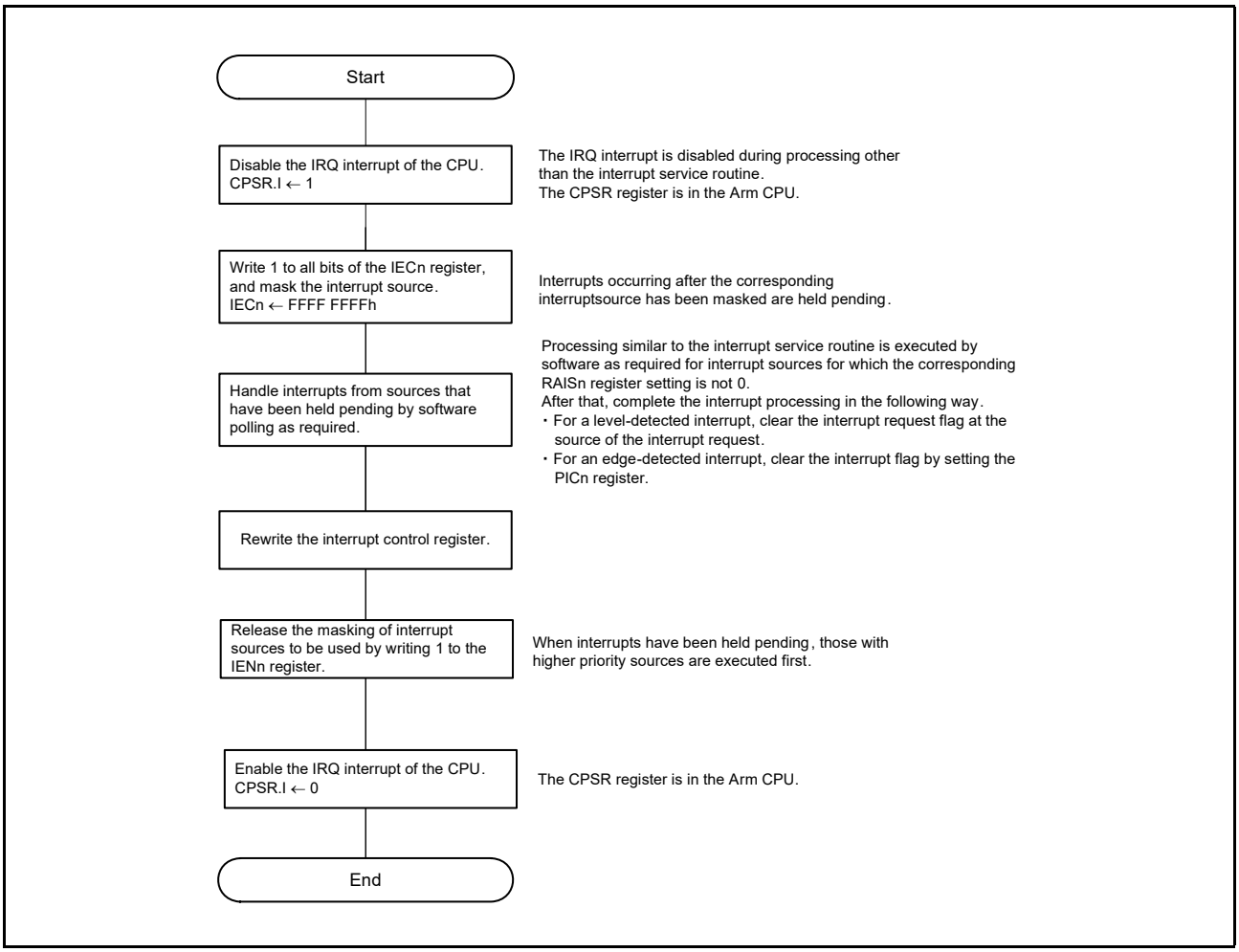


Figure 12.7 Register Rewrite Flow

12.4.4.3 Detecting Interrupts

(1) Specifying Interrupt Detection Types

When connecting external interrupts except interrupts from the NMI pin, and interrupts from on-chip peripheral modules to Cortex-R4, you must use VIC to select the edge or level detection with interrupt detection type selection register n (PLSn). Table 12.4 lists settings of the interrupt detection type for VIC. When connecting transfer completion interrupt from DMAC, always select edge detection.

Table 12.4 VIC Settings by Interrupt Detection Type

Interrupt Request Type	PLSm
Edge interrupt	1
Level interrupt	0

m: Interrupt vector number (0 to 294)

(2) IRQ Interrupt (Level interrupt)

Figure 12.8 shows level interrupt operation.

When you complete a level interrupt, stop interrupt output of the request source for the level interrupt. At that time, the applicable bit of IRQ status register n (IRQSn, n = 0 to 9) is cleared to 0. After that, make sure the interrupt is no longer requested. This operation is needed to prevent the same interrupt from being accepted after restoration because there is a delay before the interrupt output stop processing of the interrupt request source by the software is applied to the hardware. In addition, stop the interrupt output of the interrupt request source at the appropriate position of the service routine (ISR) according to the operation of the request source.

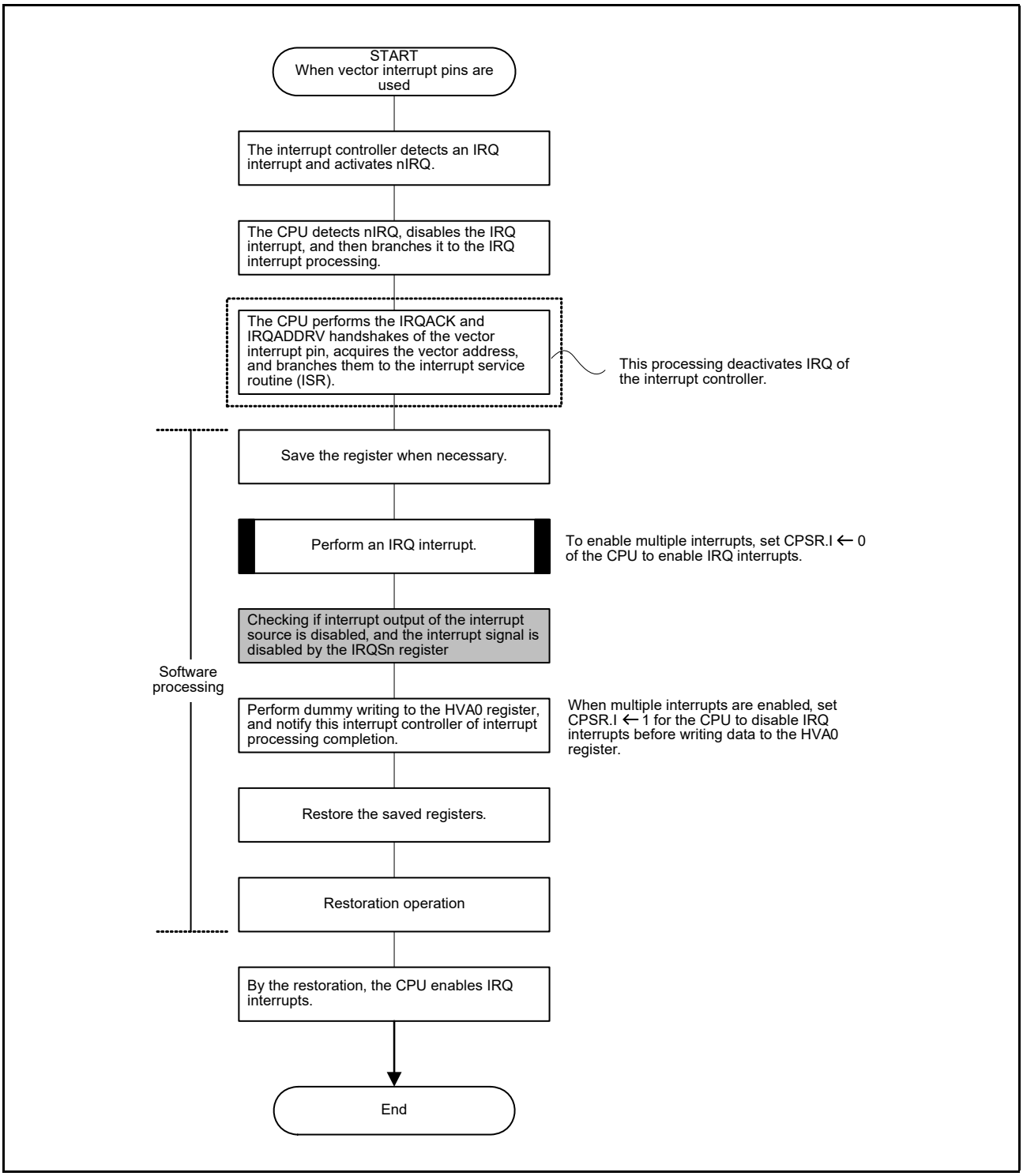


Figure 12.8 IRQ Interrupt Operation (Level Operations)

(3) IRQ Interrupt (Edge Interrupt)

Figure 12.9 shows edge interrupt operation.

Clear the edge interrupt request with edge detection bit clear register n (PICn, n = 0 to 9).

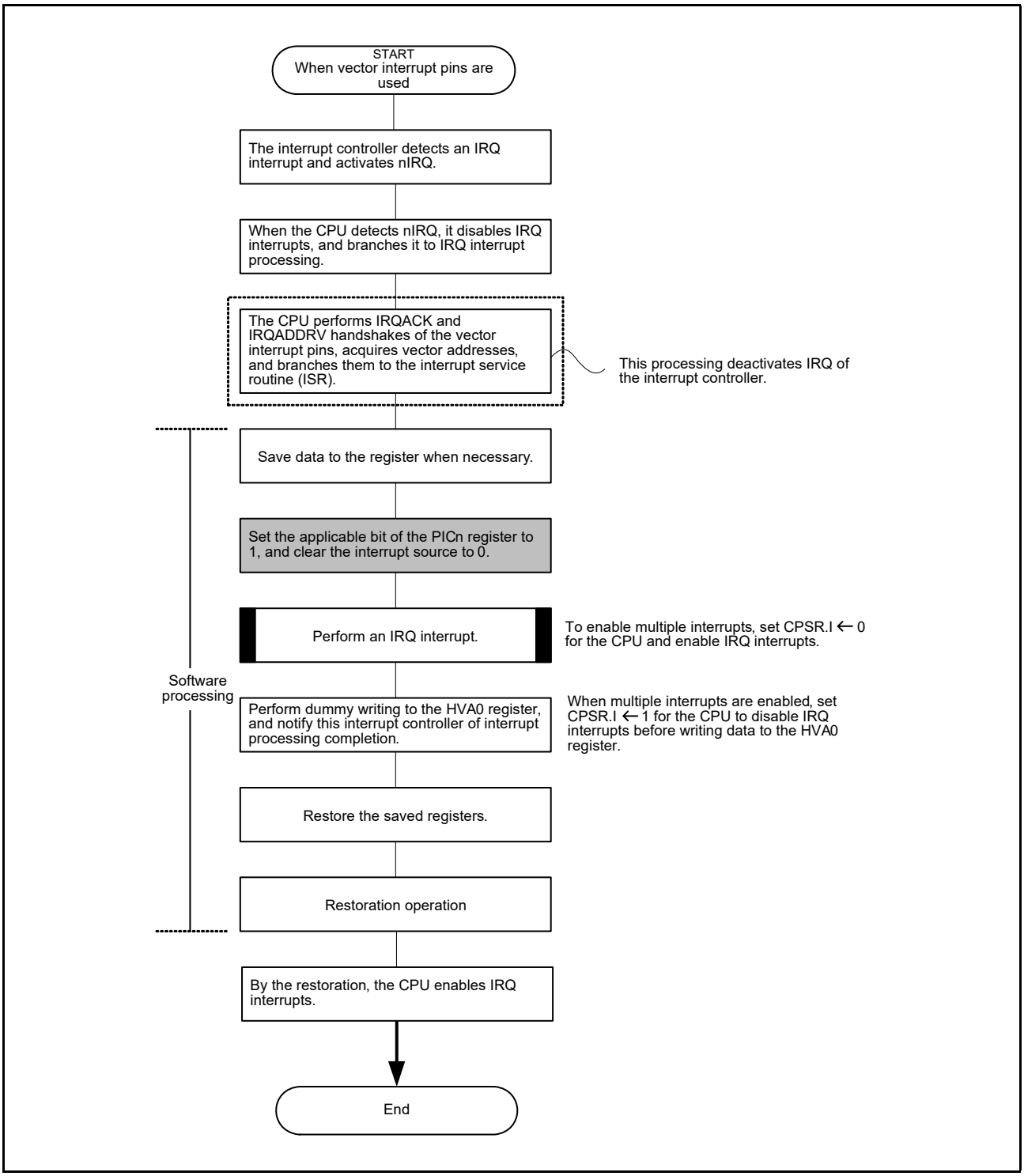


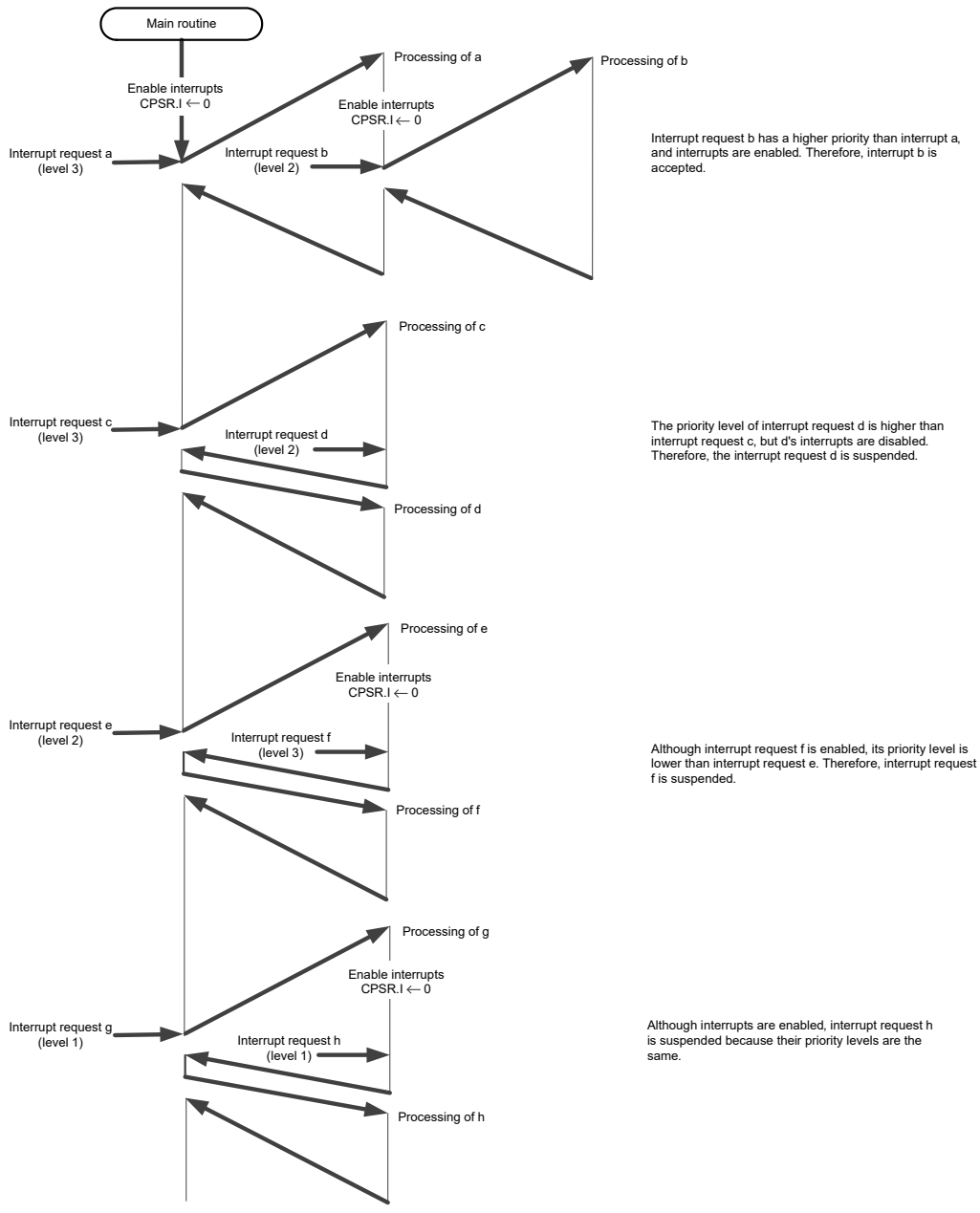
Figure 12.9 IRQ Interrupt Operation (Edge Interrupt)

12.4.4.4 Priority Level for Interrupt Multiple Control

If an interrupt is being handled (only when interrupt multiple control is being performed), only an interrupt that has higher priority than the interrupt which is being serviced is accepted. At that time, interrupts with lower priority than the currently serviced interrupt are suspended.

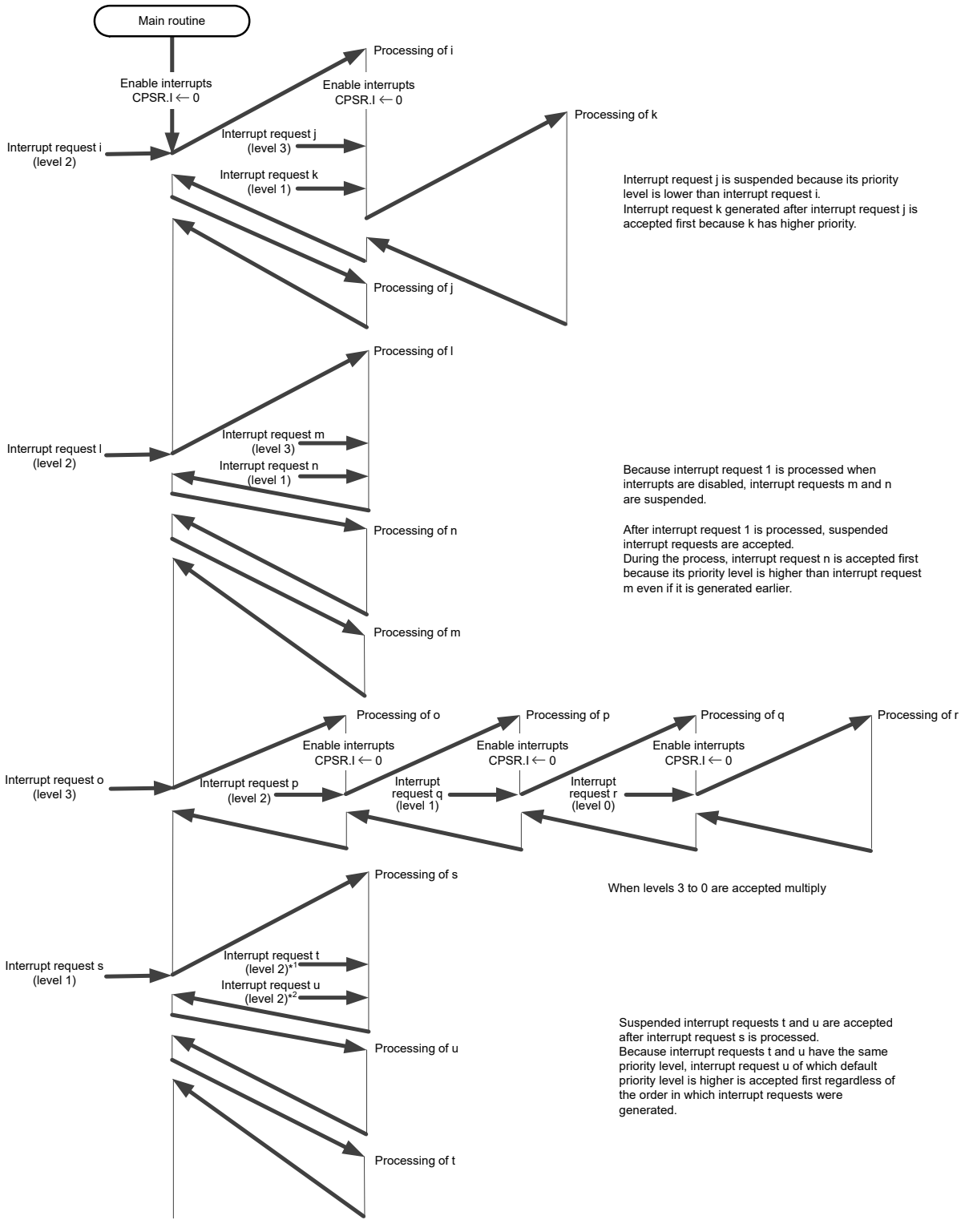
12.4.4.5 Handling Multiple Interrupts

Figure 12.10 provides an example of multiple interrupts that accept another interrupt when an interrupt is being handled.



a through u in the figure (including 2/2) are tentative names used to identify each interrupt request signal.
High/Low of the default priority level in the figure indicate the relative priority level between two interrupt request signals.
The priorities are higher for smaller vector numbers, which means, if the same value is set in two interrupts by the PRLm register, the one with smaller vector number takes priority.

Figure 12.10 Concept of Multiple Interrupts (1 / 2)



Note 1. The default priority level is low.
Note 2. The default priority level is high.
The priorities are higher for smaller vector numbers, which means, if the same value is set in two interrupts by the PRLm register, the one with smaller vector number takes priority.

Figure 12.10 Concept of Multiple Interrupts (2 / 2)

12.4.4.6 Handling IRQ Interrupt Source Conditions by Polling

Figure 12.11 shows the procedure for handling IRQ interrupt source conditions by polling the interrupt status registers (RAISn).

That is, source conditions for IRQ interrupts can be detected by checking the bits of the interrupt input status registers (RAISn). This is useful when interrupts are masked by settings in the interrupt enable registers (IENn). This enables interrupt processing without hardware forcing branches to the interrupt service routines (ISRs).

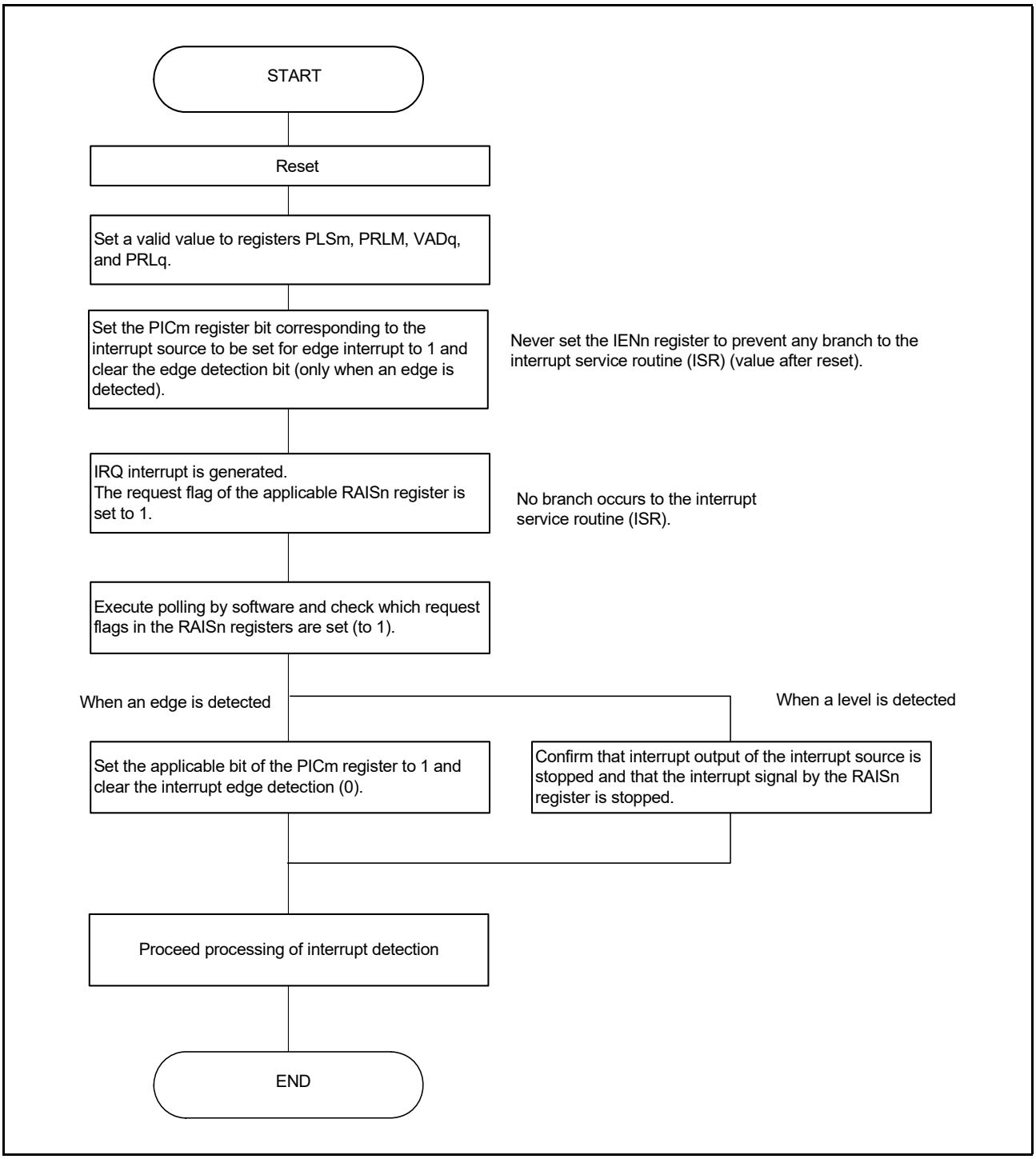


Figure 12.11 IRQ Interrupt Operation by Polling (Edge/Level Detection)

12.4.5 Return from Sleep Mode

If the interrupt controller is to return operation from sleep mode in response to an interrupt or non-maskable interrupt, make the following settings for the interrupt.

(1) The applicable interrupt request is enabled with the IENn bit.
(2) The DMAC source select register does not perform assignment to DMAC.

12.4.6 Usage Notes

12.4.6.1 Restrictions on VIC Priority Levels

VIC specifies the priority in 16 levels for each source with the interrupt priority level store register n (PRLn, n = 1 to 294). But lower priority levels are assigned to sources with vector numbers 256 and later compared to sources with vector numbers 1 to 255. Table 12.5 lists the relationship between vector numbers and priority levels.

Table 12.5 Relationship between Vector Numbers and Priority Levels

Vector Number	Priority Level*1
1 to 255	PRLn
256 to 294	PRLn + 16

n = 1 to 294
Note 1. The highest priority level is 0.

12.4.6.2 Notes on Accessing HVA0 Register

In cases of contention due to a vector interrupt being generated at the same time as writing to the HVA0 register, the AHB allows dummy-writing to the HVA0 register to proceed and blocks conveying of the IRQ interrupt to the CPU until a response to the write operation is returned so that processing to handle a new vector interrupt remains inactive over that time.
When writing to the HVA0 register, never fail to wait for the response indicating completion of the write operation before allowing IRQ interrupts to be sent to the CPU. Successful writing will be ensured by following the procedure illustrated in Figure 12.8 and executing a DMB instruction immediately after the write to the HVA0 register (as in the program example below).

- Program example

```
VIC.HVA0.LONG = 0x00000000;
asm(“dmb”);                    //DMB instruction
```

Note: The program format may vary according to the compiler. Confirm the applicable format for each compiler in the individual manuals.

12.4.6.3 Notes on Selecting Level Detection

When an interrupt request is set to level detection, do not cancel an interrupt request that was already generated except interrupt cancellation by the CPU. The correct vector address might not be output. For example, if level interrupt A is generated, the source for the output of interrupt A must be cleared by the service routine for interrupt A in a normal situation as shown in Figure 12.8 of section 12.4.4.3, Detecting Interrupts, but the source for the output of interrupt A may be transiently cleared if the service routine for another interrupt B has cleared or masked the source for the output of interrupt A.
When this interrupt controller accepts an interrupt, and the interrupt request is canceled before the CPU acquires the vector address, if another interrupt request is generated at the same time, 0000 0014h is output as the vector address. Therefore, to handle transitional cancellation of interrupt requests as mentioned above, it is recommended to use the return instruction only for 0000 0014h of the CPU (refer to the Example Program below).
In addition, if an interrupt request signal is transitionally withheld during the processing of an interrupt for which multiple interrupts are enabled, branching returns to the interrupt processing for which service is currently in progress. This interrupt handler writing to the HVA0 register causes the controller for this interrupt to recognize the completion of interrupt processing. Note that there is a gap between the actual interrupt source generating an interrupt request and the CPU recognizing the interrupt source.

- Example Program
  reserved_handler:
  subs pc, lr, #4 ; locate at 0000 0014h.

Note: The program format may differ for each complier. Confirm the applicable format for each in their manuals.

12.4.6.4 Notes when Rewriting the IECn Register

If the value of the IECn register is to be changed, do so while interrupts are disabled. To disable interrupts, set the I bit in the CPSR register of the Arm CPU to 1.

12.4.6.5 Notes on Vector Settings

In the specification of this product, use of the fixed vector by setting the SCTLR.[24]VE bit to 0 is prohibited. Only providing addresses from those of the VIC by setting SCTLR.[24]VE bit = 1 is possible. The addresses can be set by using the VADn registers (n: vector number).

12.5 Usage Notes

12.5.1 Using “Falling-Edge” or “Rising and Falling Edges” Detection with the External Pin Interrupts

Since the internal level on external pin interrupts after a reset is high, when external pin interrupts are used with low as the initial input level and the detection of “falling edge” or “rising and falling edges”, follow the procedure below.
Otherwise, follow the procedures shown in section 12.3.3, External Pin Interrupts.
In addition, make sure that a falling edge is not input to the external pin interrupts before these settings are completed.

[For IRQ pins]

1. Clear the applicable IENn bit to 0 (set the IECn bit).
2. Set the Pmn I/O select bit in the port direction register (PDR) of the I/O port to 10b (input).
3. Set and check (read) the I/O port (PmnPFS.ISEL bit).
4. Clear the IRQFLTE.FLTENi bit to 0.*1
5. Set the digital noise filter sampling clock with the IRQFLTC.FCLKSEL[1:0] bits.*1
6. Set the IRQFLTE.FLTENi bit to 1.*1
7. Select the edge for detection as “falling” or “rising and falling” by setting the IRQCRi.IRQMD[1:0] bits.
8. Set the applicable PICn register to 1 (when an edge is detected).
9. Set the applicable IENn bit to 1.

Note 1. Setting is required only when the digital filter is used.

12.5.2 Using Falling-Edge Detection with the NMI Pin

Since the internal level on the NMI pin after a reset is high, when the NMI pin is used with low as the initial input level and the detection of falling edges, follow the procedure below.
Otherwise, follow the procedure shown in section 12.3.4, NMI Pin Interrupts.
In addition, make sure that a falling edge is not input to the NMI pin before these settings are completed.

1. Select the edge for detection as rising by setting the NMICR.NMIMD bit to 1.
2. Set and check the I/O port pin (P35PFS.ISEL bit).
3. Clear the NMIFLTE.NFLTEN bit to 0.*1
4. Set the sampling clock cycle for the digital noise filter in the NMIFLTC.NFCLKSEL[1:0] bits.*1
5. Clear the NMICR.NMIMD bit to 0 (falling edge detection).
6. Set the NMICLR.NMICLR bit to 1, and clear the NMISR.NMIST flag to 0.
7. Set the NMIFLTE.NFLTEN bit to 1.*1
8. Set the P35 I/O select bit in the port direction register (PDR) of the I/O port to 10b (input).

Note 1. This setting is only required when the digital noise filter is to be used.

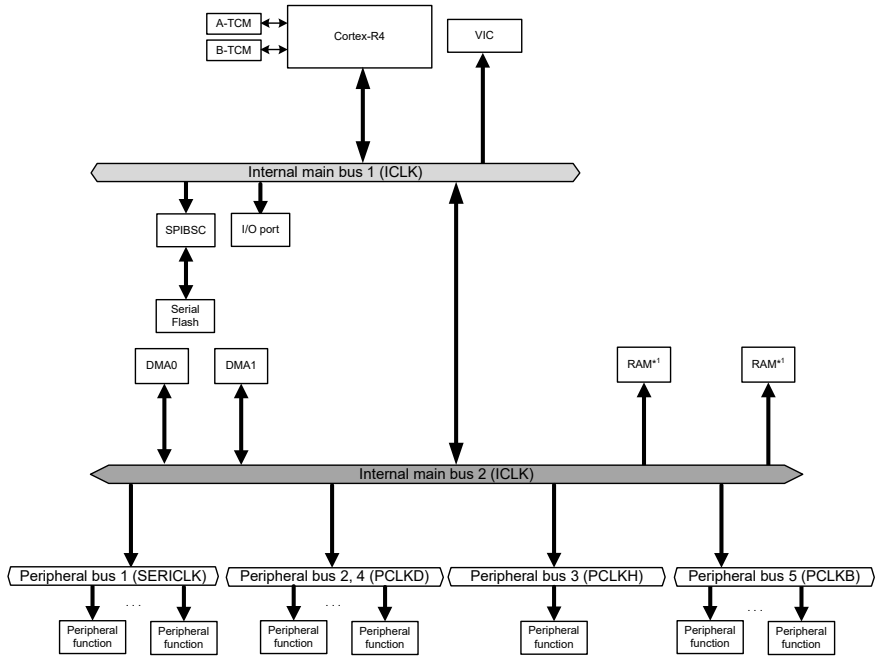
13. Internal Buses

13.1 Overview

This product contains two internal memory buses, two memory buses, and multiple internal peripheral buses. Table 13.1 lists the specifications of internal buses and Figure 13.1 shows the internal bus configuration.

Table 13.1 Specifications of Internal Buses

Internal Bus Type		Description
Internal main bus	Internal main bus 1	• Operates in synchronization with ICLK • Bus protocol: AMBA AXI • Priority order decision: Round-robin
	Internal main bus 2	• Operates in synchronization with ICLK • Bus protocol: AMBA AHB • Priority order decision: Round-robin with fixed priority order (DMA0 with top priority)
Peripheral bus 1	SCIFA, RSPIa	Operates in synchronization with SERICLK
Peripheral bus 2	CRC, ECM	Operates in synchronization with PCLKD
Peripheral bus 3	ADC	Operates in synchronization with PCLKH
Peripheral bus 4	ELC, TPU, CMT, CMTW, WDTA, IWDTa, RIIC, DOC, temperature sensor	Operates in synchronization with PCLKD
Peripheral bus 5	Clock generation circuit, CLMA	Operates in synchronization with PCLKB
External serial flash bus		Operates in synchronization with ICLK



Note 1. Optional function depending on the product

Figure 13.1 Bus Configuration

13.2 Internal Main Bus

Internal main buses 1 and 2 of this LSI both have a multiple-layer structure. If bus masters request to access different bus slaves respectively, multiple accesses are processed in parallel. If bus masters request to access the same bus slave, priority order decision is performed and accesses are processed sequentially according to the priority order.
Table 13.2 lists the connection between the bus master and the bus slave for internal main bus 1, and Table 13.3 lists the bus master-slave connection for internal main bus 2.

Table 13.2 Internal Main Bus 1: Connection between Bus Master and Bus Slave

Bus Slave \ Bus Master	CPU (Cortex-R4)	Internal Main Bus 2
CPU (Cortex-R4)	A	A
I/O port	A	A
SPIBSC	A	A
VIC	A	A
Internal Main Bus 2	A	—

A: Accessible
—: Inaccessible

Table 13.3 Internal Main Bus 2: Connection between Bus Master and Bus Slave

Bus Slave \ Bus Master	Internal Main Bus 1	DMA0	DMA1
Internal main bus 1	—	A	A
DMA0	A	—	—
DMA1	A	—	—
Extended internal RAM	A	A	A
Peripheral bus 1	A	A	A
Peripheral bus 2	A	A	A
Peripheral bus 3	A	A	A
Peripheral bus 4	A	A	A
Peripheral bus 5	A	A	A

A: Accessible
—: Inaccessible

14. DMA Controller (DMACAa)

This LSI contains DMAC (Direct Memory Access Controller) consisting of two units, DMAC0 and DMAC1. DMAC transfers data without using the CPU. When a transfer is requested, DMAC transfers data stored at the transfer source address to the transfer destination address.

14.1 Overview

Table 14.1 lists specifications of DMAC.

Table 14.1 Specifications of DMAC

Item		Description	
		DMAC0	**DMAC1**
Number of channels		16 channels	16 channels
Address space		4 Gbytes	
DMAC activation source		External interrupts (IRQ) On-chip peripheral module requests/software requests*1	
Channel priority		• Selectable from fixed priority or round-robin for channels 0 to 7 and for channels 8 to 15. • Round-robin arbitration between channels 0 to 7 and channels 8 to 15	
Transfer data unit		8, 16, 32, 128, 256, and 512 bits	8, 16, 32, and 128 bits
Maximum transfer size		$2^{32} - 1$ bytes	
Transfer mode	Single transfer	Performs DMA transfer for each DMA transfer request.	
	Block transfer	Performs DMA transfer of the specified size for a single DMAC activation request.	
DMA mode	Register mode	• DMA transfer setting value: Control register value within the DMA controller • DMA transfer to the source/destination specified by a register	
	Link mode	• DMA transfer setting value: Descriptors in the internal RAM or an external memory. • Various DMA transfers specified by descriptors can be performed (responsiveness: register mode > link mode).	
Interval function		The DMA transfer interval can be specified (bus occupation ratio can be adjusted).	
Skip function		• For the area to be accessed by DMA transfer, the continuous access size and the discrete access size (skip) can be set separately. • After the size of data specified for the continuous access is transferred, the specified number of addresses to be accessed next can be skipped.	
Suspending function		Current DMA transfer can be paused.	
Buffer flush function		When DMAC is stopped forcibly, data in the buffer can be flushed.	
Interrupt request		Each channel has the following interrupt requests: • Transfer completion: Indicates completion of transfer of the specified size; each channel has this source. • Transfer error: Indicates a bus error; total of two sources, one for unit 0 and the other for unit 1.	

Note 1. A software request is output as a source of an on-chip peripheral module request from the interrupt controller. For how to set a software request, see section 12, Interrupt Controller (ICUA).

14.2 Register Descriptions

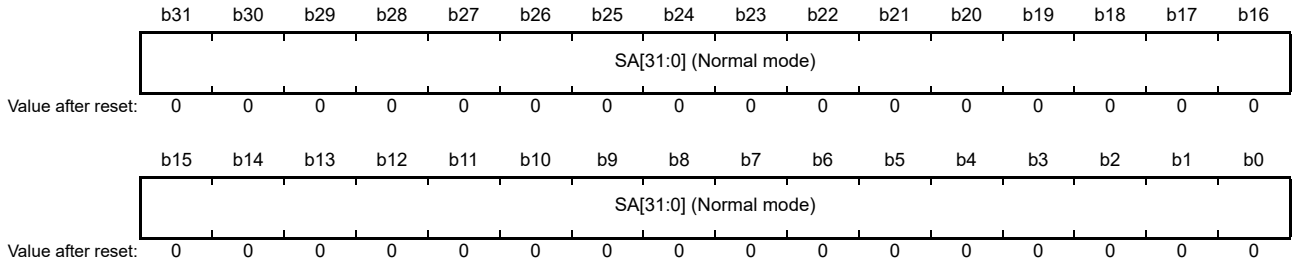
14.2.1 Next Source Address Register n (N0SA_n_N, N0SA_n_W, N1SA_n_N, N1SA_n_W)

The N0SA_n and N1SA_n registers set the DMA transfer source address of DMA channel n (n = 15 to 0).
The N0SA_n register is for Next0 Register Set, and the N1SA_n register is for Next1 Register Set.
In write-only mode (CHCFG_n register WONLY = 1), this register is used to set writing data.

- For N0SA_n_N and N1SA_n_N (normal mode)

Address(es): DMAC0
N0SA_0_N: A006 2000h, N0SA_1_N: A006 2040h, N0SA_2_N: A006 2080h, N0SA_3_N: A006 20C0h,
N0SA_4_N: A006 2100h, N0SA_5_N: A006 2140h, N0SA_6_N: A006 2180h, N0SA_7_N: A006 21C0h,
N0SA_8_N: A006 2400h, N0SA_9_N: A006 2440h, N0SA_10_N: A006 2480h, N0SA_11_N: A006 24C0h,
N0SA_12_N: A006 2500h, N0SA_13_N: A006 2540h, N0SA_14_N: A006 2580h, N0SA_15_N: A006 25C0h
DMAC1
N0SA_0_N: A006 3000h, N0SA_1_N: A006 3040h, N0SA_2_N: A006 3080h, N0SA_3_N: A006 30C0h,
N0SA_4_N: A006 3100h, N0SA_5_N: A006 3140h, N0SA_6_N: A006 3180h, N0SA_7_N: A006 31C0h,
N0SA_8_N: A006 3400h, N0SA_9_N: A006 3440h, N0SA_10_N: A006 3480h, N0SA_11_N: A006 34C0h,
N0SA_12_N: A006 3500h, N0SA_13_N: A006 3540h, N0SA_14_N: A006 3580h, N0SA_15_N: A006 35C0h

DMAC0
N1SA_0_N: A006 200Ch, N1SA_1_N: A006 204Ch, N1SA_2_N: A006 208Ch, N1SA_3_N: A006 20CCh,
N1SA_4_N: A006 210Ch, N1SA_5_N: A006 214Ch, N1SA_6_N: A006 218Ch, N1SA_7_N: A006 21CCh,
N1SA_8_N: A006 240Ch, N1SA_9_N: A006 244Ch, N1SA_10_N: A006 248Ch, N1SA_11_N: A006 24CCh,
N1SA_12_N: A006 250Ch, N1SA_13_N: A006 254Ch, N1SA_14_N: A006 258Ch, N1SA_15_N: A006 25CCh
DMAC1
N1SA_0_N: A006 300Ch, N1SA_1_N: A006 304Ch, N1SA_2_N: A006 308Ch, N1SA_3_N: A006 30CCh,
N1SA_4_N: A006 310Ch, N1SA_5_N: A006 314Ch, N1SA_6_N: A006 318Ch, N1SA_7_N: A006 31CCh,
N1SA_8_N: A006 340Ch, N1SA_9_N: A006 344Ch, N1SA_10_N: A006 348Ch, N1SA_11_N: A006 34CCh,
N1SA_12_N: A006 350Ch, N1SA_13_N: A006 354Ch, N1SA_14_N: A006 358Ch, N1SA_15_N: A006 35CCh



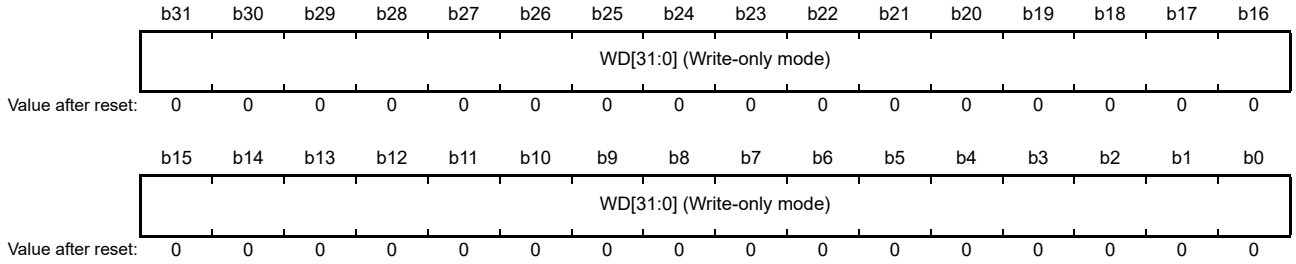
Bit	Symbol	Bit Name	Description	R/W
b31 to b0	SA[31:0] (Normal mode)	Source Address	Sets the start address of the DMA transfer source.	R/W

Note: During link mode transfer, descriptor read data is set to the N0SA_n_N register automatically.

- For N0SA_n_W and N1SA_n_W (write-only mode)

Address(es): DMAC0
N0SA_0_W: A006 2000h, N0SA_1_W: A006 2040h, N0SA_2_W: A006 2080h, N0SA_3_W: A006 20C0h,
N0SA_4_W: A006 2100h, N0SA_5_W: A006 2140h, N0SA_6_W: A006 2180h, N0SA_7_W: A006 21C0h,
N0SA_8_W: A006 2400h, N0SA_9_W: A006 2440h, N0SA_10_W: A006 2480h, N0SA_11_W: A006 24C0h,
N0SA_12_W: A006 2500h, N0SA_13_W: A006 2540h, N0SA_14_W: A006 2580h, N0SA_15_W: A006 25C0h
DMAC1
N0SA_0_W: A006 3000h, N0SA_1_W: A006 3040h, N0SA_2_W: A006 3080h, N0SA_3_W: A006 30C0h,
N0SA_4_W: A006 3100h, N0SA_5_W: A006 3140h, N0SA_6_W: A006 3180h, N0SA_7_W: A006 31C0h,
N0SA_8_W: A006 3400h, N0SA_9_W: A006 3440h, N0SA_10_W: A006 3480h, N0SA_11_W: A006 34C0h,
N0SA_12_W: A006 3500h, N0SA_13_W: A006 3540h, N0SA_14_W: A006 3580h, N0SA_15_W: A006 35C0h

DMAC0
N1SA_0_W: A006 200Ch, N1SA_1_W: A006 204Ch, N1SA_2_W: A006 208Ch, N1SA_3_W: A006 20CCh,
N1SA_4_W: A006 210Ch, N1SA_5_W: A006 214Ch, N1SA_6_W: A006 218Ch, N1SA_7_W: A006 21CCh,
N1SA_8_W: A006 240Ch, N1SA_9_W: A006 244Ch, N1SA_10_W: A006 248Ch, N1SA_11_W: A006 24CCh,
N1SA_12_W: A006 250Ch, N1SA_13_W: A006 254Ch, N1SA_14_W: A006 258Ch, N1SA_15_W: A006 25CCh
DMAC1
N1SA_0_W: A006 300Ch, N1SA_1_W: A006 304Ch, N1SA_2_W: A006 308Ch, N1SA_3_W: A006 30CCh,
N1SA_4_W: A006 310Ch, N1SA_5_W: A006 314Ch, N1SA_6_W: A006 318Ch, N1SA_7_W: A006 31CCh,
N1SA_8_W: A006 340Ch, N1SA_9_W: A006 344Ch, N1SA_10_W: A006 348Ch, N1SA_11_W: A006 34CCh,
N1SA_12_W: A006 350Ch, N1SA_13_W: A006 354Ch, N1SA_14_W: A006 358Ch, N1SA_15_W: A006 35CCh



Bit	Symbol	Bit Name	Description	R/W
b31 to b0	WD[31:0] (Write-only mode)	Write Data	Sets write data for write-only mode.	R/W

Note: During link mode transfer, descriptor read data is set to the N0SA_n_W register automatically.

14.2.2 Next Destination Address Register n (N0DA_n and N1DA_n)

The N0DA_n and N1DA_n registers set the DMA transfer destination address of DMA channel n (n = 15 to 0).
The N0DA_n register is for Next0 Register Set, and the N1DA_n register is for the Next1 Register Set.

Address(es): DMAC0
N0DA_0: A006 2004h, N0DA_1: A006 2044h, N0DA_2: A006 2084h, N0DA_3: A006 20C4h,
N0DA_4: A006 2104h, N0DA_5: A006 2144h, N0DA_6: A006 2184h, N0DA_7: A006 21C4h,
N0DA_8: A006 2404h, N0DA_9: A006 2444h, N0DA_10: A006 2484h, N0DA_11: A006 24C4h,
N0DA_12: A006 2504h, N0DA_13: A006 2544h, N0DA_14: A006 2584h, N0DA_15: A006 25C4h
DMAC1
N0DA_0: A006 3004h, N0DA_1: A006 3044h, N0DA_2: A006 3084h, N0DA_3: A006 30C4h,
N0DA_4: A006 3104h, N0DA_5: A006 3144h, N0DA_6: A006 3184h, N0DA_7: A006 31C4h,
N0DA_8: A006 3404h, N0DA_9: A006 3444h, N0DA_10: A006 3484h, N0DA_11: A006 34C4h,
N0DA_12: A006 3504h, N0DA_13: A006 3544h, N0DA_14: A006 3584h, N0DA_15: A006 35C4h

DMAC0
N1DA_0: A006 2010h, N1DA_1: A006 2050h, N1DA_2: A006 2090h, N1DA_3: A006 20D0h,
N1DA_4: A006 2110h, N1DA_5: A006 2150h, N1DA_6: A006 2190h, N1DA_7: A006 21D0h,
N1DA_8: A006 2410h, N1DA_9: A006 2450h, N1DA_10: A006 2490h, N1DA_11: A006 24D0h,
N1DA_12: A006 2510h, N1DA_13: A006 2550h, N1DA_14: A006 2590h, N1DA_15: A006 25D0h
DMAC1
N1DA_0: A006 3010h, N1DA_1: A006 3050h, N1DA_2: A006 3090h, N1DA_3: A006 30D0h,
N1DA_4: A006 3110h, N1DA_5: A006 3150h, N1DA_6: A006 3190h, N1DA_7: A006 31D0h,
N1DA_8: A006 3410h, N1DA_9: A006 3450h, N1DA_10: A006 3490h, N1DA_11: A006 34D0h,
N1DA_12: A006 3510h, N1DA_13: A006 3550h, N1DA_14: A006 3590h, N1DA_15: A006 35D0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	DA[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	DA[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	DA[31:0]	Destination Address	Sets the start address of the DMA transfer destination.	R/W

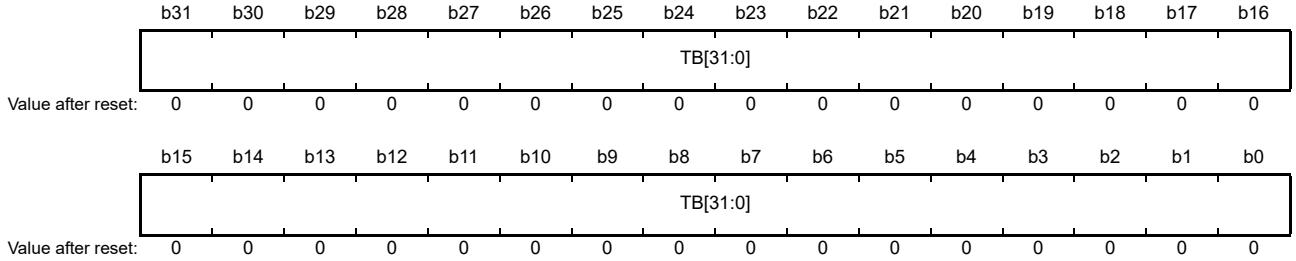
Note: During link mode transfer, descriptor read data is set to the N0DA_n register automatically.

14.2.3 Next Transaction Byte Register n (N0TB_n and N1TB_n)

The N0TB_n and N1TB_n registers set the total number of transfer bytes of DMA channel n (n = 15 to 0).
The N0TB_n register is for Next0 Register Set, and the N1TB_n register is for Next1 Register Set.

Address(es): DMAC0
N0TB_0: A006 2008h, N0TB_1: A006 2048h, N0TB_2: A006 2088h, N0TB_3: A006 20C8h,
N0TB_4: A006 2108h, N0TB_5: A006 2148h, N0TB_6: A006 2188h, N0TB_7: A006 21C8h,
N0TB_8: A006 2408h, N0TB_9: A006 2448h, N0TB_10: A006 2488h, N0TB_11: A006 24C8h,
N0TB_12: A006 2508h, N0TB_13: A006 2548h, N0TB_14: A006 2588h, N0TB_15: A006 25C8h
DMAC1
N0TB_0: A006 3008h, N0TB_1: A006 3048h, N0TB_2: A006 3088h, N0TB_3: A006 30C8h,
N0TB_4: A006 3108h, N0TB_5: A006 3148h, N0TB_6: A006 3188h, N0TB_7: A006 31C8h,
N0TB_8: A006 3408h, N0TB_9: A006 3448h, N0TB_10: A006 3488h, N0TB_11: A006 34C8h,
N0TB_12: A006 3508h, N0TB_13: A006 3548h, N0TB_14: A006 3588h, N0TB_15: A006 35C8h

DMAC0
N1TB_0: A006 2014h, N1TB_1: A006 2054h, N1TB_2: A006 2094h, N1TB_3: A006 20D4h,
N1TB_4: A006 2114h, N1TB_5: A006 2154h, N1TB_6: A006 2194h, N1TB_7: A006 21D4h,
N1TB_8: A006 2414h, N1TB_9: A006 2454h, N1TB_10: A006 2494h, N1TB_11: A006 24D4h,
N1TB_12: A006 2514h, N1TB_13: A006 2554h, N1TB_14: A006 2594h, N1TB_15: A006 25D4h
DMAC1
N1TB_0: A006 3014h, N1TB_1: A006 3054h, N1TB_2: A006 3094h, N1TB_3: A006 30D4h,
N1TB_4: A006 3114h, N1TB_5: A006 3154h, N1TB_6: A006 3194h, N1TB_7: A006 31D4h,
N1TB_8: A006 3414h, N1TB_9: A006 3454h, N1TB_10: A006 3494h, N1TB_11: A006 34D4h,
N1TB_12: A006 3514h, N1TB_13: A006 3554h, N1TB_14: A006 3594h, N1TB_15: A006 35D4h



Bit	Symbol	Bit Name	Description	R/W
b31 to b0	TB[31:0]	Transaction Byte	Sets the total number of transfer bytes. Note 1. Do not start DMA transfer when 0 is set.	R/W

Note: During link mode transfer, descriptor read data is set to the N0TB_n register automatically.

14.2.4 Current Source Address Register (CRSA_n)

The CRSA_n register is a register that indicates the DMA transfer source address of DMA channel n (n = 15 to 0).

During DMA transfer, the value is incremented automatically (fixed when the SAD bit in the CHCFG_n register = 1, and not fixed when the WONLY bit in the CHCFG_n register = 1).

Address(es): DMAC0
CRSA_0: A006 2018h, CRSA_1: A006 2058h, CRSA_2: A006 2098h, CRSA_3: A006 20D8h,
CRSA_4: A006 2118h, CRSA_5: A006 2158h, CRSA_6: A006 2198h, CRSA_7: A006 21D8h,
CRSA_8: A006 2418h, CRSA_9: A006 2458h, CRSA_10: A006 2498h, CRSA_11: A006 24D8h,
CRSA_12: A006 2518h, CRSA_13: A006 2558h, CRSA_14: A006 2598h, CRSA_15: A006 25D8h
DMAC1
CRSA_0: A006 3018h, CRSA_1: A006 3058h, CRSA_2: A006 3098h, CRSA_3: A006 30D8h,
CRSA_4: A006 3118h, CRSA_5: A006 3158h, CRSA_6: A006 3198h, CRSA_7: A006 31D8h,
CRSA_8: A006 3418h, CRSA_9: A006 3458h, CRSA_10: A006 3498h, CRSA_11: A006 34D8h,
CRSA_12: A006 3518h, CRSA_13: A006 3558h, CRSA_14: A006 3598h, CRSA_15: A006 35D8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	CRSA[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	CRSA[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	CRSA[31:0]	Current Source Address	Indicates the read address of the next DMA transfer.	R

The value after a reset is loaded from the following registers:

In register mode:
Loads the transfer source address from the Next0/1 register.

In link mode:
Loads the transfer source address from the descriptor read data (The hardware inputs descriptor read data to the N0SA_n register automatically, and loads it to the CRSA_n register when transfer starts).

The value is incremented when reading data from the transfer source is completed.

Read this register after DMA stops (when the TACT bit in the CHSTAT_n register = 0). (Handle the value during DMA operation as a reference value.)

14.2.5 Current Destination Address Register (CRDA_n)

The CRDA_n register indicates the DMA transfer destination address of DMA channel n (n = 15 to 0).
During DMA transfer, the value is incremented automatically (fixed when the DAD bit in the CHCFG_n register = 1).

Address(es): DMAC0
CRDA_0: A006 201Ch, CRDA_1: A006 205Ch, CRDA_2: A006 209Ch, CRDA_3: A006 20DCh,
CRDA_4: A006 211Ch, CRDA_5: A006 215Ch, CRDA_6: A006 219Ch, CRDA_7: A006 21DCh,
CRDA_8: A006 241Ch, CRDA_9: A006 245Ch, CRDA_10: A006 249Ch, CRDA_11: A006 24DCh,
CRDA_12: A006 251Ch, CRDA_13: A006 255Ch, CRDA_14: A006 259Ch, CRDA_15: A006 25DCh
DMAC1
CRDA_0: A006 301Ch, CRDA_1: A006 305Ch, CRDA_2: A006 309Ch, CRDA_3: A006 30DCh,
CRDA_4: A006 311Ch, CRDA_5: A006 315Ch, CRDA_6: A006 319Ch, CRDA_7: A006 31DCh,
CRDA_8: A006 341Ch, CRDA_9: A006 345Ch, CRDA_10: A006 349Ch, CRDA_11: A006 34DCh,
CRDA_12: A006 351Ch, CRDA_13: A006 355Ch, CRDA_14: A006 359Ch, CRDA_15: A006 35DCh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	CRDA[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	CRDA[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	CRDA[31:0]	Current Destination Address	Indicates the write address of the next DMA transfer.	R

The value after a reset is loaded from the following registers:

In register mode:
Loads the transfer destination address from the Next0/1 register.

In link mode:
Loads the transfer destination address from the descriptor read data (The hardware inputs the descriptor read data to the N0DA_n register automatically, and loads it to the CRDA_n register when transfer starts).

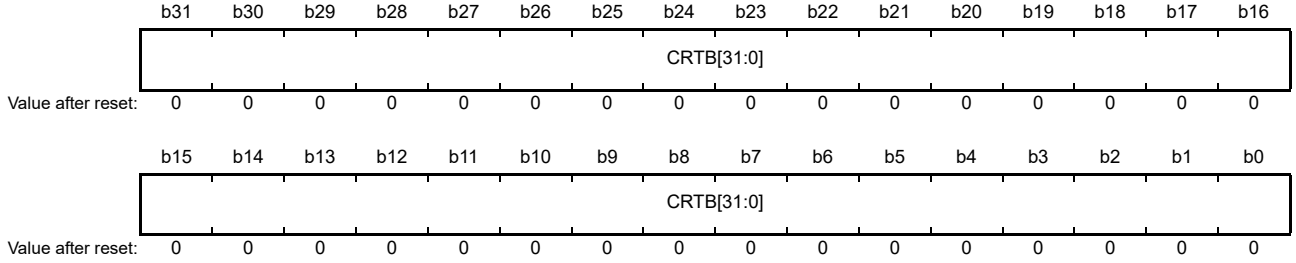
The value is incremented when writing data to the transfer destination completes.

Read this register after DMA stops (when the TACT bit in the CHSTAT_n register = 0). (Handle the value during DMA operation as a reference value.)

14.2.6 Current Transaction Byte Register (CRTB_n)

The CRTB_n register indicates the total number of transfer bytes of DMA channel n (n = 15 to 0). The value is cleared to 0 when transfer completes.
During DMA transfer, the value is decremented automatically.

Address(es): DMAC0
CRTB_0: A006 2020h, CRTB_1: A006 2060h, CRTB_2: A006 20A0h, CRTB_3: A006 20E0h,
CRTB_4: A006 2120h, CRTB_5: A006 2160h, CRTB_6: A006 21A0h, CRTB_7: A006 21E0h,
CRTB_8: A006 2420h, CRTB_9: A006 2460h, CRTB_10: A006 24A0h, CRTB_11: A006 24E0h,
CRTB_12: A006 2520h, CRTB_13: A006 2560h, CRTB_14: A006 25A0h, CRTB_15: A006 25E0h
DMAC1
CRTB_0: A006 3020h, CRTB_1: A006 3060h, CRTB_2: A006 30A0h, CRTB_3: A006 30E0h,
CRTB_4: A006 3120h, CRTB_5: A006 3160h, CRTB_6: A006 31A0h, CRTB_7: A006 31E0h,
CRTB_8: A006 3420h, CRTB_9: A006 3460h, CRTB_10: A006 34A0h, CRTB_11: A006 34E0h,
CRTB_12: A006 3520h, CRTB_13: A006 3560h, CRTB_14: A006 35A0h, CRTB_15: A006 35E0h



Bit	Symbol	Bit Name	Description	R/W
b31 to b0	CRTB[31:0]	Current Transaction Byte	Indicates the remaining number of transfer bytes that is currently performed.	R

The value after a reset is loaded from the following registers:

In register mode:
Loads the number of transfer bytes from the Next0/1 register.

In link mode:
Loads the number of transfer bytes from descriptor read data (The hardware inputs the descriptor read data to the N0TB_n register automatically, and loads it to the CRTB_n register when transfer starts).

The value is decremented when writing data to the transfer destination completes.

Read this register after DMA stops (when the TACT bit in the CHSTAT_n register = 0). (Handle the value during DMA operation as a reference value.)

14.2.7 Channel Status Register n (CHSTAT_n)

The CHSTAT_n register indicates the status of DMA channel n (n = 15 to 0).

Address(es): DMAC0
CHSTAT_0: A006 2024h, CHSTAT_1: A006 2064h, CHSTAT_2: A006 20A4h, CHSTAT_3: A006 20E4h,
CHSTAT_4: A006 2124h, CHSTAT_5: A006 2164h, CHSTAT_6: A006 21A4h, CHSTAT_7: A006 21E4h,
CHSTAT_8: A006 2424h, CHSTAT_9: A006 2464h, CHSTAT_10: A006 24A4h, CHSTAT_11: A006 24E4h,
CHSTAT_12: A006 2524h, CHSTAT_13: A006 2564h, CHSTAT_14: A006 25A4h, CHSTAT_15: A006 25E4h
DMAC1
CHSTAT_0: A006 3024h, CHSTAT_1: A006 3064h, CHSTAT_2: A006 30A4h, CHSTAT_3: A006 30E4h,
CHSTAT_4: A006 3124h, CHSTAT_5: A006 3164h, CHSTAT_6: A006 31A4h, CHSTAT_7: A006 31E4h,
CHSTAT_8: A006 3424h, CHSTAT_9: A006 3464h, CHSTAT_10: A006 34A4h, CHSTAT_11: A006 34E4h,
CHSTAT_12: A006 3524h, CHSTAT_13: A006 3564h, CHSTAT_14: A006 35A4h, CHSTAT_15: A006 35E4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	DNUM[7:0]								—	—	—	—	—	SWPRQ	DMARQM	INTM
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	MODE	DER	DW	DL	SR	—	END	ER	SUS	TACT	RQST	EN
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	EN	DMA Activation Enable	Indicates the status of operation enable/disable of DMA channel n. 0: Operation is disabled. 1: Operation is enabled. [Setting condition] • Writes 1 to the SETEN bit in the CHCTRL_n register. [Clearing conditions] When any of the following conditions is met: • Writes 1 to the CLREN bit in the CHCTRL_n register. • When a bus error was reported during transfer • When all DMA transfers complete in register mode (transfers complete when the REN bit in the CHCFG_n register = 0) • In link mode, when DMA transfer (write back when WBD = 0) of the last descriptor (LE = 1) completes • When reading the descriptor in link mode stopped (when LV = 0, and DRRP in the CHCFG_n register = 0)	R

Bit	Symbol	Bit Name	Description	R/W
b1	RQST	DMA Transfer Request	This bit indicates that the transfer request is accepted. 0: The DMA transfer request is not accepted. 1: The DMA transfer request is accepted. [Setting condition] • The transfer request is accepted. [Clearing conditions] When any of the following conditions is met: • Writes 1 to the SWRST bit in the CHCTRL_n register. • Writes 1 to the CLRRQ bit in the CHCTRL_n register. • When all DMA transfers complete in register mode (transfer completes when the REN bit in the CHCFG_n register = 0) • In link mode, DMA transfer of the last descriptor (LE = 1) completes • In link mode, reading the descriptor is disabled (when LV = 0, and DRRP in the CHCFG_n register = 0) • In link mode, the DEM bit in the CHCFG_n register = 0, and DMA transfer completes. • When a buss error was reported to the master interface	R
b2	TACT	DMAC Operating Status	This bit indicates that DMAC is running. This bit is used to make sure the channel stops completely. For details, see section 14.3.8, DMA Transfer Status. 0: DMA in Channel_n stops. 1: DMA in Channel_n is running. [Setting condition] • Write 1 to the SETEN bit in the CHCTRL_n register (Reading the descriptor starts, or the DMA request is being waited). [Clearing condition] • When the internal state is idling (the EN bit in the CHSTAT_n register is cleared to 0, and all DMA transfer completes).	R
b3	SUS	Suspend	This bit indicates that the channel is suspended. For details, see section 14.3.9, Suspending a Transfer. 0: Channel_n is not suspended. 1: Channel_n is suspended. [Setting condition] • Writes 1 to the SETSUS bit in the CHCTRL_n register during DMA transfer of Channel_n, and the internal state is changed to the suspended state. [Clearing conditions] When any of the following conditions is met: • Writes 1 to the CLRSUS bit in the CHCTRL_n register. • Writes 1 to the CLREN bit in the CHCTRL_n register. • Condition for clearing the EN bit in the CHSTAT_n register	R
b4	ER	DMA Error	This bit indicates that a DMA error interrupt is generated as a result of the bus error during DMA transfer. 0: No bus error occurred. 1: A bus error occurred. [Setting condition] • A buss error is reported to the bus cycle [Clearing condition] • Writes 1 to the SWRST bit in the CHCTRL_n register.	R

Bit	Symbol	Bit Name	Description	R/W
b5	END	DMA Transfer Completion Interrupt	This bit indicates that DMA transfer completes, and a DMA interrupt is generated. 0: DMA transfer is not completed. 1: DMA transfer is completed. [Setting conditions] When any of the following conditions is met: • The following conditions being met while the DEM bit in the CHCFG_n register = 0 (1) When a transfer for the total number of transfer bytes loaded to the CRTB register completes in register mode (2) When a transfer for the total number of transfer bytes loaded to the CRTB register completes in link mode while the WBD bit in the header of the descriptor is 1 (3) Completion of descriptor write-back in link mode while the WBD bit in the header of the descriptor is 0 • In link mode, the descriptor is read, LV of header = 0, and DRRP in the CHCFG_n register = 0, and DIM = 0. [Clearing conditions] When any of the following conditions is met: • Writes 1 to the CLREND bit in the CHCTRL_n register. • Writes 1 to the SWRST bit in the CHCTRL_n register.	R
b6	—	Reserved	This bit is always read as 0.	R
b7	SR	Next Register Select	In register mode, indicates the selected register set. 0: Next0 Register Set 1: Next1 Register Set [Setting condition] • Set 1 to the RSEL bit in the CHCFG_n register. [Clearing condition] • Clears the RSEL bit in the CHCFG_n register to 0.	R
b8	DL	Descriptor Load	Indicates that the descriptor is being read. In addition, if a bus error is reported while the descriptor is being read, 1 is retained. 0: The descriptor is not being read. 1: (When ER = 0) The descriptor is being read in link mode. (When ER = 1) A buss error occurred while the descriptor is being read in link mode. [Setting condition] • Reading the descriptor in link mode is started [Clearing conditions] When any of the following conditions is met: • Reading the descriptor in link mode completes with the OK response. • Writes 1 to the SWRST bit in the CHCTRL_n register. (If 1 is retained for the bus error, it can be cleared to 0 only by the SWRST bit.)	R

Bit	Symbol	Bit Name	Description	R/W
b9	DW	Descriptor Write Back	Indicates that the descriptor is being written back. In addition, if a bus error is reported while the descriptor is written back, 1 is retained. 0: Header is not written back in link mode. 1: (The ER bit in the CHSTAT_n register = 0) Header is written back in link mode. (The ER bit in the CHSTAT_n register = 1) A buss error occurred when header is written back in link mode. [Setting condition] • Writing back of header in link mode started. [Clearing conditions] • Writing back of header in link mode completes with the OK response. • Writes 1 to the SWRST bit in the CHCTRL_n register (If 1 is retained due to the buss error, it can be cleared to 0 only by the SWRST bit).	R
b10	DER	Descriptor Error	Indicates that the read descriptor is invalid (LV = 0). (Does not depend on the value of the DIM bit in the CHCFG_n register.) 0: No descriptor error occurred. 1: Descriptor error occurred. [Setting condition] • In link mode, the DRRP bit in the CHCFG_n register = 0, and the read descriptor's LV is 0. [Clearing conditions] When any of the following conditions is met: • Writes 1 to the CLRDE bit in the CHCTRL_n register. • Writes 1 to the SWRST bit in the CHCTRL_n register.	R
b11	MODE	DMA Mode	This bit indicates DMA mode. Indicates the setting value of the DMS bit in the CHCFG_n register. 0: Register mode 1: Link mode	R
b15 to b12	—	Reserved	These bits are always read as 0.	R
b16	INTM	Interrupt Request Mask	Indicates the temporary mask status of DMA interrupt output. 1: Temporary mask status 0: The temporary mask status is cleared. [Setting condition] • Writes 1 to the SETINTM bit in the CHCTRL_n register. [Clearing conditions] When any of the following conditions is met: • Writes 1 to the CLRINTM bit in the CHCTRL_n register. • Writes 1 to the SWRST bit in the CHCTRL_n register.	R

Bit	Symbol	Bit Name	Description	R/W
b17	DMARQM	DMA Activation Request Mask	Indicates the temporary mask status of the DMA request. 1: Temporary mask status 0: Temporary mask status is cleared. [Setting condition] • Sets 1 to the SETDMARQM bit in the CHCTRL_n register. [Clearing conditions] When any of the following conditions is met: • Writes 1 to the CLRDMARQM bit in the CHCTRL_n register. • Writes 1 to the SWRST bit in the CHCTRL_n register.	R
b18	SWPRQ	Forced Ejection Request	Indicates the forced ejection request status. Indicates the software forced ejection request (a request activated by the SETSSWPRQ bit in the CHCTRL_n register). 1: Forced ejection is requested. 0: Forced ejection is not requested. [Setting condition] • Sets 1 to the SETSSWPRQ bit in the CHCTRL_n register. [Clearing conditions] When any of the following conditions is met: • The amount of data in the buffer becomes 0 because of forced ejection. • Writes 1 to the SWRST bit in the CHCTRL_n register.	R
b23 to b19	—	Reserved	These bits are always read as 0.	R
b31 to b24	DNUM	Amount of Data in the Buffer	These bits indicate the amount of valid data in the buffer. Read data from the DMA transfer source, and indicate the amount of data that is not written to the transfer destination (in bytes). [Increment condition] • When DMA read transfer completes [Decrement condition] • When DMA write transfer completes [Clearing conditions] When any of the following conditions is met: • Condition for clearing the EN bit • Writes 1 to the SWRST bit in the CHCTRL_n register.	R

Note 1. Handle the transfer when the ER bit in the CHSTAT_n register is set to 1 as invalid.
Note 2. To suspend DMA transfer, mask or clear the transfer request, or clear the EN bit in the CHSTAT_n register. See section 14.3.10, Aborting a Transfer for the procedure).
Note 3. To request a transfer by software, make sure that the previously requested DMA transfer operation completes (by checking Current Register), set the DMREQ bit in the DMAC software activation register (DMASTG), and then activate DMA.

14.2.8 DMAC Unit 0 Source Select Register i (DMA0SELi) (i = 0 to 15)

The DMA0SELi register selects the activation trigger source of channel i (i = 0 to 15) for DMAC unit 0. For details on numbers that are selected by this source selection, see the vector numbers listed in Table 12.3, Cortex-R4/DMAC Interrupt Vector Table. In addition, do not set the same source for multiple DMA0SELi and DMA1SELi registers. If the same source is set, the operation of this LSI cannot be guaranteed.

Address(es): DMA0.DMA0SEL0 A009 4000h, DMA0.DMA0SEL1 A009 4004h, DMA0.DMA0SEL2 A009 4008h, DMA0.DMA0SEL3 A009 400Ch, DMA0.DMA0SEL4 A009 4010h, DMA0.DMA0SEL5 A009 4014h, DMA0.DMA0SEL6 A009 4018h, DMA0.DMA0SEL7 A009 401Ch, DMA0.DMA0SEL8 A009 4020h, DMA0.DMA0SEL9 A009 4024h, DMA0.DMA0SEL10 A009 4028h, DMA0.DMA0SEL11 A009 402Ch, DMA0.DMA0SEL12 A009 4030h, DMA0.DMA0SEL13 A009 4034h, DMA0.DMA0SEL14 A009 4038h, DMA0.DMA0SEL15 A009 403Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	IFC0[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	IFC0[7:0]	DMA channel source select	Select the trigger source of the DMA channel.	R/W
b31 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

14.2.9 DMAC Unit 1 Source Select Register i (DMA1SELi) (i = 0 to 15)

The DMA1SELi register selects the activation trigger source of channel i (i = 0 to 15) for DMAC unit 1. For details on numbers that are selected for this source selection, see the vector numbers listed in Table 12.3, Cortex-R4/DMAC Interrupt Vector Table. In addition, do not set the same source for multiple DMA0SELi and DMA1SELi registers. If the same source is set, the operation cannot be guaranteed.

Address(es): DMA1.DMA1SEL0 A009 4040h, DMA1.DMA1SEL1 A009 4044h, DMA1.DMA1SEL2 A009 4048h, DMA1.DMA1SEL3 A009 404Ch, DMA1.DMA1SEL4 A009 4050h, DMA1.DMA1SEL5 A009 4054h, DMA1.DMA1SEL6 A009 4058h, DMA1.DMA1SEL7 A009 405Ch, DMA1.DMA1SEL8 A009 4060h, DMA1.DMA1SEL9 A009 4064h, DMA1.DMA1SEL10 A009 4068h, DMA1.DMA1SEL11 A009 406Ch, DMA1.DMA1SEL12 A009 4070h, DMA1.DMA1SEL13 A009 4074h, DMA1.DMA1SEL14 A009 4078h, DMA1.DMA1SEL15 A009 407Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	IFC1[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	IFC1[7:0]	DMA channel source select	Select the trigger source of the DMA channel.	R/W
b31 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

14.2.10 DMAC Software Activation Register (DMASTG)

The DMASTG register controls activation of DMAC by software.

Address(es): DMAC.DMASTG A009 4080h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DMREQ1	DMREQ0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	DMREQ0	DMA Unit 0 Software Activation	0: DMA transfer is not requested. 1: DMA transfer is requested.	W
b1	DMREQ1	DMA Unit 1 Software Activation	0: DMA transfer is not requested. 1: DMA transfer is requested.	W
b31 to b2	—	Reserved	The write value should be 0.	W

DMREQ0, DMREQ1 Bits (DMA Unit 0/1 Software Activation)

DMA transfer is requested if you select DMA activation by software for the DMA0SELi register and the DMA1SELi register (i = 0 to 15), and then write 1 to DMREQ0 and DMREQ1 bits.

These bits are write only. These bits are read as 0.

14.2.11 Channel Control Register n (CHCTRL_n)

The CHCTRL_n register controls DMA transfer of DMA channel n (n = 15 to 0).
This register is used to activate each function, and it does not retain the written value. These bits are always read as 0.
Only resources for channel n are masked temporary for forced ejection request and DMA transfer request inputs by the CLRDMARQM and SETDMARQM bits.

Address(es): DMAC0
CHCTRL_0: A006 2028h, CHCTRL_1: A006 2068h, CHCTRL_2: A006 20A8h, CHCTRL_3: A006 20E8h,
CHCTRL_4: A006 2128h, CHCTRL_5: A006 2168h, CHCTRL_6: A006 21A8h, CHCTRL_7: A006 21E8h,
CHCTRL_8: A006 2428h, CHCTRL_9: A006 2468h, CHCTRL_10: A006 24A8h, CHCTRL_11: A006 24E8h,
CHCTRL_12: A006 2528h, CHCTRL_13: A006 2568h, CHCTRL_14: A006 25A8h, CHCTRL_15: A006 25E8h
DMAC1
CHCTRL_0: A006 3028h, CHCTRL_1: A006 3068h, CHCTRL_2: A006 30A8h. CHCTRL_3: A006 30E8h,
CHCTRL_4: A006 3128h, CHCTRL_5: A006 3168h, CHCTRL_6: A006 31A8h, CHCTRL_7: A006 31E8h,
CHCTRL_8: A006 3428h, CHCTRL_9: A006 3468h, CHCTRL_10: A006 34A8h, CHCTRL_11: A006 34E8h,
CHCTRL_12: A006 3528h, CHCTRL_13: A006 3568h, CHCTRL_14: A006 35A8h, CHCTRL_15: A006 35E8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	CLRDMARQM	SETDMARQM	CLRINTM	SETINTM
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	SETSSWPRQ	—	SETREN	—	—	CLRSUS	SETSUS	CLRDE	—	CLRENDD	CLRRQ	SWRST	—	CLREN	SETEN
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SETEN	DMA Activation Enable	Enables DMA transfer of DMA channel n. When this bit is set with the SWRST bit, clearing by the SWRST bit takes precedence, and a transfer does not start. This bit is always read as 0. 1: DMA transfer is enabled (The EN bit in the CHSTAT_n register is set). 0: Operation is not affected. Note: For resetting the DMA register, stop the ongoing DMA transfer by setting the CLREN bit and then set the SETEN bit.	R/W
b1	CLREN	DMA Activation Enable Clear	Clears the EN bit in the CHSTAT_n register (For details, see section 14.3.10, Aborting a Transfer). This bit is always read as 0. 1: DMA transfer is disabled (The EN bit in the CHSTAT_n register is cleared). 0: Operation is not affected.	R/W
b2	—	Reserved	This bit is always read as 0. The write value should be 0.	R/W
b3	SWRST	Software Reset	Clears each bit in the CHSTAT_n register (For details on the bit to be cleared, see descriptions of the applicable bit). Set this bit to 0 when the EN bit and the TACT bit are cleared to 0. This bit is always read as 0. 1: Clears each bit in the CHSTAT_n register. 0: Operation is not affected.	R/W
b4	CLRRQ	DMA Transfer Request Clear	Clears the RQST bit in the CHSTAT_n register to 0. This bit is always read as 0. 1: Clears the RQST bit in the CHSTAT_n register. 0: Operation is not affected.	R/W
b5	CLREND	END Clear	Clears the END bit in the CHSTAT_n register to 0. This bit is always read as 0. 1: Clears the END bit. 0: Operation is not affected.	R/W
b6	—	Reserved	This bit is always read as 0. The write value should be 0.	R/W

Bit	Symbol	Bit Name	Description	R/W
b7	CLRDE	DER Clear	Clears the descriptor error bit (DER) in the CHSTAT_n register to 0. This bit is always read as 0. 1: Clears the DER bit. 0: Operation is not affected.	R/W
b8	SETSUS	Suspend Request	If 1 is set to this bit when the EN bit in the CHSTAT_n register is 1, the current DMA transfer is suspended. This bit is always read as 0. 1: Suspends the current DMA transfer. 0: Operation is not affected.	R/W
b9	CLRSUS	Suspend Clear	If 1 is set to this bit when the SUS bit in the CHSTAT_n register is 1, the temporary suspend status is cleared. This bit is always read as 0. 1: Clears the temporary suspend status of the current DMA transfer. 0: Operation is not affected.	R/W
b11, b10	—	Reserved	These bits are always read as 0. The write value should be 0.	R/W
b12	SETREN	REN Set Enable	Sets the register set enable bit (REN) in the CHCFG_n register. This bit is always read as 0. 1: Sets the REN bit in the CHCFG_n register. 0: Operation is not affected.	R/W
b13	—	Reserved	This bit is always read as 0. The write value should be 0.	R/W
b14	SETSSWPRQ	Software Forced Ejection Request	Forcibly ejects data in the buffer to the transfer destination (See section 14.3.5, Forced Ejection Request). This bit is always read as 0. 1: Writes data in the buffer, which is not written, to the transfer destination. 0: Operation is not affected.	R/W
b15	—	Reserved	This bit is always read as 0. The write value should be 0.	R/W
b16	SETINTM	Interrupt Request Mask	Masks DMA transfer completion interrupt output temporary. Besides, the INTM bit in the CHSTATn register is set to 1. This bit is always read as 0. 1: Masks a DMA transfer completion interrupt temporary. 0: Operation is not affected.	R/W
b17	CLRINTM	Interrupt Request Mask Clear	Clears the mask status of DMA transfer completion interrupt output. Besides, the INTM bit in the CHSTATn register is cleared to 0. If the mask state is cleared when the LVINT bit in the DCTRL register = 1, and the END bit in the CHSTAT_n register = 1, DMA transfer completion interrupt is deactivated. (It is not activated when LVINT = 0.) This bit is always read as 0. 1: Clears the mask state set by the SETINTM bit. 0: Operation is not affected.	R/W
b18	SETDMARQM	DMA Activation Request Mask	Masks DMA transfer request input temporary. Besides, the DMARQM bit in the CHSTATn register is set to 1. This bit is always read as 0. 1: Masks DMA transfer request input. 0: Operation is not affected.	R/W
b19	CLRDMARQM	DMA Activation Request Mask Clear	Clears the mask state of DMA transfer request input. Besides, the DMARQM bit in the CHSTATn register is cleared to 0. This bit is always read as 0. 1: Clears the mask state set by the SETDMARQM bit. 0: Operation is not affected.	R/W
b31 to b20	—	Reserved	These bits are always read as 0. The write value should be 0.	R/W

14.2.12 Channel Configuration Register n (CHCFG_n)

The CHCFG_n register controls DMA transfer of DMA channel n (n = 15 to 0).

Specify the detection method depending on the DMA transfer source to be used. For details on the DMA request signals, see section 14.3.4.1, Specifying Detection Operation of DMA Transfer Requests for Each Source.

Address(es): DMAC0
CHCFG_0: A006 202Ch, CHCFG_1: A006 206Ch, CHCFG_2: A006 20ACh, CHCFG_3: A006 20ECh,
CHCFG_4: A006 212Ch, CHCFG_5: A006 216Ch, CHCFG_6: A006 21ACh, CHCFG_7: A006 21ECh,
CHCFG_8: A006 242Ch, CHCFG_9: A006 246Ch, CHCFG_10: A006 24ACh, CHCFG_11: A006 24ECh,
CHCFG_12: A006 252Ch, CHCFG_13: A006 256Ch, CHCFG_14: A006 25ACh, CHCFG_15: A006 25ECh
DMAC1
CHCFG_0: A006 302Ch, CHCFG_1: A006 306Ch, CHCFG_2: A006 30ACh, CHCFG_3: A006 30ECh,
CHCFG_4: A006 312Ch, CHCFG_5: A006 316Ch, CHCFG_6: A006 31ACh, CHCFG_7: A006 31ECh,
CHCFG_C: A006 342Ch, CHCFG_9: A006 346Ch, CHCFG_10: A006 34ACh, CHCFG_11: A006 34ECh,
CHCFG_12: A006 352Ch, CHCFG_13: A006 356Ch, CHCFG_14: A006 35ACh, CHCFG_15: A006 35ECh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	DMS	REN	RSW	RSEL	SBE	DIM	—	DEM	WONLY	TM	DAD	SAD	DDS[3:0]			
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SDS[3:0]				DRRP	—	—	—	—	LVL	HIEN	LOEN	—	SEL[2:0]		
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	SEL[2:0]	Pin Select	Sets channels of DMAC. Set the following values so that the channels of CHCFG_n (n = 0 to 15) and the channel set by SEL become the same: For example, set 001b to the SEL bit in CHCFG_1. Similarly, set 001b to the SEL bit in CHCFG_1. DMAC0/1 b2 b0 000: Channel 0/8 001: Channel 1/9 010: Channel 2/10 011: Channel 3/11 100: Channel 4/12 101: Channel 5/13 110: Channel 6/14 111: Channel 7/15	R/W
b3	—	Reserved	This bit is always read as 0. The write value should be 0.	R/W
b4	LOEN	L Detection Enable	Specifies the detection method of the DMA request signal. When LVL = 0 LOEN = 1: If DMA transfer request input detects a falling edge, it is regarded as requested. LOEN = 0: If the DMA transfer request input falls, the request is not recognized (a value after a reset). When LVL = 1 LOEN = 1: If DMA transfer request input detects the Low level, it is regarded as requested. LOEN = 0: If the DMA transfer request input is the Low level, the request is not recognized (a value after a reset).	R/W
b5	HIEN	H Detection Enable	Specifies the detection method of the DMA request signal. When LVL = 0 HIEN = 1: If DMA transfer request input detects a rising edge, it is regarded as requested. HIEN = 0: If the DMA transfer request input rises, the request is not recognized (a value after a reset). When LVL = 1 HIEN = 1: If DMA transfer request input detects the High level, it is regarded as requested. HIEN = 0: If the DMA transfer request input is the High level, the request is not recognized (a value after a reset).	R/W

Bit	Symbol	Bit Name	Description	R/W
b6	LVL	Level Detection Enable	Specifies the detection method of the DMA request signal. 0: Detects the edge (a value after a reset). 1: Detects the level.	R/W
b8, b7	—	Reserved	These bits are always read as 0. The write value should be 0.	R/W
b9	—	Reserved	Be sure to write 1 to this bit at the initial setting.	R/W
b10	—	Reserved	This bit is always read as 0. The write value should be 0.	R/W
b11	DRRP	Descriptor Reload Enable	Specifies the operation when LV of header = 0 while during descriptor read (see section 14.3.1.2, Link Mode (1) Operation flows in link mode). 0: Sets the DER bit in the CHSTAT_n register, and stops the operation (a value after a reset). 1: Continues reading the same descriptor until LV becomes 1. When LV becomes 1, DMA transfer using the descriptor value starts. The DSCITVL register controls the descriptor read interval.	R/W
b15 to b12	SDS[3:0]	Source Data Size	Sets the size of data in the transfer source to be transferred at a time. For a single transfer, the specified amount of data is transferred by a single request. For a block transfer, data is transferred as many times as the setting value × N times until the CRTB register is cleared to 0. For a transfer of 32 bits or more, 32 bits × N times burst transfers are performed. The SDS[3] bit switches between normal mode and skip mode. 0: Normal mode (a value after a reset). 1: Skip mode The SDS[2:0] bits set the transfer size. b14 b12 000: 8 bits (a value after a reset) 001: 16 bits 010: 32 bits 011: Setting is prohibited. 100: 128 bits*2 101: 256 bits (Can be set only for DMAC0) 110: 512 bits (Can be set only for DMAC0)*1 111: Setting is prohibited.	R/W
b19 to b16	DDS[3:0]	Destination Data Size	Sets the size of data in the transfer destination to be transferred at a time. The DDS[3] bit switches between normal mode and skip mode. 0: Normal mode (a value after a reset). 1: Skip mode The DDS[2:0] bits set the transfer size. b18 b16 000: 8 bits (a value after a reset) 001: 16 bits 010: 32 bits 011: Setting is prohibited. 100: 128 bits*2 101: 256 bits (Can be set only for DMAC0) 110: 512 bits (Can be set only for DMAC0)*1 111: Setting is prohibited.	R/W
b20	SAD	Source Address Count Direction	Sets the count direction of the transfer source address of DMA channel n. 0: Increment (a value after a reset) 1: Fixed To use SKIP mode on the transfer source, do not specify SAD = 1 (fixed). If SAD = 1 (fixed) is specified, the address of the transfer source should be aligned with a boundary of the size specified by the SDS[3:0] bits.	R/W

Bit	Symbol	Bit Name	Description	R/W
b21	DAD	Destination Address Count Direction	Sets the count direction of the transfer destination address of DMA channel n. 0: Increment (a value after a reset) 1: Fixed To use SKIP mode on the transfer destination, do not specify DAD = 1 (fixed). If DAD = 1 (fixed) is specified, the address of the transfer destination should be aligned with a boundary of the size specified by the DDS[3:0] bits.	R/W
b22	TM	Transfer Mode	Sets DMA transfer mode. 0: Single transfer mode (a value after a reset) 1: Block transfer mode	R/W
b23	WONLY	Write-Only Mode	Sets the write-only mode (see section 14.3.1.3, Write-Only Mode). 0: Normal mode (a value after a reset). 1: Write-only mode	R/W
b24	DEM	Transfer Completion Interrupt Mask	Masks DMA transfer completion interrupt detection. If this bit is set to 1 when DMA transfer completion interrupt is output, DMA transfer completion interrupt is not activated. Besides, the END bit in the CHSTAT_n register is not set. In register mode, the DEM bit is automatically cleared to 0. In link mode, it is not cleared. 0: Does not mask (a value after a reset). 1: Masks. [Clearing condition] DEM = 1, and DMA transfer completes.	R/W
b25	—	Reserved	This bit is always read as 0. The write value should be 0.	R/W
b26	DIM	Descriptor Interrupt Mask	Sets the DMA transfer completion interrupt mask if LV = 0 when header of the descriptor is read. 0: Does not mask a DMA transfer completion interrupt (a value after reset). 1: Masks a DMA transfer completion interrupt.	R/W
b27	SBE	Buffer Flush Enable	If the EN bit in the CHSTAT_n register is cleared to 0 during DMA transfer, selects whether to stop flushing (writing) data that is already read and stored in the buffer. 0: Stops transfer without flushing data in the buffer (a value after a reset). 1: Stops transfer after flushing data in the buffer.	R/W
b28	RSEL	Next Register Select	Selects the Next register set to be executed next. This bit is valid in register mode only. When RSW = 1, the value is reversed automatically (0 is reversed to 1, 1 is reversed to 0) after DMA transfer completes. 0: Executes Next0 Register Set (a value after a reset). 1: Executes Next1 Register Set. [Transition condition] RSW = 1, and DMA transfer completes.	R/W
b29	RSW	RSEL Reverse	When DMA transfer completes, the RSEL bit is automatically reversed (0 is reversed to 1, 1 is reversed to 0). This bit is valid in register mode only. 0: Does not reverse the RSEL bit when DMA transfer completes (a value after a reset). 1: Reverses the RSEL bit when DMA transfer completes.	R/W

Bit	Symbol	Bit Name	Description	R/W
b30	REN	Register Set Enable	When DMA transfer completes, performs DMA transfer of the Next register set selected by the RSEL bit accordingly. This bit is valid in register mode only. 0: Does not perform DMA transfer accordingly. 1: Performs DMA transfer accordingly. [Setting conditions] When any of the following conditions is met: • Writes 1 to this bit. • Writes 1 to the SETREN bit in the CHCTRL_n register. [Clearing conditions] • Writes 0 to this bit. • REN = 1, and DMA transfer completes. To reset the REN bit during DMA transfer, use the SETREN bit in the CHCTRL_n register. Also, reset the CHCFG_n.DEM bit to mask detection of DMA transfer completion interrupt.	R/W
b31	DMS	DMA Mode Select	Sets DMA mode. 0: Register mode (a value after a reset). 1: Link mode	R/W

Note 1. If the size is set to 512 bits, the address of the transfer source/destination should be aligned with 512-bit boundaries.
Note 2. If the size is set to 128 bits when DMAC1 is used, the address of the transfer source/destination should be aligned with 128-bit boundaries.

14.2.13 Channel Interval Register n (CHITVL_n)

The CHITVL_n register sets the DMA transfer interval of DMA channel n (n = 15 to 0).
For details, see section 14.3.6, Interval Count Function.

Address(es): DMAC0
CHITVL_0: A006 2030h, CHITVL_1: A006 2070h, CHITVL_2: A006 20B0h, CHITVL_3: A006 20F0h,
CHITVL_4: A006 2130h, CHITVL_5: A006 2170h, CHITVL_6: A006 21B0h, CHITVL_7: A006 21F0h,
CHITVL_8: A006 2430h, CHITVL_9: A006 2470h, CHITVL_10: A006 24B0h, CHITVL_11: A006 24F0h,
CHITVL_12: A006 2530h, CHITVL_13: A006 2570h, CHITVL_14: A006 25B0h, CHITVL_15: A006 25F0h
DMAC1
CHITVL_0: A006 3030h, CHITVL_1: A006 3070h, CHITVL_2: A006 30B0h, CHITVL_3: A006 30F0h,
CHITVL_4: A006 3130h, CHITVL_5: A006 3170h, CHITVL_6: A006 31B0h, CHITVL_7: A006 31F0h,
CHITVL_8: A006 3430h, CHITVL_9: A006 3470h, CHITVL_10: A006 34B0h, CHITVL_11: A006 34F0h,
CHITVL_12: A006 3530h, CHITVL_13: A006 3570h, CHITVL_14: A006 35B0h, CHITVL_15: A006 35F0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ITVL[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	ITVL	Interval	These bits set the DMA transfer internal.	R/W
b31 to b16	—	Reserved	These bits are always read as 0. The write value should be 0.	R/W

14.2.14 Next Link Address Register n (NXLA_n)

The NXLA_n register sets the link address of DMA channel n (n = 15 to 0).

Address(es): DMAC0
NXLA_0: A006 2038h, NXLA_1: A006 2078h, NXLA_2: A006 20B8h, NXLA_3: A006 20F8h,
NXLA_4: A006 2138h, NXLA_5: A006 2178h, NXLA_6: A006 21B8h, NXLA_7: A006 21F8h,
NXLA_8: A006 2438h, NXLA_9: A006 2478h, NXLA_10: A006 24B8h, NXLA_11: A006 24F8h,
NXLA_12: A006 2538h, NXLA_13: A006 2578h, NXLA_14: A006 25B8h, NXLA_15: A006 25F8h
DMAC1
NXLA_0: A006 3038h, NXLA_1: A006 3078h, NXLA_2: A006 30B8h, NXLA_3: A006 30F8h,
NXLA_4: A006 3138h, NXLA_5: A006 3178h, NXLA_6: A006 31B8h, NXLA_7: A006 31F8h,
NXLA_8: A006 3438h, NXLA_9: A006 3478h, NXLA_10: A006 34B8h, NXLA_11: A006 34F8h,
NXLA_12: A006 3538h, NXLA_13: A006 3578h, NXLA_14: A006 35B8h, NXLA_15: A006 35F8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	NXLA[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	NXLA[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	NXLA[31:0]	Next Link Address	Sets the address of the link destination. Because the two lower-order bits are fixed to 0, only word-aligned addresses can be set.	R/W
b31 to b2				R/W

14.2.15 Current Link Address Register n (CRLA_n)

The CRLA_n register sets the link address of DMA channel n (n = 15 to 0).

Address(es): DMAC0
CRLA_0: A006 203Ch, CRLA_1: A006 207Ch, CRLA_2: A006 20BCh, CRLA_3: A006 20FCh,
CRLA_4: A006 213Ch, CRLA_5: A006 217Ch, CRLA_6: A006 21BCh, CRLA_7: A006 21FCh,
CRLA_8: A006 243Ch, CRLA_9: A006 247Ch, CRLA_10: A006 24BCh, CRLA_11: A006 24FCh,
CRLA_12: A006 253Ch, CRLA_13: A006 257Ch, CRLA_14: A006 25BCh, CRLA_15: A006 25FCh
DMAC1
CRLA_0: A006 303Ch, CRLA_1: A006 307Ch, CRLA_2: A006 30BCh, CRLA_3: A006 30FCh,
CRLA_4: A006 313Ch, CRLA_5: A006 317Ch, CRLA_6: A006 31BCh, CRLA_7: A006 31FCh,
CRLA_8: A006 343Ch, CRLA_9: A006 347Ch, CRLA_10: A006 34BCh, CRLA_11: A006 34FCh,
CRLA_12: A006 353Ch, CRLA_13: A006 357Ch, CRLA_14: A006 35BCh, CRLA_15: A006 35FCh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	CRLA[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	CRLA[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	CRLA[31:0]	Current Link Address	Indicates the address of the descriptor which is being executed.	R

14.2.16 Source Continuous Register n (SCNT_n)

The SCNT_n register sets the space size for continuous access during read access to the DMA transfer source (n = 15 to 0).

Use this register together with the SSKP_n register (see Figure 14.1).

When setting this register, set the SDS[3] bit in the CHCFG_n register to 1.

Address(es): DMAC0
SCNT_0: A006 2200h, SCNT_1: A006 2220h, SCNT_2: A006 2240h, SCNT_3: A006 2260h,
SCNT_4: A006 2280h, SCNT_5: A006 22A0h, SCNT_6: A006 22C0h, SCNT_7: A006 22E0h,
SCNT_8: A006 2600h, SCNT_9: A006 2620h, SCNT_10: A006 2640h, SCNT_11: A006 2660h,
SCNT_12: A006 2680h, SCNT_13: A006 26A0h, SCNT_14: A006 26C0h, SCNT_15: A006 26E0h
DMAC1
SCNT_0: A006 3200h, SCNT_1: A006 3220h, SCNT_2: A006 3240h, SCNT_3: A006 3260h,
SCNT_4: A006 3280h, SCNT_5: A006 32A0h, SCNT_6: A006 32C0h, SCNT_7: A006 32E0h,
SCNT_8: A006 3600h, SCNT_9: A006 3620h, SCNT_10: A006 3640h, SCNT_11: A006 3660h,
SCNT_12: A006 3680h, SCNT_13: A006 36A0h, SCNT_14: A006 36C0h, SCNT_15: A006 36E0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	SCNT[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SCNT[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	SCNT[31:0]	Source Continuous Access Size	Sets the size of the continuous access space during read access to the DMA transfer source in bytes.	R/W

When performing a skip transfer on the transfer source, do not set the SAD bit in the CHCFG_n register to 1 (fixed). In addition, do not perform a skip transfer when this register is set to 0000 0000h.

14.2.17 Source Skip Register n (SSKP_n)

The SSKP_n register sets the skip amount during read access to the DMA transfer source.
During read access to the DMA transfer source, after data of the size set by using the SCNT_n register is accessed, the next DMA transfer source address of the size set by using this register is skipped (n = 15 to 0).
Use this register together with the SCNT_n register (see Figure 14.1).
When setting this register, set the SDS[3] bit in the CHCFG_n register to 1.

Address(es): DMAC0
SSKP_0: A006 2204h, SSKP_1: A006 2224h, SSKP_2: A006 2244h, SSKP_3: A006 2264h,
SSKP_4: A006 2284h, SSKP_5: A006 22A4h, SSKP_6: A006 22C4h, SSKP_7: A006 22E4h,
SSKP_8: A006 2604h, SSKP_9: A006 2624h, SSKP_10: A006 2644h, SSKP_11: A006 2664h,
SSKP_12: A006 2684h, SSKP_13: A006 26A4h, SSKP_14: A006 26C4h, SSKP_15: A006 26E4h
DMAC1
SSKP_0: A006 3204h, SSKP_1: A006 3224h, SSKP_2: A006 3244h, SSKP_3: A006 3264h,
SSKP_4: A006 3284h, SSKP_5: A006 32A4h, SSKP_6: A006 32C4h, SSKP_7: A006 32E4h,
SSKP_8: A006 3604h, SSKP_9: A006 3624h, SSKP_10: A006 3644h, SSKP_11: A006 3664h,
SSKP_12: A006 3684h, SSKP_13: A006 36A4h, SSKP_14: A006 36C4h, SSKP_15: A006 36E4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	SSKP[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SSKP[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	SSKP[31:0]	Source Skip Size	Sets the skip amount during read access to the DMA transfer source in bytes.	R/W

When performing a skip transfer on the transfer source, do not set the SAD bit in the CHCFG_n register to 1 (fixed).

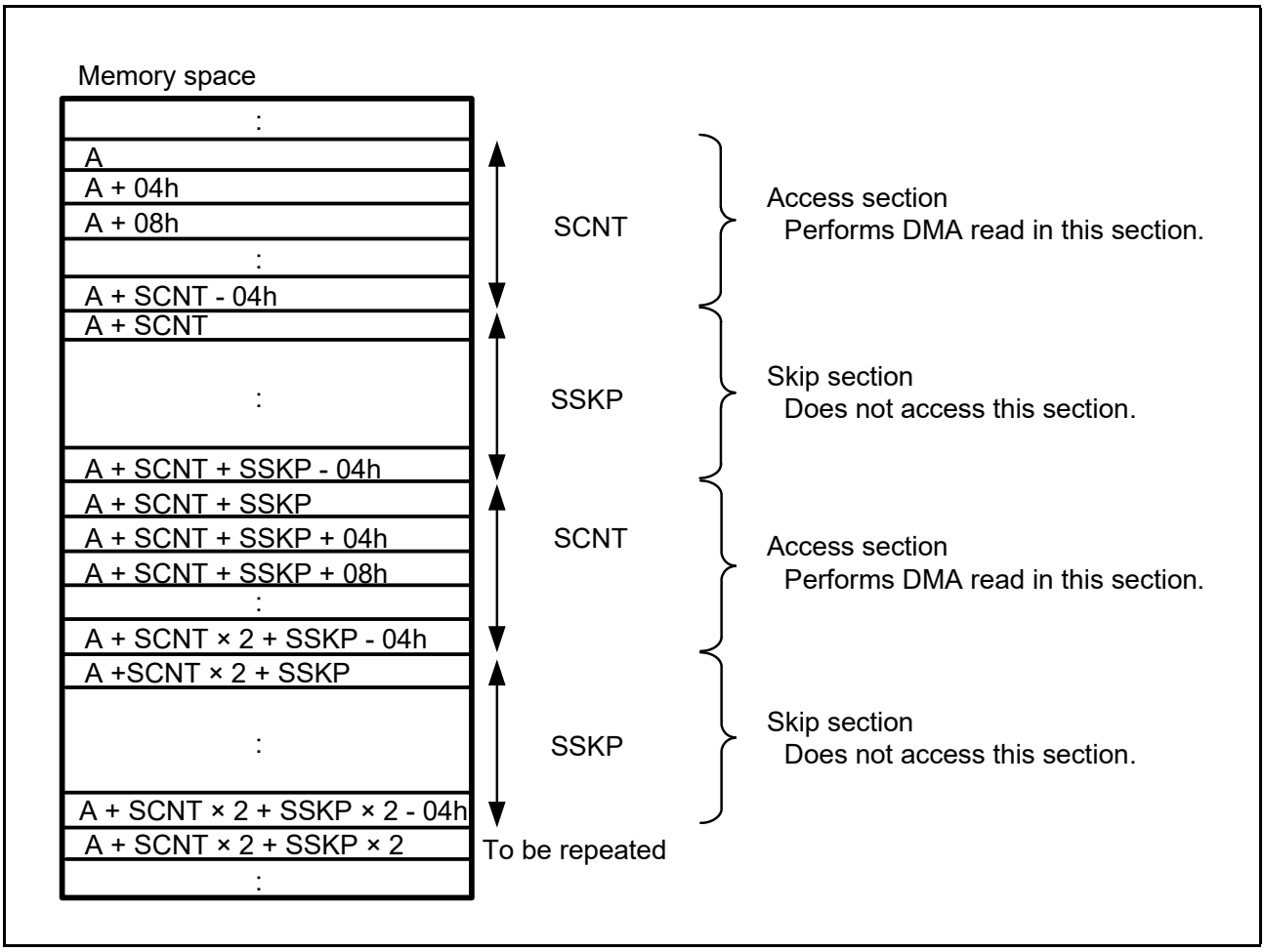


Figure 14.1 Relationship between SSKP and SCNT

Regardless of the source address and the setting value of the RDS field in the CHCFG_n register, the values for SCNT and SSKP can be set. DMAC accesses in the size set in the SDS field in the CHCFG_n register, and acquires the buffer of the valid data only.

14.2.18 Destination Continuous Register n (DCNT_n)

The DCNT_n register sets the space size for continuous access during write access to the DMA transfer destination (n = 15 to 0).

Use this register together with the DSKP_n register (see Figure 14.2).

To set this register, set the DDS[3] bit in the CHCFG_n register to 1.

Address(es): DMAC0
DCNT_0: A006 2208h, DCNT_1: A006 2228h, DCNT_2: A006 2248h, DCNT_3: A006 2268h,
DCNT_4: A006 2288h, DCNT_5: A006 22A8h, DCNT_6: A006 22C8h, DCNT_7: A006 22E8h,
DCNT_8: A006 2608h, DCNT_9: A006 2628h, DCNT_10: A006 2648h, DCNT_11: A006 2668h,
DCNT_12: A006 2688h, DCNT_13: A006 26A8h, DCNT_14: A006 26C8h, DCNT_15: A006 26E8h
DMAC1
DCNT_0: A006 3208h, DCNT_1: A006 3228h, DCNT_2: A006 3248h, DCNT_3: A006 3268h,
DCNT_4: A006 3288h, DCNT_5: A006 32A8h, DCNT_6: A006 32C8h, DCNT_7: A006 32E8h,
DCNT_8: A006 3608h, DCNT_9: A006 3628h, DCNT_10: A006 3648h, DCNT_11: A006 3668h,
DCNT_12: A006 3688h, DCNT_13: A006 36A8h, DCNT_14: A006 36C8h, DCNT_15: A006 36E8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	DCNT[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	DCNT[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	DCNT[31:0]	Destination Continuous Access Size	Sets the size of the continuous access space during write access to the DMA transfer destination in bytes.	R/W

When performing a skip transfer on the destination, do not set the DAD bit in the CHCFG_n register to 1 (fixed).

In addition, do not perform a skip transfer when this register is set to 0000 0000h.

14.2.19 Destination Skip Register n (DSKP_n)

The DSKP_n register sets the skip amount during write access to the DMA transfer destination.
During write access to the DMA transfer destination, accesses data of the size set with the DCNT_n register, and then skips the number of the next DMA transfer destination addresses set with this register (n = 15 to 0).
Use this register together with the DCNT_n register (see Figure 14.2).
To set this register, set the DDS[3] bit in the CHCFG_n register to 1.

Address(es): DMAC0
DSKP_0: A006 220Ch, DSKP_1: A006 222Ch, DSKP_2: A006 224Ch, DSKP_3: A006 226Ch,
DSKP_4: A006 228Ch, DSKP_5: A006 22ACh, DSKP_6: A006 22CCh, DSKP_7: A006 22ECh,
DSKP_8: A006 260Ch, DSKP_9: A006 262Ch, DSKP_10: A006 264Ch, DSKP_11: A006 266Ch,
DSKP_12: A006 268Ch, DSKP_13: A006 26ACh, DSKP_14: A006 26CCh, DSKP_15: A006 26ECh
DMAC1
DSKP_0: A006 320Ch, DSKP_1: A006 322Ch, DSKP_2: A006 324Ch, DSKP_3: A006 326Ch,
DSKP_4: A006 328Ch, DSKP_5: A006 32ACh, DSKP_6: A006 32CCh, DSKP_7: A006 32ECh,
DSKP_8: A006 360Ch, DSKP_9: A006 362Ch, DSKP_10: A006 364Ch, DSKP_11: A006 366Ch,
DSKP_12: A006 368Ch, DSKP_13: A006 36ACh, DSKP_14: A006 36CCh, DSKP_15: A006 36ECh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	DSKP[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	DSKP[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	DSKP[31:0]	Destination Skip Size	Sets the skip amount during write access to the DMA transfer destination in bytes.	R/W

To perform skip transfer on the destination, do not set the DAD bit in the CHCFG_n register to 1 (fixed).

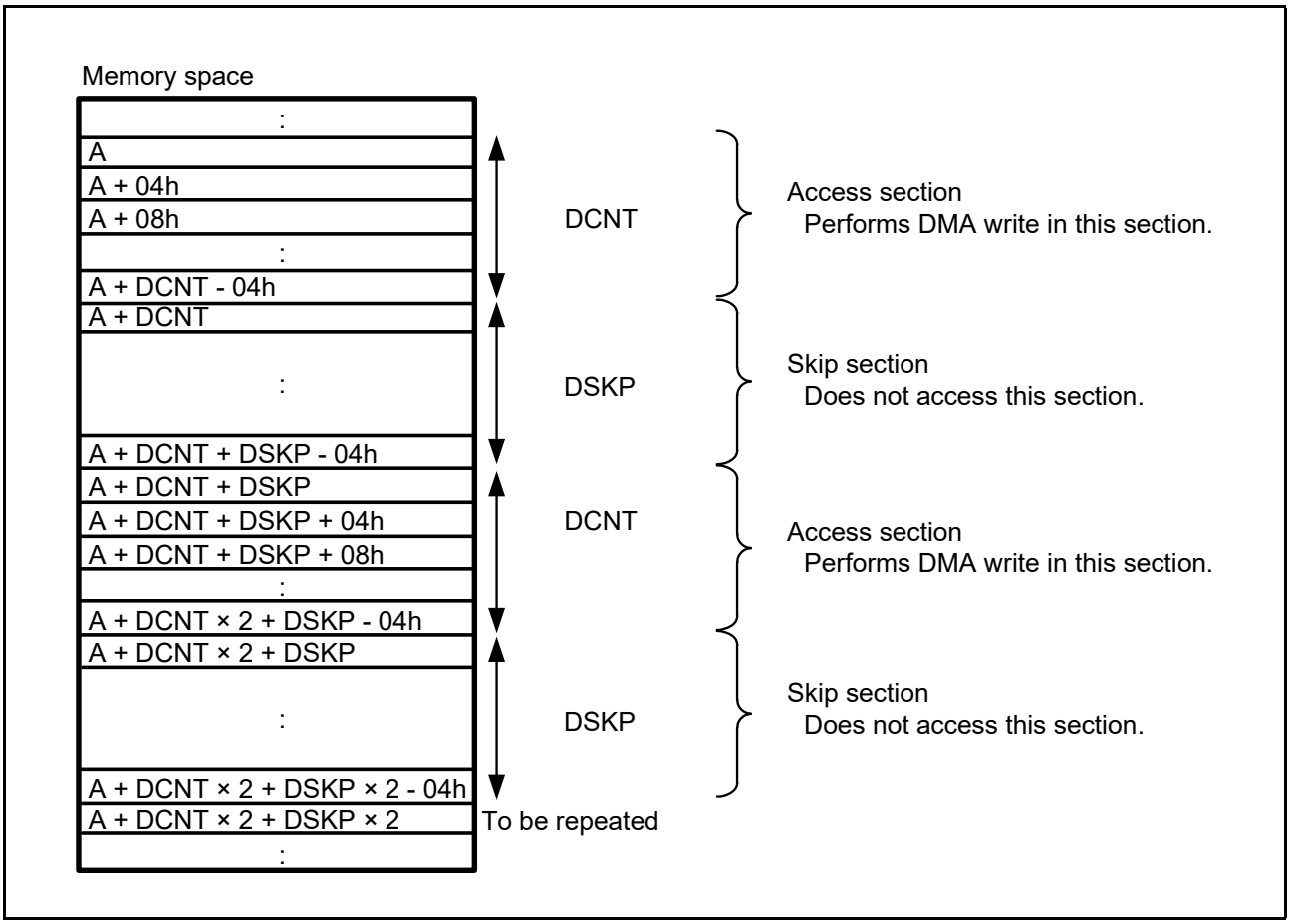


Figure 14.2 Relationship between DSKP and DCNT

Regardless of the destination address and the setting value of the DDS field in the CHCFG_n register, the values for DCNT and DSKP can be set. DMAC performs write access only to the specified space by the size equal to or smaller than the value set in the DDS field in the CHCFG_n register.

14.2.20 DMA Control Register (DCTRL_X (X = A or B))

The DCTRL_X register sets the arbitration between channels in all channels (DCTRL_A = channels 0 to 7, DCTRL_B = channels 8 to 15).

Address(es): DMAC0
DCTRL_A: A006 2300h, DCTRL_B: A006 2700h
DMAC1
DCTRL_A: A006 3300h, DCTRL_B: A006 3700h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PR
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	PR	Priority Control Select	Sets the transfer priority control mode (see section 14.3.3, DMA Channel Priority Control). 0: Fixed priority mode 1: Round-robin mode	R/W
b31 to b1	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W

14.2.21 Descriptor Interval Register n (DSCITVL_X (X = A or B))

The DSCITVL_X register sets the descriptor read interval in all channels (DSCITVL_A = channels 0 to 7, DSCITVL_B = channels 8 to 15).

By setting the DRRP bit in the CHCFG_n register to 1, the descriptor continues to read the descriptor until it reaches LV = 1. This register sets the read interval.

Address(es): DMAC0
DSCITVL_A: A006 2304h, DSCITVL_B: A006 2704h
DMAC1
DSCITVL_A: A006 3304h, DSCITVL_B: A006 3704h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	DITVL								—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	—	Reserved	These bits are always read as 0. The write value should be 0.	R/W
b15 to b8	DITVL	Descriptor Interval	Sets the descriptor read interval. The descriptor is read again in the interval of (DITVL × 256) cycles. The descriptor read interval will be (the set value in the DITVL bit × 256 × ICLK) cycles.	R/W
b31 to b16	—	Reserved	These bits are always read as 0. The write value should be 0.	R/W

14.2.22 DMA Status EN Register (DST_EN_X (X = A or B))

The DST_EN_X register indicates the status for the EN bit of all channels (DST_EN_A = channels 0 to 7, DST_EN_B = channels 8 to 15).

Writing data to this register does not change the value of each bit.

Address(es): DMAC0
DST_EN_A: A006 2310h, DST_EN_B: A006 2710h
DMAC1
DST_EN_A: A006 3310h, DST_EN_B: A006 3710h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	EN 7/15	EN 6/14	EN 5/13	EN 4/12	EN 3/11	EN 2/10	EN 1/9	EN 0/8
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	EN0/8	Channel 0/8EN	Indicates the status for the EN bit of DMA channel 0/8.	R
b1	EN1/9	Channel 1/9EN	Indicates the status for the EN bit of DMA channel 1/9.	R
b2	EN2/10	Channel 2/10EN	Indicates the status for the EN bit of DMA channel 2/10.	R
b3	EN3/11	Channel 3/11EN	Indicates the status for the EN bit of DMA channel 3/11.	R
b4	EN4/12	Channel 4/12EN	Indicates the status for the EN bit of DMA channel 4/12.	R
b5	EN5/13	Channel 5/13EN	Indicates the status for the EN bit of DMA channel 5/13.	R
b6	EN6/14	Channel 6/14EN	Indicates the status for the EN bit of DMA channel 6/14.	R
b7	EN7/15	Channel 7/15EN	Indicates the status for the EN bit of DMA channel 7/15.	R
b31 to b8	—	Reserved	These bits are always read as 0.	R

14.2.23 DMA Status ER Register (DST_ER_X (X = A or B))

The DST_ER_X register indicates the status for the ER bit of all channels (DST_ER_A = channels 0 to 7, DST_ER_B = channels 8 to 15).

Writing data to this register does not change the value of each bit.

Address(es): DMAC0
DST_ER_A: A006 2314h, DST_ER_B: A006 2714h
DMAC1
DST_ER_A: A006 3314h, DST_ER_B: A006 3714h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	ER 7/15	ER 6/14	ER 5/13	ER 4/12	ER 3/11	ER 2/10	ER 1/9	ER 0/8
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ER0/8	Channel 0/8ER	Indicates the status for the ER bit of DMA channel 0/8.	R
b1	ER1/9	Channel 1/9ER	Indicates the status for the ER bit of DMA channel 1/9.	R
b2	ER2/10	Channel 2/10ER	Indicates the status for the ER bit of DMA channel 2/10.	R
b3	ER3/11	Channel 3/11ER	Indicates the status for the ER bit of DMA channel 3/11.	R
b4	ER4/12	Channel 4/12ER	Indicates the status for the ER bit of DMA channel 4/12.	R
b5	ER5/13	Channel 5/13ER	Indicates the status for the ER bit of DMA channel 5/13.	R
b6	ER6/14	Channel 6/14ER	Indicates the status for the ER bit of DMA channel 6/14.	R
b7	ER7/15	Channel 7/15ER	Indicates the status for the ER bit of DMA channel 7/15.	R
b31 to b8	—	Reserved	These bits are always read as 0.	R

14.2.24 DMA Status END Register (DST_END_X (X = A or B))

The DST_END_X register indicates the status for the END bit of all channels (DST_END_A = channels 0 to 7, DST_END_B = channels 8 to 15). Writing data to this register does not change the value of each bit.

Address(es): DMAC0
DST_END_A: A006 2318h, DST_END_B: A006 2718h
DMAC1
DST_END_A: A006 3318h, DST_END_B: A006 3718h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	END 7/15	END 6/14	END 5/13	END 4/12	END 3/11	END 2/10	END 1/9	END 0/8
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	END0/8	Channel 0/8END	Indicates the status for the END bit of DMA channel 0/8.	R
b1	END1/9	Channel 1/9END	Indicates the status for the END bit of DMA channel 1/9.	R
b2	END2/10	Channel 2/10END	Indicates the status for the END bit of DMA channel 2/10.	R
b3	END3/11	Channel 3/11END	Indicates the status for the END bit of DMA channel 3/11.	R
b4	END4/12	Channel 4/12END	Indicates the status for the END bit of DMA channel 4/12.	R
b5	END5/13	Channel 5/13END	Indicates the status for the END bit of DMA channel 5/13.	R
b6	END6/14	Channel 6/14END	Indicates the status for the END bit of DMA channel 6/14.	R
b7	END7/15	Channel 7/15END	Indicates the status for the END bit of DMA channel 7/15.	R
b31 to b8	—	Reserved	These bits are always read as 0.	R

14.2.25 DMA Status SUS Register (DST_SUS_X (X = A or B))

The DST_SUS_X register indicates the status for the SUS bit of all channels (DST_SUS_A = channels 0 to 7, DST_SUS_B = channels 8 to 15). Writing data to this register does not change the value of each bit.

Address(es): DMAC0
DST_SUS_A: A006 2320h, DST_SUS_B: A006 2720h
DMAC1
DST_SUS_A: A006 3320h, DST_SUS_B: A006 3720h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	SUS 7/15	SUS 6/14	SUS 5/13	SUS 4/12	SUS 3/11	SUS 2/10	SUS 1/9	SUS 0/8
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SUS0/8	Channel 0/8SUS	Indicates the status for the SUS bit of DMA channel 0/8.	R
b1	SUS1/9	Channel 1/9SUS	Indicates the status for the SUS bit of DMA channel 1/9.	R
b2	SUS2/10	Channel 2/10SUS	Indicates the status for the SUS bit of DMA channel 2/10.	R
b3	SUS3/11	Channel 3/11SUS	Indicates the status for the SUS bit of DMA channel 3/11.	R
b4	SUS4/12	Channel 4/12SUS	Indicates the status for the SUS bit of DMA channel 4/12.	R
b5	SUS5/13	Channel 5/13SUS	Indicates the status for the SUS bit of DMA channel 5/13.	R
b6	SUS6/14	Channel 6/14SUS	Indicates the status for the SUS bit of DMA channel 6/14.	R
b7	SUS7/15	Channel 7/15SUS	Indicates the status for the SUS bit of DMA channel 7/15.	R
b31 to b8	—	Reserved	These bits are always read as 0.	R

14.3 Operation

14.3.1 DMA Mode

With the DMS bit in the CHCFG_n register, DMA mode can be switched between register mode and link mode.

Table 14.2 DMA Mode Setting

DMS (CHCFG_n)	Description	Applications
0	Register mode	With the values set for the next register set, performs a DMA transfer.
1	Link mode	Accesses the descriptor area, and performs a DMA transfer with the value set for the descriptor. Repeats descriptor read and DMA transfer unless you set the descriptor or use the control register to stop them.

14.3.1.1 Register Mode

In register mode, you can perform a DMA transfer with the value set in the internal register.
You can set two sets (Next0 Register Set and Next1 Register Set) of transfer source addresses, the transfer destination addresses, and the numbers of transfer bytes. You can select the next register set to perform a transfer, or transfer two next register sets successively.

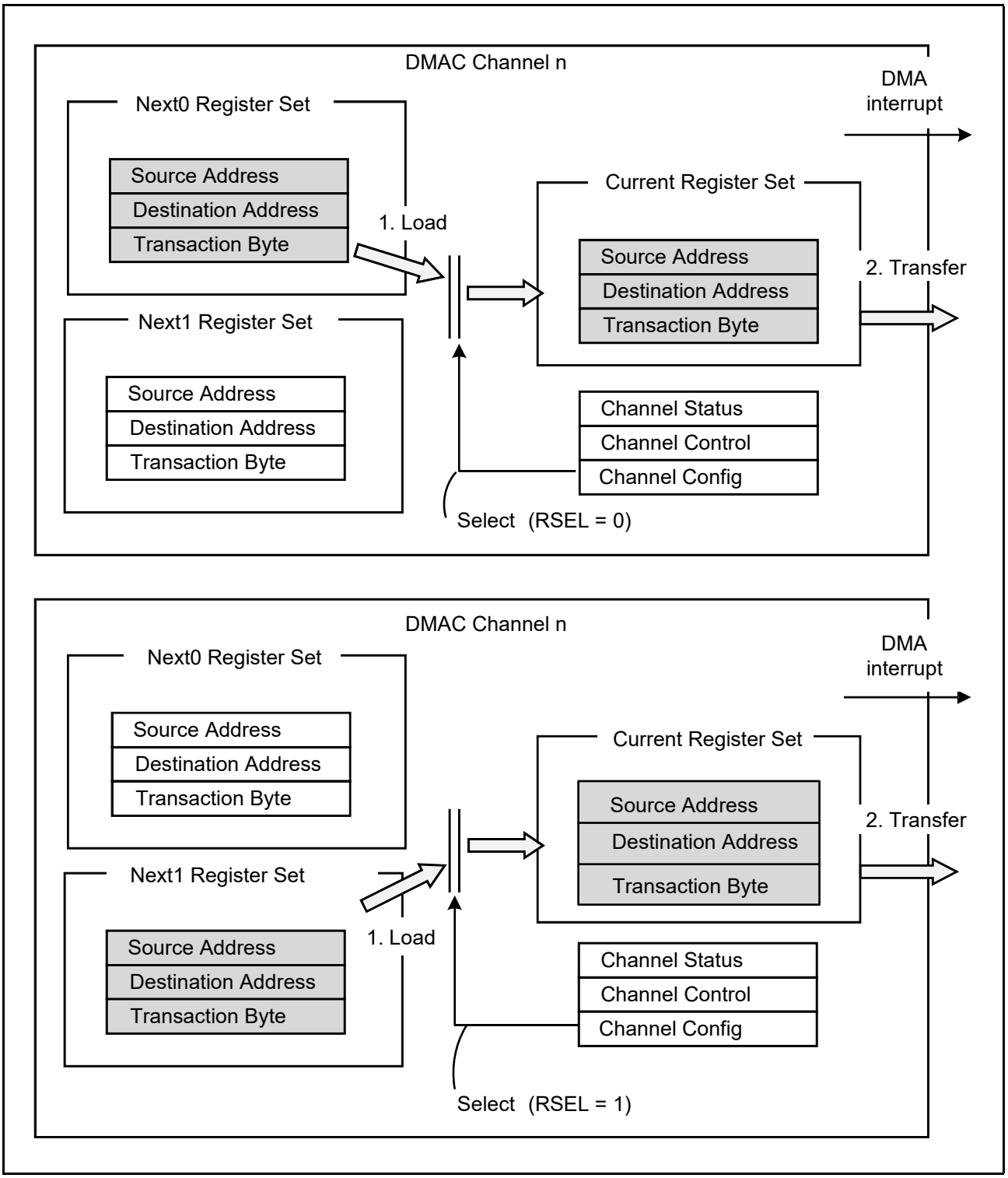


Figure 14.3 Overview of Register Normal Mode

Figure 14.3 illustrates operation when Next0 Register Set is executed (above), and when Next1 Register Set is executed (below).

(1) Operation Flow in Register Mode

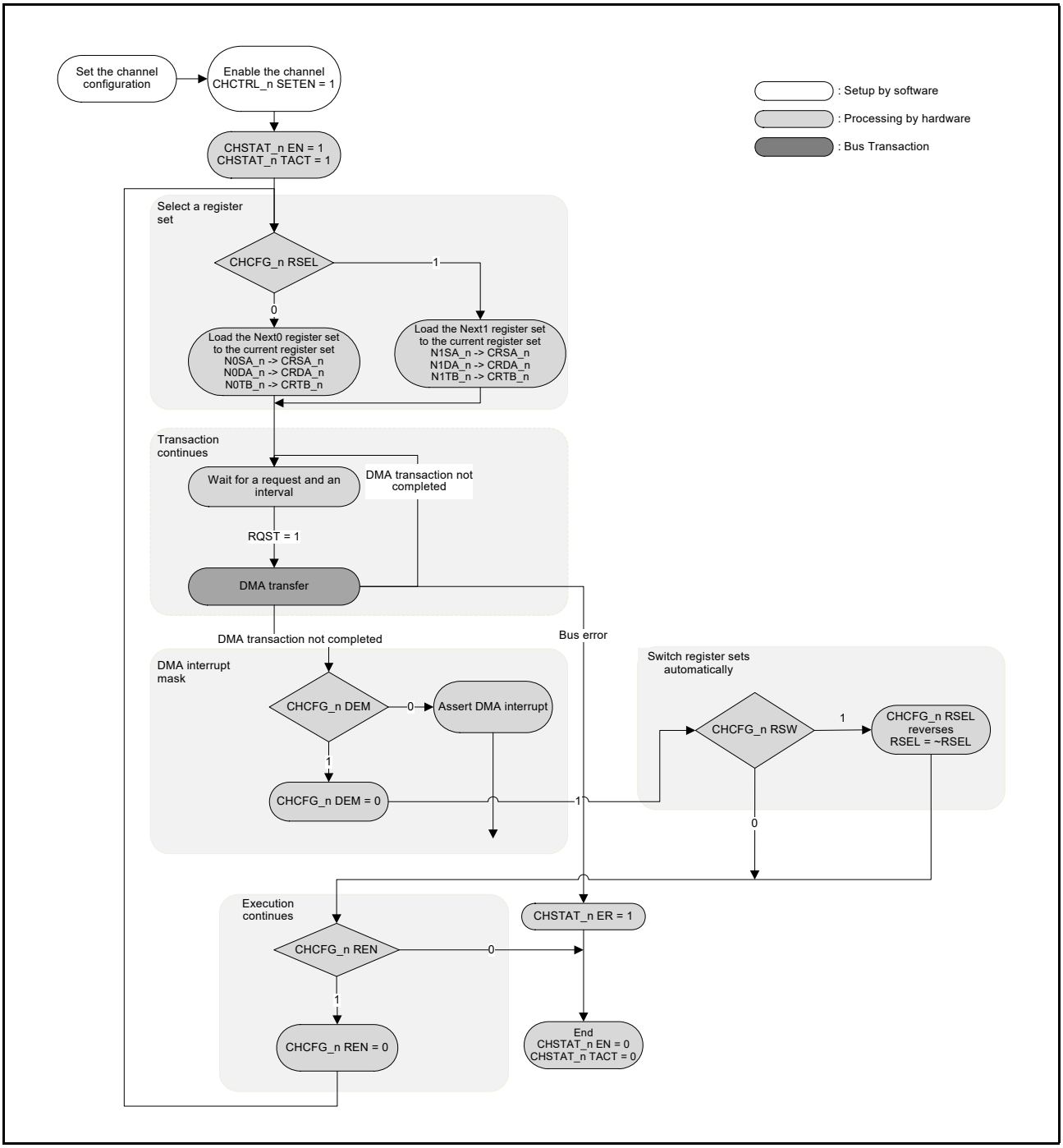


Figure 14.4 Register Mode Flow

<Register Mode Flow Description>

1. Channel Setting
   Set Next0 or Next1 Register Set (the transfer destination address, the transfer source address, and the total number of transfer bytes).
   In addition, use setting registers CHCTRL_n or CHCFG_n for each channel to set detection method of the DMA transfer request and the amount of data for a transfer.
2. Selecting the register set
   When 1 is written to the SETEN bit in the CHCTRL_n register, the EN bit and the TACT bit in the CHSTAT_n register are set to 1, and the setting value of the next register set selected by the RSEL bit in the CHCFG_n register is loaded to the current register set.
3. DMA Transfer
   According to the set value, a DMA transfer is performed. For details on the transfer, see from section 14.3.2, Transfer Mode to section 14.3.10, Aborting a Transfer.
4. DMA Transfer Completion Mask
   According to the value set for the DEM bit in the CHCFG_n register, the DMA transfer completion interrupt is masked. When DEM = 1, the DMA transfer completion interrupt is masked. Besides, immediately after the DMA transfer completion interrupt conditions are satisfied, the DEM bit is cleared to 0 automatically.
5. Switching Register Sets Automatically
   According to the value set for the RSW bit in the CHCFG_n register, the register set is switched to the other next register set.
6. Successive Execution
   According to the value set for the REN bit in the CHCFG_n register, DMA transfers are performed successively. When REN = 0, the EN bit and the TACT bit in the CHSTAT_n register are cleared to 0, and DMAC stops operation. When REN = 1, the DMA transfer is continued. Besides, immediately after the conditions for performing DMA transfer again are satisfied with the REN bit, the REN bit is cleared to 0 automatically.

(2) Setting the Register Mode

- Register mode settings
  Select the register set to execute.

Table 14.3 Register Mode Settings

DMS (CHCFG_n)	RSEL (CHCFG_n)	Description
0	0	Executes Next0 Register Set.
	1	Executes Next1 Register Set.

- DMA transfer completion interrupt mask settings
  DMA transfer completion interrupts can be masked.

Table 14.4 DMA Transfer Completion Interrupt Mask Settings

DEM (CHCFG_n)	Description
0	When a DMA transfer completes, the DMA transfer completion interrupt is generated.
1	Even if a DMA transfer completes, no DMA transfer completion interrupt is generated. When a DMA transfer completes, the DEM bit is cleared to 0 automatically.

- Settings for executing register sets automatically
  After a DMA transfer, another DMA transfer can be performed successively.

Table 14.5 Settings for Automatic Execution of Register Sets

REN (CHCFG_n)	Operation	Remarks
0	When a DMA transfer of the register set set for the RSEL bit completes, the EN bit is cleared to 0, and the DMA operation ends.	Set this value if you want to perform a DMA transfer once.
1	After a DMA transfer completes, a DMA transfer is performed for the register set that was selected to be performed successively. When the successive transfer is performed, the REN bit is cleared to 0.	Set this register if you want to perform register sets successively.

- Settings for switching register sets automatically
  When a DMA transfer completes, the register set can be switched to the next register set.

Table 14.6 Settings for Switching Register Sets Automatically

RSW (CHCFG_n)	Operation	Remarks
0	When a DMA transfer completes, register sets are not switched.	Set this value when you want to use one register set only.
1	When a DMA transfer completes, the RSEL bit is reversed automatically, and the other register set is selected.	Set this value if you want to switch register sets.

(3) Register Mode Setting Examples

- Using the Next0 register set only

Table 14.7 Register Mode Setting Example 1

DMS (CHCFG_n)	RSEL (CHCFG_n)	DEM (CHCFG_n)	RSW (CHCFG_n)	REN (CHCFG_n)
0 (register mode)	0 (Next0)	0 (Do not mask)	0 (Do not switch)	0 (Continuous execution not in progress)

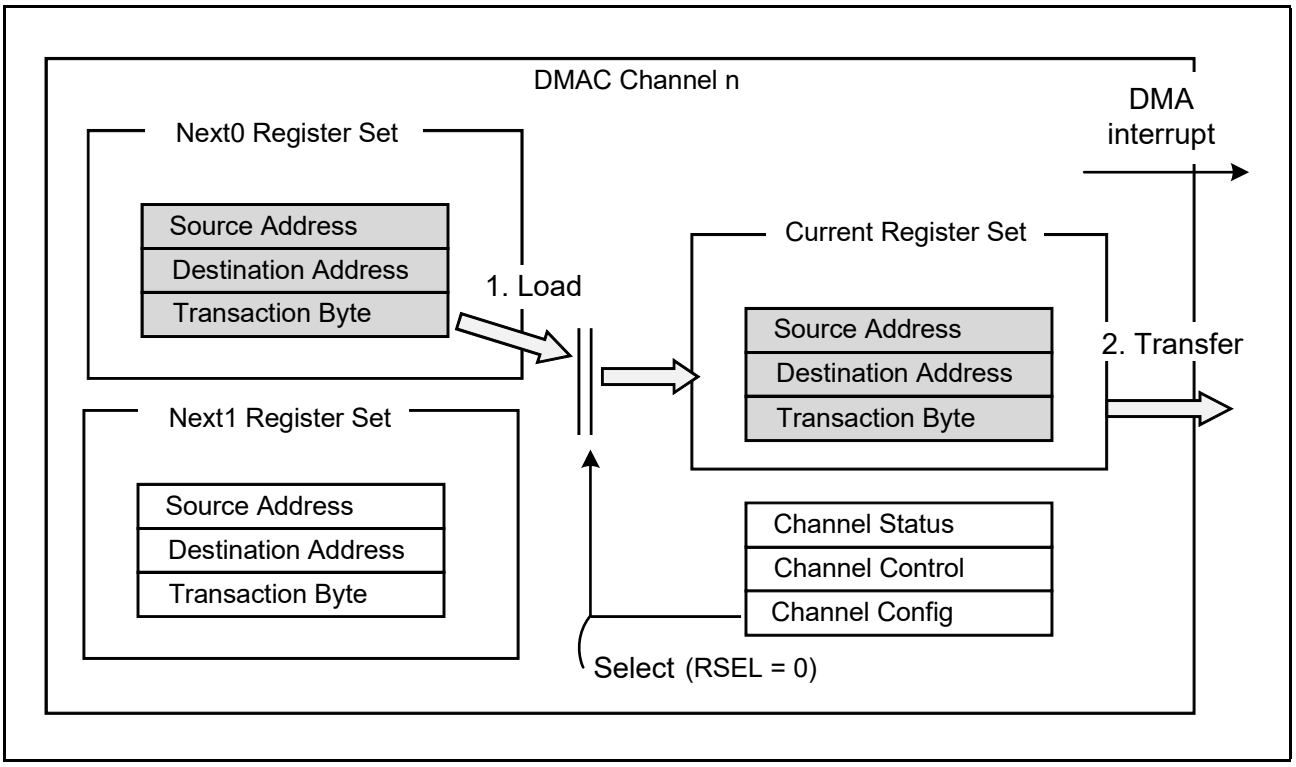


Figure 14.5 Register Mode Setting Example 1

1. Writing 1 to the SETEN bit in the CHCTRL_n register sets 1 to the EN bit in the CHSTAT_n register, and loads Next0 Register Set to the current register set.
2. According to the values set for the current register set and the channel register set, a DMA transfer is performed.
3. Because the DEM bit in the CHCFG_n register is 0, a DMA transfer completion interrupt is generated when a DMA transfer completes.
4. Because the REN bit in the CHCFG_n register is 0, the EN bit in the CHSTAT_n register is cleared to 0, and the operation ends.

- Using two register sets successively

Table 14.8 Register Mode Setting Example 2

DMS (CHCFG_n)	RSEL (CHCFG_n)	DEM (CHCFG_n)	RSW (CHCFG_n)	REN (CHCFG_n)
0 (Register mode)	0 (Next0)	1 (Mask)	1 (Switch)	1 (Continuous execution not in progress)

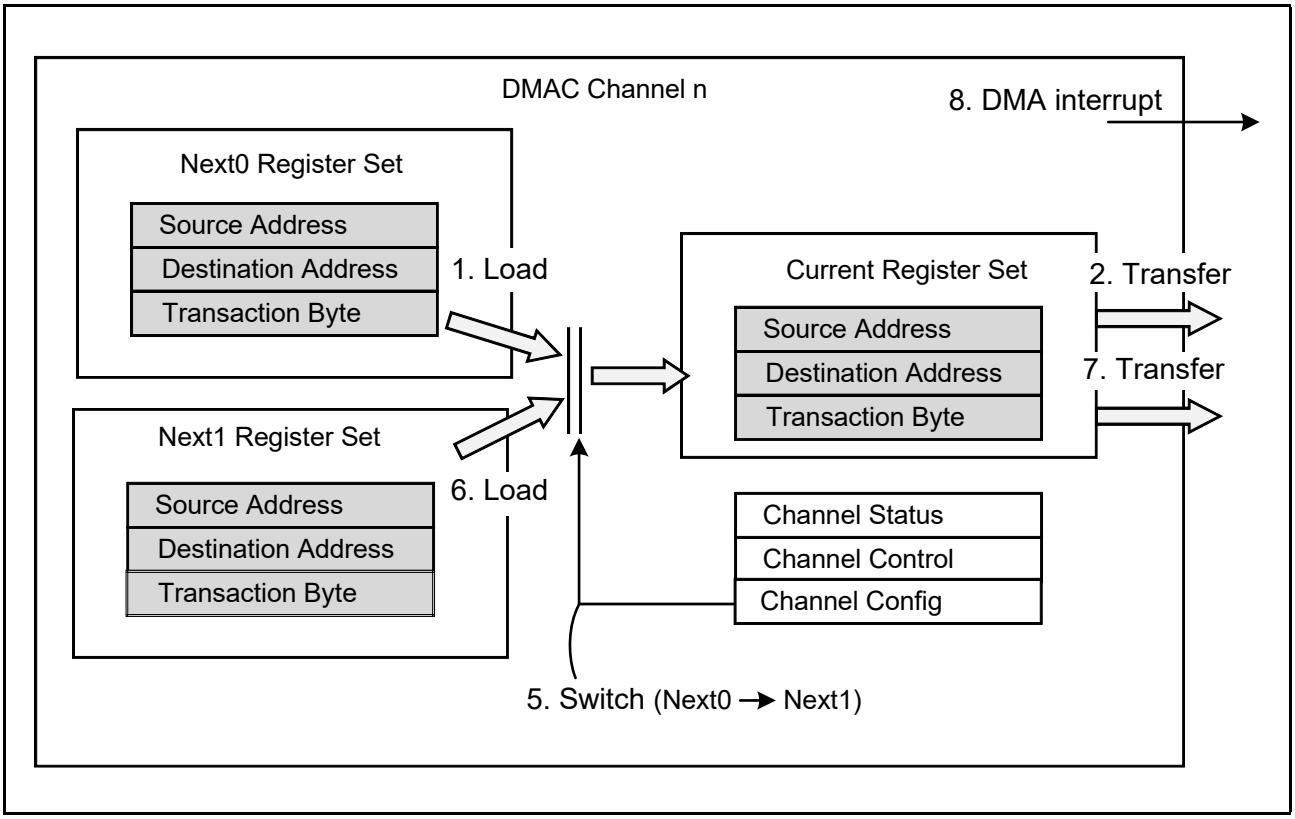


Figure 14.6 Register Mode Setting Example 2

1. Writing 1 to the SETEN bit in the CHCTRL_n register sets 1 to the EN bit in the CHSTAT_n register, and loads Next0 Register Set to the current register set.
2. According to the values set for the current register set and the channel register set, a DMA transfer is performed.
3. Because the DEM bit in the CHCFG_n register is 1, when the DMA transfer completes, no DMA transfer completion interrupt is generated. In addition, the DEM bit is cleared to 0 automatically.
4. Because the REN bit in the CHCFG_n register is 1, DMA transfers are performed successively. In addition, the REN bit is cleared to 0.
5. Because the RSW bit in the CHCFG_n register is 1, the register set to be executed next is switched (RSEL = 0 -> 1).
6. Loads Next1 Register Set to Current Register Set.
7. According to the values for Current Register Set and Channel Register Set, DMA transfers are performed.
8. Because the DEM bit in the CHCFG_n register is 0, a DMA transfer completion interrupt is generated when a DMA transfer completes.
9. Because the REN bit in the CHCFG_n register is 0, the EN bit in the CHSTAT_n register is cleared to 0 automatically.

14.3.1.2 Link Mode

In link mode, a DMA transfer is performed by reading a descriptor in the memory area outside the DMAC as the setting value. Within DMAC, each channel has the next link address (NXLA_n) register and the current link address (CRLA_n) register. Each of them is used to set the address of the descriptor to be executed next, and to indicate the descriptor address of the current DMA transfer respectively.

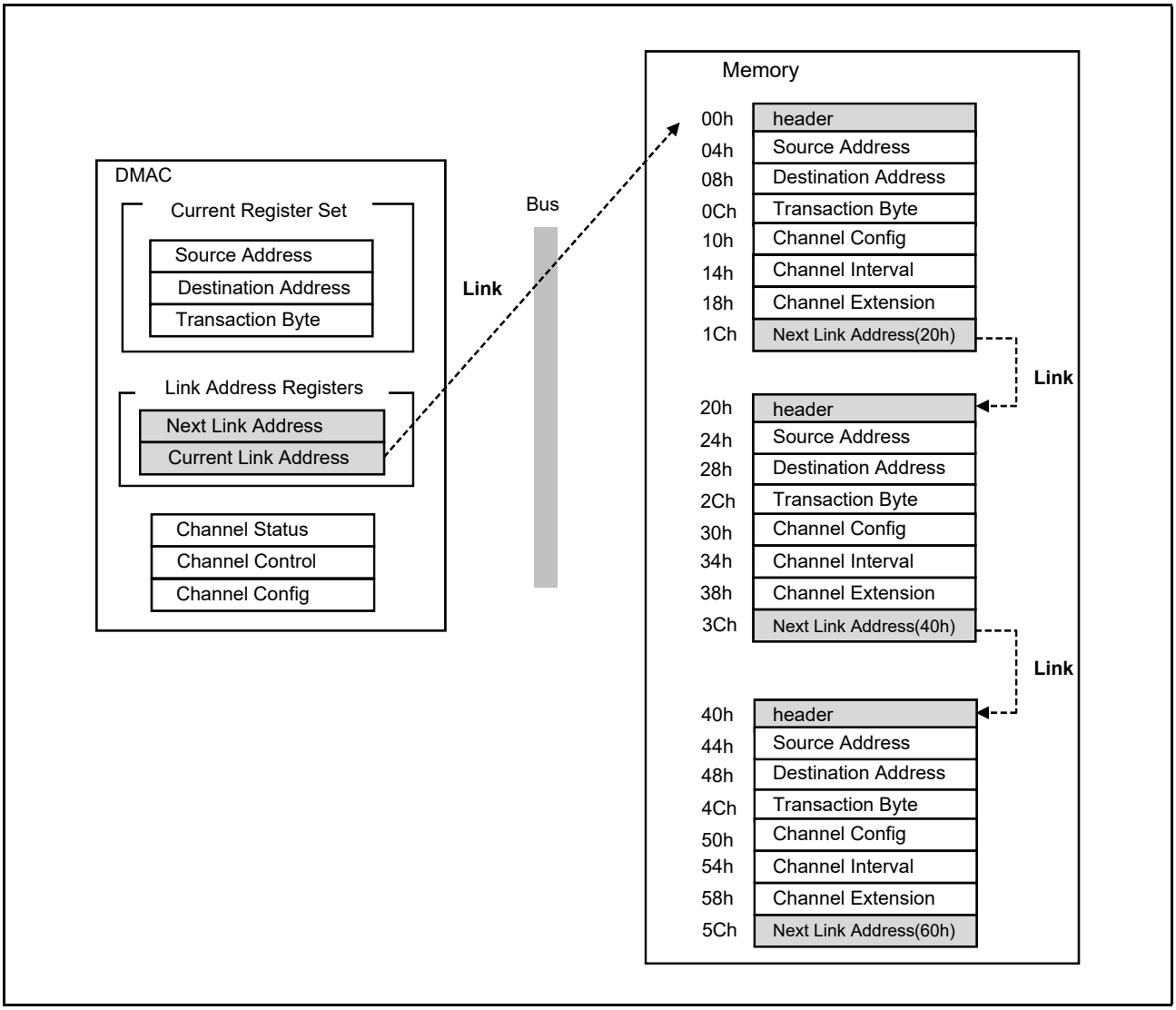


Figure 14.7 Link Mode Overview

(1) Operation flows in link mode

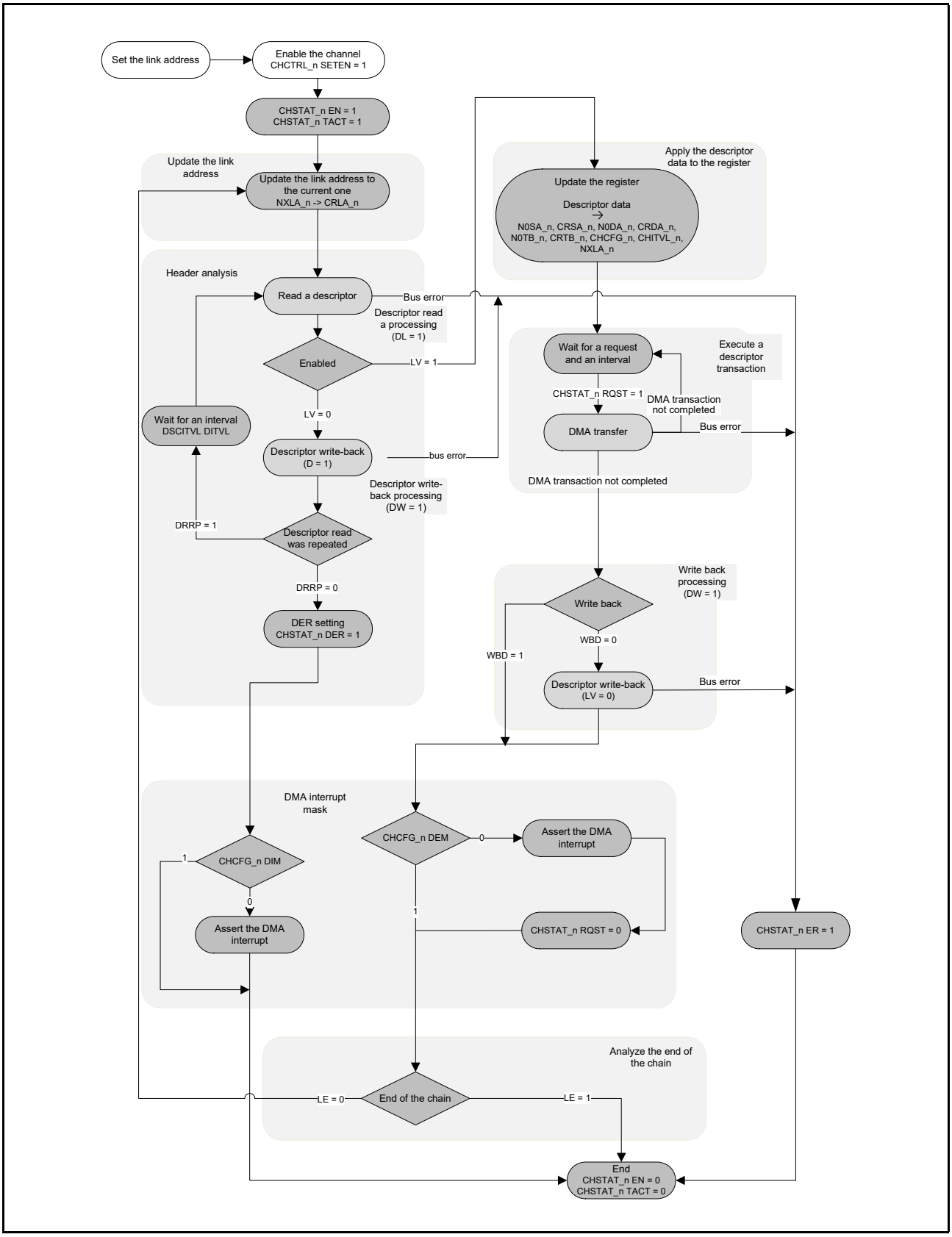


Figure 14.8 Link Mode Operation Flow

<Description of Link Mode Operation Flow>

1. Channel Setting
   Set the start address of the link destination in the NXLA_n register.
2. Updating the Link Address
   Writing 1 to the SETEN bit in the CHCTRL_n register sets 1 to the EN bit and the TACT bit in the CHSTAT_n register, and loads the link address set for the NXLA_n register to the CRLA_n register.
3. Descriptor Read and Header Judgment
   DMAC starts reading a descriptor, and checks the header contents. When LV = 0, 1 is written back to the D bit of header. After that, when the DRRP bit in the CHCFG_n register = 1, the same descriptor is read again after the number of cycles set in the DSCITVL register elapsed. When DRRP =0, DER in the CHSTAT_n register =1. This indicates the end state (the EN bit in the CHSTAT_n register = 0, and TACT = 0). At that time, if the DIM bit in the CHCFG_n register is 0, a DMA transfer completion interrupt is generated.
4. Descriptor Error
   When LV = 1, the read descriptor data is loaded to the current register set, and the channel register set. In addition, the next link destination is loaded to the NXLA_n register.
5. DMA Transfer
   According to the set value, a DMA transfer is performed. For details on the transfer, see from section 14.3.2, Transfer Mode to section 14.3.10, Aborting a Transfer.
6. Writing back of Header
   When WBD of header = 0, DMAC writes LV = 0 back to the header area.
7. DMA Interrupt Mask
   When the DEM bit in the CHCFG_n register is 0, a DMA transfer completion interrupt is generated.
8. Link End Judgment
   When LE of header = 1, the EN bit and the TACT bit in the CHSTAT_n register are cleared to 0, and DMAC stops operation. When LE = 0, the current register set is updated, and the next descriptor read is started.

(2) Register Settings

- Link mode settings
  When using link mode, set the DMS bit in the CHCFG_n register to 1.

Table 14.9 Link Mode Settings

DMS (CHCFG_n)	Description
1	Operates in link mode. This bit cannot be changed by using a descriptor.

- LINK address settings
  As registers that indicate the link destination, the next link address (NXLA_n) register, and the current link address (CRLA_n) register are available.
  To start link mode, set the link destination in the NXLA_n register.
  The NXLA_n register is updated to the next link after a descriptor is read. In addition, the CRLA_n register indicates the link address of the currently executed descriptor.

Table 14.10 Link Address Register Set

Register	Description
NXLA_n	Sets and indicates the next link destination. Before starting link mode, set the address of the link destination for this register.
CRLA_n	Indicates the currently executed link destination. This register is read-only.

(3) Descriptor settings

DMAC supports multiple descriptor formats.
To switch formats, use the DSCFM field of bits[31:28] in the first word (header) of a descriptor.
The following table describes the relationship between the value of the DSCFM bit and the descriptor format.

Table 14.11 Descriptor Format

DSCFM	Descriptor Size	Next Link Address	Channel Interval	Channel Config	Transaction Size	Destination Address	Source Address	Header
3	Four words	Y	— (reload)	— (reload)	— (header)	Y	Y	Y (with STS)
1	Eight words	Y	Y	Y	Y	Y	Y	Y (without STS)
Other than above	Setting prohibited							

Table 14.12 Description of activation in Table 14.11Table 14.11 Descriptor Format

Field	Availability	Description	Remarks
Header	Y (with STS)	Indicates the STS field of [15:0] for header is enabled. The value set in the STS field is used as the total number of transfer bytes (transaction size).	—
	Y (without STS)	The STS field of [15:0] for header is disabled. Use the transaction size of the descriptor as the total number of transfer bytes.	
Source Address	Y	Specifies the source address.	—
Destination Address	Y	Specifies the destination address.	—
Transaction Size	Y	Specifies the transaction size.	—
	— (header)	Omits the transaction size. Use the value set in the STS field of header as the total number of transfer bytes (transaction size).	Because the STS field is 16 bits, the maximum size you can set is 65,535 bytes.
Channel Config Channel Interval	Y	Specifies Channel Config and Channel Interval.	—
	— (reload)	Omits Channel Config and Channel Interval. Inherits the previous setting values (values of the CHCFG_n and CHITVL_n registers at that time).	—
Next Link Address	Y	Specifies the next descriptor address (next link address) that is read after a DMA transfer of this descriptor.	—

DMAC interprets data obtained through descriptor read in order. If a value less than eight words is specified in the DSCFM field, place descriptor data marked with Y in Table 14.11, Descriptor Format on memory.

Table 14.13 Descriptor Placement Example

	Address (Link Address + N)							
DSCFM	+1Ch	+18h	+14h	+10h	+0Ch	+08h	+04h	+00h
3h	—	—	—	—	Next Link Address	Destination Address	Source Address	Header
1h	Next Link Address	—	Interval	Config	Transaction Byte	Destination Address	Source Address	Header

- Header
  As shown below, header indicates descriptor states.
  This area is read by DMAC before starting a DMA transfer in link mode. In addition, after the DMA transfer, the transfer status is written back by DMAC.

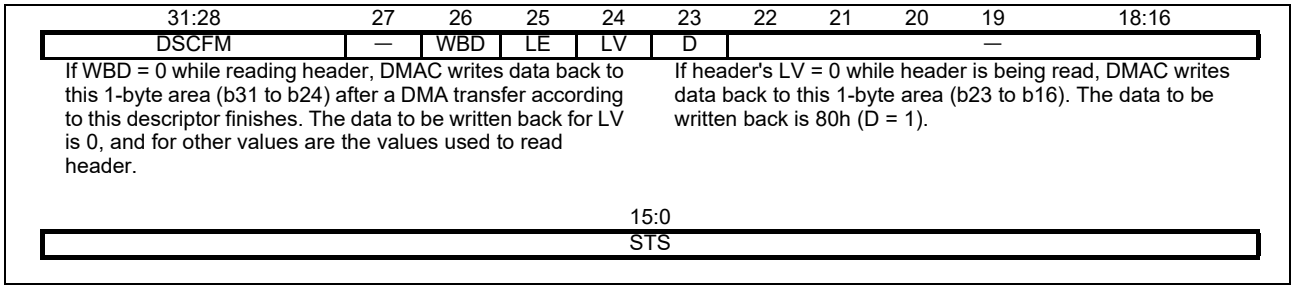


Figure 14.9 Header Area

Table 14.14 Header Area

Bit Position	Bit Name	Description
b15 to b0	STS	When DSCFM = 3, sets the transaction size in bytes. The maximum transfer bytes that can be set is 65,535 bytes. When DSCFM = 3, do not set 0 to the STS bit. If 0 is set, the operation is not guaranteed.
b22 to b16	—	A reserved area. Set 0.
b23	D	Indicates an access error of a descriptor. When LV = 0 while a descriptor is being read, DMAC writes 1 back to this bit. 0: No descriptor error 1: LV = 0 during descriptor read.
b24	LV	Indicates that this descriptor is enabled. When WBD = 0, DMAC writes 0 after the DMA transfer written in the descriptor. Set 1 when setting header. 0: The descriptor is disabled. 1: The descriptor is enabled.
b25	LE	Indicates that the link ends during DMA transfer of this descriptor. To indicate the end of the link, set this bit to 1. 0: The link continues. 1: The link ends.
b26	WBD	Masks write back execution of the LV bit. When this bit is 1, DMAC does not perform write-back operation. 0: The LV bit is written back to 0. 1: The LV bit is not written back.
b27	—	A reserved area. Set 0.
b31 to b28	DSCFM	Specifies the descriptor format (descriptor length, and combination). For details, see Table 14.11.

If you add a descriptor during DMA transfer completion processing (writing back to the descriptor), access of the CPU to set the LV bit to 1, and the access of DMAC to write 1 back to the D bit might conflict. Because of this, the data that was written first is overwritten with the data that was written later.
To avoid this problem, the byte lane of the D bit and the byte lane of the LV bit are placed differently. DMAC uses the byte write method for writing back the D bit. Therefore, to set LV = 1, also use the byte write method.

- Setting descriptors other than header
  Data in descriptors except header has the same specifications as an on-chip register. For details on the on-chip register specifications, see section 14.2, Register Descriptions.
  For descriptor setting examples, see section 14.5.3, Setting Example 3 (Link Mode).

- Descriptor areas and DMA transfer areas
  The following figure provides an overview of the descriptor area and the DMA transfer area to which DMAC accesses.

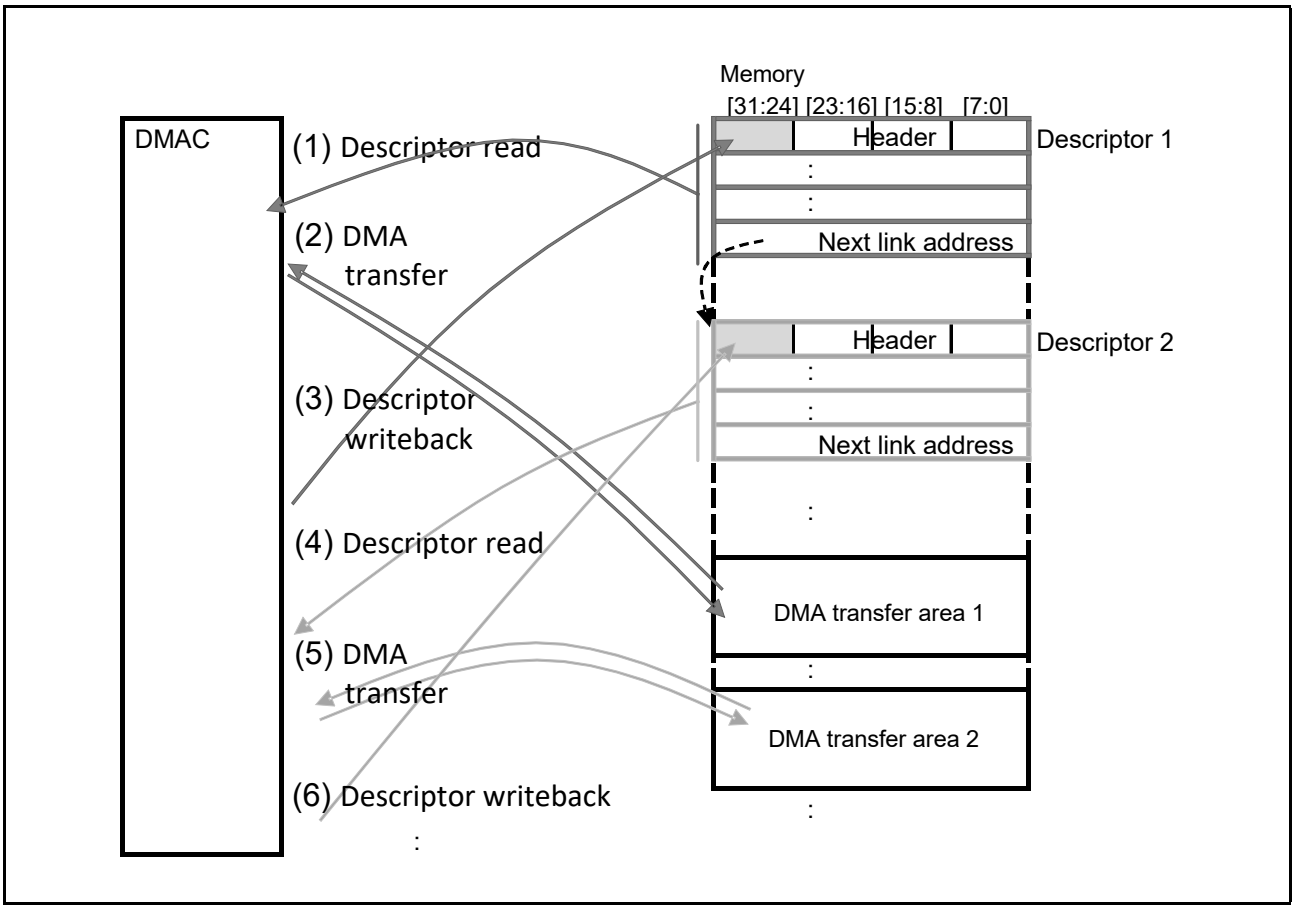


Figure 14.10 Header Area

(1) Descriptor read
Loads the value set for the on-chip NXLA_n register to the CRLA_n register, and then reads the descriptor from the memory space (descriptor 1) indicated by the CRLA_n register.

(2) DMA transfer
When the LV bit in header is 1, performs a DMA transfer according to the descriptor information.

(3) Descriptor write-back
After the DMA transfer for the set number of bytes, if the WBD bit in header is 0, the LV bit writes 0 and other fields writes the value read in step 1 as data back to header[31:24] of descriptor1 in bytes.

(4) Descriptor read
If the LE bit in header of the descriptor which was read previously (step 1) is 0, reads the next descriptor from the address (descriptor 2) indicated by the next link address in the descriptor.

(5) DMA transfer
When the LV bit in header is 1, performs a DMA transfer according to the descriptor information.

(6) Descriptor write-back
After the DMA transfer for the set number of bytes, if the WBD bit in header is 0, the LV bit writes 0 and other fields writes the value read in step 4 as data back to header[31:24] of descriptor2 in bytes.

Hereafter, repeats steps (4) through (6).

When LE of the header is 1 and WBD = 0, DMA transfer proceeds in accord with the descriptor settings, and 0 is written back to the LV bit of the header, after which operation ends.

When header's LE = 1, and WBD = 1, performs a DMA transfer with the setting, and ends the operation (no write-back is performed).

When header's LV = 0, writes 1 back to the D bit of header. And then, if the DRRP bit in the CHCFG_n register = 1, the descriptor is read again after the interval specified in the DITVL field in the DSCITVL_n register. When DRRP = 0, the operation stops.

- Notes on descriptors
  - In link mode, settings can be changed by reading a descriptor, but the timing for changing settings and for a hardware request cannot be synchronized. Because of this, if you want to issue a hardware request (an external interrupt), set the LVL, HIEN, LOEN, and SEL bits in the CHCFG_n register to 1 before setting the SETEN bit in the CHCTRL_n register. Besides, do not change these bits you set in a descriptor.

  - In a descriptor, the settings of the DMS field in the CHCFG_n register cannot be changed (always link mode). In addition, settings of the REN, RSW, and RSEL fields in the CHCFG_n register can be changed in the descriptor, but that does not affect the operation.

  - DMAC references the DSCFM field and the LV bit of header to determine if the descriptor is enabled or disabled. Therefore, initialize (DSCFM = 1 or 3, and LV = 1) the memory area equivalent to the LV bit of the DSCFM field before DMAC accesses it.

  - If you want to set the next descriptor on the memory while reading the transfer settings of DMA (during descriptor read), writes 1 to the LV bit after setting the descriptors after header (source address, destination address, ...next link address). This is to avoid DMA transfer using descriptor values (source address, destination address, and so on) before the setting if descriptor settings by the CPU and descriptor read of DMAC conflict, and DMAC's descriptor read interrupts descriptor settings by the CPU.

  - If you want to keep the information written back to the D bit of header, perform byte access to write 1 to the LV bit of header.

(4) LINK configuration examples

In link mode, descriptors can be configured as in the following figure.

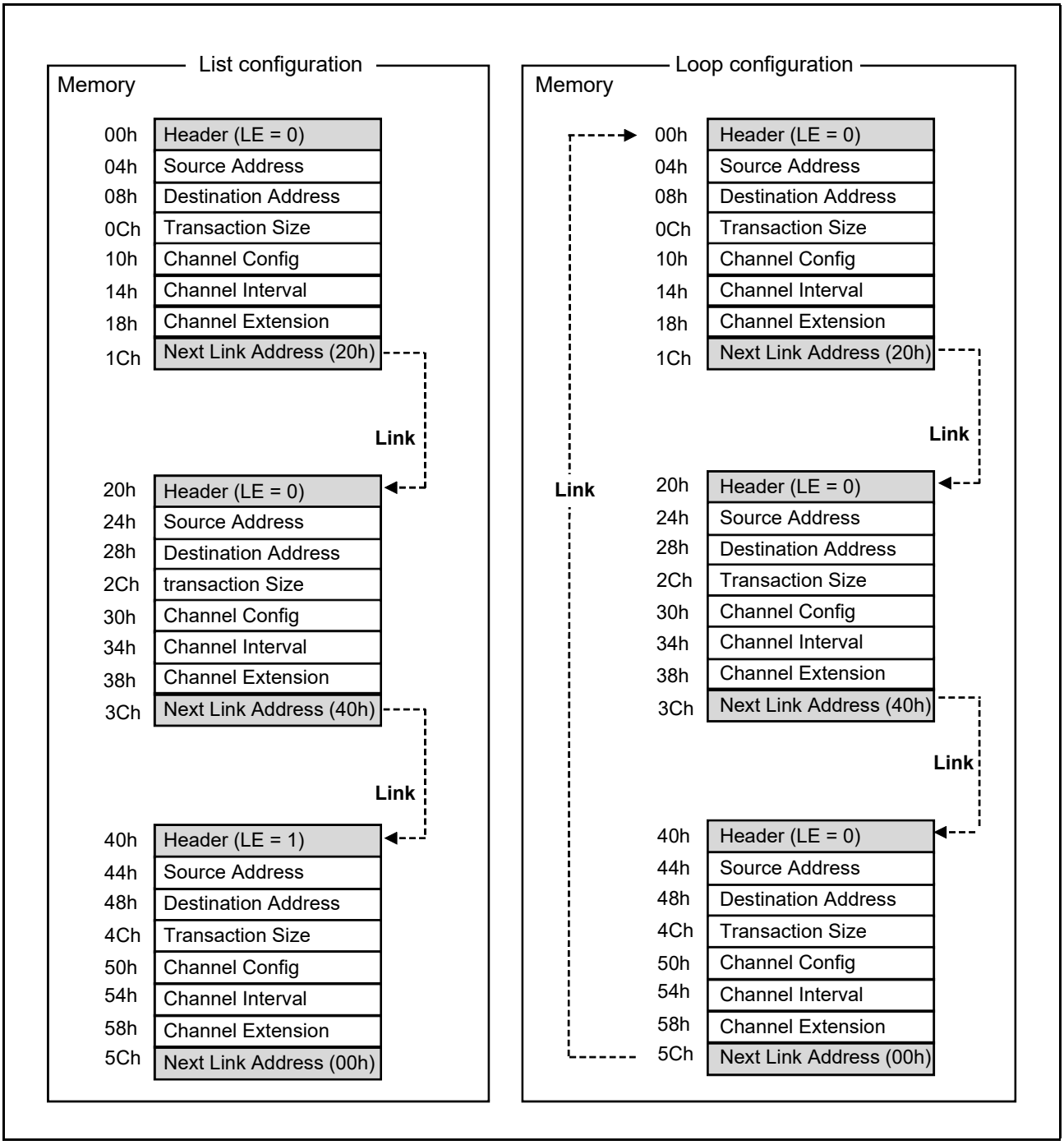


Figure 14.11 Header Area

- List configuration
  Setting 1 to the LE bit in header of the last descriptor ends the link.

- Loop configuration
  Setting the link destination of the last descriptor to the address of the previous descriptor configures the descriptors in a loop. To end the loop, change the LE bit of header to 1 before DMAC reads a descriptor, or follow the procedure to pause a transfer.

14.3.1.3 Write-Only Mode

Setting 1 to the WONLY bit in the CHCFG_n register falls in write-only mode.

Table 14.15 Write-Only Mode Settings

WONLY (CHCFG_n)	Mode	Description
0	Normal Mode	Performs a DMA transfer with the values set for the next register set
1	Write-Only Mode	Performs DMA write transfers only without performing DMA read transfers.

In write-only mode, read operation for DMA transfers is not performed (descriptor read is performed as same as in normal mode). In register mode, use the value set for the NxSA_n register (x = 0 when RSEL = 0, and x = 1 when RSEL = 1) as the write data. In link mode, use the value in the SA field of a descriptor as write data.
Use this mode for initializing a memory area.

14.3.2 Transfer Mode

Single transfer mode and block transfer mode are supported.
When selecting mode, use the TM bit in the CHCFG_n register for each channel.

Table 14.16 Basic Transfer Settings

Transfer Mode	TM (CHCFG_n)	Description
Single transfer	0	For one DMA request, performs a DMA transfer.
Block transfer	1	For one DMA request, performs transfers until a DMA transfer completes.

14.3.2.1 Single Transfer Mode

When a DMA transfer request is accepted, performs a DMA transfer. Every time a transfer request is accepted, transfer proceeds. This operation is repeated until it reaches the transfer size loaded from the N0TB_n or N1TB_n register to the CRTB_n register (arbitration between channels is performed for each DMA transfer).

14.3.2.2 Block Transfer Mode

Once a DMA transfer request is accepted, the transfer continues until transfer of the number of bytes loaded from the N0TB_n or N1TB_n register to the DMA transfer byte register (CRTB_n register) is completed (DMA transfer completion; arbitration between channels is performed for each DMA transfer).

14.3.3 DMA Channel Priority Control

As an arbitration method between channels, fixed priority mode and round-robin mode are supported. To select mode, use the PR bit in the DCTRL register. When the PR bit is 0, fixed priority mode is selected. When the PR bit is set to 1, round-robin mode is selected.

Table 14.17 Priority Control Settings

Transfer Mode	PR (DCTRL)	Description	Applications
Fixed Priority	0	Channels 0 to 7, channels 8 to 15 are fixed priority mode. CH0 (CH8) > CH1 (CH9) > CH2 (CH10) > CH3 (CH11) > CH4 (CH12) > CH5 (CH13) > CH6 (CH14) > CH7 (CH15)	Select this mode if channels have priority.
Round-Robin	1	Controls requests in round-robin mode.	Select this mode if you want to execute requests equally.

14.3.3.1 Fixed Priority Mode

In fixed priority mode, the priority levels are fixed within the group of channels 0 to 7 and within the group of channels 8 to 15. In addition, the priority levels between the group of channels 0 to 7 and the group of channels 8 to 15 are in round-robin mode.
The priority levels immediately after a reset and transfer through DMA channel 0 are shown in Figure 14.12.

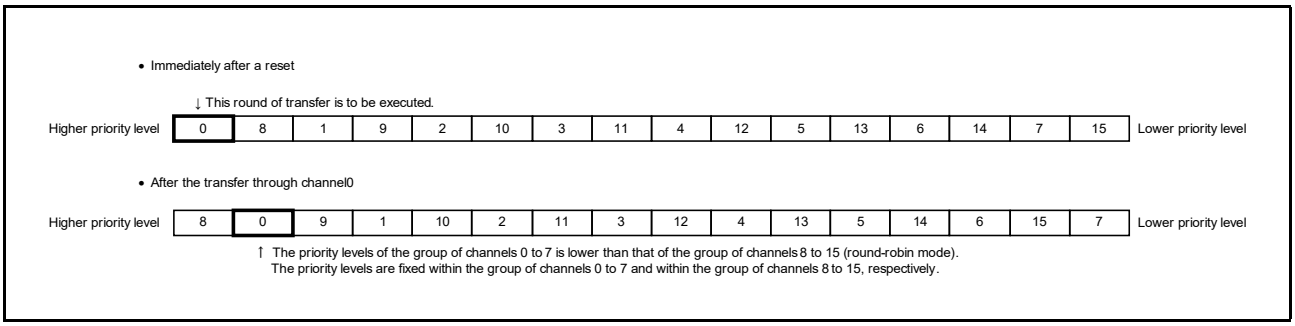


Figure 14.12 Priority Levels Immediately after a Reset and Transfer through DMA Channel 0

If DMA transfer requests are generated on multiple channels simultaneously, a DMA transfer request with a smaller channel number takes precedence.

Note: Channel 0 is handled with the highest priority, but the transfer with the next highest priority is performed because bus arbitration is performed after reading data from the transfer source of channel 0 finishes (Another read operation might interrupt the operation between reading data from and writing data to the same channel).

14.3.3.2 Round-Robin Mode

In round-robin mode, priority is changed every time a transfer of a channel is accepted so that the lowest priority is given to the channel in which the last transfer is performed.
The priority levels immediately after a reset and transfer through DMA channel 2 are shown in Figure 14.13.

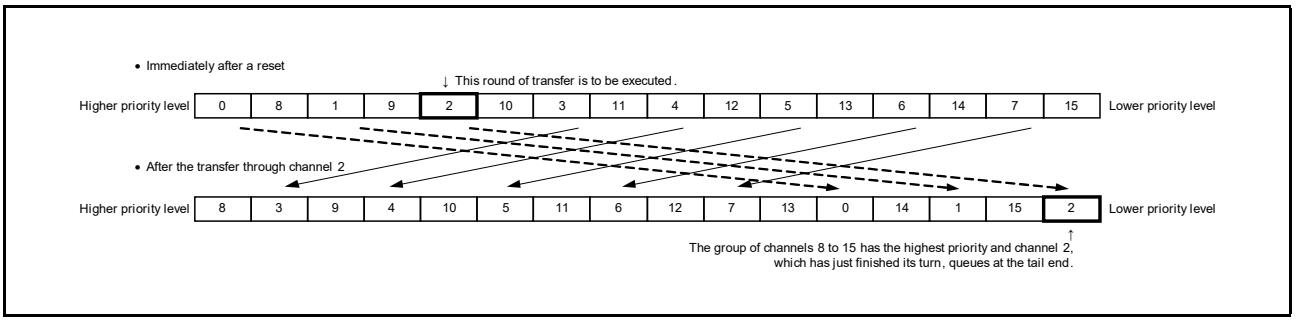


Figure 14.13 Priority Levels Immediately after a Reset and Transfer through DMA Channel 2

14.3.4 DMA Transfer Request

DMA activation requests have three types including software requests, on-chip peripheral module requests, and external interrupts.
Select the transfer request source for an on-chip peripheral module request, external interrupt, and software request with the DMAmSELn (m = 0 or 1, n = 0 to 15) register.
For details on the DMAmSELn (m = 0 or 1, n = 0 to 15) register, see section 14.2.8, DMAC Unit 0 Source Select Register i (DMA0SELi) (i = 0 to 15), and section 14.2.9, DMAC Unit 1 Source Select Register i (DMA1SELi) (i = 0 to 15).

14.3.4.1 Specifying Detection Operation of DMA Transfer Requests for Each Source

Detection methods for DMA transfer requests of on-chip peripheral module requests, external interrupts, and software requests might be specified according to sources.
For each DMA transfer source, set the LVL, HIEN, and LOEN bits in the CHCFG_n register according to Table 14.18, Table 14.19, and Table 14.20.

Table 14.18 Detection Operation Specification for Each Source of DMA Transfer Requests

DMA Transfer Request Source	Detection Operation Specification of DMA Transfer Requests
On-chip peripheral module request	Depends on specifications of the DMA transfer request source. (See Table 14.20.)
External interrupt	Detects the rising edge. Detects the high level.
Software request	Detects the rising edge.

Table 14.19 Method of Detecting DMA Transfer Request Signals

Mode	LVL (CHCFG_n)	HIEN (CHCFG_n)	LOEN (CHCFG_n)	Description
Edge detection	0	0	0	Disables detection.
			1	Detects the falling edge.
		1	0	Detects the rising edge.
			1	Setting prohibited
Level detection	1	0	0	Disables detection.
			1	Detects the low level.
		1	0	Detects the high level.
			1	Setting prohibited

Table 14.20 DMA Transfer Request Detection Operation Setting Table (1 / 2)

DMA Transfer Request Source	DMA Transfer Source	Transfer Source	Transfer Destination	DMAm SELn [7:0]	CHCFG_n				
					TM	LVL	HIEN	LOEN	SEL[2:0]
External Interrupt	IRQ0	Arbitrary	Arbitrary	04h	0/1	0/1 *1	1*1	0*1	DMAC0/1 ch0/8: 0h ch1/9: 1h ch2/10: 2h ch3/11: 3h ch4/12: 4h ch5/13: 5h ch6/14: 6h ch7/15: 7h
	IRQ1			05h					
	IRQ2			06h					
	IRQ3			07h					
	IRQ4			08h					
	IRQ6			0Ah					
	IRQ7			0Bh					
CMT Unit 0	Compare match 0	Arbitrary	Arbitrary	15h	0/1	0	1	0	
	Compare match 1			16h					
CMT Unit 1	Compare match 0	Arbitrary	Arbitrary	17h	0/1	0	1	0	
	Compare match 1			18h					
CMTW Unit 0	Compare match	Arbitrary	Arbitrary	19h	0/1	0	1	0	
	Input capture 0			1Ah					
	Input capture 1			1Bh					
	Output compare 0			1Ch					
	Output compare 1			1Dh					
CMTW Unit 1	Compare match	Arbitrary	Arbitrary	1Eh	0/1	0	1	0	
	Input capture 0			1Fh					
	Input capture 1			20h					
	Output compare 0			21h					
	Output compare 1			22h					
S12ADCa Unit 0	AD conversion completed	ADDRn	Arbitrary	23h	0/1	0	1	0	
	Group B Conversion completed	ADDRn		24h					
S12ADCa Unit 1	AD conversion completed	ADDRn	Arbitrary	26h	0/1	0	1	0	
	Group B Conversion completed	ADDRn		27h					
DMAC0	DMAC0 Software trigger	Arbitrary	Arbitrary	FBh	0/1	0	1	0	
DMAC1	DMAC1 Software trigger	Arbitrary	Arbitrary	FCh	0/1	0	1	0	
RSPI Channel 0	Reception buffer full	SPDR	Arbitrary	50h	0	0	1	0	
	Transmission buffer empty	Arbitrary	SPDR	51h	0	0	1	0	
RSPI Channel 1	Reception buffer full	SPDR	Arbitrary	54h	0	0	1	0	
	Transmission buffer empty	Arbitrary	SPDR	55h	0	0	1	0	
SCIFA Channel 0	Reception buffer full	FRDR	Arbitrary	61h	0	1	1	0	
	Transmission buffer empty	Arbitrary	FTDR	62h	0	1	1	0	
SCIFA Channel 1	Reception buffer full	FRDR	Arbitrary	65h	0	1	1	0	
	Transmission buffer empty	Arbitrary	FTDR	66h	0	1	1	0	

Table 14.20 DMA Transfer Request Detection Operation Setting Table (2 / 2)

DMA Transfer Request Source	DMA Transfer Source	Transfer Source	Transfer Destination	DMAm SELn [7:0]	CHCFG_n				
					TM	LVL	HIEN	LOEN	SEL[2:0]
SCIFA Channel 2	Reception buffer full	FRDR	Arbitrary	6Eh	0	1	1	0	DMAC0/1 ch0/8: 0h ch1/9: 1h ch2/10: 2h ch3/11: 3h ch4/12: 4h ch5/13: 5h ch6/14: 6h ch7/15: 7h
	Transmission buffer empty	Arbitrary	FTDR	6Fh	0	1	1	0	
SCIFA Channel 4	Reception buffer full	FRDR	Arbitrary	76h	0	1	1	0	
	Transmission buffer empty	Arbitrary	FTDR	77h	0	1	1	0	
RIIC Channel 0	Data reception completed	ICDRR	Arbitrary	7Ah	0	0	1	0	
	Transmission data empty	Arbitrary	ICDRT	7Bh	0	0	1	0	
RIIC Channel 1	Data reception completed	ICDRR	Arbitrary	7Dh	0	0	1	0	
	Transmission data empty	Arbitrary	ICDRT	7Eh	0	0	1	0	
TPUa Unit 0	TGI0A	Arbitrary	Arbitrary	D8h	0	0	1	0	
	TGI0B			D9h					
	TGI1A			DDh					
	TGI1B			DEh					
	TGI2A			E1h					
	TGI2B			E2h					
	TGI3A			E5h					
	TGI3B			E6h					
	TGI4A			EAh					
	TGI4B			EBh					
	TGI5A			EEh					
	TGI5B			EFh					
ELC	ELCIRQ1	Arbitrary	Arbitrary	F2h	0	0	1	0	
	ELCIRQ2			F3h					

Note 1. Set the LVL, HIEN, and LOEN bits of the external interrupt (IRQ0 to IRQ4, IRQ6, IRQ7) as follows.
For the setting of the IRQCRi (i = 0 to 4, 6, 7) register, see section 12.3.1, Selecting Interrupt Request Destinations.
LVL: Set according to the edge/level setting of the IRQCRi register.
HIEN: Set to 1 regardless of the detection level of the IRQCRi register.
LOEN: Set to 0 regardless of the detection level of the IRQCRi register.

Remarks: CHCFG_n register setting values
TM Bit
0: Single transfer
1: Block transfer
LVL bit
0: Detects the edge of a DMA request.
1: Detects the level of a DMA request.

Note: Instead of an interrupt source from peripheral modules, transfer completion sources of DMAC channels selected by the DMAC source select register are connected to the vector number selected by the DMAC source select register.

Note: Be sure to select edge detection with the PLSn register as the transfer completed interrupt detection type for the DMAC.

14.3.4.2 Edge Detection

Setting the LVL bit in the CHCFG_n register to 0 detects the edge.
Setting the HIEN bit in the CHCFG_n register to 1 detects the rising edge, and setting the LOEN bit in the CHCFG_n register to 1 detects the falling edge.

14.3.4.3 Level Detection

Setting the LVL bit in the CHCFG_n register to 1 detects the level.
If a DMA transfer request is active (according to HIEN or LOEN settings) for two consecutive clocks (ICLK) or more, it is recognized as a DMA request.

14.3.5 Forced Ejection Request

When a forced ejection request is input, data that is not yet transferred in a buffer is transferred to the DMA transfer destination address. After data is flushed, the DMA transfer resumes.

The following are notes on forced ejection requests:

- If a forced ejection request conflicts with DMA transfer request input, the forced ejection takes precedence, and then the DMA transfer is performed.

- Differences from flush mode described in section 14.3.10.2, Aborting a Transfer (Buffer Flush: SBE = 1) (The EN bit is cleared to 0 when the SBE bit in the CHCFG_n register = 1).

Flush mode: DMAC stops operation after data in a buffer is written.
Forced ejection request: A DMA transfer continues after flush operation ends.

14.3.5.1 Software Forced Ejection Request

For software forced ejection requests, use the SETSSWPRQ bit in the CHCTRL_n register.
To request a forced ejection, set 1 to the SETSSWPRQ bit. DMAC outputs data in a buffer to the DMA transfer destination.

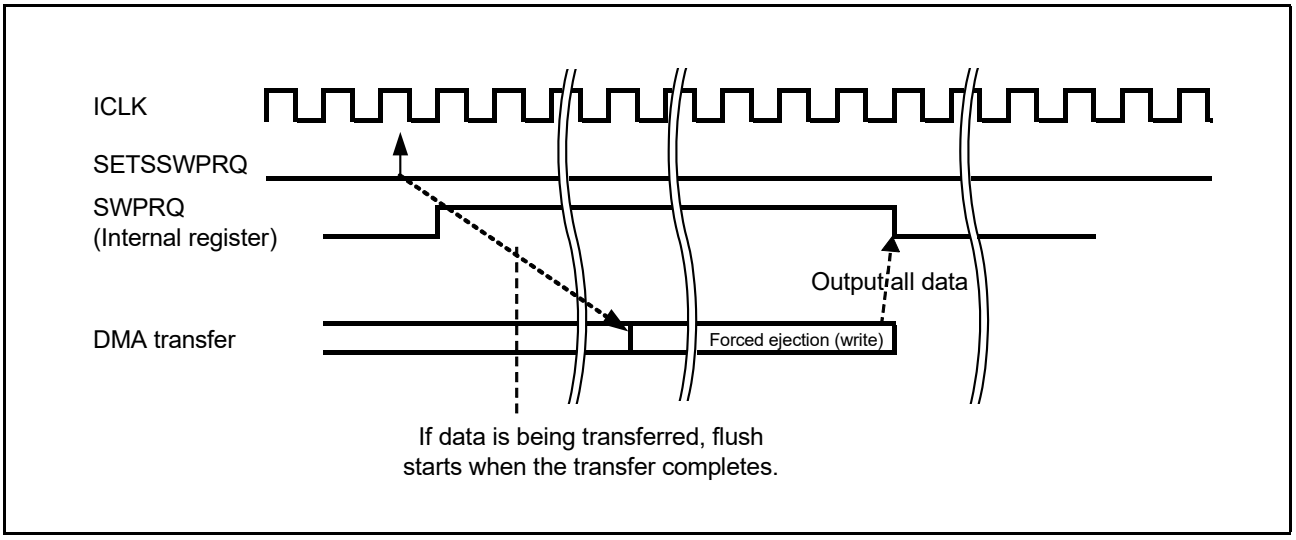


Figure 14.14 Software Forced Ejection Timing

14.3.6 Interval Count Function

By the setting for the ITVL field in the CHITVL_n register, the execution interval of DMA transfers can be adjusted. This function is used to avoid occupation of a bus by DMAC. Until the count value becomes 0, no DMA transfer for the next DMA request is performed.

14.3.7 Differences in Operation According to the Transfer Data Size

14.3.7.1 When the Transfer Data Size on the Transfer Source is Small

As the transfer data size at the destination is large, reading from the source proceeds several times, and this is followed by writing to the destination.

14.3.7.2 When the Transfer Data Size on the Transfer Destination is Small

Because the transfer data size on the transfer source is large, after a single read operation, write operation to the transfer destination is performed a few times.

14.3.7.3 When the Size of Transfer Data on the Transfer Destination and on the Transfer Destination is the Same

Every time a DMA transfer request is detected, read operation is performed on the transfer source, and write operation is performed on the transfer destination.

14.3.8 DMA Transfer Status

The CHSTAT_n register indicates the DMA transfer status of each channel.
The TACT bit in the CHSTAT_n register indicates that DMA operation is being performed on channel n. Writing 1 to the SETEN bit in the CHCTRL_n register sets 1. The TACT bit remains 1 while accessing a descriptor, or waiting for a DMA request.
The TACT bit is cleared when the EN bit in the CHSTAT_n register is cleared (for details on clear conditions, see section 14.2.7, Channel Status Register n (CHSTAT_n)), and DMA transfers for the set number of times are finished.
The TACT bit is not cleared even when a DMA transfer finishes, but the EN bit is not cleared (when the REN bit in the CHCFG_n register = 1 in register mode, or the next descriptor access is performed in link mode).

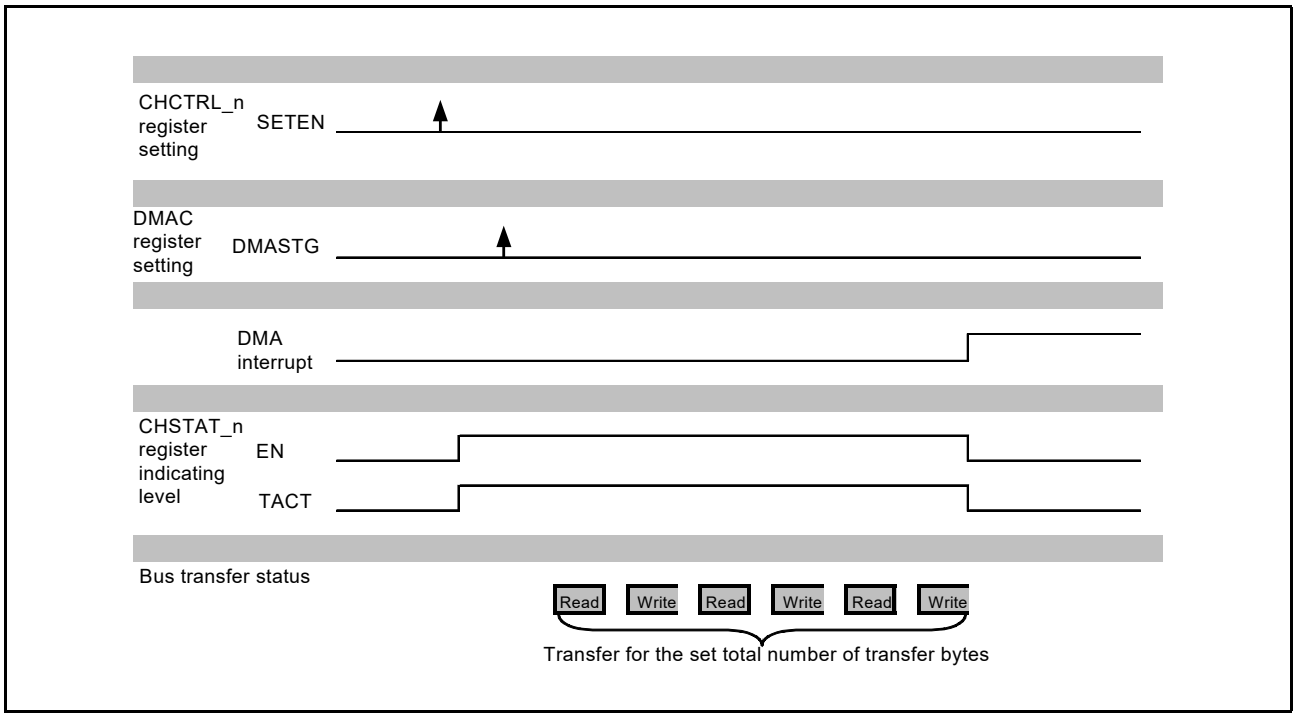


Figure 14.15 DMAC Status Example (Software Request)

14.3.9 Suspending a Transfer

You can suspend a DMA transfer at the SETSUS bit in the CHCTRL_n register. At that time, if there is an already running bus cycle, waits for the cycle to end, and then suspends the transfer. Writing 1 to the CLRSUS bit in the CHCTRL_n register resumes from the suspended state.

To check if the transfer is suspended, set the SETSUS bit in the CHCTRL_n register, and then make sure that the SUS bit in the CHSTAT_n register, or the SUS bit in the DST_SUS register on the applicable channel is set to 1.

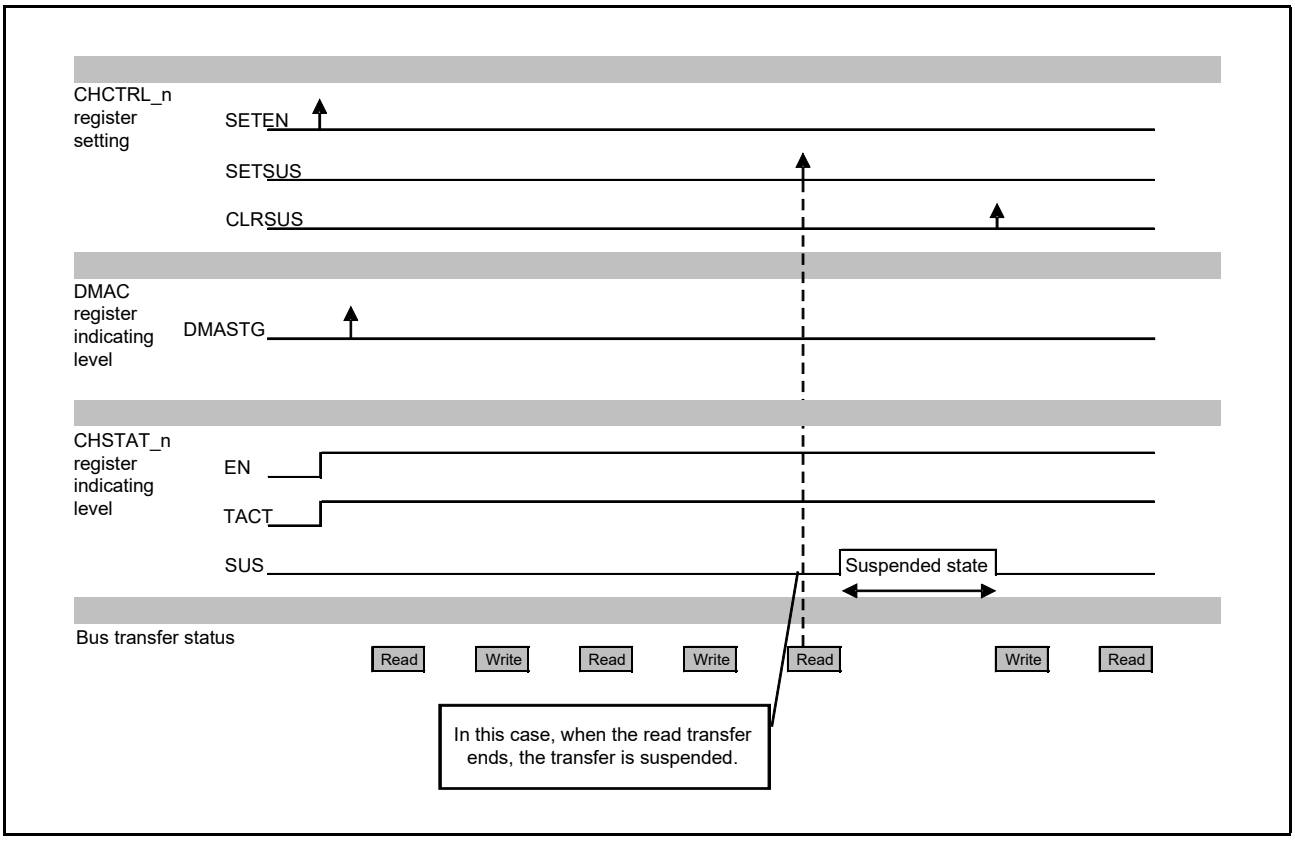


Figure 14.16 DMAC Suspended State (Software Request Block Transfer)

14.3.10 Aborting a Transfer

During a DMA transfer, if you write 1 to the CLREN bit in the CHCTRL_n register, you can abort the DMA transfer of the channel. As processing after aborting the transfer, you can use the SBE bit in the CHCFG_n register to determine whether to flush data remaining in a buffer when a transfer is suspended. By default, SBE = 0 (do not flush data) is selected.
When this mode (flush data) is activated, if a transfer which is being performed when the CLREN bit in the CHCTRL_n register = 1 is aborted, data remaining in the buffer of DMAC is flushed, and the operation stops.

14.3.10.1 Aborting a Transfer (No Buffer Flush: SBE = 0)

During a DMA transfer, if you write 1 to the CLREN bit in the CHCTRL_n register, you can abort and then stop the DMA transfer. After the transfer stops, write 1 to the SWRST bit in the CHCTRL_n register, and clear the contents within the DMAC before setting the next transfer.

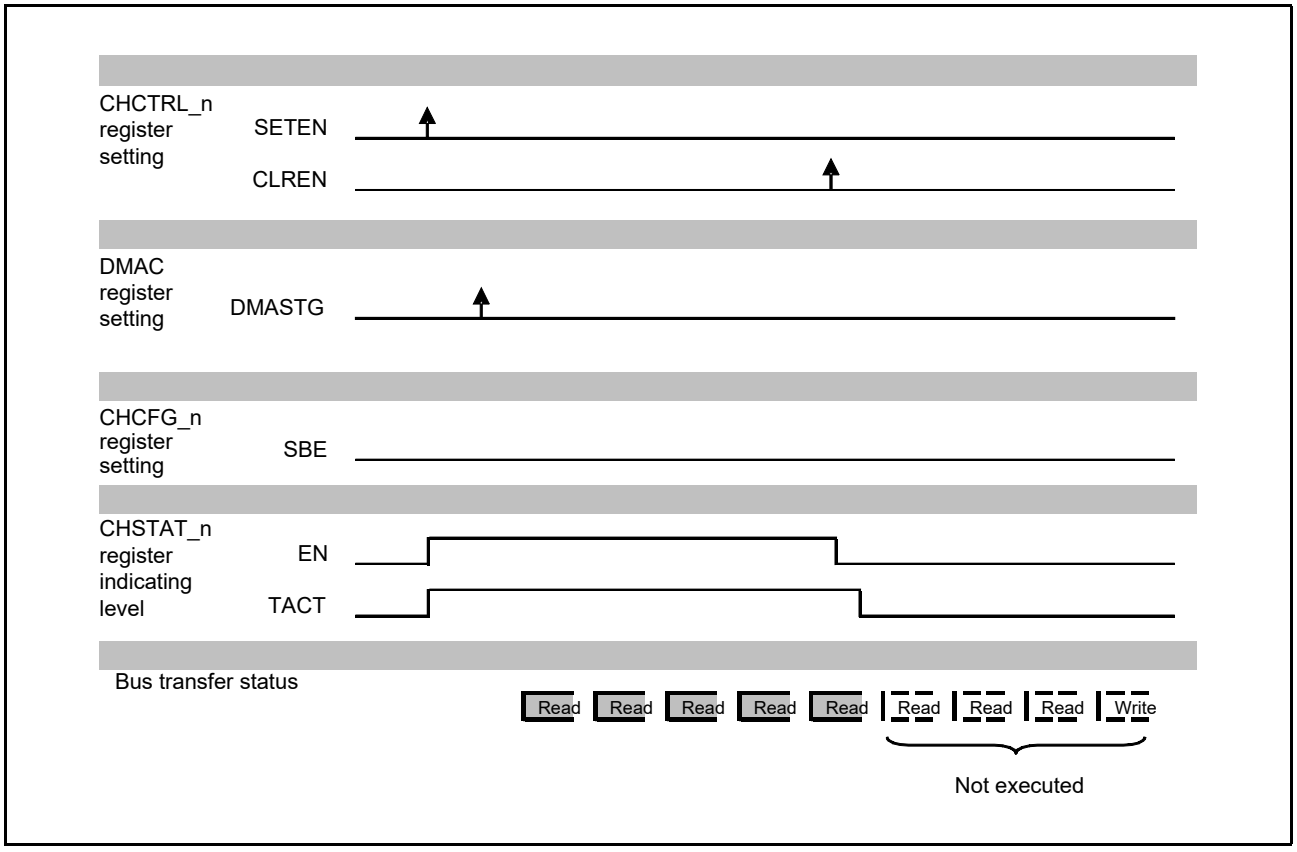


Figure 14.17 Aborting a DMA Transfer

- When the TACT bit in the CHSTAT_n register is cleared, you can confirm that the channel stops completely.
- If a DMA transfer is aborted, no DMA transfer completion interrupt is generated.
- If a DMA transfer is aborted, the transfer stops when the next read operation completes. (Note that if there is data that can be written in the buffer, the transfer stops after the data is written.)

14.3.10.2 Aborting a Transfer (Buffer Flush: SBE = 1)

During a DMA transfer, if you write 1 to the CLREN bit in the CHCTRL_n register, you can abort the DMA transfer. After the transfer stops, set the SWRST bit in the CHCTRL_n register, and clear the contents within the DMAC before setting the next transfer.

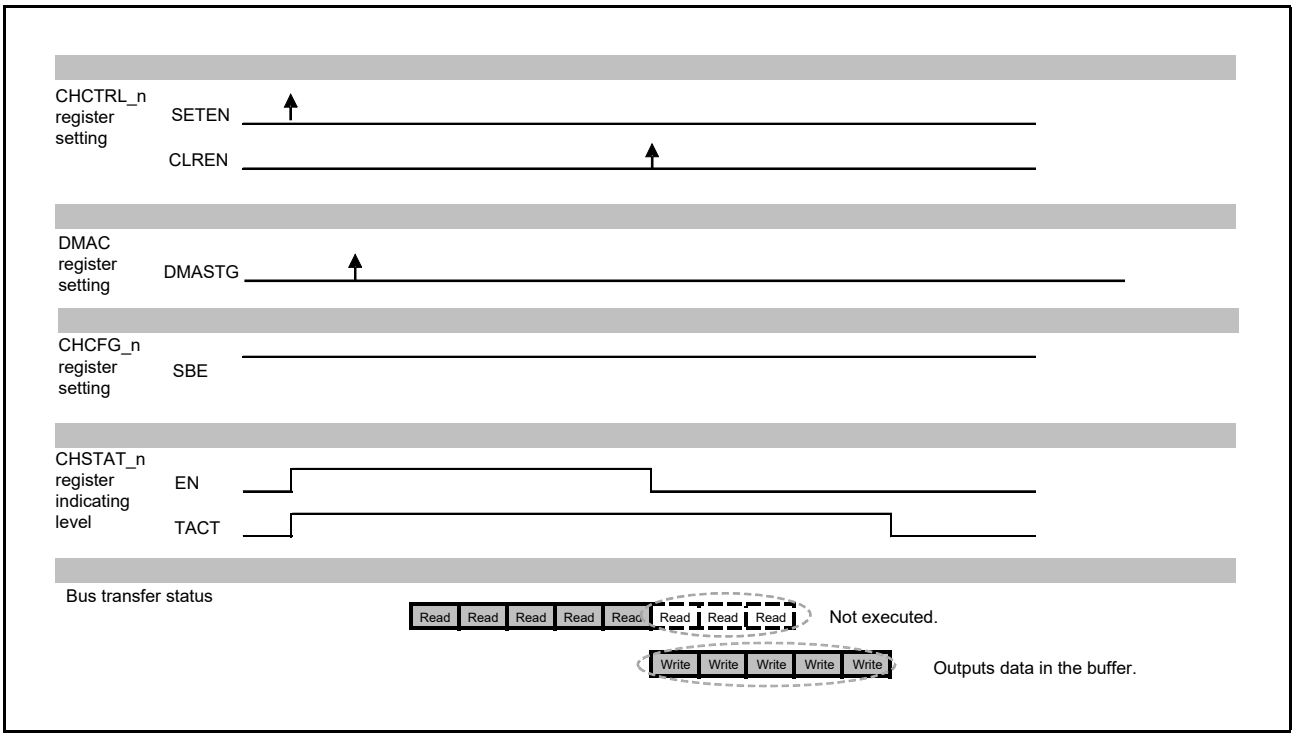


Figure 14.18 Aborting DMA Transfer (Buffer Flush Mode)

- The above figure shows an example when 1 is written to the CLREN bit in the CHCTRL_n register during the fifth read transfer in flush mode (the SBE bit in the CHCFG_n register = 1), and the transfer is aborted. It illustrates how read data is written, and the DMA transfer stops.
- When the TACT bit in the CHSTAT_n register is cleared to 0, you can confirm that the channel stops completely.

14.3.10.3 Checking If the Channel Stops

When 1 is written to the CLREN bit in the CHCTRL_n register, and the EN bit in the CHSTAT_n register is cleared to 0, if a transfer is already performed on a bus, DMAC cannot stop immediately. To check if DMA stops completely, make sure that the EN bit is cleared to 0, and the TACT bit in the CHSTAT_n register is cleared to 0.

14.3.10.4 Procedure for Aborting a Transfer

The following is the procedure for stopping a transfer:

1. Write 1 to the CLREN bit in the CHCTRL_n register.
2. By reading the CHSTAT_n register, make sure the TACT bit is cleared to 0. When TACT = 0, DMA stops completely. When TACT = 1, perform polling until the TACT bit is cleared to 0.
3. After a transfer is aborted, if you want to perform the next DMA transfer, you must set the SWRST (software reset) bit in the CHCTRL_n register immediately before the next transfer starts.

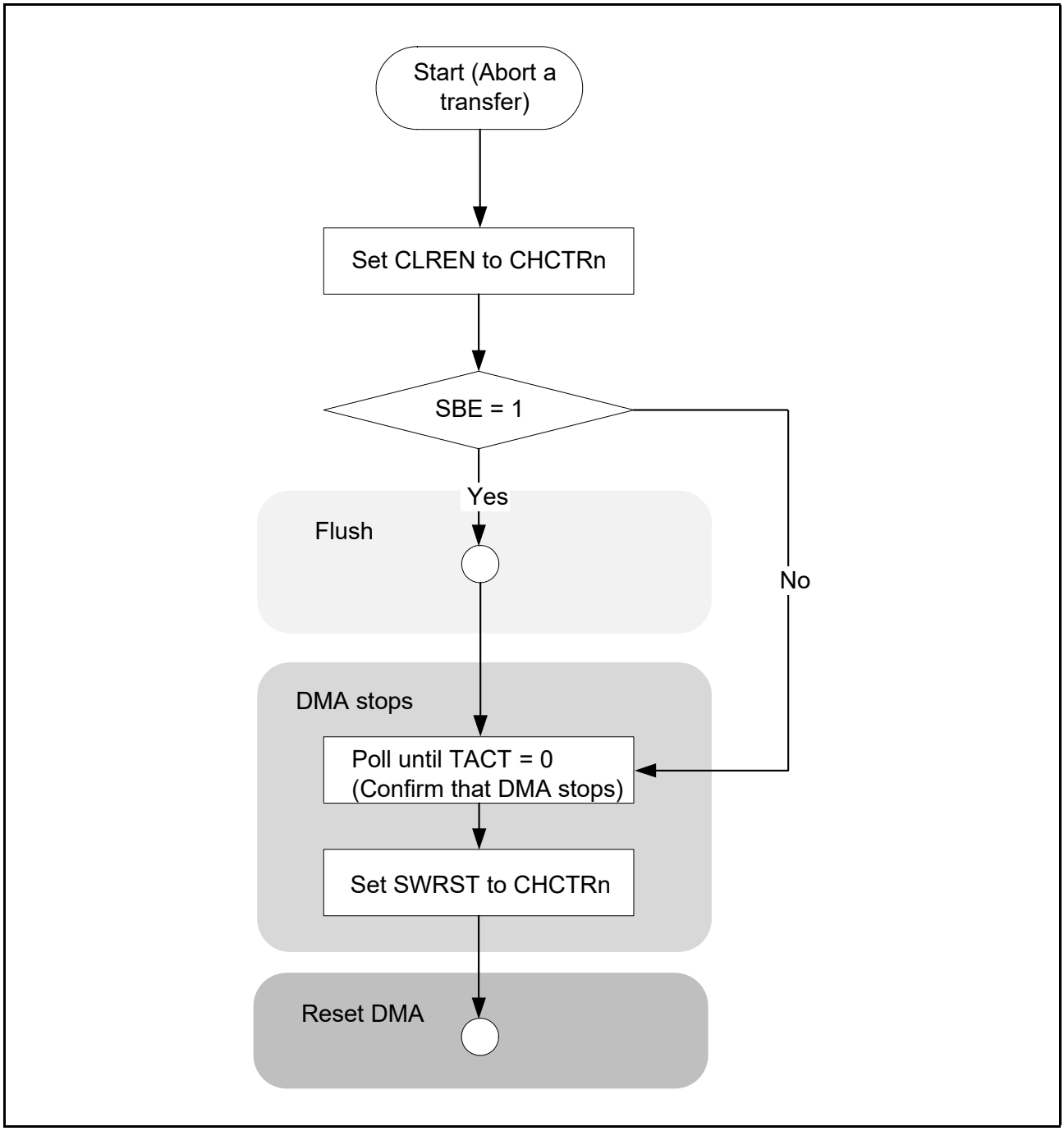


Figure 14.19 Operation Flow of Aborting a Transfer

14.4 Interrupts

14.4.1 Interrupt Sources

DMAC has two types of interrupt sources, such as DMA transfer completion interrupts and DMA error interrupts for each channel.
Table 14.21 shows the relationship among interrupt sources, enable bits, and status flags.

Table 14.21 Interrupt Sources of DMAC

Interrupt Source		Interrupt Enable Bit	Interrupt Status Flag	Output Condition
DMA transfer completion interrupt	DMA transfer completion	CHCFG_n.DEM	CHSTAT_n.END	When a transfer for the total number of transfer bytes loaded to the CRTB_n register completes (after a write back if write back is performed in link mode)
	Descriptor invalid	CHCFG_n.DIM		In link mode, when the DRRP and DIM bits in the CHCFG_n register = 0, and header of the read descriptor's LV = 0
DMA error interrupt		— (Mask disabled)	CHSTAT_n.ER	When a bus error occurs during a DMA transfer and descriptor access

14.4.2 DMA Transfer Completion Interrupts

A DMA transfer completion interrupt is an interrupt request signal indicating that the DMA transfer completes.
Each bit of the DMA transfer completion interrupt corresponds to each channel.

When the transfer of the total number of bytes for transfer loaded to the CRTB_n register is completed, the END bit in the CHSTAT_n register is set to 1. At that time, if the DEM bit in the CHCFG_n register = 0, a DMA transfer completion interrupt is generated (n = 15 to 0). To perform write back in link mode, an interrupt is generated after the write back.

In addition, when the DRRP bit in the CHCFG_n register = 0 in link mode, and header of the read descriptor is LV = 0, the DER bit in the CHSTAT_n register is set to 1. At that time, if the DIM bit in the CHCFG_n register = 0, a DMA transfer completion interrupt is generated.

Note: Instead of an interrupt source from peripheral modules, transfer completion sources of DMAC channels selected by the DMAC source select register are connected to the vector number selected by the DMAC source select registers (DMA0SELi, DMA1SELi). (The vector number selected by the DMA source selection register is handled as the vector number of the DMA transfer completed interrupt.)

Note: Be sure to select edge detection with the PLSn register as the transfer completed interrupt detection type for the DMAC.

For details, see section 12.3.1, Selecting Interrupt Request Destinations.

14.4.3 DMA Error Interrupt

If a bus error occurs during a DMA transfer or descriptor access, this module determines an error occurred, and stops the transfer. When a bus error occurs, the EN bit in the CHSTAT_n register for channel n where a transfer is performed is cleared to 0, and the ER bit is set to 1 (n = 15 to 0). In addition, a DMA error interrupt is generated.
DMA error interrupts cannot be masked.

Data for a series of error transfers cannot be guaranteed. You must perform the transfer from the beginning by using the following procedure.

1. Set the SWRST bit in the CHCTRL_n register to 1.
2. Set each register again.

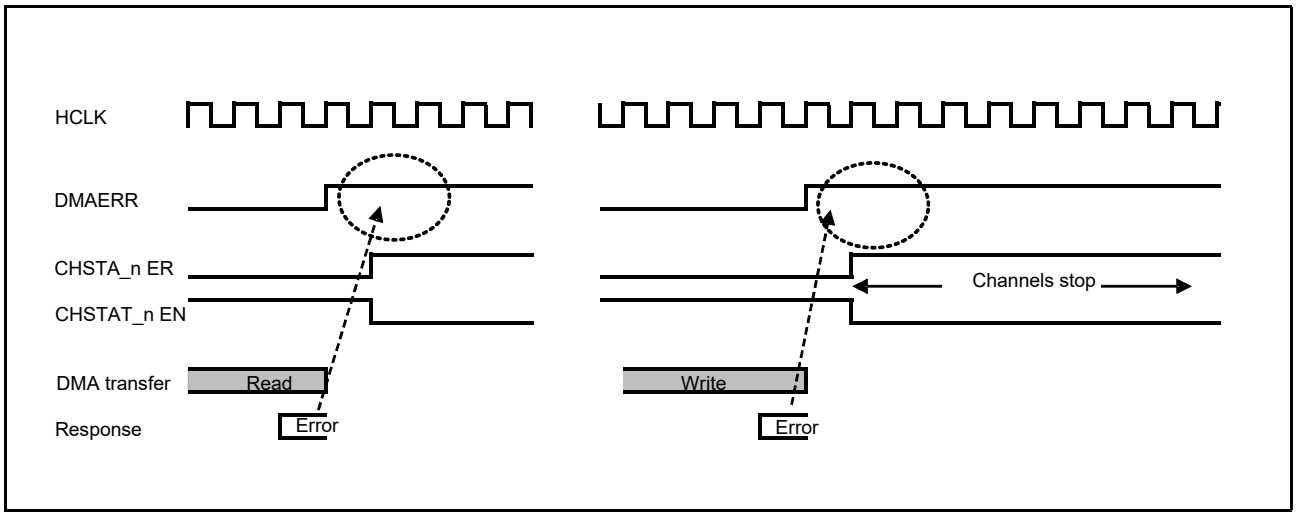


Figure 14.20 Stop Timing for Responding to a Bus Error

Note: When the CPU accesses a register of DMAC, if a bus error occurs, no DMA error interrupt is generated.

14.5 DMA Setting Examples

This section provides examples of DMA transfers. The following table lists the transfer conditions for the setting examples described in this section.

Table 14.22 List of Transfer Conditions for the DMA Transfer Setting Examples

Setting Example	DMA Mode	Transfer Mode	Transfer Request
Setting example 1	Register mode	Block transfer mode	Software
Setting example 2	Register mode (continuous execution)	Block transfer mode	Software
Setting example 3	Link mode	Block transfer mode	Software

14.5.1 Setting Example 1 (Register Mode Software Request)

This subsection provides a setting example of DMA transfers that use software requests in register mode.

Table 14.23 DMA Transfer Setting Example 1

Item		Description	
Channel to use		DMAC0 channel 2	
Priority control		Round-robin	
DMA mode		Register mode	
Transfer mode		Block transfer mode	
Register set to use		Next1 register set	
Transfer source/transfer destination		Transfer source	Transfer destination
	Start address	0400 0000h	2000 0000h
	Address direction	Increment	Increment
	Data size	8 bits	256 bits
Number of DMA transfer bytes		128 bytes	
DMA transfer request		Software request	
DMA transfer completion interrupt output mask		None	

Setting Example 1
DCTRL = 0000 0001h (DMA setting)

N1SA = 0400 0000h (Transfer source address)
N1DA = 2000 0000h (Transfer destination address)
N1TB = 0000 0080h (Number of transfer bytes)

CHCFG = 1045 0222h (Configuration)
CHITVL = 0000 0000h (Interval)

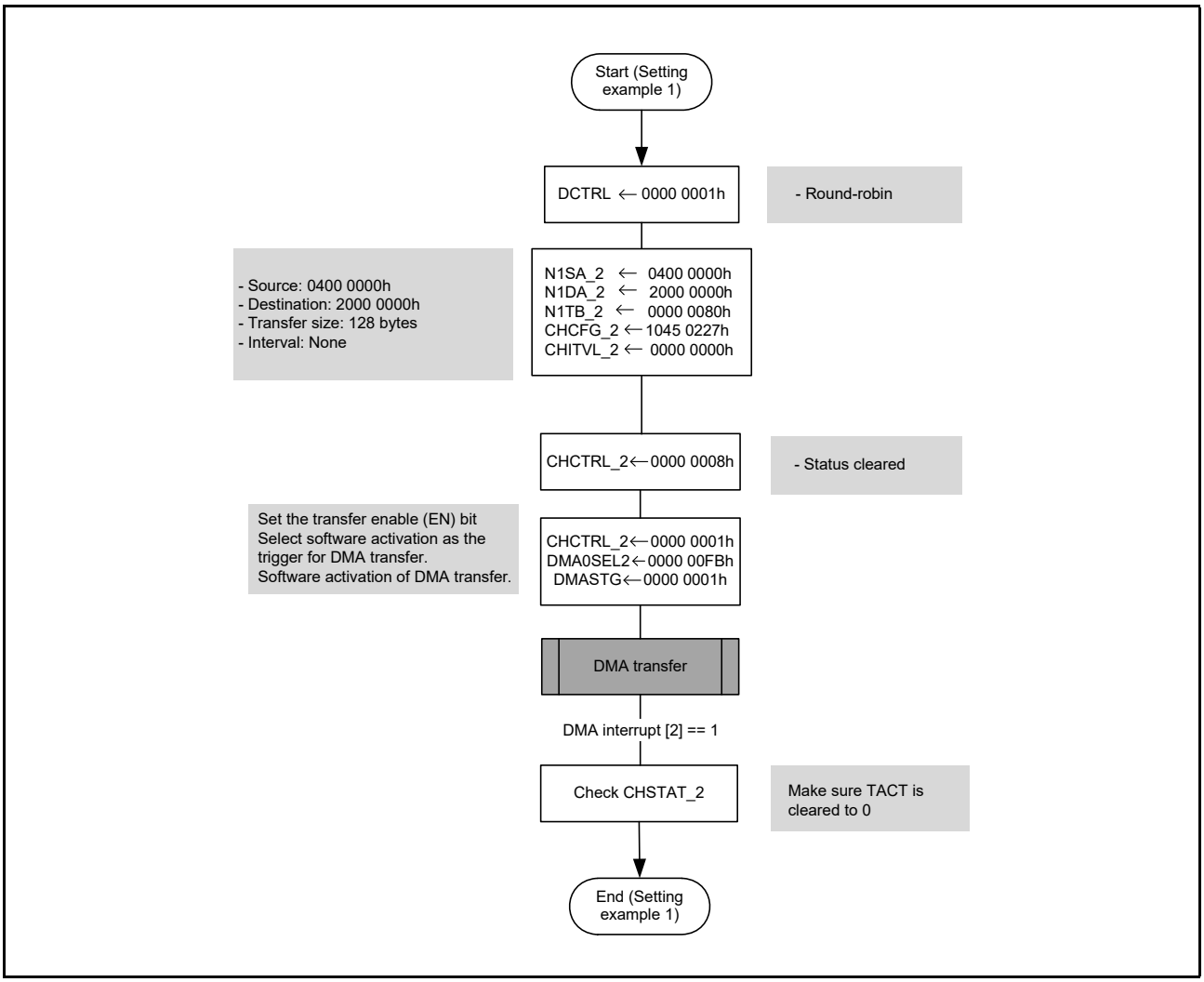


Figure 14.21 Setting Example 1

Note: DMA interrupt [2] indicates the interrupt source assigned to serve as the trigger for DMA transfer on channel 2.

14.5.2 Setting Example 2 (Register Mode Continuous Execution)

This subsection provides a setting example of DMA transfers that use Next0/1 Register Sets in series in register mode.

Table 14.24 DMA Transfer Setting Example 2

Item		Description	
Channel to use		DMAC0 channel 1	
Priority control		Round-robin	
DMA mode		Register mode	
Transfer mode		Block transfer mode	
Register set to use		Next0 register set -> Next1 register set in series	
Transfer source/transfer destination (Next0)		Transfer source	Transfer destination
	Start address	1111 0000h	2000 0000h
	Address direction	Fixed	Fixed
	Data size	32 bits	512 bits
	Number of DMA transfer bytes	512 bytes	
Transfer source/transfer destination (Next1)		Transfer source	Transfer destination
	Start address	0400 0000h	1000 0000h
	Address direction	Fixed	Fixed
	Data size	32 bits	512 bits
	Number of DMA transfer bytes	2,048 bytes	
DMA transfer request		Software request	
DMA transfer completion interrupt output mask		Masks when Next0 completes.	

Setting Example 2
DCTR = 0000 0001h (DMA setting)

N0SA = 1111 0000h (Transfer source address)
N0DA = 2000 0000h (Transfer destination address)
N0TB = 0000 0200h (Number of transfer bytes)

N1SA = 0400 0000h (Transfer source address)
N1DA = 1000 0000h (Transfer destination address)
N1TB = 0000 0800h (Number of transfer bytes)

CHCFG = 6176 2007h (Configuration)
CHITVL = 0000 0000h (Interval)

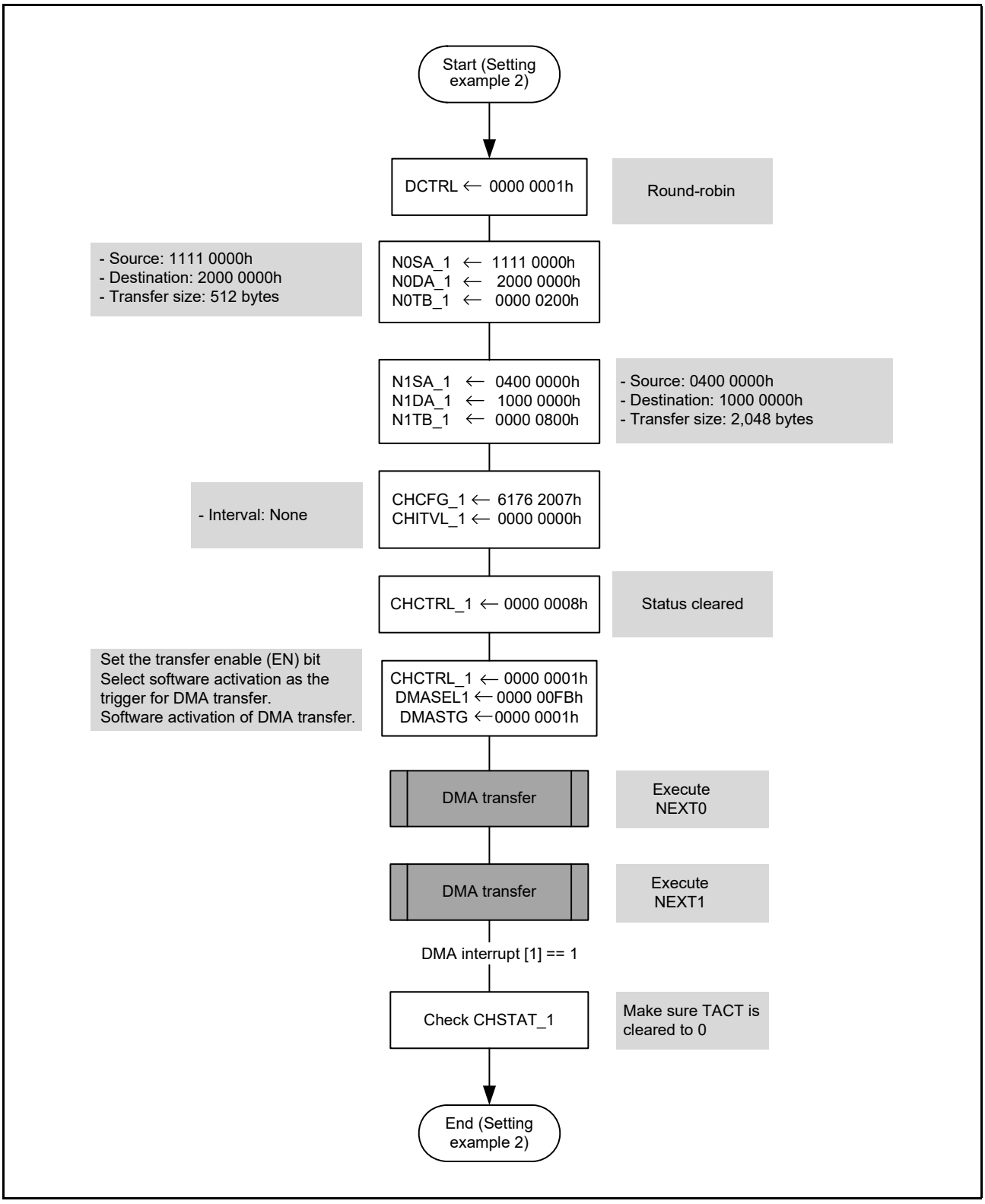


Figure 14.22 Setting Example 2

Note: DMA interrupt [1] indicates the interrupt source assigned to serve as the trigger for DMA transfer on channel 1.

14.5.3 Setting Example 3 (Link Mode)

This subsection provides a setting example when a DMA transfer is performed in link mode.

Table 14.25 DMA Transfer Setting Example 3

Item	Description
Channel to use	DMAC0 channel 0
Priority control	Round-robin
DMA mode	Link mode
Transfer mode	Block transfer mode
Descriptor start address	0080 0000h

Table 14.26 DMA Transfer Setting Example 3 (Descriptor 1)

Item		Description	
Descriptor start address		0080 0000h	
Next descriptor start address		0080 1000h	
Transfer mode		Block transfer mode	
Transfer source/transfer destination		Transfer source	Transfer destination
	Start address	1111 0000h	2000 0000h
	Address direction	Increment	Increment
	Data size	32 bits	32 bits
Number of DMA transfer bytes		2,048 bytes	
DMA transfer request		Software request	
DMA transfer completion interrupt output mask		Masks when a DMA transfer on descriptor 1 completes.	
Descriptor format		1 (8 words)	
Descriptor header			
	Write back of the LV bit	Enable (WBD = 0)	
	Next link destination	Available (LE = 0)	
	Descriptor enabled	Enabled (LV = 1)	

Table 14.27 DMA Transfer Setting Example 3 (Descriptor 2)

Item		Item	
Descriptor start address		0080 1000h	
Next descriptor start address		0080 2000h	
Transfer mode		Block transfer mode	
Transfer source/transfer destination		Transfer source	Transfer destination
	Start address	0400 0000h	2000 0000h
	Address direction	Increment	Increment
	Data size	256 bits	256 bits
Number of DMA transfer bytes		1,024 bytes	
DMA transfer request		Software request	
DMA transfer completion interrupt output mask		Masks when a DMA transfer on descriptor 2 completes.	
Descriptor format		1 (8 words)	
Descriptor header			
	Write back of the LV bit	Enable (WBD = 0)	
	Next link destination	Available (LE = 0)	
	Descriptor enabled	Enabled (LV = 1)	

Table 14.28 DMA Transfer Setting Example 3 (Descriptor 3)

Item		Item	
Descriptor start address		0080 2000h	
Next descriptor start address		—	
Transfer mode		Block transfer mode	
Transfer source/transfer destination		Transfer source	Transfer destination
	Start address	2000 0000h	0800 2000h
	Address direction	Increment	Increment
	Data size	512 bits	512 bits
Number of DMA transfer bytes		4,096 bytes	
DMA transfer request		Software request	
DMA transfer completion interrupt output mask		Do not mask.	
Descriptor format		1 (8 words)	
Descriptor header			
	Write back of the LV bit	Enable (WBD = 0)	
	Next link destination	None (LE = 1)	
	Descriptor enabled	Enabled (LV = 1)	

Setting Example 3
DCTRL = 0000 0001h (DMA setting)
NXLA = 0080 0000h (Descriptor start address)
CHCFG = 8000 0000h (Configuration)

Table 14.29 Descriptor Settings

Item	Descriptor 1	Descriptor 2	Descriptor 3
Header	1100 0000h	1100 0000h	1300 0000h
SA (Source Address)	1111 0000h	0400 0000h	2000 0000h
DA (Destination Address)	2000 0000h	2000 0000h	0800 2000h
TB (Transaction Bytes)	0000 0800h	0000 0400h	0000 1000h
CFG (Configuration)	8142 2220h	8145 5220h	8046 6220h
ITVL (Interval)	0000 0000h	0000 0000h	0000 0000h
EXT (Extension)	0000 0000h	0000 0000h	0000 0000h
NXLA (Next Link Address)	0080 1000h	0080 2000h	0000 0000h

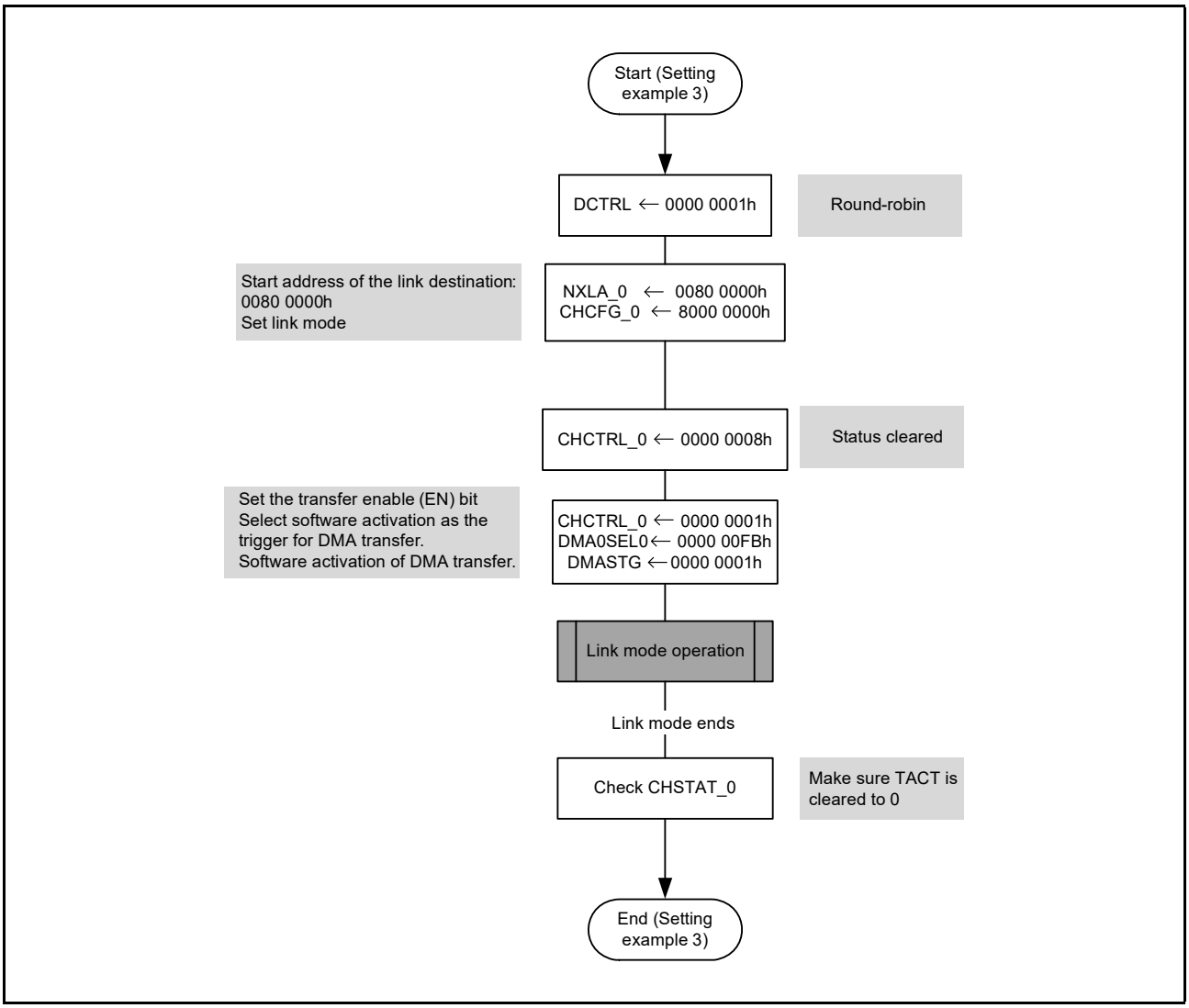


Figure 14.23 Setting Example 3

14.5.4 Next Register Continuous Execution Settings

The following figure shows a flow chart when using two Next register sets to continue DMA transfers in register mode. While performing a DMA transfer of a Next register, set the other Next register, and perform the DMA transfers in series.

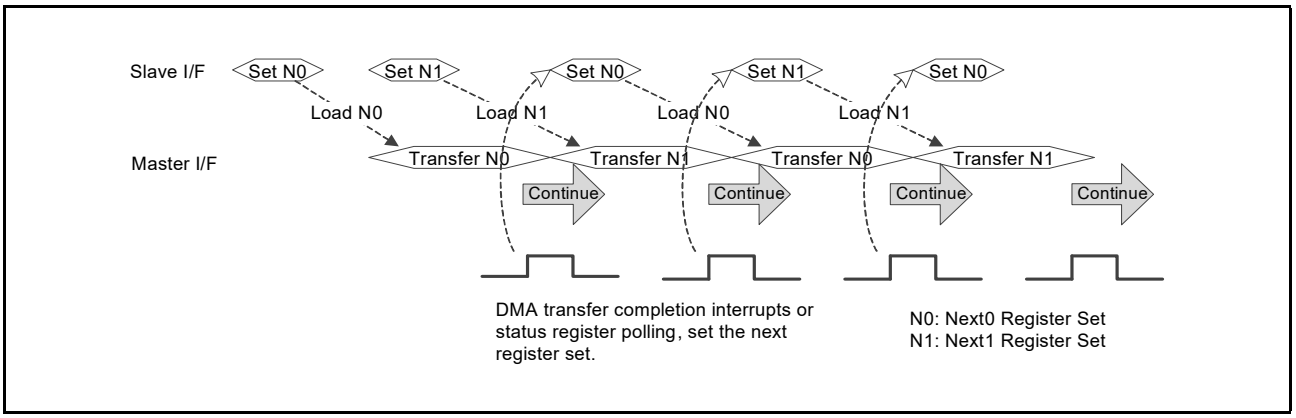


Figure 14.24 Next Register Continuous Execution Setting Image

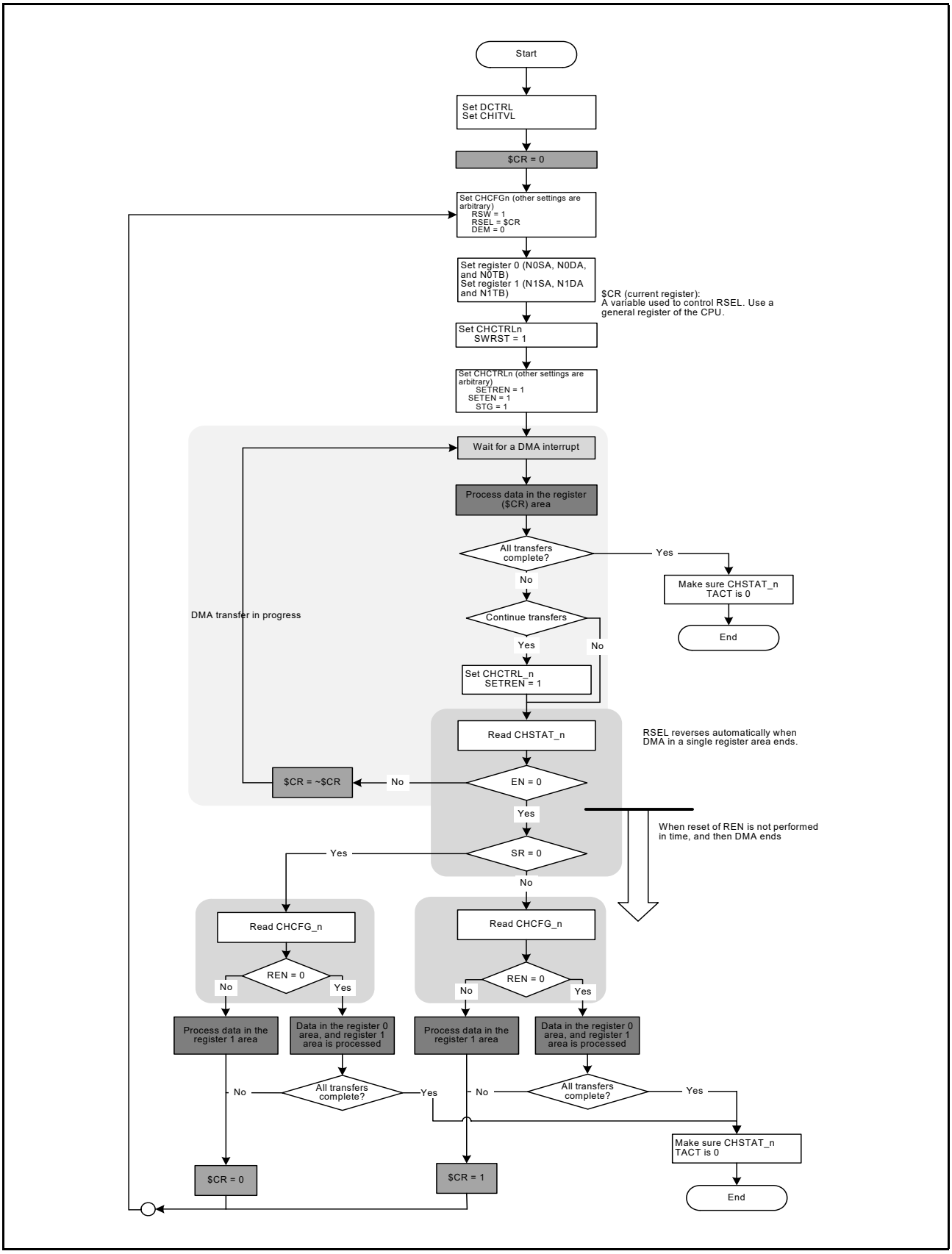


Figure 14.25 Setting Example of Next Register Continuous Execution

- Supplementary information:
  Save the register sets 0 (N0SA_n, N0DA_n, and N0TB_n registers) and 1 (N1SA_n, N1DA_n, and N1TB_n registers) in a general register (call this register value $CR for convenience).
  Every time a DMA transfer of a register set completes (a DMA transfer completion interrupt is generated), the REN bit in the CHCFG_n register is cleared to 0 automatically. To perform transfers in series, write 1 to the SETREN bit in the CHCTR_n register. By doing so, the REN bit in the CHCFG_n register is also set.

  In this mode, two Next registers are executed in series. However, if the SETREN bit is not set before a DMA transfer completes (before the next DMA transfer completion interrupt is generated), the continuous execution stops. In this case, by reading the SR and EN bits in the CHSTAT_n register, and the REN bit in the CHCFG_n register, you can check how far the transfer is performed. To resume the transfer, perform the procedure described in the above flow chart.

14.6 Usage Notes

This subsection provides notes on this module.

- When a transfer of which source and destination are in the same, or partially shared area is performed, consistency of data cannot be guaranteed. Therefore, do not perform a transfer of which source and destination address areas overlap.

- When DAD = 1 (fixed transfer source address), no skip transfer can be performed on the transfer destination. If a transfer is performed with this setting, operation cannot be guaranteed. Do not perform this type of transfers.

- When SAD = 1 (fixed transfer source address), no skip transfer can be performed on the transfer source. If a transfer is performed with this setting, operation cannot be guaranteed. Do not perform this type of transfers.

- Since access to the region from A00E 0000h to A010 0000h within the peripheral I/O register region is not supported, locations or ranges within this region should never be set as sources or destinations for transfer.

15. Event Link Controller (ELC)

15.1 Overview

The event link controller (ELC) connects (links) events generated by various peripheral modules to other modules. Event link allows direct cooperation among modules without CPU intervention.

Table 15.1 lists the specifications of the ELC, and Figure 15.1 shows a block diagram of the ELC.

Table 15.1 ELC Specifications

Item	Description
Event link function	• Event signals can be directly connected to modules. • The operation of timer modules can be selected when an event is input to the timer module. • Event link operation is possible for port E. Single port*1: An event link can be set for a single bit specified in a port. Port group*1: An event link can be set for a group of single bits specified within eight I/O ports.
Low-power consumption function	Module-stop state can be set.

Note 1. The single port and port group specified as the input generate an event according to the change in the connected signal value.

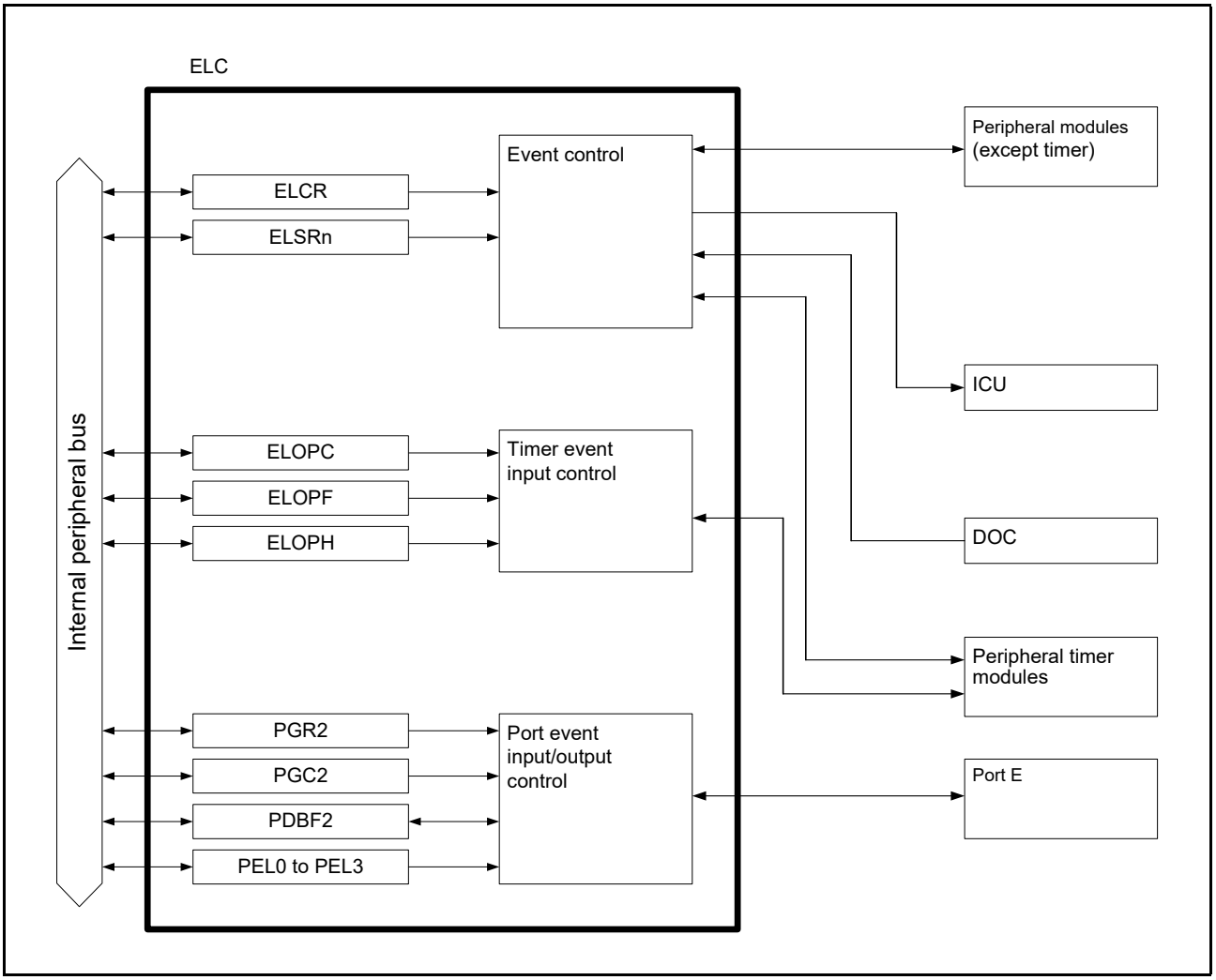


Figure 15.1 ELC Block Diagram (n = 7, 15, 18 to 27, 33, 35 to 38, 45)

15.2 Register Descriptions

15.2.1 Event Link Control Register (ELCR)

The ELCR register controls operation of the ELC.

Address(es): A008 0B00h

	b7	b6	b5	b4	b3	b2	b1	b0
	ELCON	—	—	—	—	—	—	—
Value after reset:	0	1	1	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b6 to b0	—	Reserved	These bits are read as 1. The write value should be 1.	R/W
b7	ELCON	All Event Link Enable	0: ELC function is disabled. 1: ELC function is enabled	R/W

15.2.2 Event Link Setting Register n (ELSRn) (n = 7, 15, 18 to 27, 33, 35 to 38, 45)

The ELSRn register specifies an event signal to be linked to for each peripheral module. Table 15.2 shows the correspondence between the ELSRn register and the peripheral modules. Table 15.3 shows the correspondence between the event signal names set in the ELSRn register and the signal numbers.

Address(es): ELSR7 A008 0B08h, ELSR15 A008 0B10h, ELSR18 A008 0B13h, ELSR19 A008 0B14h, ELSR20 A008 0B15h, ELSR21 A008 0B16h, ELSR22 A008 0B17h, ELSR23 A008 0B18h, ELSR24 A008 0B19h, ELSR25 A008 0B1Ah, ELSR26 A008 0B1Bh, ELSR27 A008 0B1Ch, ELSR33 A008 0B31h, ELSR35 A008 0B33h, ELSR36 A008 0B34h, ELSR37 A008 0B35h, ELSR38 A008 0B36h, ELSR45 A008 0B3Dh

	b7	b6	b5	b4	b3	b2	b1	b0
	ELS[7:0]							
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	ELS[7:0]	Event Link Select	b7 b0 00000000: Event output to the corresponding peripheral module is disabled. 00011111 to 10111101: Set the number for the event signal to be linked. Settings other than above are prohibited.	R/W

Table 15.2 Correspondence between the ELSRn Register and the Peripheral Functions

Register Name	Peripheral Function (Module)
ELSR7	CMT1
ELSR15	S12AD0
ELSR18	Interrupt 1 (ELCIRQ1)
ELSR19	Interrupt 2 (ELCIRQ2)
ELSR20	Output port group 1
ELSR21	Output port group 2
ELSR22	Input port group 1
ELSR23	Input port group 2
ELSR24	Single port 0
ELSR25	Single port 1
ELSR26	Single-port 2
ELSR27	Single-port 3
ELSR33	CMTW0
ELSR35	TPU0
ELSR36	TPU1
ELSR37	TPU2
ELSR38	TPU3
ELSR45	S12AD1

Table 15.3 Correspondence between Event Signal Names Set in ELSRn.ELS[7:0] Bits and Signal Numbers

ELS[7:0] Bit Value	Peripheral modules	Name of Event Signal Set in ELSRn
1Fh	Compare match timer	CMT1 compare match 1
4Eh	I²C bus interface	RIIC0 communication error or event generation
4Fh		RIIC0 receive data full
50h		RIIC0 transmit data empty
51h		RIIC0 transmit end
52h	Serial peripheral interface	RSPI0 error (mode fault, overrun, or parity error)
53h		RSPI0 idle
54h		RSPI0 receive data full
55h		RSPI0 transmit data empty
56h		RSPI0 transmit end
58h	12-bit A/D converter	S12AD0 A/D conversion end
63h	I/O ports	Input edge detection of input port group 1
64h		Input edge detection of input port group 2
65h		Input edge detection of single input port 0
66h		Input edge detection of single input port 1
67h		Input edge detection of single input port 2
68h		Input edge detection of single input port 3
69h	Event link controller	Software event
6Ah	Data operation circuit	DOC data operation condition met signal
6Ch	12-bit A/D converter	S12AD1 A/D conversion end
7Eh	Compare match timer W	CMTW channel 0 compare match
ACh	16-bit timer pulse unit	TPU0 compare match A
ADh		TPU0 compare match B
AEh		TPU0 compare match C
AFh		TPU0 compare match D
B0h		TPU0 overflow
B1h		TPU1 compare match A
B2h		TPU1 compare match B
B3h		TPU1 overflow
B4h		TPU1 underflow
B5h		TPU2 compare match A
B6h		TPU2 compare match B
B7h		TPU2 overflow
B8h		TPU2 underflow
B9h		TPU3 compare match A
BAh		TPU3 compare match B
BBh		TPU3 compare match C
BCh		TPU3 compare match D
BDh		TPU3 overflow
Settings other than above are prohibited.		

15.2.3 Event Link Option Setting Register C (ELOPC)

The ELOPC register determines the operation of CMT1 when an event is input. The event setting should be disabled when the ELC function is not to be used.

Address(es): A008 0B21h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	CMT1MD[1:0]		—	—
Value after reset:	1	1	1	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b1, b0	—	Reserved	These bits are read as 1. The write value should be 1.	R/W
b3, b2	CMT1MD[1:0]	CMT1 Operation Select	b3 b2 0 0: Counting is started. 0 1: The counter is cleared 1 0: Event counter 1 1: Event is disabled.	R/W
b7 to b4	—	Reserved	These bits are read as 1. The write value should be 1.	R/W

15.2.4 Port Group Setting Register n (PGRn) (n = 2)

The PGRn register specifies a group for I/O port bits. This register specifies each port bit in the same eight I/O ports as the member of a group. One to eight port bits can be specified as the members of the same group as required. Table 15.4 shows the PGRn register and corresponding ports.

Address(es): PGR2 A008 0B24h

	b7	b6	b5	b4	b3	b2	b1	b0
	PGRn7	PGRn6	PGRn5	PGRn4	PGRn3	PGRn2	PGRn1	PGRn0
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	PGRn0	Port Group Setting n 0	0: The port bit is not specified as a member of the same group. 1: The port bit is specified as a member of the same group.	R/W
b1	PGRn1	Port Group Setting n 1		R/W
b2	PGRn2	Port Group Setting n 2		R/W
b3	PGRn3	Port Group Setting n 3		R/W
b4	PGRn4	Port Group Setting n 4		R/W
b5	PGRn5	Port Group Setting n 5		R/W
b6	PGRn6	Port Group Setting n 6		R/W
b7	PGRn7	Port Group Setting n 7		R/W

Table 15.4 Registers Related to Port Groups and Corresponding Port Numbers

Port Number	Port Group Setting Register (PGR)	Port Group Control Register (PGC)	Port Buffer Register (PDBF)
Port E	PGR2 register	PGC2 register	PDBF2 register

15.2.5 Port Group Control Register n (PGCn) (n = 2)

For the output port group, the PGCn register specifies the form of outputting the signal externally via the port when the event signal is input. For the input port group, this register enables/disables overwriting of the PDBF register and specifies the conditions of event generation (edge of the externally input signal).
Refer to Table 15.4 for the PGRn register and corresponding ports.

Address(es): PGC2 A008 0B26h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	PGCOn[2:0]			—	PGCOVEn	PGCIn[1:0]	
Value after reset:	1	0	0	0	1	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	PGCIn [1:0]	Event Output Edge Select	b1 b0 0 0: Event is generated upon detection of the rising edge of the external input signal. 0 1: Event is generated upon detection of the falling edge of the external input signal. 1 X: Event is generated upon detection of both the rising and falling edges of the external input signal.	R/W
b2	PGCOVEn	PDBF Overwrite	0: Overwriting PDBFn register is disabled. 1: Overwriting PDBFn register is enabled.	R/W
b3	—	Reserved	This bit is read as 1. The write value should be 1.	R/W
b6 to b4	PGCOn [2:0]	Port Group Operation Select	b6 b4 0 0 0: 0 is output when the event is input. 0 0 1: 1 is output when the event is input. 0 1 0: The toggled (inverted) value is output when the event is input. 0 1 1: The buffer value is output when the event is input. 1 X X: The bit value is rotated out in the group (from MSB to LSB) when the event is input.	R/W
b7	—	Reserved	This bit is read as 1. The write value should be 1.	R/W

X: Don't care

15.2.6 Port Buffer Register n (PDBFn) (n = 2)

The PDBFn register is an 8-bit readable/writable register used in combination with the PGRn register. Refer to section 15.3.5, I/O Port Operation upon Event Input and Event Generation for PDBFn register operations. Refer to Table 15.4 for the PDBFn register and corresponding ports.

Address(es): PDBF2 A008 0B28h

	b7	b6	b5	b4	b3	b2	b1	b0
	PDBFn 7	PDBFn 6	PDBFn 5	PDBFn 4	PDBFn 3	PDBFn 2	PDBFn 1	PDBFn 0
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	PDBFn0	Port Buffer n0	These bits handle the following operations in response to an input to or output from the port. • As an output port The value written to the PDBFn register is transferred to the PODR register. • As an input port The signal values on the external pins are transferred to the PDBFn register. Write access to the bit specified as a member of the input port group is invalid. For details, refer to section 15.3, Operation.	R/W
b1	PDBFn1	Port Buffer n1		R/W
b2	PDBFn2	Port Buffer n2		R/W
b3	PDBFn3	Port Buffer n3		R/W
b4	PDBFn4	Port Buffer n4		R/W
b5	PDBFn5	Port Buffer n5		R/W
b6	PDBFn6	Port Buffer n6		R/W
b7	PDBFn7	Port Buffer n7		R/W

15.2.7 Event Link Port Setting Register n (PELn) (n = 0 to 3)

The PELn register specifies the single port to which an event is to be linked, the port operation upon the event signal input, and the conditions of event generation. In this LSI, a total of 4 bits in port E can be specified as single ports.

Address(es): PEL0 A008 0B29h, PEL1 A008 0B2Ah, PEL2 A008 0B2Bh, PEL3 A008 0B2Ch

	b7	b6	b5	b4	b3	b2	b1	b0
	—	PSMn[1:0]		PSPn[1:0]		PSBn[2:0]		
Value after reset:	1	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	PSBn[2:0]	Bit Number Specification	A bit number in eight I/O ports is specified.	R/W
b4, b3	PSPn[1:0]	Port Number Specification	b4 b3 0 0: Setting prohibited 0 1: Setting prohibited 1 0: Port E (corresponding to PGR2) 1 1 1: Setting prohibited	R/W
b6, b5	PSMn[1:0]	Event Link Specification	• For the output port, data to be output from the port is specified. b6 b5 0 0: 0 is output when the event is input. 0 1: 1 is output when the event is input. 1 X: The toggled (inverted) value is output when the event is input. • For the input port, the edge on which the event is to be output is specified. b6 b5 0 0: Event is output upon detection of the rising edge. 0 1: Event is output upon detection of the falling edge. 1 X: Event is output upon detection of both the rising and falling edges.	R/W
b7	—	Reserved	This bit is read as 1. The write value should be 1.	R/W

X: Don’t care

15.2.8 Event Link Software Event Generation Register (ELSEGR)

The ELSEGR register controls event generation by software.

Address(es): A008 0B2Dh

	b7	b6	b5	b4	b3	b2	b1	b0
	WI	WE	—	—	—	—	—	SEG
Value after reset:	1	0	1	1	1	1	1	0

Bit	Symbol	Bit Name	Description	R/W
b0	SEG	Software Event Generation	0: Normal operation 1: Software event is generated.	W
b5 to b1	—	Reserved	These bits are read as 1. The write value should be 1.	R/W
b6	WE	SEG Bit Write Enable	0: Write to SEG bit is disabled. 1: Write to SEG bit is enabled.	R/W
b7	WI	ELSEGR Register Write Disable	0: Write to ELSEGR register is enabled. 1: Write to ELSEGR register is disabled.	W

SEG Bit (Software Event Generation)

When 1 is written to this bit while the WE bit is 1, a software event is generated.
This bit is read as 0. Even if 1 is written to this bit, the data will not be stored.

WE Bit (SEG Bit Write Enable)

The SEG bit can be written to only when the WE bit is 1.
[Setting condition]
If 1 is written to this bit while the WI bit is 0, this bit becomes 1.
[Clearing condition]
If 0 is written to this bit while the WI bit is 0, this bit becomes 0.

WI Bit (ELSEGR Register Write Disable)

The ELSEGR register can be written to only when the value to be written to the WI bit is 0.
This bit is read as 1.

Note 1. The WE bit can only be updated by setting the WI bit and the WE bit at the same time.
Similarly, to update the SEG bit, the WI bit must be set at the same time; set WE = 1 by setting the WI bit and WE bit at the same time, and then set the WI bit and SEG bit. If WE = 1 when SEG is set, the WE bit retains the value 1.
However, setting the three bits at the same time while the current values are WI = 1, WE = 0, and SEG = 0 will not lead to the output of a software trigger. Since WE will only have the value 1 following the first time the three bits are set, a software trigger will only be generated by setting the three bits at the same time again.

15.2.9 Event Link Option Setting Register F (ELOPF)

The ELOPF register determines the operation of TPU0 to TPU3 when an event is input. The event setting should be disabled when the ELC function is not to be used.

Address(es): A008 0B3Fh

	b7	b6	b5	b4	b3	b2	b1	b0
	TPU3MD[1:0]		TPU2MD[1:0]		TPU1MD[1:0]		TPU0MD[1:0]	
Value after reset:	1	1	1	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b1, b0	TPU0MD[1:0]	TPU0 Operation Select	b1 b0 0 0: Counting is started. 0 1: Counting is cleared. 1 0: Input capture*1 1 1: Event is disabled.	R/W
b3, b2	TPU1MD[1:0]	TPU1 Operation Select	b3 b2 0 0: Counting is started. 0 1: Counting is cleared. 1 0: Input capture*2 1 1: Event is disabled.	R/W
b5, b4	TPU2MD[1:0]	TPU2 Operation Select	b5 b4 0 0: Counting is started. 0 1: Counting is cleared. 1 0: Input capture*3 1 1: Event is disabled.	R/W
b7, b6	TPU3MD[1:0]	TPU3 Operation Select	b7 b6 0 0: Counting is started. 0 1: Counting is cleared. 1 0: Input capture*4 1 1: Event is disabled.	R/W

Note 1. The TPU0.TCNT value is captured by TPU0.TGRA.
Note 2. The TPU1.TCNT value is captured by TPU1.TGRA.
Note 3. The TPU2.TCNT value is captured by TPU2.TGRA.
Note 4. The TPU3.TCNT value is captured by TPU3.TGRA

15.2.10 Event Link Option Setting Register H (ELOPH)

The ELOPH register determines the operation of channel 0 in CMTW when an event is input. The event setting should be disabled when the ELC function is not to be used.

Address(es): A008 0B41h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	CMTW0MD[1:0]	
Value after reset:	1	1	1	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b1, b0	CMTW0MD[1:0]	CMTW0 Operation Select	b1 b0 0 0: Counting is started. 0 1: Counting is cleared. 1 0: Event counter 1 1: Event is disabled.	R/W
b7 to b2	—	Reserved	These bits are read as 1. The write value should be 1.	R/W

15.3 Operation

15.3.1 Relation between Interrupt Handling and Event Linking

The peripheral modules incorporated in this LSI are provided with the interrupt request status flags and the bits to enable/disable these interrupt requests. When an interrupt request is generated in a module, the corresponding interrupt request status flag is set. If the corresponding interrupt request is enabled then, the interrupt requested is issued to the CPU.
In contrast, the event link controller (ELC) uses the interrupt requests generated by various peripheral modules as event signals to connect (link) them to different modules, allowing direct cooperation between the modules without CPU intervention. Event signals can be output regardless of the setting of the corresponding interrupt request enable bit.
Figure 15.2 shows the relation between the interrupt handling and ELC.

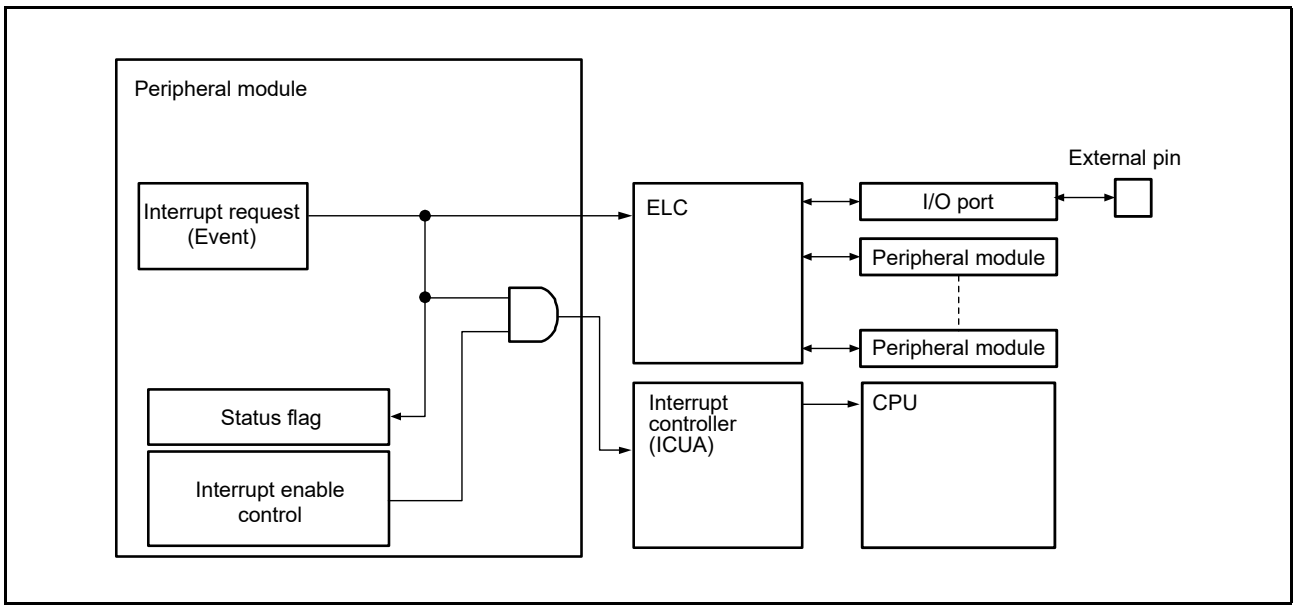


Figure 15.2 Relation between Interrupt Handling and ELC

15.3.2 Event Linkage

Set an event to the ELSRn register of the module to which the event is to be linked. On occurrence of the specified event, the link destination module performs the operation set in the ELOPm register (m = C, F, H). Only one type of event can be connected with one module. For detailed procedure for setting desired operation, see section 15.3.6, Example of Procedure for Linking Events. Table 15.5 lists the operations of modules when an event is input.

Table 15.5 Operations of Modules When Event is Input

Module	Operations When Event is Input	
CMT CMTW TPU	Each timer operates differently depending on the setting of the ELOPm register (m = C, F, H) as below. • Starts counting when an event signal is input (CMT, CMTW, TPU). • Clears counting when an event signal is input (CMT, CMTW, TPU). Restarting count is allowed when the start bit of the timer is 1. • Counts the input events (CMT, CMTW). • Performs input-capture operation when an event signal is input (TPU).	
A/D converter	Starts A/D conversion when an event signal is input.	
I/O ports (output)	Port group	• Changes the PODR value to the value specified in the PGCn register. • Transfers the PDBFn value to the PODR register. • Rotates out the bit value.
	Single port	Changes the PODR value to the value specified in the PELn register.
I/O ports (input)	Port group	Transfers the signal value of the external pin to the PDBFn register.
	Single port	Event connection is not possible.
Interrupt controller	Issues an interrupt request to the CPU and starts DMAC data transfer.	

15.3.3 Operation of Peripheral Timer Modules When Event is Input

The operation when an event signal is input is set by the ELOPm register (m = C, F, H).

(1) Counting Start Operation

When an event is input, the timer starts counting, which sets the count start bit*1 in each timer control register to 1. An event that is input while the count start bit is 1 is invalid.

(2) Counting Clear Operation

When an event is input, the timer counter is initialized. If the count start bit*1 in each timer control register is set to 1, the counting continues, thus, counting restarts.

(3) Event Counter Operation

Event input is selected as the timer clock source and the timer counts events.

(4) Input Capture Operation

When an event is input, the timer performs input-capture operation.

Note 1. Refer to the register descriptions on starting the timer in the relevant peripheral timer module section.

15.3.4 Operation of A/D Converter When Event is Input

The A/D converter starts A/D conversion when the ADCSR.ADST bit*1 is set to 1.

Note 1. Refer to the register descriptions on the A/D converter section.

15.3.5 I/O Port Operation upon Event Input and Event Generation

The I/O port operation to be performed upon event input to the port can be set and the operation causing the port to generate an event can be set.

(1) Single ports and Port Groups

There are two event link modes: event link to single ports and event link to port groups. In the former mode, events can be connected to eight I/O ports. In the latter mode, events can be connected to port groups consisting of any two or more bits in the same eight I/O ports.

A single port can be set by specifying any bit in the I/O port*1 to which an event can be connected using the PELn register. A port group can be set by specifying any one or more bits in the I/O port*1 to which an event can be connected using the PGRn register. One input port group and one output port group can be set in the same I/O port.

If the I/O port bit is specified as both a single port and a member of a port group, both functions are enabled when the relevant port is input, whereas only the port group function is enabled when the relevant port is output.

Set the PDR register to select the direction of the I/O ports.

Note 1. Port E

(2) Single Input Port Operation upon Event Generation

A single input port which is specified by the PDR register generates an event when the signal value of the external pin connected to the relevant port changes. The event generation condition is specified using PELn register. An example of event linking operation by a single input port is shown as [1] in Figure 15.3.

(3) Single Output Port Operation upon Event Input

When an event is input to a single output port which is specified by the PDR register, the signal of the external pin connected to the relevant port changes according to the settings of PELn register. This changes the signal value of the external pin connected to the relevant port. An example of event linking operation by a single output port is shown as [2] in Figure 15.3.

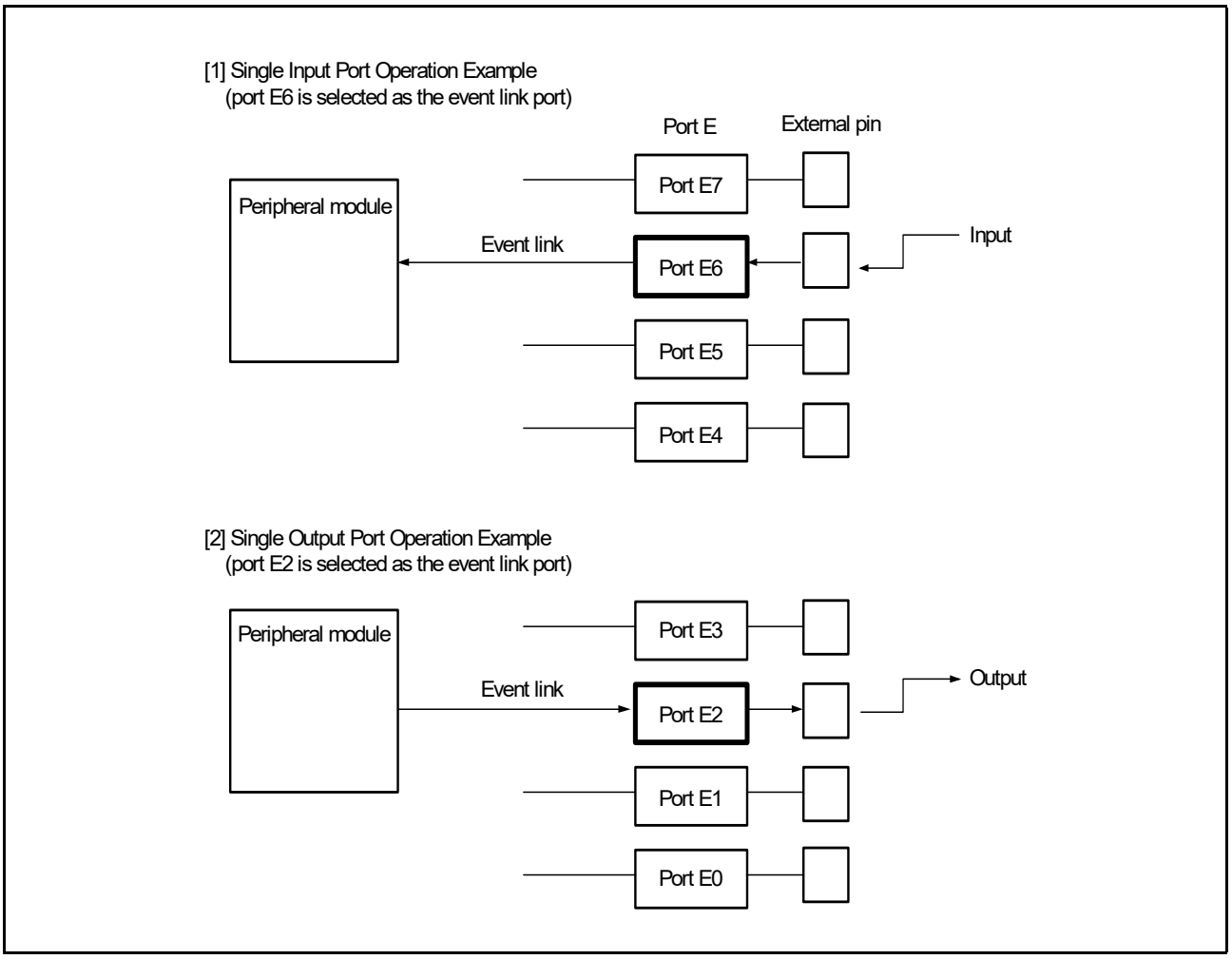


Figure 15.3 Event Linkage Related to Single Ports (Port E)

(4) Input Port Group Operation upon Event Generation

An input port group which is specified by the PDR register generates an event when the signal value of any one of the external pins connected to the relevant port group changes. The event generation condition is specified using the PCIn bit of the PGCn register.

(5) Input Port Group Operation upon Event Input

When an event is input to an input port group, the signal value of the external pin of the bit specified as a member of the input port group is transferred to the PDBFn register. If another event is input to the input port group in this state, operations are performed depending on the PGCn.PGCOVE bit setting as described below. Figure 15.4 shows the input port group operation upon an event input.

- PGCn.PGCOVEn = 0 (overwriting is disabled)
  If the value that was transferred to the PDBFn register upon the previous event input has already been read by the CPU, the signal value of the external pin is transferred to the PDBFn register. If not read, the signal value of the external pin is not transferred to the PDBFn register and the input event is invalid.
- PGCn.PGCOVEn = 1 (overwriting is enabled)
  When another event is input to an input port group, the signal value of the external pin is transferred to the PDBFn register.

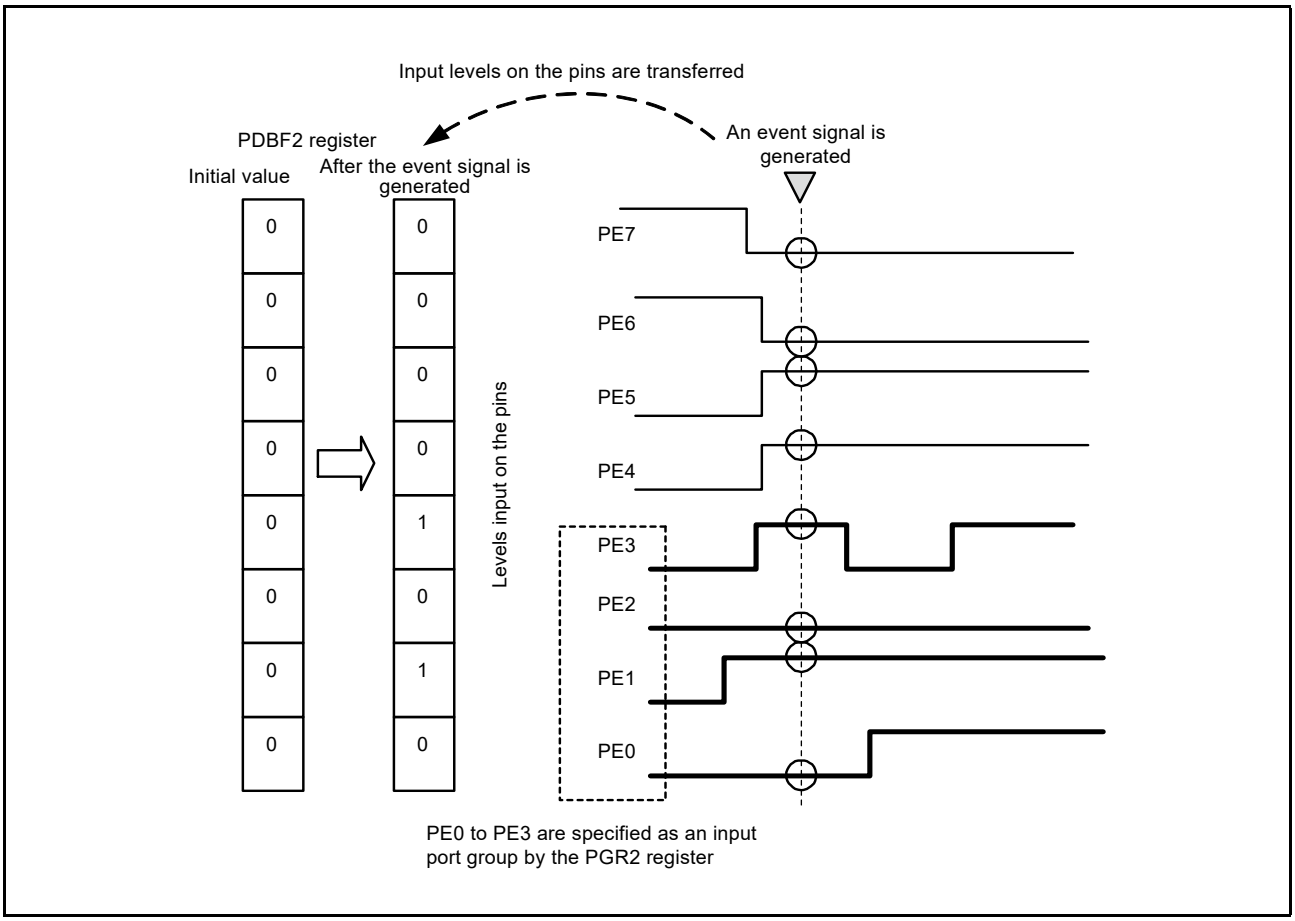


Figure 15.4 Input Port Group Operation upon Event Input (Port E)

(6) Output Port Group Operation upon Event Input

When an event is input to an output port group, the following operations are performed depending on the PGCn.PGCOn bit setting as described below.

- If an event is input to an output port group while the PGCn.PGCOn bit being 000b, 001b, or 010b, the PODR value is changed to the value which was specified in the PGCn register.
- If an event is input to an output port group while the PGCn.PGCOn bit being 011b, the PDBFn value is transferred to the PODR register of the port which was specified in the PGRn register. Example of operation of the output port group upon an event input (when PGCn.PGCOn = 011b) is shown in Figure 15.5.
- If an event is input to the output port group while the PGCn.PGCOn bit being 1XXb, the PDBFn value is transferred to the PODR register of the port which was specified in the PGRn register, and then the PODR value is rotated bit by bit from MSB to LSB. The initial value to be output to the port group should be provided in the PDBFn register in advance. Examples of bit-rotating operation of output port groups upon an event input (when PGCn.PGCOn = 1XXb) is shown in Figure 15.6.

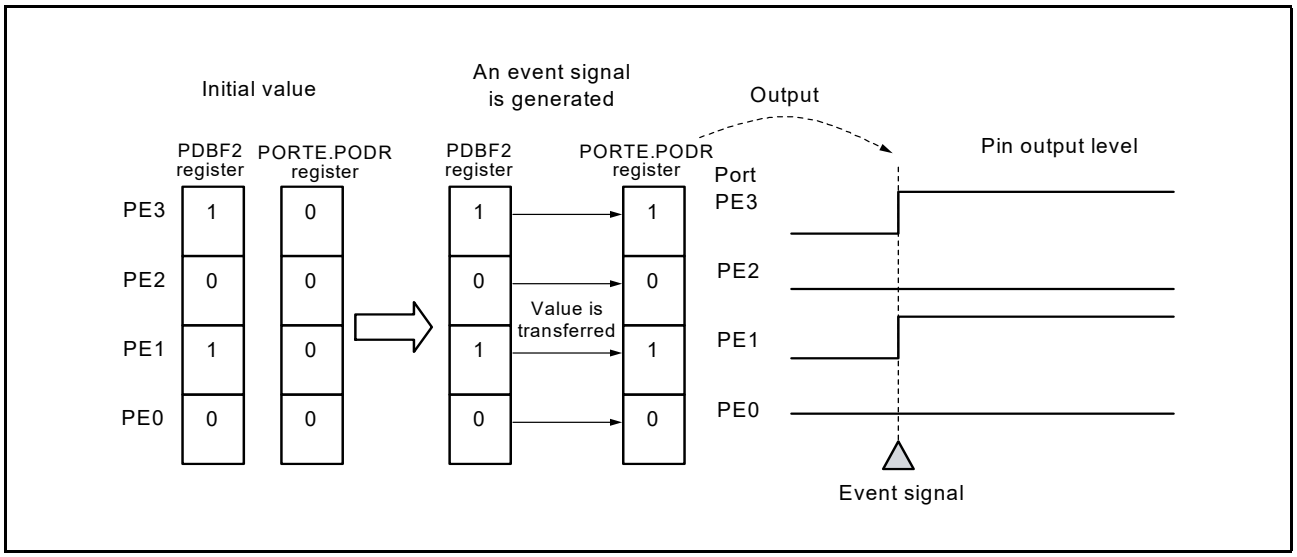


Figure 15.5 Event Linkage Operation of Output Port Group (Port E)

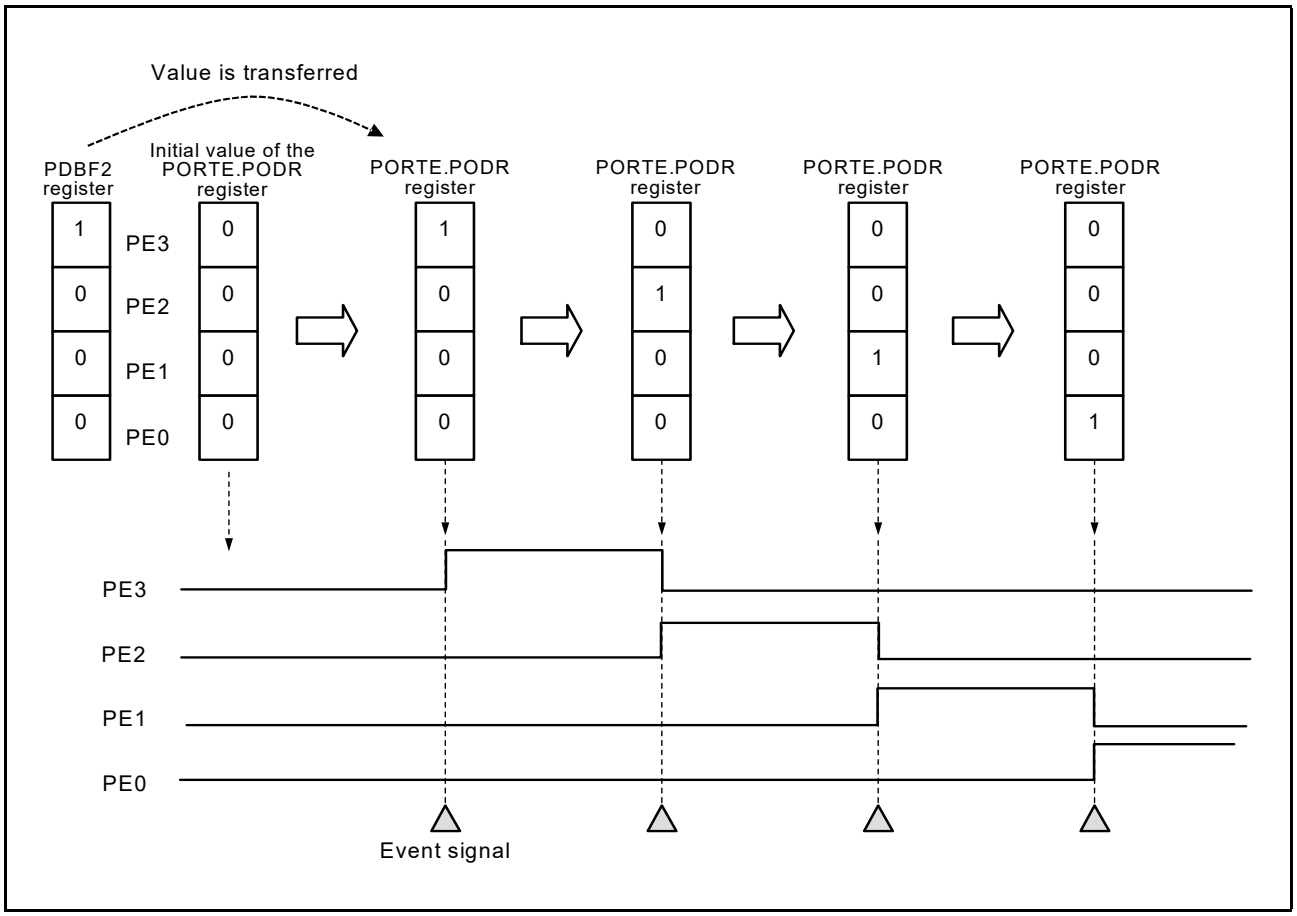


Figure 15.6 Bit-Rotating Operation of Output Port Groups (Port E)

(7) Restrictions on Writing to PODR and PDBFn Registers by a CPU

For event linkage through the I/O ports, following restrictions apply when a CPU writes the PODR and PDBFn registers.

- If port bits are specified as members of the input port group, write access to the relevant bits in the PDBFn register is invalid.
- If port bits are specified as members of the output port group, write access to the relevant bits in the PODR register is invalid.
- If a port bit is specified as a single output port and the event linkage is set (by the ELSRn register) for the port, write access to the relevant bit in the PODR register is invalid.

15.3.6 Example of Procedure for Linking Events

The following describes the procedure for linking events.

1. Set the operation of the module to which an event is to be linked.
2. If events are linked to ports, set the registers corresponding to the ports as below.

- I/O port setting
  PODR: Set the initial values of the output ports.
  PDR: Set the I/O direction of the ports.
- ELC setting
  PGRn: If ports are used as a port group, set the ports (in bit units) to be grouped.
  PGCn: Set the operation of the port group.
  PELn: If ports are used as single ports, set the ports, the operation of the ports when an event is input, and the condition when an event is generated.

Note: Setting the PDBFn register
(1) Output port groups
Set the PGCn register before setting the PDBFn register.
The value of the PGCn register can be changed if this precedes an event trigger which causes the value of the PODR register to change.
(2) Input port groups
Setting the PDBFn is not required. However, since the value after a reset is 00h, if the PDBFn register is used to confirm changes from H to L due to an event input, set the PDBFn bit for the pin for which you wish to confirm this to 1.

3. To the ELSRn register corresponding to the module to which an event signal is to be linked, set the number of the event signal.
4. If events are to be linked to timer modules, set the ELOPm register (m = C, F, H) corresponding to the timers as required.
5. Set the ELCR.ELCON bit to 1, which enables linkage of all the events.
6. Set the operation of the module from which an event is output, and activate the module. This allows the event output from the module to start the module to which an event is linked as preset.
7. To stop event linkage of independent modules, set 0000 0000b to the ELSRn.ELS[7:0] bits corresponding to the modules. To stop linkage of all the events, set the ELCR.ELCON bit to 0.

15.4 Usage Notes

15.4.1 Setting ELSR18 and ELSR19 Registers

For event linkage to an interrupt controller, specify the event signals to be set in the ELSR18 and ELSR19 registers from 63h to BDh. Setting any other values is prohibited.

15.4.2 Setting Bit-Rotating Operation of Output Port Groups

When the values of the PDBFn register are changed in the bit-rotating operation mode of the output port group, set the ELSRn register again after changing the PDBFn register value.

15.4.3 Setting Clocks

To link events, it is necessary for the ELC and the related modules to be enabled. The modules cannot operate if the related modules are in the module-stop state or if the low-power consumption mode causes the modules to stop (all-module stop mode).

15.4.4 Module Stop Function Setting

ELC operation can be disabled or enabled using module stop control register C (MSTPCRC). The initial setting allows the ELC to be stopped. Register access is enabled by canceling the module-stop state. For details, see section 9, Low-Power Consumption Function.

16. I/O Ports

16.1 Overview

The pins of an I/O port function as general I/O port pins, I/O pins for peripheral modules, or interrupt input pins.
Each pin is also configurable as an I/O pin of a peripheral module or an input pin for an interrupt. All pins are set to non-use immediately after a reset (Hi-Z input protection), and pin functions can be switched by register settings. The setting of each pin is specified by the registers for the corresponding I/O port and peripheral modules.
Each port has the port direction register (PDR) that selects non-use, input, or output, the port output data register (PODR) that holds data for output, the port input register (PIDR) that indicates the pin states, the pull-up/pull-down control register (PCR) that controls enabling and disabling of the input pull-up/pull-down resistor, and the port mode register (PMR) that specifies the pin function of each port. For details on PMR, refer to section 17, Multi-Function Pin Controller (MPC).
Table 16.1 shows the specifications of I/O ports, and Table 16.2 lists the port functions.

Table 16.1 Specifications of I/O Ports

	Package	
Port	**112 Pins**	**Number of Pins**
PORT0	P00	1
PORT1	P10	1
PORT2	P21, P22, P27	3
PORT3	P33 to P35	3
PORT4	P40, P42, P44	3
PORT5*1	P50 to P56	—
PORT6	P60 to P65	6
PORT7	P71 to P73	3
PORT9	P90 to P97	8
PORTA	PA3 to PA5	3
PORTC	PC2, PC3, PC6, PC7	4
PORTE	PE0 to PE7	8
PORTG	PG2 to PG6	5
	Total of pins	48

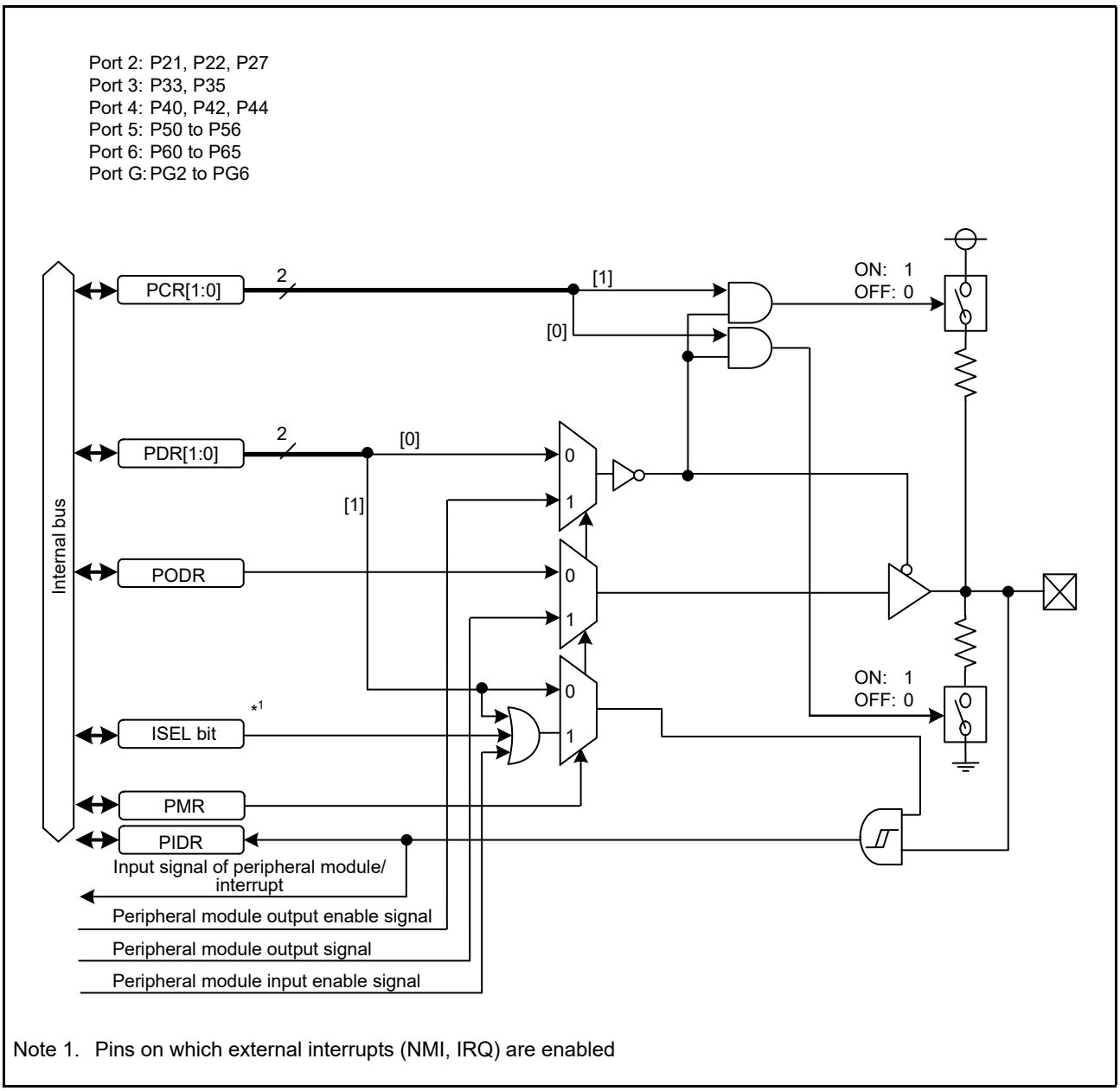
Note 1. PORT5 (P50 to P56) functions as a 1.2-V pin.

Table 16.2 Port Functions

Port	Pin	Input Pull-Up/Pull-Down	Switching of Driving Ability	5-V Tolerant	Schmitt Input
PORT0	P00	√	—	—	—
PORT1	P10	√	—	—	√
PORT2	P21, P22, P27	√	—	—	√
PORT3	P33, P35	√	—	—	√
	P34	√	—	—	—
PORT4	P40, P42, P44	√	—	—	√
PORT5	P50 to P56	√	√	—	√
PORT6	P60 to P65	√	—	—	√
PORT7	P71 to P73	√	—	—	—
PORT9	P90 to P97	√	—	—	√
PORTA	PA3 to PA5	√	—	—	—
PORTC	PC2, PC3, PC6, PC7	—	—	√	√
PORTE	PE0 to PE7	√	—	—	—
PORTG	PG2 to PG6	√	—	—	√

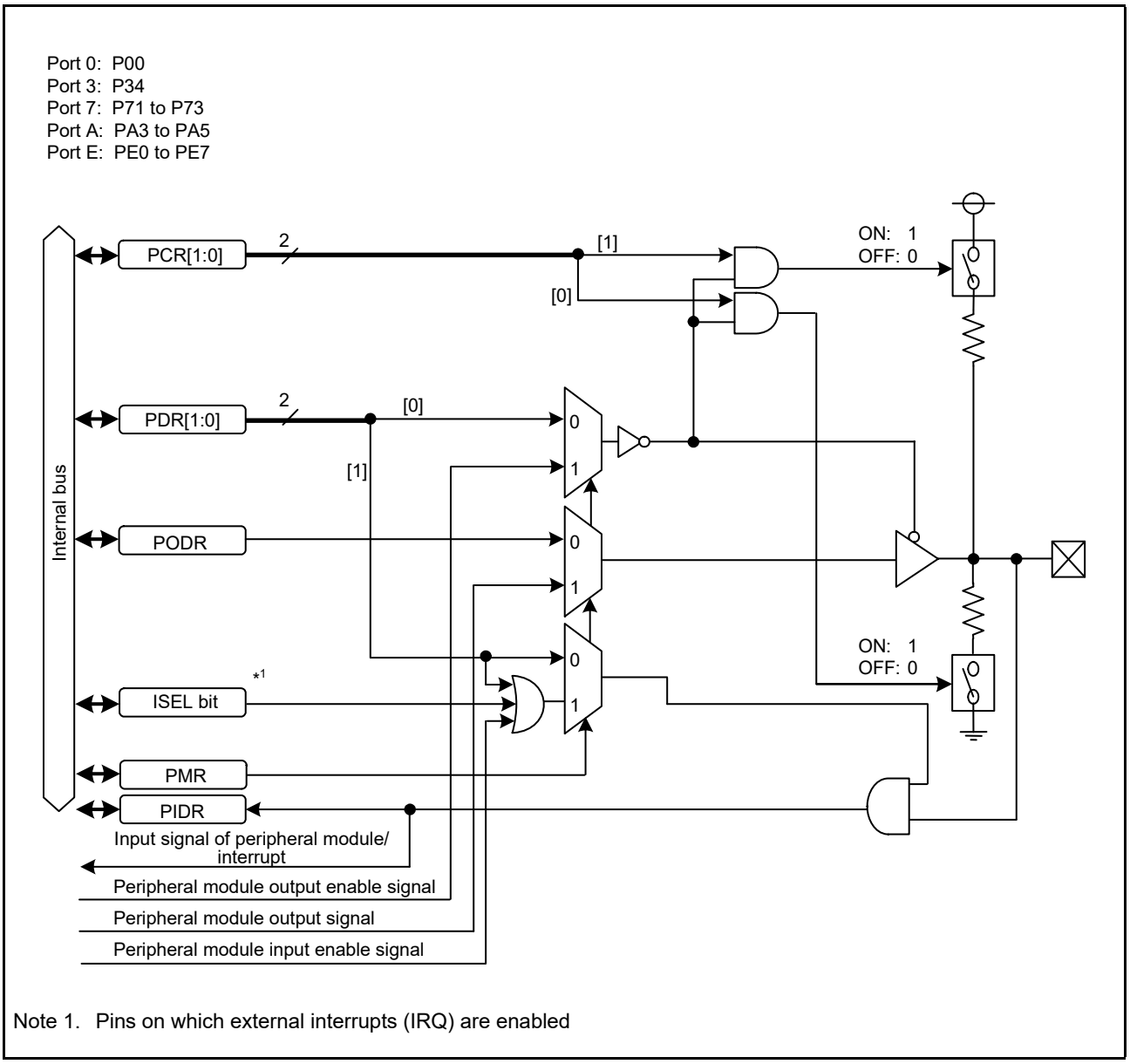
Specifying input pull-up/pull-down or switching of driving ability is available for other signals on pins that also function as general I/O pins.

16.2 I/O Port Configuration



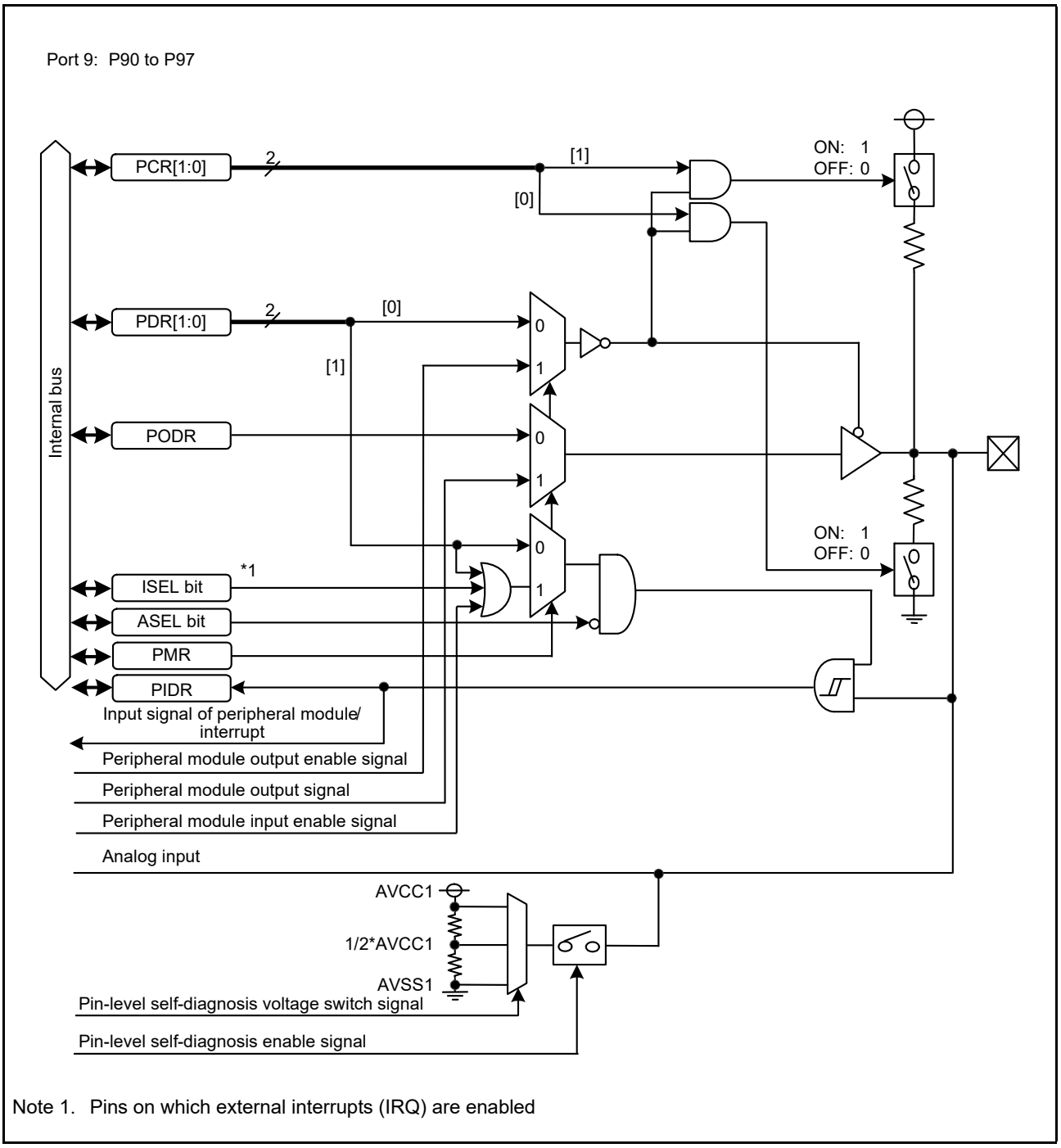
Note 1. Pins on which external interrupts (NMI, IRQ) are enabled

Figure 16.1 I/O Port Configuration (1)



Note 1. Pins on which external interrupts (IRQ) are enabled

Figure 16.2 I/O Port Configuration (2)



Note 1. Pins on which external interrupts (IRQ) are enabled

Figure 16.3 I/O Port Configuration (3)

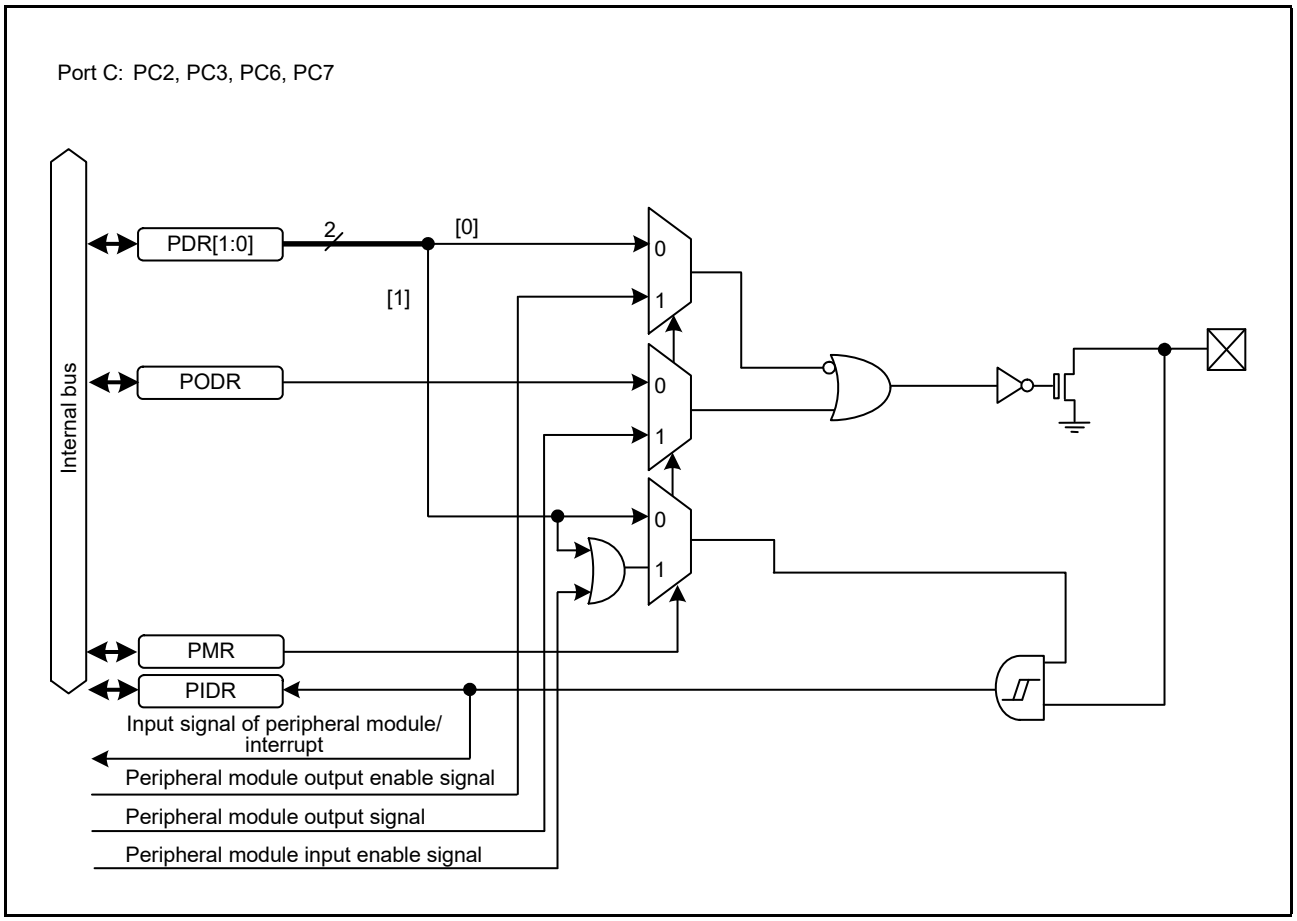


Figure 16.4 I/O Port Configuration (4)

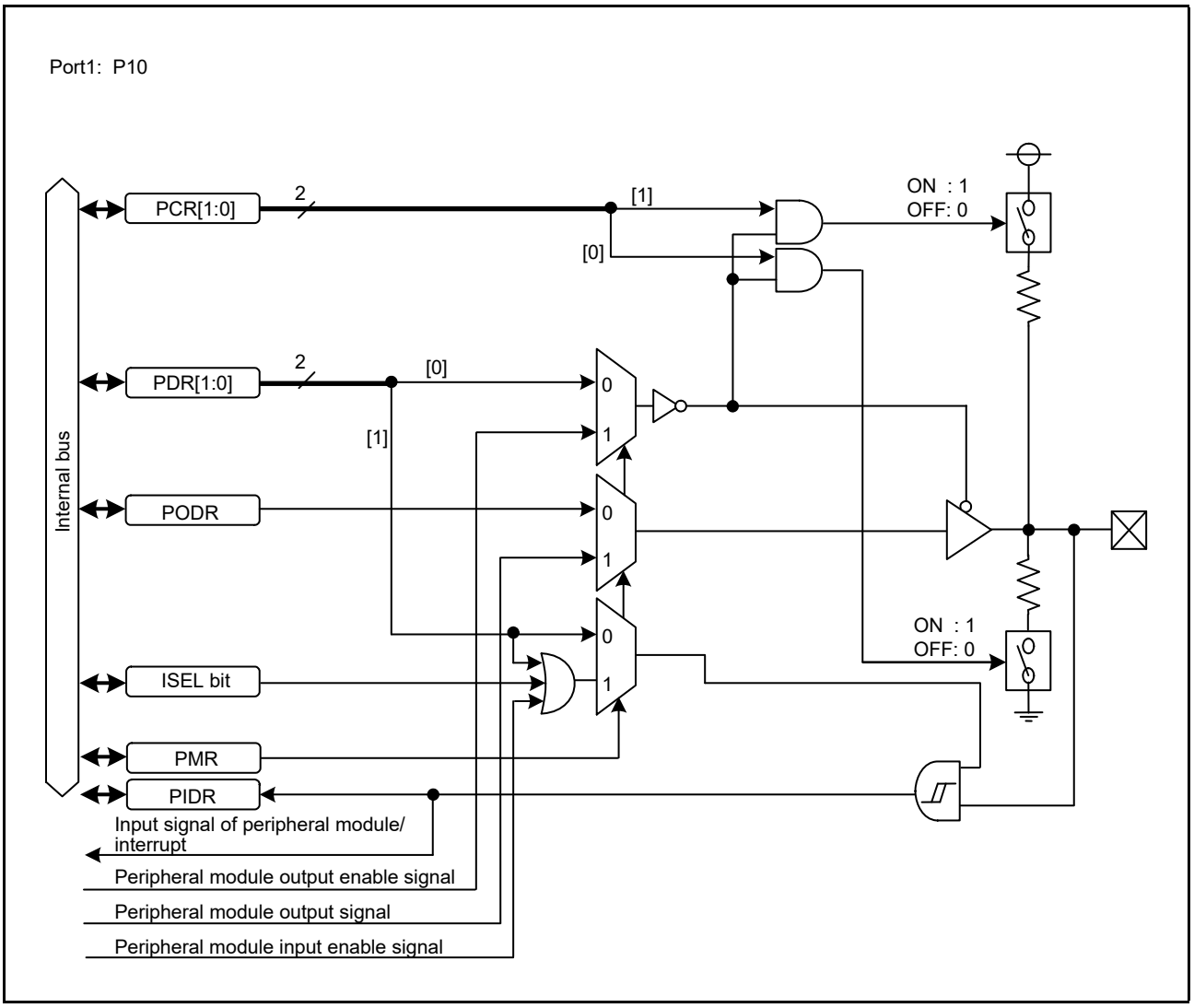


Figure 16.5 I/O Port Configuration (5)

16.3 Register Descriptions

16.3.1 Port Direction Register (PDR)

The PDR register is used to select non-use, input, or output (input enable) for individual pins of the corresponding port when the pins are configured as the general I/O pins. When 00 (non-use) is set to this register, this LSI can be protected from input Hi-Z state.

Address(es): PORT0.PDR A000 0000h, PORT1.PDR A000 0002h, PORT2.PDR A000 0004h, PORT3.PDR A000 0006h, PORT4.PDR A000 0008h, PORT5.PDR A000 000Ah, PORT6.PDR A000 000Ch, PORT7.PDR A000 000Eh, PORT9.PDR A000 0012h, PORTA.PDR A000 0014h, PORTC.PDR A000 0018h, PORTE.PDR A000 001Ch, PORTG.PDR A000 0020h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	B7[1:0]		B6[1:0]		B5[1:0]		B4[1:0]		B3[1:0]		B2[1:0]		B1[1:0]		B0[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	B0[1:0]	Pm0 I/O Select	Odd bit Even bit 0 0: Non-use (Hi-Z input protection) 0 1: Setting prohibited 1 0: Input (functions as an input pin) 1 1: Output (functions as an output pin (port read enable))	R/W
b3, b2	B1[1:0]	Pm1 I/O Select		R/W
b5, b4	B2[1:0]	Pm2 I/O Select		R/W
b7, b6	B3[1:0]	Pm3 I/O Select		R/W
b9, b8	B4[1:0]	Pm4 I/O Select		R/W
b11, b10	B5[1:0]	Pm5 I/O Select		R/W
b13, b12	B6[1:0]	Pm6 I/O Select		R/W
b15, b14	B7[1:0]	Pm7 I/O Select		R/W

m = 0 to 7, 9, A, C, E, and G

Each bit of PORTm.PDR corresponds to each pin of port m; pin function can be specified in 2-bit units.
The bit corresponding to a pin that does not exist is also reserved. A reserved bit is always read as 0. The write value should always be 0.

16.3.2 Port Output Data Register (PODR)

The PODR register holds the data to be output from the pins used for general I/O.

Address(es): PORT0.PODR A000 0040h, PORT1.PODR A000 0041h, PORT2.PODR A000 0042h, PORT3.PODR A000 0043h, PORT4.PODR A000 0044h, PORT6.PODR A000 0046h, PORT7.PODR A000 0047h, PORT9.PODR A000 0049h, PORTA.PODR A000 004Ah, PORTC.PODR A000 004Ch, PORTE.PODR A000 004Eh, PORTG.PODR A000 0050h

	b7	b6	b5	b4	b3	b2	b1	b0
	B7	B6	B5	B4	B3	B2	B1	B0
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	B0	Pm0 Output Data Store	0: Low output 1: High output	R/W
b1	B1	Pm1 Output Data Store		R/W
b2	B2	Pm2 Output Data Store		R/W
b3	B3	Pm3 Output Data Store		R/W
b4	B4	Pm4 Output Data Store		R/W
b5	B5	Pm5 Output Data Store		R/W
b6	B6	Pm6 Output Data Store		R/W
b7	B7	Pm7 Output Data Store		R/W

m = 0 to 4, 6, 7, 9, A, C, E, and G

The bit corresponding to a pin that does not exist is reserved. A reserved bit is always read as 0. The write value should always be 0.

16.3.3 Port Input Data Register (PIDR)

The PIDR register reflects the states of the individual input port pins.

Address(es): PORT0.PIDR A000 0060h, PORT1.PIDR A000 0061h, PORT2.PIDR A000 0062h, PORT3.PIDR A000 0063h, PORT4.PIDR A000 0064h, PORT6.PIDR A000 0066h, PORT7.PIDR A000 0067h, PORT9.PIDR A000 0069h, PORTA.PIDR A000 006Ah, PORTC.PIDR A000 006Ch, PORTE.PIDR A000 006Eh, PORTG.PIDR A000 0070h

	b7	b6	b5	b4	b3	b2	b1	b0
	B7	B6	B5	B4	B3	B2	B1	B0
Value after reset:	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	R/W
b0	B0	Pm0	0: Low input 1: High input	R
b1	B1	Pm1		R
b2	B2	Pm2		R
b3	B3	Pm3		R
b4	B4	Pm4		R
b5	B5	Pm5		R
b6	B6	Pm6		R
b7	B7	Pm7		R

m = 0 to 4, 6, 7, 9, A, C, E, and G

If PORTm.PDR is set to 10 or 11, the pin states of ports m can be read with PORTm.PIDR, regardless of the values of PORTm.PMR.

The bit corresponding to a pin that does not exist is reserved. A reserved bit is read as undefined, and cannot be modified.

16.3.4 Port Mode Register (PMR)

The PMR register specifies the function of the pins of the port.

Address(es): PORT0.PMR A000 0080h, PORT1.PMR A000 0081h, PORT2.PMR A000 0082h, PORT3.PMR A000 0083h, PORT4.PMR A000 0084h, PORT5.PMR A000 0085h, PORT6.PMR A000 0086h, PORT7.PMR A000 0087h, PORT9.PMR A000 0089h, PORTA.PMR A000 008Ah, PORTC.PMR A000 008Ch, PORTE.PMR A000 008Eh, PORTG.PMR A000 0090h

	b7	b6	b5	b4	b3	b2	b1	b0
	B7	B6	B5	B4	B3	B2	B1	B0
Value after reset: *1	0	0	0	0	0	0	0	0

Note 1. The PMR register value for port 3 after a reset is 18h.

Bit	Symbol	Bit Name	Description	R/W
b0	B0	Pm0 Pin Mode Control	0: Uses the pin as a general I/O pin 1: Uses the pin as an I/O port for peripheral functions	R/W
b1	B1	Pm1 Pin Mode Control		R/W
b2	B2	Pm2 Pin Mode Control		R/W
b3	B3	Pm3 Pin Mode Control		R/W
b4	B4	Pm4 Pin Mode Control		R/W
b5	B5	Pm5 Pin Mode Control		R/W
b6	B6	Pm6 Pin Mode Control		R/W
b7	B7	Pm7 Pin Mode Control		R/W

m = 0 to 7, 9, A, C, E, and G

Each bit of PORTm.PMR corresponds to each pin of port m; pin function can be specified in 1-bit units. The bit corresponding to a pin that does not exist is reserved. A reserved bit is always read as 0. The write value should always be 0.

16.3.5 Pull-Up/Pull-Down Control Register (PCR)

The PCR register enables or disables an input pull-up or pull-down resistor for each pin of the port.
When a pin for the general port or peripheral module is in the input state, the input pull-up resistor connected to the pin with the corresponding bit in PORTm.PCR set to 10 is enabled, and the input pull-down resistor connected to the pin with the corresponding bit in PORTm.PCR set to 01 is enabled.
When a pin is set as a general port output pin or a peripheral module output pin, the pull-up resistor for the pin is disabled regardless of the settings of PCR. Note that the PC2, PC3, PC6, and PC7 pins do not have his function.
The pull-up and pull-down resistors are also disabled in the reset state.

Address(es): PORT0.PCR A000 0100h, PORT1.PCR A000 0102h, PORT2.PCR A000 0104h, PORT3.PCR A000 0106h, PORT4.PCR A000 0108h, PORT5.PCR A000 010Ah, PORT6.PCR A000 010Ch, PORT7.PCR A000 010Eh, PORT9.PCR A000 0112h, PORTA.PCR A000 0114h, PORTE.PCR A000 011Ch, PORTG.PCR A000 0120h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	B7[1:0]		B6[1:0]		B5[1:0]		B4[1:0]		B3[1:0]		B2[1:0]		B1[1:0]		B0[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	B0[1:0]	Pm0 Input Pull-Up/Pull-Down Resistor Control	Odd bit Even bit 0 0: Disables an input pull-up and pull-down resistors 0 1: Enables an input pull-down resistor 1 0: Enables an input pull-up resistor 1 1: Setting prohibited	R/W
b3, b2	B1[1:0]	Pm1 Input Pull-Up/Pull-Down Resistor Control		R/W
b5, b4	B2[1:0]	Pm2 Input Pull-Up/Pull-Down Resistor Control		R/W
b7, b6	B3[1:0]	Pm3 Input Pull-Up/Pull-Down Resistor Control		R/W
b9, b8	B4[1:0]	Pm4 Input Pull-Up/Pull-Down Resistor Control		R/W
b11, b10	B5[1:0]	Pm5 Input Pull-Up/Pull-Down Resistor Control		R/W
b13, b12	B6[1:0]	Pm6 Input Pull-Up/Pull-Down Resistor Control		R/W
b15, b14	B7[1:0]	Pm7 Input Pull-Up/Pull-Down Resistor Control		R/W

m = 0 to 7, 9, A, C, E, and G

The bit corresponding to a pin that does not exist is also reserved. A reserved bit is always read as 00b. The write value should always be 00b.

16.3.6 Driving Ability Control Register (DSCR)

The DSCR register controls driving ability of the P50 to P56.
The bit corresponding to a pin whose driving ability has been fixed is readable and writable, but the driving ability cannot be changed.

Address(es): PORT5.DSCR A000 014Ah

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	B6[1:0]		B5[1:0]		B4[1:0]		B3[1:0]		B2[1:0]		B1[1:0]		B0[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	B0[0]	P50 Driving Ability Control Bit 0	0 0: Value after reset 0 1: Setting prohibited 1 0: Setting prohibited 1 1: 1.2-V driving output Note: In actual operation, set these bits to 11b.	R/W
b1	B0[1]	P50 Driving Ability Control Bit 1		R/W
b2	B1[0]	P51 Driving Ability Control Bit 0		R/W
b3	B1[1]	P51 Driving Ability Control Bit 1		R/W
b4	B2[0]	P52 Driving Ability Control Bit 0		R/W
b5	B2[1]	P52 Driving Ability Control Bit 1		R/W
b6	B3[0]	P53 Driving Ability Control Bit 0		R/W
b7	B3[1]	P53 Driving Ability Control Bit 1		R/W
b8	B4[0]	P54 Driving Ability Control Bit 0		R/W
b9	B4[1]	P54 Driving Ability Control Bit 1		R/W
b10	B5[0]	P55 Driving Ability Control Bit 0		R/W
b11	B5[1]	P55 Driving Ability Control Bit 1		R/W
b12	B6[0]	P56 Driving Ability Control Bit 0		R/W
b13	B6[1]	P56 Driving Ability Control Bit 1		R/W
b15, b14	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

The bit corresponding to a pin that does not exist is also reserved. A reserved bit is always read as 0. The write value should always be 0.

16.4 Handling of Unused Pins

Table 16.3 lists the details of handling of unused pins.

Table 16.3 Handling of Unused Pins

Pin Name	Handling
MD0, MD1	— (Use this as a mode pin.)
RSTOUT#	Keep this pin open.
TRST#	Connect these pins to VSS via a resistor (pulling down), or input the same signal as that on the RES# pin.
TCK	Connect these pins to VSS via a resistor (pulling down).
TMS	Connect this pin to VCCQ33 via a resistor (pulling up).
Port 34 (TDI)	Connect this pin to VCCQ33 via a resistor (pulling up).
Port 0 to port 4, port 6, port 7, port 9, port A, port C, port E, and port G (except port 34)*1	Keep these pins open, connect them to VCCQ33 via a resistor (pulling up), or connect them to VSS via a resistor (pulling down).
Port 5*1	Keep these pins open, connect them to VCCQ12 via a resistor (pulling up), or connect them to VSS via a resistor (pulling down).
VREFH0	Connect this pin to AVCC0.
VREFL0	Connect this pin to AVSS0.
VREFH1	Connect this pin to AVCC1.
VREFL1	Connect this pin to AVSS1.
AN000 to AN007	Connect these pins to AVSS0 via a resistor (pulling down).

Note 1. When handling them as unused pins, set the corresponding bits of the port direction register (PDR) to “Non-use (Hi-Z input protection)” which is the value after reset release.

17. Multi-Function Pin Controller (MPC)

17.1 Overview

This LSI configures an I/O pin or an interrupt pin of peripheral functions to be multiplexed with multiple ports. The multi-function pin controller (MPC) is a module that selects I/O pins and interrupt pins for the peripheral function to use from multiple ports, and then assigns the function to the selected pins.
Table 17.1 lists the multiplexed pin configurations.Selecting a single function for multiple pins is prohibited.

Table 17.1 List of Multiplexed Pin Configurations (1 / 3)

Module/Function	Channel	Pin Function	Allocation Port
Debugging interface		TDI (input)	P34
		TDO (output)	P33
		TRACECLK (output)	P10
		TRACECTL (output)	P00
			P71
		TRACEDATA0 (output)	P72
			PE0
		TRACEDATA1 (output)	P73
			PE1
		TRACEDATA2 (output)	PE2
		TRACEDATA3 (output)	PE3
		TRACEDATA4 (output)	PE4
		TRACEDATA5 (output)	PE5
		TRACEDATA6 (output)	PE6
		TRACEDATA7 (output)	PE7
Interrupt	NMI	NMI (input)	P35
	IRQ0	IRQ0 (input)	P10
	IRQ1	IRQ1 (input)	P21
	IRQ2	IRQ2 (input)	P22
			PE2
	IRQ3	IRQ3 (input)	P73
			PE3
	IRQ4	IRQ4 (input)	P94
	IRQ6	IRQ6 (input)	PE6
	IRQ7	IRQ7 (input)	P97
16-bit timer pulse unit	TPU0	TIOCA0 (input/output)	P10
		TIOCB0 (input/output)	P27
		TIOCC0 (input/output)	PE4
		TIOCD0 (input/output)	P22
			PE6
	TPU1	TIOCA1 (input/output)	P00
		TIOCB1 (input/output)	P21
	TPU2	TIOCA2 (input/output)	PA3
		TIOCB2 (input/output)	PE0

Table 17.1 List of Multiplexed Pin Configurations (2 / 3)

Module/Function	Channel	Pin Function	Allocation Port
16-bit timer pulse unit	TPU3	TIOCA3 (input/output)	PA4
		TIOCB3 (input/output)	PE1
		TIOCC3 (input/output)	PE5
		TIOCD3 (input/output)	PE7
	TPU4	TIOCA4 (input/output)	PA5
		TIOCB4 (input/output)	PE2
	TPU5	TIOCA5 (input/output)	P90
		TIOCB5 (input/output)	PE3
	TPU	TCLKA (input)	PG5
		TCLKB (input)	PG6
		TCLKC (input)	PC6
		TCLKD (input)	P44
Compare match timer W	CMTW0	TOC0 (output)	PG2
		TIC0 (input)	PC7
	CMTW1	TOC1 (output)	PG4
		TIC1 (input)	PG3
	CMTW2	TOC2 (output)	P71
		TIC2 (input)	P72
	CMTW3	TOC3 (output)	P92
		TIC3 (input)	P93
FIFO on-chip serial communication interface (SCIFA)	SCI0	RXD0 (input)	P42
		TXD0 (output)	P40
		SCK0 (input/output)	P22
		CTS0# (input/output)	P21
			P44
		RTS0# (output)	P27
	SCI1	RXD1 (input)	P73
			PE6
		TXD1 (output)	P72
			PE5
		SCK1 (input/output)	P71
			PE7
		CTS1# (input/output)	PE3
		RTS1# (output)	PE4
	SCI2	RXD2 (input)	P92
			PA4
		TXD2 (output)	P91
			PA5
		SCK2 (input/output)	P93
			PA3
		CTS2# (input/output)	P95
		RTS2# (output)	P94
	SCI4	RXD4 (input)	PC3
		TXD4 (output)	P90

Table 17.1 List of Multiplexed Pin Configurations (3 / 3)

Module/Function	Channel	Pin Function	Allocation Port
I²C bus interface	RIIC0	SCL0 (input/output)	PC3
		SDA0 (input/output)	PC2
	RIIC1	SCL1 (input/output)	PC6
		SDA1 (input/output)	PC7
Management data input/output interface		PRTADR4 (input)	P50
		PRTADR2 (input)	P51
		PRTADR1 (input)	P52
		PRTADR3 (input)	P53
		PRTADR0 (input)	P54
		MDIO (I/O)	P55
		MDC (input)	P56
		MMDC1 (output)	PA3
		MMDIO1 (input/output)	PA5
Serial peripheral interface	RSPI0	RSPCK0 (input/output)	PE7
		MOSI0 (input/output)	PE5
		MISO0 (input/output)	PE6
		SSL00 (input/output)	PE4
		SSL01 (output)	PE3
		SSL02 (output)	PE2
		SSL03 (output)	PE1
	RSPI1	RSPCK1 (input/output)	PG2
		MOSI1 (input/output)	PG4
		MISO1 (input/output)	PG3
		SSL10 (input/output)	PG5
		SSL11 (output)	PG6
Quad serial peripheral interface		SPBCLK (output)	P62
		SPBMO/SPBIO0 (input/output)	P63
		SPBMI/SPBIO1 (input/output)	P64
		SPBIO2 (input/output)	P65
		SPBIO3 (input/output)	P61
		SPBSSL (output)	P60
12-bit A/D converter	Unit 0	ADTRG0 (input)	P44
			PA4
	Unit 1	AN100 (input)*1	P90
		AN101 (input)*1	P91
		AN102 (input)*1	P92
		AN103 (input)*1	P93
		AN104 (input)*1	P94
		AN105 (input)*1	P95
		AN106 (input)*1	P96
		AN107 (input)*1	P97
		ADTRG1 (input)	P00
			P97

Note 1. To use this pin, make the applicable pin unavailable, and configure it as a general input/output port (by setting the PORTm.PDR.Bn bit to 00, and the PORTm.PMR.Bn bit to 0).

17.2 Register Descriptions

17.2.1 Write-Protect Register (PWPR)

The PWPR register enables or disables writing to the PFS register and the PFSWE bit of the PWPR register.

Address(es): A000 02FFh

	b7	b6	b5	b4	b3	b2	b1	b0
	B0WI	PFSWE	—	—	—	—	—	—
Value after reset:	1	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b5 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b6	PFSWE	PFS Register Write Enable	0: Writing to the PFS register is disabled. 1: Writing to the PFS register is enabled.	R/W
b7	B0WI	PFSWE Bit Write Disable	0: Writing to the PFSWE bit is enabled. 1: Writing to the PFSWE bit is disabled.	R/W

PFSWE Bit (PFS Register Write Enable)

Writing to the PmnPFS register (m = 0 to 7, 9, A, C, E, G, and n = 0 to 7) is enabled only when the PFSWE bit is set to 1. To set the PFSWE bit to 1, write 0 to the B0WI bit, and then set 1 to the PFSWE bit.

B0WI Bit (PFSWE Bit Write Disable)

Only when the B0WI bit is set to 0, writing to the PFSWE bit is enabled.

17.2.2 P0n Pin Function Control Register (P0nPFS) (n = 0)

The P0n pin function control register (P0nPFS) selects the function of the pin to use.
The P0nPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection.

Address(es): P00PFS A000 0200h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	PSEL[5:0]					
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each function, see Table 17.2.	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

Table 17.2 Register Settings for the Input/Output Function

PSEL[5:0] Setting	Pin
	P00
000000b (Value after reset)	Hi-Z
000011b	TIOCA1
001001b	ADTRG1
100111b	TRACECTL

17.2.3 P1n Pin Function Control Register (P1nPFS) (n = 0)

The P1n pin function control register (P1nPFS) selects the function of the pin to use.
The P1nPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection. The ISEL bit of the pin without the IRQn function is reserved. A value after a reset must be written.

Address(es): P10PFS A000 0208h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	ISEL	PSEL[5:0]					
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each pin function, see Table 17.3.	R/W
b6	ISEL	Interrupt Input Function Select	0: Do not use as the IRQn input pin. 1: Use as the IRQn input pin.	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

ISEL Bit (Interrupt Input Function Select)

Set this bit to 1 to use the given pin as the IRQ input pin. This bit can be used with peripheral functions. Note that IRQn (external pin interrupt) with the same number cannot be enabled for two or more pins.

Table 17.3 Register Settings for the Input/Output Function

	Pin
PSEL[5:0] Setting	**P10**
000000b (Value after reset)	Hi-Z
000011b	TIOCA0
100111b	TRACECLK

17.2.4 P2n Pin Function Control Register (P2nPFS) (n = 1, 2, 7)

The P2n pin function control register (P2nPFS) selects the function of the pin to use. The P2nPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection. The ISEL bit of the pin without the IRQn function is reserved. A value after a reset must be written.

Address(es): P21PFS A000 0211h, P22PFS A000 0212h, P27PFS A000 0217h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	ISEL	PSEL[5:0]					
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each function, see Table 17.4.	R/W
b6	ISEL	Interrupt Input Function Select	0: Do not use as the IRQn input pin. 1: Use as the IRQn input pin.	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

ISEL Bit (Interrupt Input Function Select)

Set this bit to 1 to use the given pin as the IRQ pin. This bit can be used with peripheral functions. Note that IRQn (external pin interrupt) with the same number cannot be enabled for two or more pins.

Table 17.4 Register Settings for the Input/Output Function

	Pin		
PSEL[5:0] Setting	**P21**	**P22**	**P27**
000000b (Value after reset)	Hi-Z		
000011b	TIOCB1	TIOCD0	TIOCB0
001010b	—	—	RTS0#
001011b	CTS0#	SCK0	—

Note: —: Do not set.

17.2.5 P3n Pin Function Control Register (P3nPFS) (n = 3 to 5)

The P3n pin function control register (P3nPFS) selects the function of the pin to use. The P3nPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection. The ISEL bit of the pin without the IRQn function is reserved. A value after a reset must be written. No peripheral functions are assigned to P35. To use P35 as the NMI pin, see section 12.3.4, NMI Pin Interrupts.

Address(es): P33PFS A000 021Bh, P34PFS A000 021Ch, P35PFS A000 021Dh

	b7	b6	b5	b4	b3	b2	b1	b0	
	—	ISEL	PSEL[5:0]						
Value after reset:	0	0	0	0	0	0	0	0	*1
	0	0	1	0	0	1	1	1	*2

Note 1. P35PFS
Note 2. P33PFS, P34PFS

Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each pin function, see Table 17.5.	R/W
b6	ISEL	Interrupt Input Function Select	0: Do not use as the IRQn input pin or NMI input pin. 1: Use as the IRQn input pin or NMI input pin.	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

ISEL Bit (Interrupt Input Function Select)

Set this bit to 1 to use the given pin as the IRQ input pin or the NMI input pin (P35). This bit can be used with peripheral functions. Note that IRQn (external pin interrupt) with the same number cannot be enabled for two or more pins. To use the pin as the NMI pin, see section 12.3.4, NMI Pin Interrupts.

Table 17.5 Register Settings for the Input/Output Function

	Pin	
PSEL[5:0]Setting	P33	P34
000000b	Hi-Z	
100111b (Value after reset)	TDO	TDI

17.2.6 P4n Pin Function Control Register (P4nPFS) (n = 0, 2, 4)

The P4n pin function control register (P4nPFS) selects the function of the pin to use. The P4nPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection.

Address(es): P40PFS A000 0220h, P42PFS A000 0222h, P44PFS A000 0224h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	PSEL[5:0]					
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each pin function, see Table 17.6.	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

Table 17.6 Register Settings for the Input/Output Function

	Pin		
PSEL[5:0]Setting	**P40**	**P42**	**P44**
000000b (Value after reset)	Hi-Z		
000011b	—	—	TCLKD
001001b	—	—	ADTRG0
001010b	TXD0	RXD0	CTS0#

Note: —: Do not set.

17.2.7 P5n Pin Function Control Register (P5nPFS) (n = 0 to 6)

The P5n pin function control register (P5nPFS) selects the function of the pin to use. The P5nPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection.

Address(es): P50PFS A000 0228h, P51PFS A000 0229h, P52PFS A000 022Ah, P53PFS A000 022Bh, P54PFS A000 022Ch, P55PFS A000 022Dh, P56PFS A000 022Eh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	PSEL[5:0]					
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each pin function, see Table 17.7.	R/W
b7, b6	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

Table 17.7 Register Settings for the Input/Output Function in the 320-pin FBGA Pin

PSEL[5:0] Setting	Pin						
	P50	P51	P52	P53	P54	P55	P56
000000b (Value after reset)	Hi-Z						
101011b	PRTADR4	PRTADR2	PRTADR1	PRTADR3	PRTADR0	MDIO	MDC

17.2.8 P6n Pin Function Control Register (P6nPFS) (n = 0 to 5)

The P6n pin function control register (P6nPFS) selects the function of the pin to use. The P6nPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection.

Address(es): P60PFS A000 0230h, P61PFS A000 0231h, P62PFS A000 0232h, P63PFS A000 0233h, P64PFS A000 0234h, P65PFS A000 0235h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	PSEL[5:0]					
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each pin function, see Table 17.8.	R/W
b7-b6	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

Table 17.8 Register Settings for the Input/Output Function

	Pin					
PSEL[5:0] Setting	**P60**	**P61**	**P62**	**P63**	**P64**	**P65**
000000b (Value after reset)	Hi-Z					
011011b	SPBSSL	SPBIO3	SPBCLK	SPBMO/ SPBIO0	SPBMI/SPBIO1	SPBIO2

17.2.9 P7n Pin Function Control Register (P7nPFS) (n = 1 to 3)

The P7n pin function control register (P7nPFS) selects the function of the pin to use. The P7nPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection. The ISEL bit of the pin without the IRQn function is reserved. A value after a reset must be written.

Address(es): P71PFS A000 0239h, P72PFS A000 023Ah, P73PFS A000 023Bh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	ISEL	PSEL[5:0]					
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each function, see Table 17.9.	R/W
b6	ISEL	Interrupt Input Function Select	0: Do not use as the IRQn input pin. 1: Use as the IRQn input pin.	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

ISEL Bit (Interrupt Input Function Select)

Set this bit to 1 to use the given pin as the IRQ pin. This bit can be used with peripheral functions. Note that IRQn (external pin interrupt) with the same number cannot be enabled for two or more pins.

Table 17.9 Register Settings for the Input/Output Function

	Pin		
PSEL[5:0]Setting	**P71**	**P72**	**P73**
000000b (Value after reset)	Hi-Z		
001010b	SCK1	TXD1	RXD1
011101b	TOC2	TIC2	—
100111b	TRACECTL	TRACEDATA0	TRACEDATA1

Note: —: Do not set.

17.2.10 P9n Pin Function Control Register (P9nPFS) (n = 0 to 7)

The P9n pin function control register (P9nPFS) selects the function of the pin to use. The P9nPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection. The ISEL bit of a pin without the IRQn function, and the ASEL bit of a pin without the analog input function are reserved. A value after a reset must be written.

Address(es): P90PFS A000 0248h, P91PFS A000 0249h, P92PFS A000 024Ah, P93PFS A000 024Bh, P94PFS A000 024Ch, P95PFS A000 024Dh, P96PFS A000 024Eh, P97PFS A000 024Fh

	b7	b6	b5	b4	b3	b2	b1	b0
	ASEL	ISEL	PSEL[5:0]					
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each function, see Table 17.10.	R/W
b6	ISEL	Interrupt Input Function Select	0: Do not use as the IRQn input pin. 1: Use as the IRQn input pin.	R/W
b7	ASEL	Analog Input Function Select	0: Do not use as an analog input. 1: Use as an analog input.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

ISEL Bit (Interrupt Input Function Select)

Set this bit to 1 to use the given pin as an IRQ input pin. This bit can be used with peripheral functions. Note that IRQn (external pin interrupt) with the same number cannot be enabled for two or more pins.
If the setting of the ASEL bit is 1, the given pin does not function as an IRQn input pin even if this bit is set to 1.

ASEL Bit (Analog Input Function Select)

Set this bit to 1 to use the given pin as an analog pin. To set the ASEL bit as an analog pin, select the general input/output port by the port mode register (PORTm.PMR), and then disable it by the port direction register (PORTm.PDR). At that time, the pin status cannot be read.

Table 17.10 Register Settings for the Input/Output Function

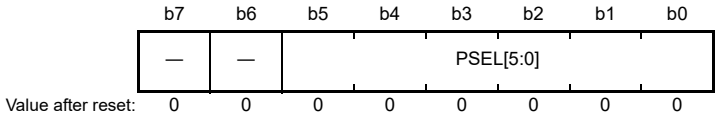
	Pin							
PSEL[5:0] Setting	P90	P91	P92	P93	P94	P95	P96	P97
000000b (Value after reset)	Hi-Z							
000011b	TIOCA5	—	—	—	—	—	—	—
001001b	—	—	—	—	—	—	—	ADTRG1
001011b	—	TXD2	RXD2	SCK2	RTS2#	CTS2#	—	—
001100b	TXD4	—	—	—	—	—	—	—
011101b	—	—	TOC3	TIC3	—	—	—	—

Note: —: Do not set.

17.2.11 PAn Pin Function Control Register (PAnPFS) (n = 3 to 5)

The PAn pin function control register (PAnPFS) selects the function of the pin to use. The PAnPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection.

Address(es): PA3PFS A000 0253h, PA4PFS A000 0254h, PA5PFS A000 0255h



Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each function, see Table 17.11.	R/W
b7, b6	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

Table 17.11 Register Settings for the Input/Output Function

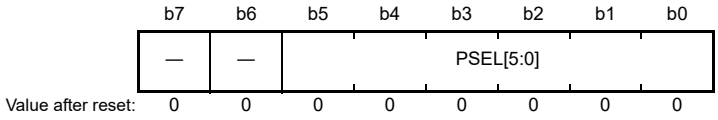
	Pin		
PSEL[5:0] Setting	**PA3**	**PA4**	**PA5**
000000b (Value after reset)	Hi-Z		
000011b	TIOCA2	TIOCA3	TIOCA4
001001b	—	ADTRG0	—
001010b	SCK2	RXD2	TXD2
101011b	MMDC1*1	—	MMDIO1*1

Note: —: Do not set.
Note 1. Only for products with the MDIO master module (optional).

17.2.12 PCn Pin Function Control Register (PCnPFS) (n = 2, 3, 6, 7)

The PCn pin function control register (PCnPFS) selects the function of the pin to use. The PCnPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection.

Address(es): PC2PFS A000 0262h, PC3PFS A000 0263h, PC6PFS A000 0266h, PC7PFS A000 0267h



Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each pin function, see Table 17.12.	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

Table 17.12 Register Settings for the Input/Output Function

	Pin			
PSEL[5:0] Setting	**PC2**	**PC3**	**PC6**	**PC7**
000000b (Value after reset)	Hi-Z			
000011b	—	—	TCLKC	—
001010b	—	RXD4	—	—
001111b	SDA0	SCL0	SCL1	SDA1
011101b	—	—	—	TIC0

Note: —: Do not set.

17.2.13 PEn Pin Function Control Register (PEnPFS) (n = 0 to 7)

The PEn pin function control register (PEnPFS) selects the function of the pin to use. The PEnPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection. The ISEL bit of the pin without the IRQn function is reserved. A value after a reset must be written.

Address(es): PE0PFS A000 0270h, PE1PFS A000 0271h, PE2PFS A000 0272h, PE3PFS A000 0273h, PE4PFS A000 0274h, PE5PFS A000 0275h, PE6PFS A000 0276h, PE7PFS A000 0277h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	ISEL	PSEL[5:0]					
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each function, see Table 17.13.	R/W
b6	ISEL	Interrupt Input Function Select	0: Do not use as the IRQn input pin. 1: Use as the IRQn input pin.	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

ISEL Bit (Interrupt Input Function Select)

Set this bit to 1 to use the given pin as an IRQ input pin. This bit can be used with peripheral functions. Note that IRQn (external pin interrupt) with the same number cannot be enabled for two or more pins.

Table 17.13 Register Settings for the Input/Output Function

PSEL[5:0] Setting	Pin							
	PE0	PE1	PE2	PE3	PE4	PE5	PE6	PE7
000000b (Value after reset)	Hi-Z							
000011b	TIOCB2	TIOCB3	TIOCB4	TIOCB5	TIOCC0	TIOCC3	TIOCD0	TIOCD3
001100b	—	—	—	CTS1#	RTS1#	TXD1	RXD1	SCK1
001110b	—	SSL03	SSL02	SSL01	SSL00	MOSI0	MISO0	RSPCK0
100111b	TRACEDATA0	TRACEDATA1	TRACEDATA2	TRACEDATA3	TRACEDATA4	TRACEDATA5	TRACEDATA6	TRACEDATA7

Note: —: Do not set.

17.2.14 PGn Pin Function Control Register (PGnPFS) (n = 2 to 6)

The PGn pin function control register (PGnPFS) selects the function of the pin to use. The PGnPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection.

Address(es): PG2PFS A000 0282h, PG3PFS A000 0283h, PG4PFS A000 0284h, PG5PFS A000 0285h, PG6PFS A000 0286h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	PSEL[5:0]					
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each function, see Table 17.14.	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

Table 17.14 Register Settings for the Input/Output Function

	Pin				
PSEL[5:0] Setting	**PG2**	**PG3**	**PG4**	**PG5**	**PG6**
000000b (Value after reset)	Hi-Z				
000011b	—	—	—	TCLKA	TCLKB
001101b	RSPCK1	MISO1	MOSI1	SSL10	SSL11
011101b	TOC0	TIC1	TOC1	—	—

Note: —: Do not set.

17.3 Usage Notes

17.3.1 Procedure for Specifying the Pin Input/Output Function

To specify the pin input/output function:

1. Set the port direction register (PDR) of the applicable pin to 00, and clear the port mode register (PMR) to 0 to set them as a general I/O port.
2. For each peripheral module, set the I/O signal to assign to the applicable pin.
3. Clear the PWPR.B0WI bit to 0, and then set the PWPR.PFSWE bit to 1. By doing so, make the Pmn pin function control register (PmnPFS) (m = 0 to 7, 9, A, C, E, G, n = 0 to 7) writable.
4. By using the PmnPFS.PSEL[5:0] bits, set the pin input/output function.
5. Clear the PWPR.PFSWE bit to 0 to disable writing to the PmnPFS register.
6. Set the applicable bit of the PMR register corresponding to the selected pin to 1 as necessary to switch to the pin input/output function of the peripheral function.
7. Set the PDR register to 10 as necessary to enable reading the port status.

17.3.2 Notes on MPC Register Setting

1. Settings of the Pmn pin function control register (PmnPFS) (m = 0 to 7, 9, A, C, E, G, n = 0 to 7) should be made only while the PMR register for the target pin is cleared to 0. If the PmnPFS is set while the applicable bit of the PMR register is 1, unexpected edges might be input for the input function. Besides, unexpected pulses might be output for the output function.
2. When setting the PmnPFS.ISEL bit to use the IRQ or NMI pin interrupt, follow the procedure described in [For IRQ pins] of section 12.3.3, External Pin Interrupts and section 12.3.4, NMI Pin Interrupts. If the PmnPFS.ISEL bit is set using a different procedure, unexpected edges may be input, leading to malfunction.
3. Only the allowed functions should be specified for the PmnPFS register. If a value that is not allowed for the register is specified, correct operation is not guaranteed.
4. Do not assign a single function to multiple pins through the MPC settings.
5. 9 also function as analog input/output pins for the A/D converter. When using these ports as analog input/output pins, disable the pin by clearing the bit of the corresponding pin in the port mode register (PMR) to 0, and setting the port direction register (PDR) to 00, and set the PmnPFS.ASEL bit to 1, and then setting the PmnPFS.ASEL bit to 1 to avoid degradation of accuracy.
6. Points to note regarding the port mode register (PMR), port direction register (PDR), and Pmn pin function control register (PmnPFS) settings for pins that have multiplexed pin functions are listed in Table 17.15. The pin states are readable if the value of the ASEL bit of the Pmn pin is 0. Ensure that the bit corresponding to the applicable pin of the PMR register is 0 when changes to the PSEL[5:0] bits are made.

Table 17.15 Register Settings

Item	PMR.Bn	PDR.Bn[1:0]	PmnPFS			Note
			ASEL	ISEL	PSEL[5:0]	
After a reset is canceled	0 *1	00	0	0	000000b *1	In the disabled (Hi-z input protection) state after a reset is canceled.
Not used	0	00	0	0	N	
General I/O port	0	10/11 *2	0	0/1 *3	N	Set the PmnPFS.ISEL bit to 1 if these are multiplexed with interrupt inputs.
Peripheral functions	1	00/10 *4	0	0/1 *3	Peripheral functions (See Table 17.2 to Table 17.14.)	Set the PmnPFS.ISEL bit to 1 if it is multiplexed with interrupt inputs. Set the PDR.Bn[1:0] bits to 10 if it is multiplexed with the port read function (reading the pin status of the port in the PIDR.Bn bit).
Interrupt input (NMI, IRQ0 to IRQ4, IRQ6, IRQ7)	0	10	0	1	N	
Analog inputs	0	00	1	N *5	N	For use as analog input, set the given pin to “non-use” by setting the associated bits of the port mode register (PMR) and the port direction register (PDR) to 0 and 00, respectively. After that, set the PmnPFS.ASEL bit to 1. This procedure avoids deterioration of the precision of values.

N: Setting is not required.

Note 1. Values after reset for PORT3.PMR, P33PFS.PSEL[5:0], and P34PFS.PSEL[5:0] are different.
For details on the PSEL[5:0] bits, see section 17.2.5, P3n Pin Function Control Register (P3nPFS) (n = 3 to 5).
For details on PORT3.PMR, see section 16.3.4, Port Mode Register (PMR).

Note 2. Setting the PDR.Bn[1:0] bits to 10 makes the register function as a general input port.
Setting the PDR.Bn[1:0] bits to 11 makes the register function as a general output port.

Note 3. If the PmnPFS.ISEL bit is set to 0, the register does not function as an IRQ pin.
Setting the PmnPFS.ISEL bit to 1 makes the register function as an IRQ pin (when the IRQ function is multiplexed).

Note 4. If the PDR.Bn[1:0] bits is set to 00, the port read function (reading the pin status of the port with the PIDR.Bn bit) cannot be used.
Setting the PDR.Bn[1:0] bits to 10 enables reading of the port pin status.

Note 5. Setting the PmnPFS.ISEL bit to 1 does not make the register function as an IRQn input pin.

17.3.3 Usage Notes on Port Read Function

When a peripheral module for a pin which is bidirectional or an output is in use and the value of the PDR bits for the given pin is changed from 00 (initial value) to 10 (input enabled), the state of the pin can be read (port read function) from the PIDR register while the peripheral function is in use.

When a peripheral module for a pin which is an input or any of the input/output pin functions listed in Table 17.16 is in use, input is always enabled and the pin function can be used in parallel with port reading without changing the setting of the PDR register.

A shoot-through current flows when an external pin enters the Hi-Z state. An external pin must be pulled down or up when in the Hi-Z state.

Table 17.16 List of Modules and Associated Pin Functions for Which Input is Always Enabled (1 / 2)

Module/Function	Channel	Pin Function
16-bit timer pulse unit	TPU0	TIOCA0 (Input/output)
		TIOCB0 (Input/output)
		TIOCC0 (Input/output)
		TIOCD0 (Input/output)
	TPU1	TIOCA1 (Input/output)
		TIOCB1 (Input/output)
	TPU2	TIOCA2 (Input/output)
		TIOCB2 (Input/output)
	TPU3	TIOCA3 (Input/output)
		TIOCB3 (Input/output)
		TIOCC3 (Input/output)
		TIOCD3 (Input/output)
	TPU4	TIOCA4 (Input/output)
		TIOCB4 (Input/output)
	TPU5	TIOCA5 (Input/output)
		TIOCB5 (Input/output)
Serial Communications Interface with FIFO (SCIFA)	SCI0	RXD0 (Input)
		SCK0 (Input/output)
		CTS0# (Input/output)
		RTS0# (Output)
	SCI1	RXD1 (Input)
		SCK1 (Input/output)
		CTS1# (Input/output)
		RTS1# (Output)
	SCI2	RXD2 (Input)
		SCK2 (Input/output)
		CTS2# (Input/output)
		RTS2# (Output)
	SCI4	RXD4 (Input)
		SCK4 (Input/output)
		CTS4# (Input/output)
		RTS4# (Output)

Table 17.16 List of Modules and Associated Pin Functions for Which Input is Always Enabled (2 / 2)

Module/Function	Channel	Pin Function
Serial peripheral interface	RSPI0	RSPCK0 (Input/output)
		MOSI0 (Input/output)
		MISO0 (Input/output)
		SSL00 (Input/output)
	RSPI1	RSPCK1 (Input/output)
		MOSI1 (Input/output)
		MISO1 (Input/output)
		SSL10 (Input/output)
		SSL11 (Output)
SPI multi I/O bus controller		SPBMO/SPBIO0 (Input/output)
		SPBMI/SPBIO1 (Input/output)
		SPBIO2 (Input/output)
		SPBIO3 (Input/output)

18. 16-Bit Timer Pulse Unit (TPUa)

This LSI has one unit (unit 0) of on-chip 16-bit timer pulse unit (TPU) comprising six-channel 16-bit timers, and thus has 6 channels (TPU0 to TPU5) in total.

18.1 Overview

Specifications of the TPU are listed in Table 18.1. Functions of the TPU are listed in Table 18.2.
A block diagram of TPU is shown in Figure 18.1.

Table 18.1 Specifications of TPU

Item	Description
Pulse input/output	Maximum 16
Count clock	Seven or eight types are provided for each channel.
Settable operations	• Waveform output at compare match • Input capture function (noise filters can be set) • Counter clear operation • Simultaneous writing to multiple timer counters (TCNT) • Simultaneous clearing by compare match and input capture • Synchronous input/output for registers by counter synchronous operation • Maximum of 15-phase PWM output by combination with synchronous operation • Cascaded operation
Channels 0 and 3	Buffer operation can be set.
Channels 1, 2, 4, and 5	Phase counting mode can be set for individual channels
Interrupt source	26 sources
Buffer operation	Automatic transfer of register data
Generation of trigger	Conversion start trigger for the A/D converter can be generated.
Event linking (output)	Six types of event signal can be output to the ELC. • Compare match A (TPU0 to TPU3) • Compare match B (TPU0 to TPU3) • Compare match C (TPU0, TPU3) • Compare match D (TPU0, TPU3) • Overflow (TPU0 to TPU3) • Underflow (TPU1, TPU2)
Event linking (input)	Any of the three operations in response to event reception is possible. • Starting counts (TPU0 to TPU3) • Clearing counts (TPU0 to TPU3) • Input capture operation (TPU0 to TPU3)
Low-power consumption function	Module-stop state can be set.

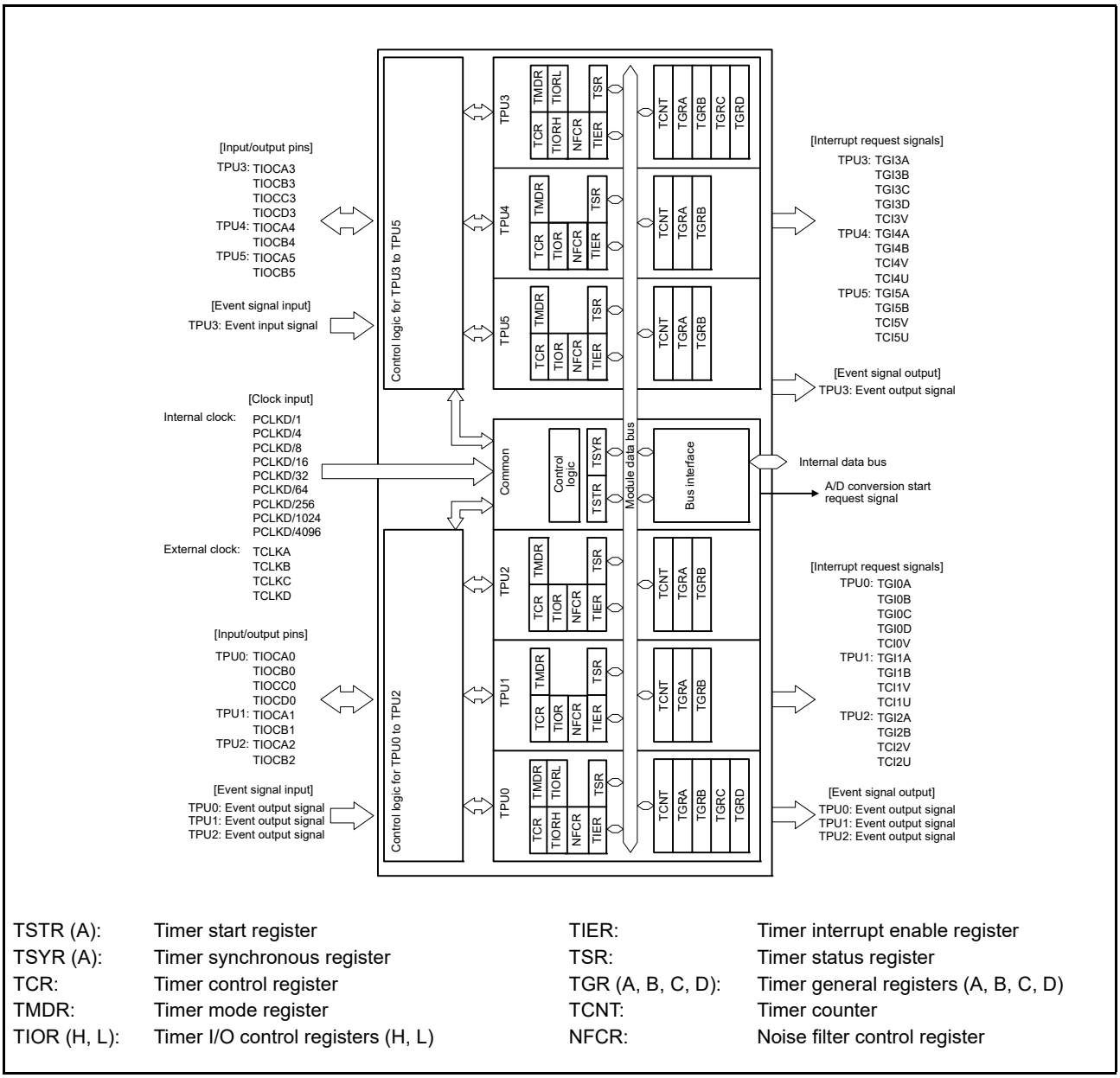
Table 18.2 TPU Functions (1 / 2)

Item		TPU0	TPU1	TPU2	TPU3	TPU4	TPU5
Count clock*1		PCLKD/1 PCLKD/4 PCLKD/16 PCLKD/64 TCLKA TCLKB TCLKC TCLKD	PCLKD/1 PCLKD/4 PCLKD/16 PCLKD/64 PCLKD/256 TCLKA TCLKB	PCLKD/1 PCLKD/4 PCLKD/16 PCLKD/64 PCLKD/1024 TCLKA TCLKB TCLKC	PCLKD/1 PCLKD/4 PCLKD/16 PCLKD/64 PCLKD/256 PCLKD/1024 PCLKD/4096 TCLKA	PCLKD/1 PCLKD/4 PCLKD/16 PCLKD/64 PCLKD/1024 TCLKA TCLKC	PCLKD/1 PCLKD/4 PCLKD/16 PCLKD/64 PCLKD/256 TCLKA TCLKC TCLKD
Timer general registers		TGRA TGRB TGRC*2 TGRD*2	TGRA TGRB	TGRA TGRB	TGRA TGRB TGRC*2 TGRD*2	TGRA TGRB	TGRA TGRB
I/O pins		TIOCA0 TIOCB0 TIOCC0 TIOCD0	TIOCA1 TIOCB1	TIOCA2 TIOCB2	TIOCA3 TIOCB3 TIOCC3 TIOCD3	TIOCA4 TIOCB4	TIOCA5 TIOCB5
Counter clear function		TGRy compare match or input capture	TGRy compare match or input capture	TGRy compare match or input capture	TGRy compare match or input capture	TGRy compare match or input capture	TGRy compare match or input capture
Compare match output	Low output	Possible	Possible	Possible	Possible	Possible	Possible
	High output	Possible	Possible	Possible	Possible	Possible	Possible
	Toggle output	Possible	Possible	Possible	Possible	Possible	Possible
Input capture function		Possible	Possible	Possible	Possible	Possible	Possible
Synchronous operation		Possible	Possible	Possible	Possible	Possible	Possible
PWM mode		Possible	Possible	Possible	Possible	Possible	Possible
Phase counting mode		Not possible	Possible	Possible	Not possible	Possible	Possible
Buffer operation		Possible	Not possible	Not possible	Possible	Not possible	Not possible
DMAC activation		TGRA/TGRB compare match or input capture	TGRA/TGRB compare match or input capture	TGRA/TGRB compare match or input capture	TGRA/TGRB compare match or input capture	TGRA/TGRB compare match or input capture	TGRA/TGRB compare match or input capture
A/D conversion start trigger		TGRA compare match or input capture	TGRA compare match or input capture	TGRA compare match or input capture	TGRA compare match or input capture	TGRA compare match or input capture	Not possible
Interrupt sources		5 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 • 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	4 sources • Compare match or input capture 4A • Compare match or input capture 4B • Overflow • Underflow	4 sources • Compare match or input capture 5A • Compare match or input capture 5B • Overflow • Underflow
Event linking (output)		5 sources • Compare match 0A • Compare match 0B • Compare match 0C • Compare match 0D • Overflow	4 sources • Compare match 1A • Compare match 1B • Overflow • Underflow	4 sources • Compare match 2A • Compare match 2B • Overflow • Underflow	5 sources • Compare match 3A • Compare match 3B • Compare match 3C • Compare match 3D • Overflow	Not possible	Not possible

Table 18.2 TPU Functions (2 / 2)

Item	TPU0	TPU1	TPU2	TPU3	TPU4	TPU5
Event linking (input)	• Starting counts • Clearing counts • Input capture operation (data is captured in TGRA)	• Starting counts • Clearing counts • Input capture operation (data is captured in TGRA)	• Starting counts • Clearing counts • Input capture operation (data is captured in TGRA)	• Starting counts • Clearing counts • Input capture operation (data is captured in TGRA)	Not possible	Not possible
Module-stop setting*3	MSTPCRA.MSTPCRA8 bit					

Note 1. The external count clocks not listed in this table are available in phase counting mode. For details, see section 18.3.6, Phase Counting Mode.
Note 2. TGRC and TGRD can be set as a buffer register.
Note 3. For details, see section 9, Low-Power Consumption Function.



TSTR (A):	Timer start register	TIER:	Timer interrupt enable register
TSYR (A):	Timer synchronous register	TSR:	Timer status register
TCR:	Timer control register	TGR (A, B, C, D):	Timer general registers (A, B, C, D)
TMDR:	Timer mode register	TCNT:	Timer counter
TIOR (H, L):	Timer I/O control registers (H, L)	NFCR:	Noise filter control register

Figure 18.1 Block Diagram of TPU

Table 18.3 lists the input/output pins of the TPU.

Table 18.3 Pin Configuration of TPU

Channel	Pin Name	I/O	Description
Common	TCLKA	Input	External clock A input pin (TPU1 and TPU5 phase counting mode A phase input)
	TCLKB	Input	External clock B input pin (TPU1 and TPU5 phase counting mode B phase input)
	TCLKC	Input	External clock C input pin (TPU2 and TPU4 phase counting mode A phase input)
	TCLKD	Input	External clock D input pin (TPU2 and TPU4 phase counting mode B phase input)
TPU0	TIOCA0	I/O	TPU0.TGRA input capture input/output compare output/PWM output pin
	TIOCB0	I/O	TPU0.TGRB input capture input/output compare output/PWM output pin
	TIOCC0	I/O	TPU0.TGRC input capture input/output compare output/PWM output pin
	TIOCD0	I/O	TPU0.TGRD input capture input/output compare output/PWM output pin
TPU1	TIOCA1	I/O	TPU1.TGRA input capture input/output compare output/PWM output pin
	TIOCB1	I/O	TPU1.TGRB input capture input/output compare output/PWM output pin
TPU2	TIOCA2	I/O	TPU2.TGRA input capture input/output compare output/PWM output pin
	TIOCB2	I/O	TPU2.TGRB input capture input/output compare output/PWM output pin
TPU3	TIOCA3	I/O	TPU3.TGRA input capture input/output compare output/PWM output pin
	TIOCB3	I/O	TPU3.TGRB input capture input/output compare output/PWM output pin
	TIOCC3	I/O	TPU3.TGRC input capture input/output compare output/PWM output pin
	TIOCD3	I/O	TPU3.TGRD input capture input/output compare output/PWM output pin
TPU4	TIOCA4	I/O	TPU4.TGRA input capture input/output compare output/PWM output pin
	TIOCB4	I/O	TPU4.TGRB input capture input/output compare output/PWM output pin
TPU5	TIOCA5	I/O	TPU5.TGRA input capture input/output compare output/PWM output pin
	TIOCB5	I/O	TPU5.TGRB input capture input/output compare output/PWM output pin

18.2 Register Descriptions

18.2.1 Timer Control Register (TCR)

TPU has 6 TCR registers in total (one register per channel).
TCR controls the TCNT counter for each channel. TCR settings should be made while TCNT counter operation is stopped.

Address(es): TPU0.TCR A008 0110h, TPU1.TCR A008 0120h, TPU2.TCR A008 0130h, TPU3.TCR A008 0140h, TPU4.TCR A008 0150h, TPU5.TCR A008 0160h

	b7	b6	b5	b4	b3	b2	b1	b0
	CCLR[2:0]			CKEG[1:0]		TPSC[2:0]		
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	TPSC[2:0]	Timer Prescaler Select	See Table 18.4 to Table 18.9.	R/W
b4, b3	CKEG[1:0]	Input Clock Edge Select	See Table 18.10.	R/W
b7 to b5	CCLR[2:0]*1	Counter Clear Source Select	See Table 18.11 and Table 18.12.	R/W

Note 1. Bit 7 in TPU1.TCR, TPU2.TCR, TPU4.TCR, and TPU5.TCR is reserved. This bit is read as 0. The write value should be 0.

TPSC[2:0] Bits (Timer Prescaler Select)

These bits select the TCNT counter clock. The clock source can be selected independently for each channel.
To select the external clock as the clock source, set the bit in the port direction register (PDR) for the corresponding pin to 0 (input port), and set the bit in the port mode register (PMR) to 1 (uses the pin as an I/O port for peripheral functions). For details, see section 16, I/O Ports.

CKEG[1:0] Bits (Input Clock Edge Select)

These bits select the input clock edge.
When the internal clock is counted using both edges, the input clock period is halved (e.g. Both edges of PCLKD/4 = PCLKD/2 rising edge).
Internal clock edge selection is valid when the input clock is PCLKD/4 or slower. This setting is ignored if the input clock is PCLKD/1, or when overflow/underflow of another channel is selected.

CCLR[2:0] Bits (Counter Clear Source Select)

These bits select the clear source of the TCNT counter.

Table 18.4 Bits TPSC[2:0] (TPU0)

Channel	Bits TPSC[2:0] b2	b1	b0	Description
TPU0	0	0	0	Internal clock: Counts on PCLKD/1
	0	0	1	Internal clock: Counts on PCLKD/4
	0	1	0	Internal clock: Counts on PCLKD/16
	0	1	1	Internal clock: Counts on PCLKD/64
	1	0	0	External clock • TPU0: Counts on TCLKA pin input
	1	0	1	External clock • TPU0: Counts on TCLKB pin input
	1	1	0	External clock • TPU0: Counts on TCLKC pin input
	1	1	1	External clock • TPU0: Counts on TCLKD pin input

Table 18.5 Bits TPSC[2:0] (TPU1)

Channel	Bits TPSC[2:0] b2	b1	b0	Description
TPU1	0	0	0	Internal clock: Counts on PCLKD/1
	0	0	1	Internal clock: Counts on PCLKD/4
	0	1	0	Internal clock: Counts on PCLKD/16
	0	1	1	Internal clock: Counts on PCLKD/64
	1	0	0	External clock • TPU1: Counts on TCLKA pin input
	1	0	1	External clock • TPU1: Counts on TCLKB pin input
	1	1	0	Internal clock: Counts on PCLKD/256
	1	1	1	• TPU1: Counts on TPU2.TCNT counter overflow/underflow

Note: This setting is invalid when TPU1 is in phase counting mode.

Table 18.6 Bits TPSC[2:0] (TPU2)

Channel	Bits TPSC[2:0] b2	b1	b0	Description
TPU2	0	0	0	Internal clock: Counts on PCLKD/1
	0	0	1	Internal clock: Counts on PCLKD/4
	0	1	0	Internal clock: Counts on PCLKD/16
	0	1	1	Internal clock: Counts on PCLKD/64
	1	0	0	External clock • TPU2: Counts on TCLKA pin input
	1	0	1	External clock • TPU2: Counts on TCLKB pin input
	1	1	0	External clock • TPU2: Counts on TCLKC pin input
	1	1	1	Internal clock: Counts on PCLKD/1024

Note: This setting is invalid when TPU2 is in phase counting mode.

Table 18.7 Bits TPSC[2:0] (TPU3)

Channel	Bits TPSC[2:0] b2	b1	b0	Description
TPU3	0	0	0	Internal clock: Counts on PCLKD/1
	0	0	1	Internal clock: Counts on PCLKD/4
	0	1	0	Internal clock: Counts on PCLKD/16
	0	1	1	Internal clock: Counts on PCLKD/64
	1	0	0	External clock • TPU3: Counts on TCLKA pin input
	1	0	1	Internal clock: Counts on PCLKD/1024
	1	1	0	Internal clock: Counts on PCLKD/256
	1	1	1	Internal clock: Counts on PCLKD/4096

Table 18.8 Bits TPSC[2:0] (TPU4)

Channel	Bits TPSC[2:0] b2	b1	b0	Description
TPU4	0	0	0	Internal clock: Counts on PCLKD/1
	0	0	1	Internal clock: Counts on PCLKD/4
	0	1	0	Internal clock: Counts on PCLKD/16
	0	1	1	Internal clock: Counts on PCLKD/64
	1	0	0	External clock • TPU4: Counts on TCLKA pin input
	1	0	1	External clock • TPU4: Counts on TCLKC pin input
	1	1	0	Internal clock: Counts on PCLKD/1024
	1	1	1	• TPU4: Counts on TPU5.TCNT counter overflow/underflow

Note: This setting is invalid when TPU4 is in phase counting mode.

Table 18.9 Bits TPSC[2:0] (TPU5)

Channel	Bits TPSC[2:0] b2	b1	b0	Description
TPU5	0	0	0	Internal clock: Counts on PCLKD/1
	0	0	1	Internal clock: Counts on PCLKD/4
	0	1	0	Internal clock: Counts on PCLKD/16
	0	1	1	Internal clock: Counts on PCLKD/64
	1	0	0	External clock • TPU5: Counts on TCLKA pin input
	1	0	1	External clock • TPU5: Counts on TCLKC pin input
	1	1	0	Internal clock: Counts on PCLKD/256
	1	1	1	External clock • TPU5: Counts on TCLKD pin input

Note: This setting is invalid when TPU5 is in phase counting mode.

Table 18.10 Bits CKEG[1:0]

Bits CKEG[1:0]		Input Clock	
b4	**b3**	**Internal Clock**	**External clock**
0	0	Counted at falling edge	Counted at rising edge
0	1	Counted at rising edge	Counted at falling edge
1	0	Counted at both edges	Counted at both edges
1	1	Counted at both edges	Counted at both edges

Table 18.11 Bits CCLR[2:0] (TPU0, TPU3)

	Bits CCLR[2:0]			
Channel	**b7**	**b6**	**b5**	**Description**
TPU0, TPU3	0	0	0	TCNT counter clearing disabled
	0	0	1	TCNT counter cleared by TGRA compare match/input capture
	0	1	0	TCNT counter cleared by TGRB compare match/input capture
	0	1	1	TCNT counter cleared by counter clearing for another channel performing synchronous clearing/synchronous operation*2
	1	0	0	TCNT counter clearing disabled
	1	0	1	TCNT counter cleared by TGRC compare match/input capture*1
	1	1	0	TCNT counter cleared by TGRD compare match/input capture*1
	1	1	1	TCNT counter cleared by counter clearing for another channel performing synchronous clearing/synchronous operation*2

Note 1. When TGRC or TGRD is used as a buffer register, TCNT counter is not cleared because the buffer register setting has priority, and compare match/input capture does not occur.
Note 2. Synchronous operation is selected by setting the TSYRA.SYNCj bit (j = 0, 3) to 1.

Table 18.12 Bits CCLR[2:0] (TPU1, TPU2, TPU4, TPU5)

	Bits CCLR[2:0]*1			
Channel	**b7**	**b6**	**b5**	**Description**
TPU1, TPU2, TPU4, TPU5	0	0	0	TCNT counter clearing disabled
	0	0	1	TCNT counter cleared by TGRA compare match/input capture
	0	1	0	TCNT counter cleared by TGRB compare match/input capture
	0	1	1	TCNT counter cleared by counter clearing for another channel performing synchronous clearing/synchronous operation*2
	1	0	0	Setting prohibited
	1	0	1	Setting prohibited
	1	1	0	Setting prohibited
	1	1	1	Setting prohibited

Note 1. Bit 7 in TPU1.TCR, TPU2.TCR, TPU4.TCR, and TPU5.TCR is reserved. This bit is read as 0. The write value should be 0.
Note 2. Synchronous operation is selected by setting the TSYRA.SYNCj bit (j = 1, 2, 4, 5) to 1.

18.2.2 Timer Mode Register (TMDR)

TMDR sets the operating mode of each channel.
TPU has 6 TMDR registers in total (one register per channel). TMDR settings should be made while TCNT counter operation is stopped.

Address(es): TPU0.TMDR A008 0111h, TPU1.TMDR A008 0121h, TPU2.TMDR A008 0131h, TPU3.TMDR A008 0141h, TPU4.TMDR A008 0151h, TPU5.TMDR A008 0161h

	b7	b6	b5	b4	b3	b2	b1	b0
	ICSELD	ICSELB	BFB	BFA	MD[3:0]			
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b3 to b0	MD[3:0]	Mode Select	b3*1 b0 0 0 0 0: Normal operation 0 0 1 0: PWM mode 1 0 0 1 1: PWM mode 2 0 1 0 0: Phase counting mode 1*2 0 1 0 1: Phase counting mode 2*2 0 1 1 0: Phase counting mode 3*2 0 1 1 1: Phase counting mode 4*2 Settings other than above are prohibited.	R/W
b4	BFA*3	Buffer Operation A	0: TPUm.TGRA operates normally 1: TPUm.TGRA and TPUm.TGRC used together for buffer operation (m = 0, 3)	R/W
b5	BFB*4	Buffer Operation B	0: TGRB operates normally 1: TGRB and TGRD used together for buffer operation (m = 0, 3)	R/W
b6	ICSELB	TGRB Input Capture Input Select	0: Input capture input source is TIOCBn pin 1: Input capture input source is TIOCAn pin (n = 0 to 5)	R/W
b7	ICSELD*4	TGRD Input Capture Input Select	0: Input capture input source is TIOCDn pin 1: Input capture input source is TIOCCn pin (n = 0, 3)	R/W

Note 1. Bit 3 is reserved. This bit is read as 0. The write value should be 0.
Note 2. Phase counting mode cannot be set for TPU0 and TPU3. A 0 should be written to bit 2 for them.
Note 3. Bit 4 of TPU1, TPU2, TPU4, and TPU5 that do not have TGRC is reserved. This bit is read as 0. The write value should be 0.
Note 4. Bits 5 and 7 of TPU1, TPU2, TPU4, and TPU5 that do not have TGRD are reserved. These bits are read as 0. The write value should be 0.

MD[3:0] Bits (Mode Select)

Specifies the operating mode of the timer.

BFA Bit (Buffer Operation A)

Specifies whether TPUm.TGRA is to normally operate, or TPUm.TGRA and TPUm.TGRC are to be used together for buffer operation (m = 0, 3).
When TGRC is used as a buffer register, TGRC input capture/output compare is not generated.

BFB Bit (Buffer Operation B)

Specifies whether TPUm.TGRB is to normally operate, or TPUm.TGRB and TPUm.TGRD are to be used together for buffer operation (m = 0, 3).
When TGRD is used as a buffer register, TGRD input capture/output compare is not generated.

ICSELB Bit (TGRB Input Capture Input Select)

Selects the input capture input for TPUm.TGRB (m = 0 to 5). This function allows measurement of high-level width and period of the input pulse on a TIOCAn input pin.

ICSELD Bit (TGRD Input Capture Input Select)

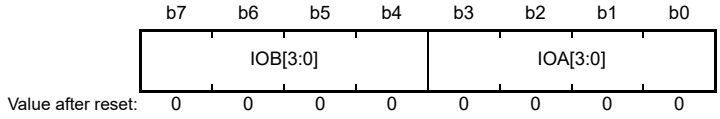
Selects the input capture input for TPUm.TGRD (m = 0, 3).
This function allows measurement of high-level width and period of the input pulse on a TIOCCn input pin.

18.2.3 Timer I/O Control Register (TIORH, TIORL, TIOR)

TPU has two TIORH registers (one for TPU0 and TPU3), two TIORL registers (one for TPU0 and TPU3), and four TIOR registers (one for TPU1, TPU2, TPU4, and TPU5). Thus, the TPU has 8 timer I/O control registers in total.
TIORH, TIORL, and TIOR control registers TGRA, TGRB, TGRC, and TGRD.
Note that TIORH, TIORL, and TIOR are affected by the TMDR setting.
For details, see Table 18.13 to Table 18.20.
The initial output specified by TIORH, TIORL, and TIOR is valid when the counter is stopped (the TPUA.TSTRA.CSTj bit (j = 0 to 5) is cleared to 0). In PWM mode 2, the output at the time when the TCNT counter is cleared to 0 is specified as the initial output.
When buffer operation has been selected for register TGRC or TGRD, the settings of the corresponding set of IOC[3:0] or IOD[3:0] bits becomes ineffective, and the TGRC or TGRD register simply operates as a buffer.
To specify the input capture pin in TIORH, TIORL, or TIOR, set the bit in the port direction register (PDR) for the corresponding pin to 0 (input port), and set the bit in the port mode register (PMR) to 1 (uses the pin as an I/O port for peripheral functions). For details, see section 16, I/O Ports.

- TPU0.TIORH, TPU1.TIOR, TPU2.TIOR, TPU3.TIORH, TPU4.TIOR, TPU5.TIOR

Address(es): TPU0.TIORH A008 0112h, TPU1.TIOR A008 0122h, TPU2.TIOR A008 0132h, TPU3.TIORH A008 0142h, TPU4.TIOR A008 0152h, TPU5.TIOR A008 0162h

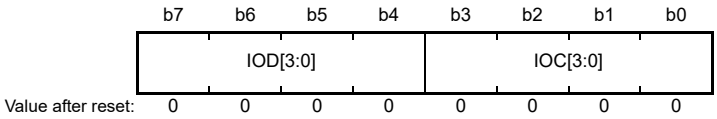


Bit	Symbol	Bit Name	Description	R/W
b3 to b0	IOA[3:0]	TGRA Control	See Table 18.13 to Table 18.18.*1	R/W
b7 to b4	IOB[3:0]	TGRB Control	See Table 18.13 to Table 18.18.*1	R/W

Note 1. If the IOn[3:0] (n = A, B) bits are changed to an “output prohibited” setting (0000b or 0100b) while the output of 0, output of 1, or toggling of the output in response to compare matches is in progress, the output becomes high impedance.

- TPU0.TIORL, TPU3.TIORL

Address(es): TPU0.TIORL A008 0113h, TPU3.TIORL A008 0143h



Bit	Symbol	Bit Name	Description	R/W
b3 to b0	IOC[3:0]	TGRC Control	See Table 18.19 and Table 18.20.*1	R/W
b7 to b4	IOD[3:0]	TGRD Control	See Table 18.19 and Table 18.20.*1	R/W

Note 1. If the IOn[3:0] (n = C, D) bits are changed to an “output prohibited” setting (0000b or 0100b) while the output of 0, output of 1, or toggling of the output in response to compare matches is in progress, the output becomes high impedance.

IOA[3:0] Bits (TGRA Control)

Select the function of TPUm.TGRA (m = 0 to 5).

IOB[3:0] Bits (TGRB Control)

Select the function of TPUm.TGRB (m = 0 to 5).

IOC[3:0] Bits (TGRC Control)

Select the function of TPUm.TGRC (m = 0, 3).

IOD[3:0] Bits (TGRD Control)

Select the function of TPUm.TGRD (m = 0, 3).

Table 18.13 TPU0.TIORH

Bits IOA[3:0]				Description	
b3	b2	b1	b0	TPUm.TGRA (m = 0) Function	TIOCAn Pin (n = 0) Function and Related Issue
0	0	0	0	Output compare register	Output disabled
0	0	0	1		Initial output is low output; low output at compare match
0	0	1	0		Initial output is low output; high output at compare match
0	0	1	1		Initial output is low output; toggle output at compare match
0	1	0	0		Output disabled
0	1	0	1		Initial output is high output; low output at compare match
0	1	1	0		Initial output is high output; high output at compare match
0	1	1	1		Initial output is high output; toggle output at compare match
1	0	0	0	Input capture register	Capture input source is TIOCAn pin; input capture at rising edge
1	0	0	1		Capture input source is TIOCAn pin; input capture at falling edge
1	0	1	x		Capture input source is TIOCAn pin; input capture at both edges
1	1	x	x		• TPU0 Capture input source is TPU1 count clock Input capture at TPU1.TCNT count-up/count-down*1

Bits IOB[3:0]				Description	
b7	b6	b5	b4	TPUm.TGRB (m = 0) Function	TIOCBn Pin (n = 0) Function and Related Issue
0	0	0	0	Output compare register	Output disabled
0	0	0	1		Initial output is low output; low output at compare match
0	0	1	0		Initial output is low output; high output at compare match
0	0	1	1		Initial output is low output; toggle output at compare match
0	1	0	0		Output disabled
0	1	0	1		Initial output is high output; low output at compare match
0	1	1	0		Initial output is high output; high output at compare match
0	1	1	1		Initial output is high output; toggle output at compare match
1	0	0	0	Input capture register	Capture input source is TIOCBn or TIOCAn pin*2; input capture at rising edge
1	0	0	1		Capture input source is TIOCBn or TIOCAn pin*2; input capture at falling edge
1	0	1	x		Capture input source is TIOCBn or TIOCAn pin*2; input capture at both edges
1	1	x	x		• TPU0 Capture input source is TPU1 count clock Input capture at TPU1.TCNT count-up/count-down*1

x: Don't care

Note 1. When the TPUm.TCR.TPSC[2:0] bits are set to 000b and PCLKD/1 is used as the TPUm.TCNT count clock, this setting is invalid and input capture is not generated (m = 1).

Note 2. Selected by the TPUm.TMDR.ICSELB bit (m = 0).

Table 18.14 TPU1.TIOR

Bits IOA[3:0]				Description	
b3	b2	b1	b0	TPUm.TGRA (m = 1) Function	TIOCAn Pin (n = 1) Function and Related Issue
0	0	0	0	Output compare register	Output disabled
0	0	0	1		Initial output is low output; low output at compare match
0	0	1	0		Initial output is low output; high output at compare match
0	0	1	1		Initial output is low output; toggle output at compare match
0	1	0	0		Output disabled
0	1	0	1		Initial output is high output; low output at compare match
0	1	1	0		Initial output is high output; high output at compare match
0	1	1	1		Initial output is high output; toggle output at compare match
1	0	0	0	Input capture register	Capture input source is TIOCAn pin; input capture at rising edge
1	0	0	1		Capture input source is TIOCAn pin; input capture at falling edge
1	0	1	x		Capture input source is TIOCAn pin; input capture at both edges
1	1	x	x		• TPU1 Capture input source is TPU0.TGRA compare match/input capture Input capture at generation of TPU0.TGRA compare match/input capture

Bits IOB[3:0]				Description	
b7	b6	b5	b4	TPUm.TGRB (m = 1) Function	TIOCBn Pin (n = 1) Function and Related Issue
0	0	0	0	Output compare register	Output disabled
0	0	0	1		Initial output is low output; low output at compare match
0	0	1	0		Initial output is low output; high output at compare match
0	0	1	1		Initial output is low output; toggle output at compare match
0	1	0	0		Output disabled
0	1	0	1		Initial output is high output; low output at compare match
0	1	1	0		Initial output is high output; high output at compare match
0	1	1	1		Initial output is high output; toggle output at compare match
1	0	0	0	Input capture register	Capture input source is TIOCBn or TIOCAn pin*1; input capture at rising edge
1	0	0	1		Capture input source is TIOCBn or TIOCAn pin*1; input capture at falling edge
1	0	1	x		Capture input source is TIOCBn or TIOCAn pin*1; input capture at both edges
1	1	x	x		• TPU1 Capture input source is TPU0.TGRC compare match/input capture Input capture at generation of TPU0.TGRC compare match/input capture

x: Don’t care

Note 1. Selected by the TPUm.TMDR.ICSELB bit (m = 1).

Table 18.15 TPU2.TIOR

Bits IOA[3:0]				Description	
b3	b2	b1	b0	TPUm.TGRA (m = 2) Function	TIOCAn Pin (n = 2) Function and Related Issue
0	0	0	0	Output compare register	Output disabled
0	0	0	1		Initial output is low output; low output at compare match
0	0	1	0		Initial output is low output; high output at compare match
0	0	1	1		Initial output is low output; toggle output at compare match
0	1	0	0		Output disabled
0	1	0	1		Initial output is high output; low output at compare match
0	1	1	0		Initial output is high output; high output at compare match
0	1	1	1		Initial output is high output; toggle output at compare match
1	x	0	0	Input capture register	Capture input source is TIOCAn pin; input capture at rising edge
1	x	0	1		Capture input source is TIOCAn pin; input capture at falling edge
1	x	1	x		Capture input source is TIOCAn pin; input capture at both edges

Bits IOB[3:0]				Description	
b7	b6	b5	b4	TPUm.TGRB (m = 2) Function	TIOCBn Pin (n = 2) Function and Related Issue
0	0	0	0	Output compare register	Output disabled
0	0	0	1		Initial output is low output; low output at compare match
0	0	1	0		Initial output is low output; high output at compare match
0	0	1	1		Initial output is low output; toggle output at compare match
0	1	0	0		Output disabled
0	1	0	1		Initial output is high output; low output at compare match
0	1	1	0		Initial output is high output; high output at compare match
0	1	1	1		Initial output is high output; toggle output at compare match
1	x	0	0	Input capture register	Capture input source is TIOCBn or TIOCAn pin*1; input capture at rising edge
1	x	0	1		Capture input source is TIOCBn or TIOCAn pin*1; input capture at falling edge
1	x	1	x		Capture input source is TIOCBn or TIOCAn pin*1; input capture at both edges

x: Don’t care

Note 1. Selected by the TPUm.TMDR.ICSELB bit (m = 2).

Table 18.16 TPU3.TIORH

Bits IOA[3:0]				Description	
b3	b2	b1	b0	TPUm.TGRA (m = 3) Function	TIOCAn Pin (n = 3) Function and Related Issue
0	0	0	0	Output compare register	Output disabled
0	0	0	1		Initial output is low output; low output at compare match
0	0	1	0		Initial output is low output; high output at compare match
0	0	1	1		Initial output is low output; toggle output at compare match
0	1	0	0		Output disabled
0	1	0	1		Initial output is high output; low output at compare match
0	1	1	0		Initial output is high output; high output at compare match
0	1	1	1		Initial output is high output; toggle output at compare match
1	0	0	0	Input capture register	Capture input source is TIOCAn pin; input capture at rising edge
1	0	0	1		Capture input source is TIOCAn pin; input capture at falling edge
1	0	1	x		Capture input source is TIOCAn pin; input capture at both edges
1	1	x	x		• TPU3 Capture input source is TPU4 count clock Input capture at TPU4.TCNT count-up/count-down*1

Bits IOB[3:0]				Description	
b7	b6	b5	b4	TPUm.TGRB (m = 3) Function	TIOCBn Pin (n = 3) Function and Related Issue
0	0	0	0	Output compare register	Output disabled
0	0	0	1		Initial output is low output; low output at compare match
0	0	1	0		Initial output is low output; high output at compare match
0	0	1	1		Initial output is low output; toggle output at compare match
0	1	0	0		Output disabled
0	1	0	1		Initial output is high output; low output at compare match
0	1	1	0		Initial output is high output; high output at compare match
0	1	1	1		Initial output is high output; toggle output at compare match
1	0	0	0	Input capture register	Capture input source is TIOCBn or TIOCAn pin*2; input capture at rising edge
1	0	0	1		Capture input source is TIOCBn or TIOCAn pin*2; input capture at falling edge
1	0	1	x		Capture input source is TIOCBn or TIOCAn pin*2; input capture at both edges
1	1	x	x		• TPU3 Capture input source is TPU4 count clock Input capture at TPU4.TCNT count-up/count-down*1

x: Don't care

Note 1. When the TPUm.TCR.TPSC[2:0] bits are set to 000b and PCLKD/1 is used as the TPUm.TCNT count clock, this setting is invalid and input capture is not generated (m = 4).

Note 2. Selected by the TPUm.TMDR.ICSELB bit (m = 3).

Table 18.17 TPU4.TIOR

Bits IOA[3:0]				Description	
b3	b2	b1	b0	TPUm.TGRA (m = 4) Function	TIOCAn Pin (n = 4) Function and Related Issue
0	0	0	0	Output compare register	Output disabled
0	0	0	1		Initial output is low output; low output at compare match
0	0	1	0		Initial output is low output; high output at compare match
0	0	1	1		Initial output is low output; toggle output at compare match
0	1	0	0		Output disabled
0	1	0	1		Initial output is high output; low output at compare match
0	1	1	0		Initial output is high output; high output at compare match
0	1	1	1		Initial output is high output; toggle output at compare match
1	0	0	0	Input capture register	Capture input source is TIOCAn pin; input capture at rising edge
1	0	0	1		Capture input source is TIOCAn pin; input capture at falling edge
1	0	1	x		Capture input source is TIOCAn pin; input capture at both edges
1	1	x	x		• TPU4 Capture input source is TPU3.TGRA compare match/input capture Input capture at generation of TPU3.TGRA compare match/input capture

Bits IOB[3:0]				Description	
b7	b6	b5	b4	TPUm.TGRB (m = 4) Function	TIOCBn Pin (n = 4) Function and Related Issue
0	0	0	0	Output compare register	Output disabled
0	0	0	1		Initial output is low output; low output at compare match
0	0	1	0		Initial output is low output; high output at compare match
0	0	1	1		Initial output is low output; toggle output at compare match
0	1	0	0		Output disabled
0	1	0	1		Initial output is high output; low output at compare match
0	1	1	0		Initial output is high output; high output at compare match
0	1	1	1		Initial output is high output; toggle output at compare match
1	0	0	0	Input capture register	Capture input source is TIOCBn or TIOCAn pin*1; input capture at rising edge
1	0	0	1		Capture input source is TIOCBn or TIOCAn pin*1; input capture at falling edge
1	0	1	x		Capture input source is TIOCBn or TIOCAn pin*1; input capture at both edges
1	1	x	x		• TPU4 Capture input source is TPU3.TGRC compare match/input capture Input capture at generation of TPU3.TGRC compare match/input capture

x: Don’t care

Note 1. Selected by the TPUm.TMDR.ICSELB bit (m = 4).

Table 18.18 TPU5.TIOR

Bits IOA[3:0]				Description	
b3	**b2**	**b1**	**b0**	**TPUm.TGRA (m = 5) Function**	**TIOCAn Pin (n = 5) Function and Related Issue**
0	0	0	0	Output compare register	Output disabled
0	0	0	1		Initial output is low output; low output at compare match
0	0	1	0		Initial output is low output; high output at compare match
0	0	1	1		Initial output is low output; toggle output at compare match
0	1	0	0		Output disabled
0	1	0	1		Initial output is high output; low output at compare match
0	1	1	0		Initial output is high output; high output at compare match
0	1	1	1		Initial output is high output; toggle output at compare match
1	x	0	0	Input capture register	Capture input source is TIOCAn pin; input capture at rising edge
1	x	0	1		Capture input source is TIOCAn pin; input capture at falling edge
1	x	1	x		Capture input source is TIOCAn pin; input capture at both edges

Bits IOB[3:0]				Description	
b7	**b6**	**b5**	**b4**	**TPUm.TGRB (m = 5) Function**	**TIOCBn Pin (n = 5) Function and Related Issue**
0	0	0	0	Output compare register	Output disabled
0	0	0	1		Initial output is low output; low output at compare match
0	0	1	0		Initial output is low output; high output at compare match
0	0	1	1		Initial output is low output; toggle output at compare match
0	1	0	0		Output disabled
0	1	0	1		Initial output is high output; low output at compare match
0	1	1	0		Initial output is high output; high output at compare match
0	1	1	1		Initial output is high output; toggle output at compare match
1	x	0	0	Input capture register	Capture input source is TIOCBn or TIOCAn pin*1; input capture at rising edge
1	x	0	1		Capture input source is TIOCBn or TIOCAn pin*1; input capture at falling edge
1	x	1	x		Capture input source is TIOCBn or TIOCAn pin*1; input capture at both edges

x: Don't care

Note 1. Selected by the ITPUm.TMDR.ICSELB bit (m = 5).

Table 18.19 TPU0.TIORL

Bits IOC[3:0]				Description	
b3	b2	b1	b0	TPUm.TGRC (m = 0) Function	TIOCCn Pin (n = 0) Function and Related Issue
0	0	0	0	Output compare register*1	Output disabled
0	0	0	1		Initial output is low output; low output at compare match
0	0	1	0		Initial output is low output; high output at compare match
0	0	1	1		Initial output is low output; toggle output at compare match
0	1	0	0		Output disabled
0	1	0	1		Initial output is high output; low output at compare match
0	1	1	0		Initial output is high output; high output at compare match
0	1	1	1		Initial output is high output; toggle output at compare match
1	0	0	0	Input capture register*1	Capture input source is TIOCCn pin; input capture at rising edge
1	0	0	1		Capture input source is TIOCCn pin; input capture at falling edge
1	0	1	x		Capture input source is TIOCCn pin; input capture at both edges
1	1	x	x		• TPU0 Capture input source is TPU1 count clock Input capture at TPU1.TCNT count-up/count-down*3

Bits IOD[3:0]				Description	
b7	b6	b5	b4	TPUm.TGRD (m = 0) Function	TIOCDn Pin (n = 0) Function and Related Issue
0	0	0	0	Output compare register*2	Output disabled
0	0	0	1		Initial output is low output; low output at compare match
0	0	1	0		Initial output is low output; high output at compare match
0	0	1	1		Initial output is low output; toggle output at compare match
0	1	0	0		Output disabled
0	1	0	1		Initial output is high output; low output at compare match
0	1	1	0		Initial output is high output; high output at compare match
0	1	1	1		Initial output is high output; toggle output at compare match
1	0	0	0	Input capture register*2	Capture input source is TIOCDn or TIOCCn pin*4; input capture at rising edge
1	0	0	1		Capture input source is TIOCDn or TIOCCn pin*4; input capture at falling edge
1	0	1	x		Capture input source is TIOCDn or TIOCCn pin*4; input capture at both edges
1	1	x	x		• TPU0 Capture input source is TPU1 count clock Input capture at TPU1.TCNT count-up/count-down*3

x: Don't care

Note 1. When the TPUm.TMDR.BFA bit is set to 1 (TPUm.TGRA and TPUm.TGRC are used for buffer operation) and TPUm.TGRC is used as a buffer register, this setting is invalid and input capture/output compare is not generated (m = 0).

Note 2. When the TPUm.TMDR.BFB bit is set to 1 (TPUm.TGRB and TPUm.TGRD are used for buffer operation) and TPUm.TGRD is used as a buffer register, this setting is invalid and input capture/output compare is not generated (m = 0).

Note 3. When the TPUm.TCR.TPSC[2:0] bits are set to 000b and PCLKD/1 is used as the TPUm.TCNT count clock, this setting is invalid and input capture is not generated. (m = 1)

Note 4. Selected by the TPUm.TMDR.ICSELD bit (m = 0).

Table 18.20 TPU3.TIORL

Bits IOC[3:0]				Description	
b3	b2	b1	b0	TPUm.TGRC (m = 3) Function	TIOCCn Pin (n = 3) Function and Related Issue
0	0	0	0	Output compare register*1	Output disabled
0	0	0	1		Initial output is low output; low output at compare match
0	0	1	0		Initial output is low output; high output at compare match
0	0	1	1		Initial output is low output; toggle output at compare match
0	1	0	0		Output disabled
0	1	0	1		Initial output is high output; low output at compare match
0	1	1	0		Initial output is high output; high output at compare match
0	1	1	1		Initial output is high output; toggle output at compare match
1	0	0	0	Input capture register*1	Capture input source is TIOCCn pin; input capture at rising edge
1	0	0	1		Capture input source is TIOCCn pin; input capture at falling edge
1	0	1	x		Capture input source is TIOCCn pin; input capture at both edges
1	1	x	x		• TPU3 Capture input source is TPU4 count clock Input capture at TPU4.TCNT count-up/count-down*3

Bits IOD[3:0]				Description	
b7	b6	b5	b4	TPUm.TGRD (m = 3) Function	TIOCDn Pin (n = 3) Function and Related Issue
0	0	0	0	Output compare register*2	Output disabled
0	0	0	1		Initial output is low output; low output at compare match
0	0	1	0		Initial output is low output; high output at compare match
0	0	1	1		Initial output is low output; toggle output at compare match
0	1	0	0		Output disabled
0	1	0	1		Initial output is high output; low output at compare match
0	1	1	0		Initial output is high output; high output at compare match
0	1	1	1		Initial output is high output; toggle output at compare match
1	0	0	0	Input capture register*2	Capture input source is TIOCDn or TIOCCn pin*4; input capture at rising edge
1	0	0	1		Capture input source is TIOCDn or TIOCCn pin*4; input capture at falling edge
1	0	1	x		Capture input source is TIOCDn or TIOCCn pin*4; input capture at both edges
1	1	X	x		• TPU3 Capture input source is TPU4 count clock Input capture at TPU4.TCNT count-up/count-down*3

x: Don't care

Note 1. When the TPUm.TMDR.BFA bit is set to 1 (TPUm.TGRA and TPUm.TGRC are used for buffer operation) and TPUm.TGRC is used as a buffer register, this setting is invalid and input capture/output compare is not generated (m = 3).

Note 2. When the TPUm.TMDR.BFB bit is set to 1 (TPUm.TGRB and TPUm.TGRD are used for buffer operation) and TPUm.TGRD is used as a buffer register, this setting is invalid and input capture/output compare is not generated (m = 3).

Note 3. When the TPUm.TCR.TPSC[2:0] bits are set to 000b and PCLKD/1 is used as the TPUm.TCNT count clock, this setting is invalid and input capture is not generated (m = 4).

Note 4. Selected by the TPUm.TMDR.ICSELD bit (m = 3).

18.2.4 Timer Interrupt Enable Register (TIER)

TPU has 6 TIER registers in total (one register per channel).

TPUAm.TIER controls enabling or disabling of interrupt requests to individual channels (m = 0 to 5).

Address(es): TPU0.TIER A008 0114h, TPU1.TIER A008 0124h, TPU2.TIER A008 0134h, TPU3.TIER A008 0144h, TPU4.TIER A008 0154h, TPU5.TIER A008 0164h

	b7	b6	b5	b4	b3	b2	b1	b0
	TTGE	—	TCIEU	TCIEV	TGIED	TGIEC	TGIEB	TGIEA
Value after reset:	0	1	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	TGIEA	TGRA Interrupt Enable	0: Interrupt requests (TGImA) disabled 1: Interrupt requests (TGImA) enabled (m = 0 to 5)	R/W
b1	TGIEB	TGRB Interrupt Enable	0: Interrupt requests (TGImB) disabled 1: Interrupt requests (TGImB) enabled (m = 0 to 5)	R/W
b2	TGIEC*1	TGRC Interrupt Enable	0: Interrupt requests (TGImC) disabled 1: Interrupt requests (TGImC) enabled (m = 0, 3)	R/W
b3	TGIED*1	TGRD Interrupt Enable	0: Interrupt requests (TGImD) disabled 1: Interrupt requests (TGImD) enabled (m = 0, 3)	R/W
b4	TCIEV	Overflow Interrupt Enable	0: Interrupt requests (TCImV) disabled 1: Interrupt requests (TCImV) enabled (m = 0 to 5)	R/W
b5	TCIEU*2	Underflow Interrupt Enable	0: Interrupt requests (TCImU) disabled 1: Interrupt requests (TCImU) enabled (m = 1, 2, 4, 5)	R/W
b6	—	Reserved	This bit is read as 1. The write value should be 1.	R/W
b7	TTGE*3	A/D Conversion Start Request Enable	0: A/D conversion start request generation disabled 1: A/D conversion start request generation enabled	R/W

Note 1. Bits 3 and 2 in TPU1.TIER, TPU2.TIER, TPU4.TIER, and TPU5.TIER are reserved. These bits are read as 0. The write value should be 0.
Note 2. Bit 5 in TPU0.TIER and TPU3.TIER is reserved. This bit is read as 0. The write value should be 0.
Note 3. Bit 7 in TPU5.TIER is reserved. This bit is read as 0. The write value should be 0.

TGIEA Bit (TGRA Interrupt Enable)

This bit enables or disables interrupt TGImA (m = 0 to 5).

TGIEB Bit (TGRB Interrupt Enable)

This bit enables or disables interrupt TGImB (m = 0 to 5).

TGIEC Bit (TGRC Interrupt Enable)

This bit enables or disables interrupt TGImC (m = 0, 3).

TGIED Bit (TGRD Interrupt Enable)

This bit enables or disables interrupt TGImD (m = 0, 3).

TCIEV Bit (Overflow Interrupt Enable)

This bit enables or disables interrupt TCImV (m = 0 to 5).

TCIEU Bit (Underflow Interrupt Enable)

This bit enables or disables interrupt TCImU (m = 1, 2, 4, 5).

TTGE Bit (A/D Conversion Start Request Enable)

Enables/disables generation of A/D conversion start requests by TPUm.TGRA (m = 0 to 4) input capture/compare match.

18.2.5 Timer Status Register (TSR)

TPU has 6 TSR registers in total (one register per channel).

TPUm.TSR indicates the status of individual channels, and the count direction of TPUm.TCNT counter (m = 0 to 5).

Address(es): TPU0.TSR A008 0115h, TPU1.TSR A008 0125h, TPU2.TSR A008 0135h, TPU3.TSR A008 0145h, TPU4.TSR A008 0155h, TPU5.TSR A008 0165h

	b7	b6	b5	b4	b3	b2	b1	b0
	TCFD	—	TCFU	TCFV	TGFD	TGFC	TGFB	TGFA
Value after reset:	1	1	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	TGFA	Input Capture/Output Compare Flag A	0: Input capture to TPUm.TGRA or compare match with TPUm.TGRA has not occurred. 1: Input capture to TPUm.TGRA or compare match with TPUm.TGRA has occurred. (m = 0 to 5)	R/W*2
b1	TGFB	Input Capture/Output Compare Flag B	0: Input capture to TPUm.TGRB or compare match with TPUm.TGRB has not occurred. 1: Input capture to TPUm.TGRB or compare match with TPUm.TGRB has occurred. (m = 0 to 5)	R/W*2
b2	TGFC*4	Input Capture/Output Compare Flag C	0: Input capture to TPUm.TGRC or compare match with TPUm.TGRC has not occurred. 1: Input capture to TPUm.TGRC or compare match with TPUm.TGRC has occurred. (m = 0, 3)	R/W*2
b3	TGFD*4	Input Capture/Output Compare Flag D	0: Input capture to TPUm.TGRD or compare match with TPUm.TGRD has not occurred. 1: Input capture to TPUm.TGRD or compare match with TPUm.TGRD has occurred. (m = 0, 3)	R/W*2
b4	TCFV	Overflow Flag	0: TPUm.TCNT has not overflowed. 1: TPUm.TCNT has overflowed. (m = 0 to 5)	R/W*2
b5	TCFU*3	Underflow Flag	0: TPUm.TCNT has not underflowed. 1: TPUm.TCNT has underflowed. (m = 1, 2, 4, 5)	R/W*2
b6	—	Reserved	This bit is read as 1.	R
b7	TCFD*1	Counting Direction Flag	0: TPUm.TCNT counter counts down. 1: TPUm.TCNT counter counts up. (m = 1, 2, 4, 5)	R

Note 1. Bit 7 in TPU0.TSR and TPU3.TSR is reserved. The bit is read as 1. The write value should be 1.
Note 2. Only writing 0 to this bit is possible; this clears the flag.
Note 3. Bit 5 in TPU0.TSR and TPU3.TSR is reserved. The bit is read as 0. The write value should be 0.
Note 4. Bits 2 and 3 in TPU1.TSR, TPU2.TSR, TPU4.TSR, and TPU5.TSR are reserved. These bits are read as 0. The write value should be 0.

TGFA Flag (Input Capture/Output Compare Flag A)

This status flag indicates that input capture to TPUm.TGRA or compare match with TPUm.TGRA (m = 0 to 5) has occurred.

[Setting conditions]

- When TPUm.TGRA holds the value for comparison in output-compare operations, TPUm.TCNT matches TPUm.TGRA.
- When TPUm.TGRA is serving as an input-capture register, the input-capture signal has caused transfer of the value in TPUm.TCNT to TPUm.TGRA.

[Clearing condition]

- Writing 0 to TGFA after reading its value as 1.

TGFB Flag (Input Capture/Output Compare Flag B)

This status flag indicates that input capture to TPUm.TGRB or compare match with TPUm.TGRB (m = 0 to 5) has occurred.

[Setting conditions]

- When TPUm.TGRB holds the value for comparison in output-compare operations, TPUm.TCNT matches TPUm.TGRB.
- When TPUm.TGRB is serving as an input-capture register, the input-capture signal has caused transfer of the value in TPUm.TCNT to TPUm.TGRB.

[Clearing condition]

- Writing 0 to TGFB after reading its value as 1.

TGFC Flag (Input Capture/Output Compare Flag C)

This status flag indicates that input capture to TPUm.TGRC or compare match with TPUm.TGRC (m = 0, 3) has occurred.

[Setting conditions]

- When TPUm.TGRC holds the value for comparison in output-compare operations, TPUm.TCNT matches TPUm.TGRC.
- When TPUm.TGRC is serving as an input-capture register, the input-capture signal has caused transfer of the value in TPUm.TCNT to TPUm.TGRC.

[Clearing condition]

- Writing 0 to TGFC after reading its value as 1.

TGFD Flag (Input Capture/Output Compare Flag D)

This status flag indicates that input capture to TPUm.TGRD or compare match with TPUm.TGRD (m = 0, 3) has occurred.

[Setting conditions]

- When TPUm.TGRD holds the value for comparison in output-compare operations, TPUm.TCNT matches TPUm.TGRD.
- When TPUm.TGRD is serving as an input-capture register, the input-capture signal has caused transfer of the value in TPUm.TCNT to TPUm.TGRD.

[Clearing condition]

- Writing 0 to TGFD after reading its value as 1.

TCFV Flag (Overflow Flag)

This status flag indicates an overflow of TPUm.TCNT (m = 0 to 5).

[Setting condition]

- Overflow of the value in TPUm.TCNT (TPUm.TCNT counted from FFFFh to 0000h).

[Clearing condition]

- Writing 0 to TCFV after reading its value as 1.

TCFU Flag (Underflow Flag)

This status flag indicates an underflow of TPUm.TCNT (m = 1, 2, 4, 5).

[Setting condition]

- Underflow of the value in TPUm.TCNT (TPUm.TCNT counted from 0000h to FFFFh).

[Clearing condition]

- Writing 0 to TCFU after reading its value as 1.

18.2.6 Timer Counter (TCNT)

TPUm.TCNT is a 16-bit counter that counts the internal clock or external events (m = 0 to 5).
This counter can be read/written in 16-bit units.
This counter is initialized to 0000h by a reset.

Address(es): TPU0.TCNT A008 0116h, TPU1.TCNT A008 0126h, TPU2.TCNT A008 0136h, TPU3.TCNT A008 0146h, TPU4.TCNT A008 0156h, TPU5.TCNT A008 0166h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

18.2.7 Timer General Register A (TGRA)
Timer General Register B (TGRB)
Timer General Register C (TGRC)
Timer General Register D (TGRD)

TPU has 16 TGR registers in total, four each for TPU0 and TPU3 and two each for TPU1, TPU2, TPU4, and TPU5.
TPUm.TGRA (m = 0 to 5), TPUm.TGRB (m = 0 to 5), TPUm.TGRC (m = 0, 3), and TPUm.TGRD (m = 0, 3) are 16-bit registers with a dual function as output compare and input capture registers.
These registers can be read/written in 16-bit units.
TPUm.TGRC and TPUm.TGRD can also be specified for operation as buffer registers. Register combinations during buffer operations are TPUm.TGRA-TPUm.TGRC and TPUm.TGRB-TPUm.TGRD.

Address(es): TPU0.TGRA A008 0118h, TPU0.TGRB A008 011Ah, TPU0.TGRC A008 011Ch, TPU0.TGRD A008 011Eh, TPU1.TGRA A008 0128h, TPU1.TGRB A008 012Ah, TPU2.TGRA A008 0138h, TPU2.TGRB A008 013Ah, TPU3.TGRA A008 0148h, TPU3.TGRB A008 014Ah, TPU3.TGRC A008 014Ch, TPU3.TGRD A008 014Eh, TPU4.TGRA A008 0158h, TPU4.TGRB A008 015Ah, TPU5.TGRA A008 0168h, TPU5.TGRB A008 016Ah

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

18.2.8 Timer Start Register (TSTRA)

TSTRA starts or stops count operation for TCNT counter of TPU0 to TPU5.
Before setting the operating mode in TPUm.TMDR or setting the TPUm.TCNT count clock in TPUm.TCR, stop the TPUm.TCNT counter.

Address(es): TPUA.TSTRA A008 0100h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	CST5	CST4	CST3	CST2	CST1	CST0
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CST0	Counter Start 0	0: TCNT count operation is stopped 1: TCNT performs count operation	R/W
b1	CST1	Counter Start 1		R/W
b2	CST2	Counter Start 2		R/W
b3	CST3	Counter Start 3		R/W
b4	CST4	Counter Start 4		R/W
b5	CST5	Counter Start 5		R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

CSTn Bits (Counter Start) (n = 0 to 5)

These bits start or stop the TCNT counter.
When the CSTn bit is cleared to 0 with CSTn = 1 and the corresponding TIOCyn pin (y = A to D, n = 0 to 5) specified for output, the counter stops but the output compare output level of the corresponding TIOCyn pin is retained.
If TIORH, TIORL, or TIOR is written to when the CSTn bit is 0, the pin output level will be changed to the set initial output value.

18.2.9 Timer Synchronous Register (TSYRA)

TSYRA selects independent operation or synchronous operation for the TCNT counters of TPU0 to TPU5.

Address(es): TPUA.TSYRA A008 0101h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	SYNC5	SYNC4	SYNC3	SYNC2	SYNC1	SYNC0
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SYNC0	Timer Synchronization 0	0: TCNT operates independently (TCNT presetting/clearing is unrelated to other channels) 1: TCNT performs synchronous operation*1 (TCNT synchronous presetting/synchronous clearing is possible)	R/W
b1	SYNC1	Timer Synchronization 1		R/W
b2	SYNC2	Timer Synchronization 2		R/W
b3	SYNC3	Timer Synchronization 3		R/W
b4	SYNC4	Timer Synchronization 4		R/W
b5	SYNC5	Timer Synchronization 5		R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. To set synchronous operation, the SYNCn bit (n = 0 to 5) for at least two channels must be set to 1. To set synchronous clearing, the TCNT clearing source must also be set by the TCR.CCLR[2:0] bits in addition to the SYNCn bit.

SYNCn Bits (Timer Synchronization) (n = 0 to 5)

These bits select whether the TCNT operation is independent of or synchronized with TCNT of other channels. When synchronous operation is selected, synchronous presetting of multiple TCNT counters and synchronous clearing through counter clearing on another channel are possible.

18.2.10 Noise Filter Control Register (NFCR)

TPU has 6 noise filter control registers in total (one register per channel).
TPUm.NFCR controls noise filtering of input capture signal on each channel. Only set the TPUm.NFCR register while the TPUm.TCNT counter is stopped (m = 0 to 5).

Address(es): TPU0.NFCR A008 0108h, TPU1.NFCR A008 0109h, TPU2.NFCR A008 010Ah, TPU3.NFCR A008 010Bh, TPU4.NFCR A008 010Ch, TPU5.NFCR A008 010Dh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	NFCS[1:0]		NFDEN	NFCEN	NFBEN	NFAEN
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	NFAEN	Noise Filter Enable A	0: The noise filter for TIOCAm is disabled. 1: The noise filter for TIOCAm is enabled. (m = 0 to 5)	R/W
b1	NFBEN	Noise Filter Enable B	0: The noise filter for TIOCBm is disabled. 1: The noise filter for TIOCBm is enabled. (m = 0 to 5)	R/W
b2	NFCEN*1	Noise Filter Enable C	0: The noise filter for TIOCCm is disabled. 1: The noise filter for TIOCCm is enabled. (m = 0, 3)	R/W
b3	NFDEN*1	Noise Filter Enable D	0: The noise filter for TIOCDm is disabled. 1: The noise filter for TIOCDm is enabled. (m = 0, 3)	R/W
b5, b4	NFCS[1:0]	Noise Filter Clock Select	00: PCLKD/1 01: PCLKD/8 10: PCLKD/32 11: Clock source that drives counting	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Bits 2 and 3 in TPU1.NFCR, TPU2.NFCR, TPU4.NFCR, and TPU5.NFCR are reserved. The bits are read as 0. Writing to these bits is not possible.

NFAEN Bit (Noise Filter Enable A)

This bit disables or enables the noise filter for the TIOCAm pin (m = 0 to 5).
Since unexpected edges may be internally generated when the value of NFAEN is changed, select the output compare function in the timer I/O control register before changing the NFAEN value.

NFBEN Bit (Noise Filter Enable B)

This bit disables or enables the noise filter for the TIOCBm pin (m = 0 to 5).
Since unexpected edges may be internally generated when the value of NFBEN is changed, select the output compare function in the timer I/O control register before changing the NFBEN value.

NFCEN Bit (Noise Filter Enable C)

This bit disables or enables the noise filter for the TIOCCm pin (m = 0, 3).
Since unexpected edges may be internally generated when the value of NFCEN is changed, select the output compare function in the timer I/O control register before changing the NFCEN value.

NFDEN Bit (Noise Filter Enable D)

This bit disables or enables the noise filter for the TIOCDm pin (m = 0, 3).
Since unexpected edges may be internally generated when the value of NFDEN is changed, select the output compare function in the timer I/O control register before changing the NFDEN value.

NFCS[1:0] Bits (Noise Filter Clock Select)

These bits select the sampling clock for the noise filter.
When the count source is selected with NFCS[1:0] bits set to 11b, the clock that can be used as sampling clock are the internal clocks other than PCLKD/1 specified with the TPSC[2:0] bits and the external clock. To select the PCLKD/1 as both the counter clock and the sampling clock, set the NFCS[1:0] bits to 00b.
The input-capture signal is sampled on rising edges of the selected clock signal. If the sampled levels match three times in a row, the given level is passed through as the input-capture signal. If the levels do not match, the existing value is retained.
After setting the NFCS[1:0] bits, wait for two selected sampling periods before setting the input capture function.

18.3 Operation

18.3.1 Basic Functions

Each channel has a TPUm.TCNT counter and a TPUm.TGRy register (y = A to D).
TCNT is a 16-bit up-counter, which can function as a free-running counter, periodic counter, or event counter.
TGRy can be used as an input capture register or output compare register.

(1) Counter Operation

When the TSTRA.CSTj bit (j = 0 to 5) is set to 1, the TCNT counter for the corresponding channel starts counting.

(a) Example of count operation setting procedure

Figure 18.2 shows an example of the count operation setting procedure.

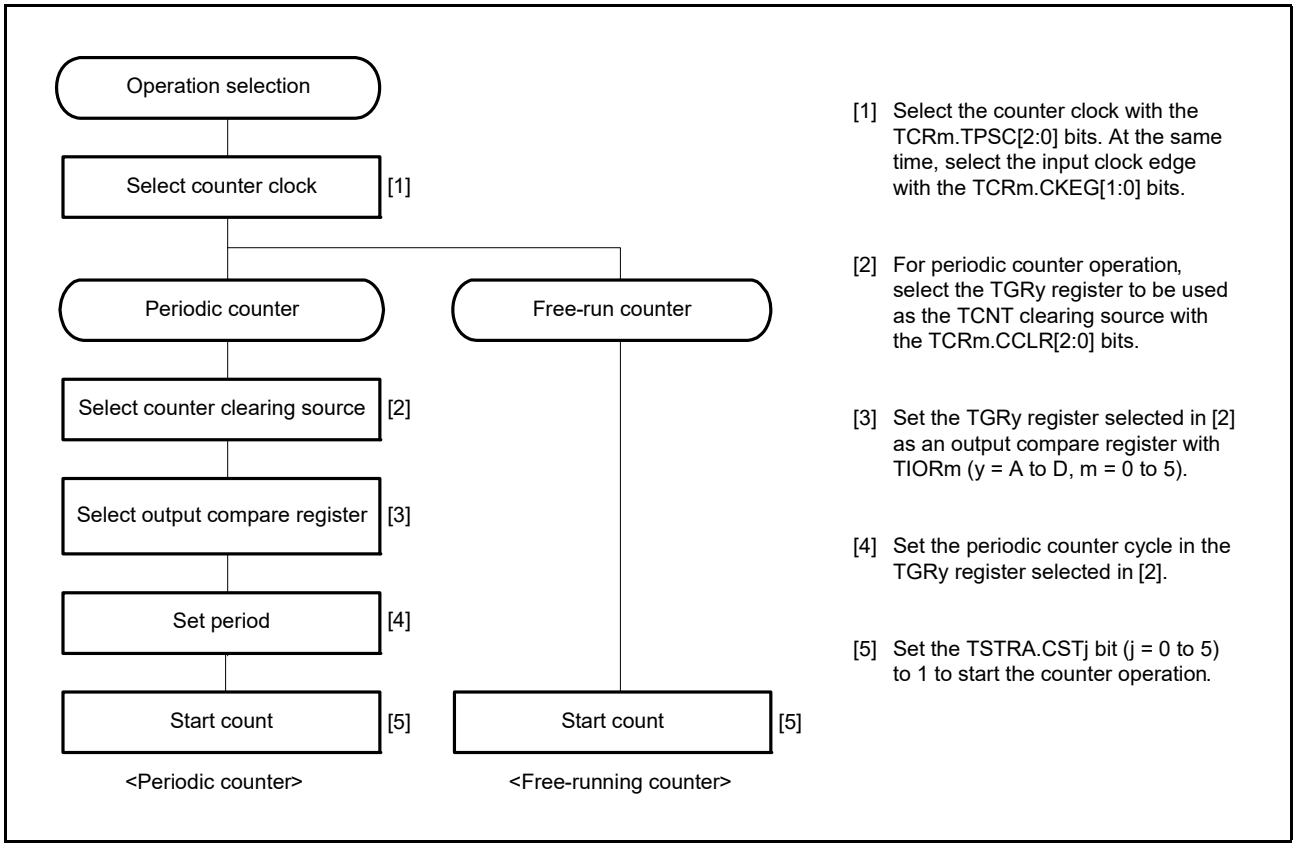


Figure 18.2 Example of Counter Operation Setting Procedure

(b) Free-running count operation and periodic count operation

Immediately after a reset, the TPUm.TCNT counters are all set as free-running counters. When the relevant bit in TSTRA is set to 1, the corresponding TCNT counter starts up-count operation as a free-running counter. When TCNT overflows (changes from FFFFh to 0000h), the TPU requests an overflow interrupt (TCImV). After an overflow, TCNT restarts counting up from 0000h (m = 0 to 5).

Figure 18.3 shows free-running counter operation.

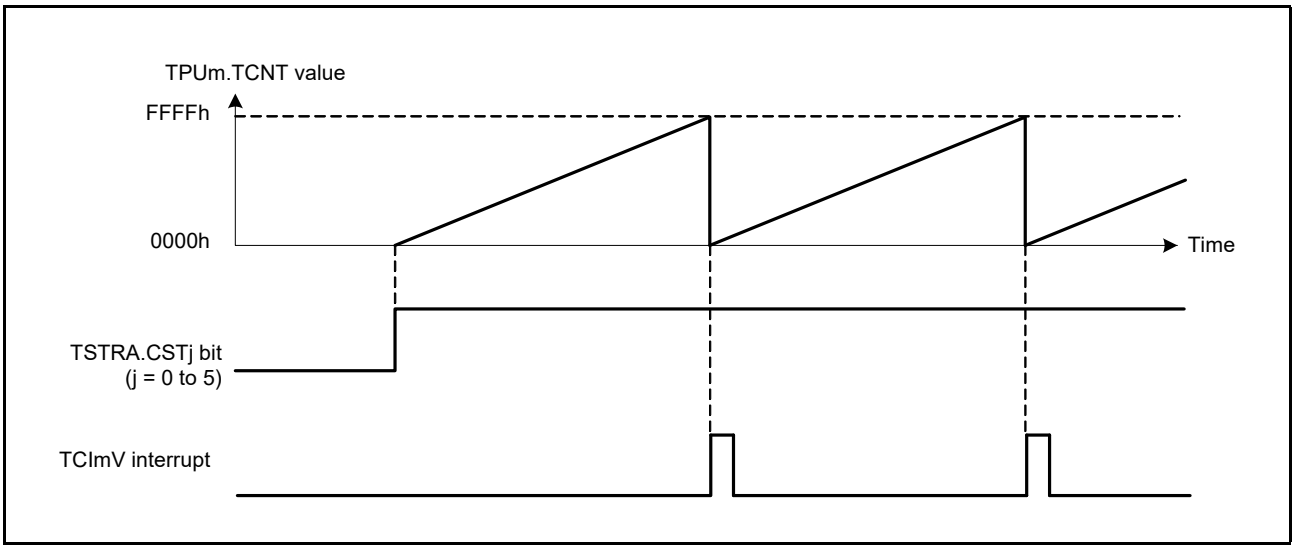


Figure 18.3 Free-Running Counter Operation

When compare match is selected as the TCNT clearing source, the TCNT counter for the relevant channel performs periodic count operation. The TPUm.TGRy register for setting the period is set as an output compare register, and counter clearing by compare match is selected by the TPUm.TCR.CCLR[2:0] bits. After the settings have been made, TCNT starts count-up operation as a periodic counter when the corresponding bit in TSTRA is set to 1. When the count value matches the TGRy value, TCNT is cleared to 0000h.

At this time, the TPU requests an interrupt (TGImy). After a compare match, TCNT restarts counting up from 0000h (m = 0 to 5).

Figure 18.4 shows periodic counter operation.

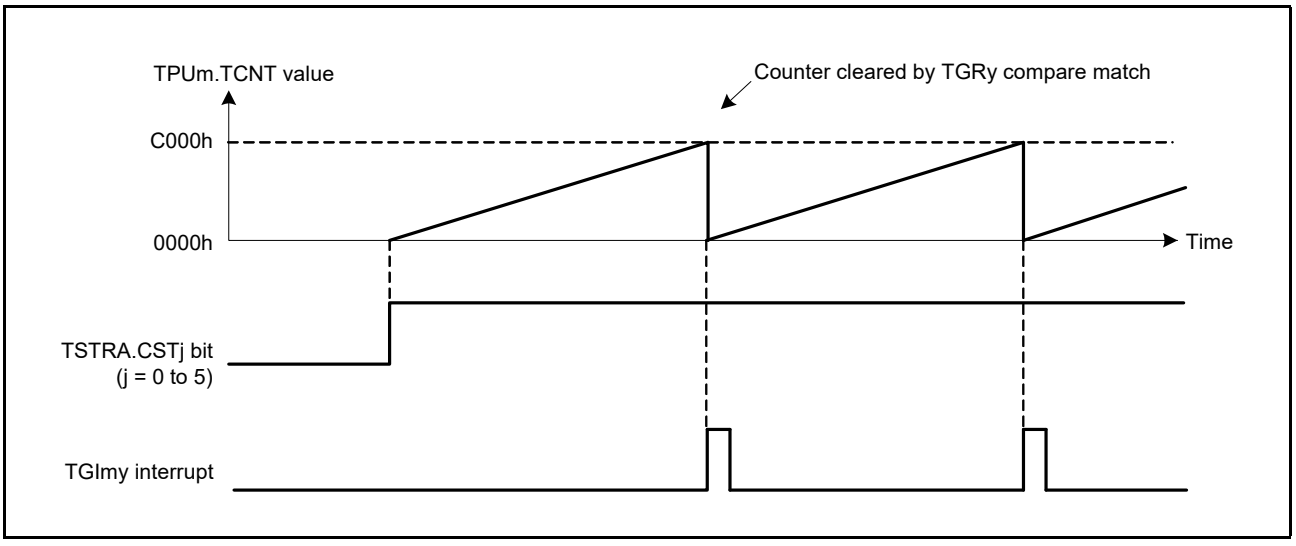


Figure 18.4 Periodic Counter Operation

(2) Waveform Output by Compare Match

The TPU can perform low, high, or toggle output from the corresponding output pin using a compare match.

(a) Example of setting procedure for waveform output by compare match

Figure 18.5 shows an example of the setting procedure for waveform output by a compare match.

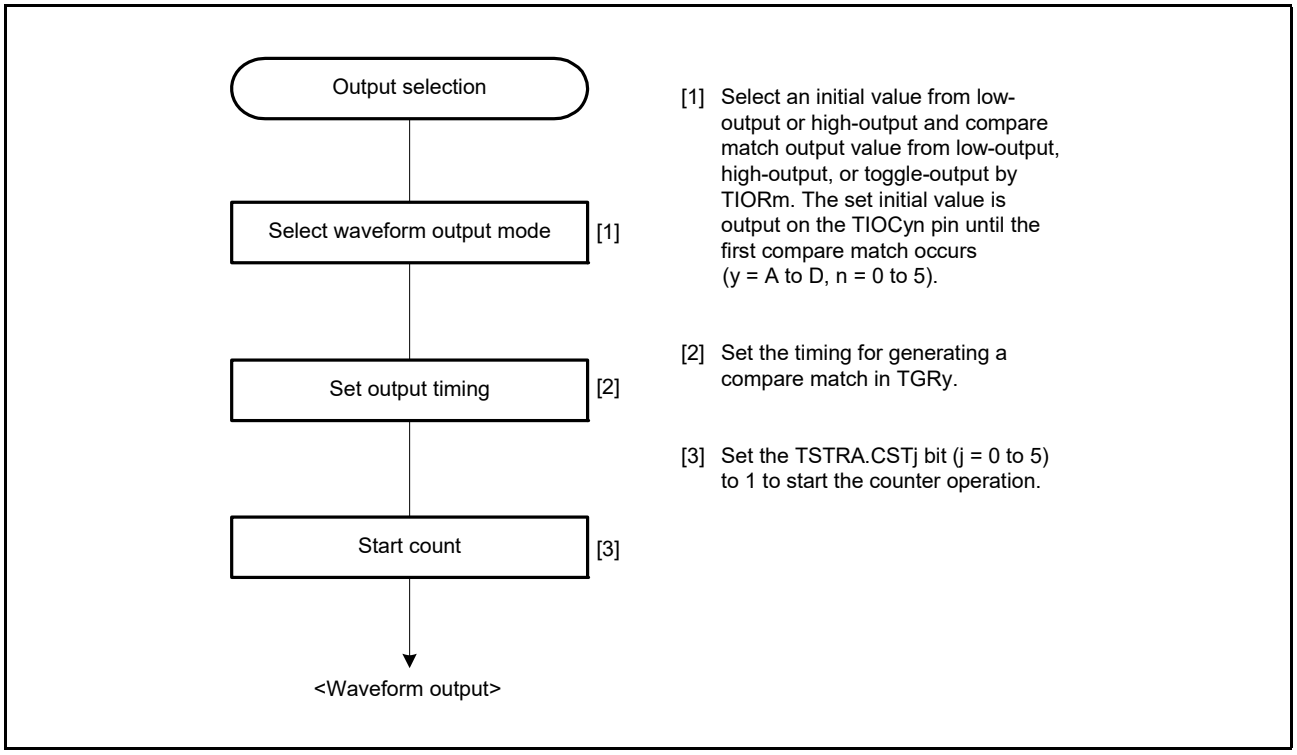


Figure 18.5 Example of Setting Procedure for Waveform Output by Compare Match

(b) Examples of waveform output operation

Figure 18.6 shows an example of low output/high output.

In this example, TPUm.TCNT (m = 0 to 5) has been set as a free-running counter, and settings have been made so that high is output by compare match A and low is output by compare match B. When the set level and the pin level match, the pin level does not change.

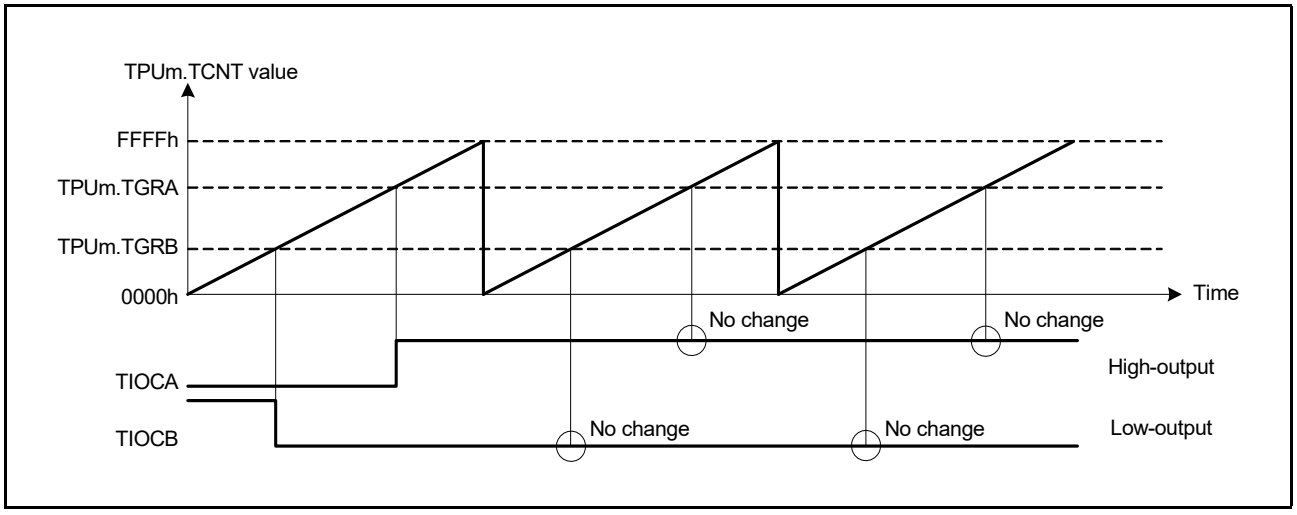


Figure 18.6 Example of Low-Output/High-Output Operation

Figure 18.7 shows an example of toggle output.
In this example, TPUm.TCNT (m = 0 to 5) has been set as a periodic counter (with counter clearing performed by compare match B), and settings have been made so that output is toggled by both compare match A and compare match B.

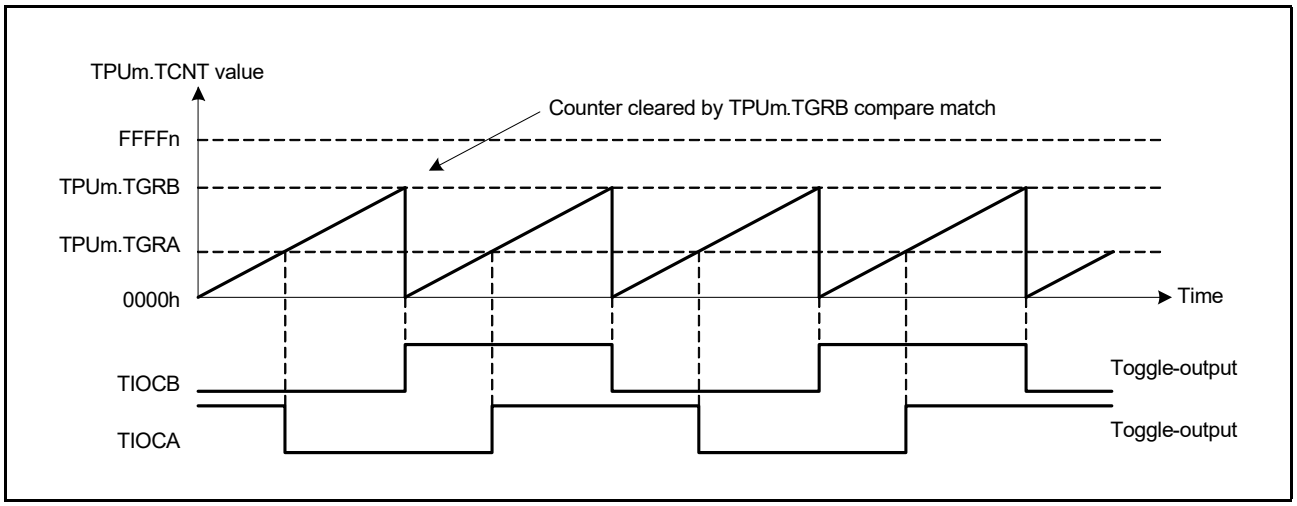


Figure 18.7 Example of Toggle Output Operation

(3) Input Capture Function

The TCNT value can be transferred to TGRy on detection of the TIOCyn pin (n = 0 to 5 when y = A, B; n = 0, 3 when y = C, D) input edge.

The rising edge, the falling edge, or both edges can be selected as the detection edge. It is also possible to specify the counter input clock or compare match signal of TPU0, TPU1, TPU3, and TPU4 as the input capture source. Noise filtering can be applied to the input capture input.

Note: Even if the counter is halted, an input capture is generated, and flag and interrupt signals are generated.

Note: When another channel's counter input clock is used as the input capture input for TPU0 and TPU3, PCLKD/1 should not be selected as the counter input clock used for input capture input. Input capture will not be generated if PCLKD/1 is selected.

(a) Example of setting procedure for input capture operation

Figure 18.8 shows an example of the setting procedure for input capture operation.

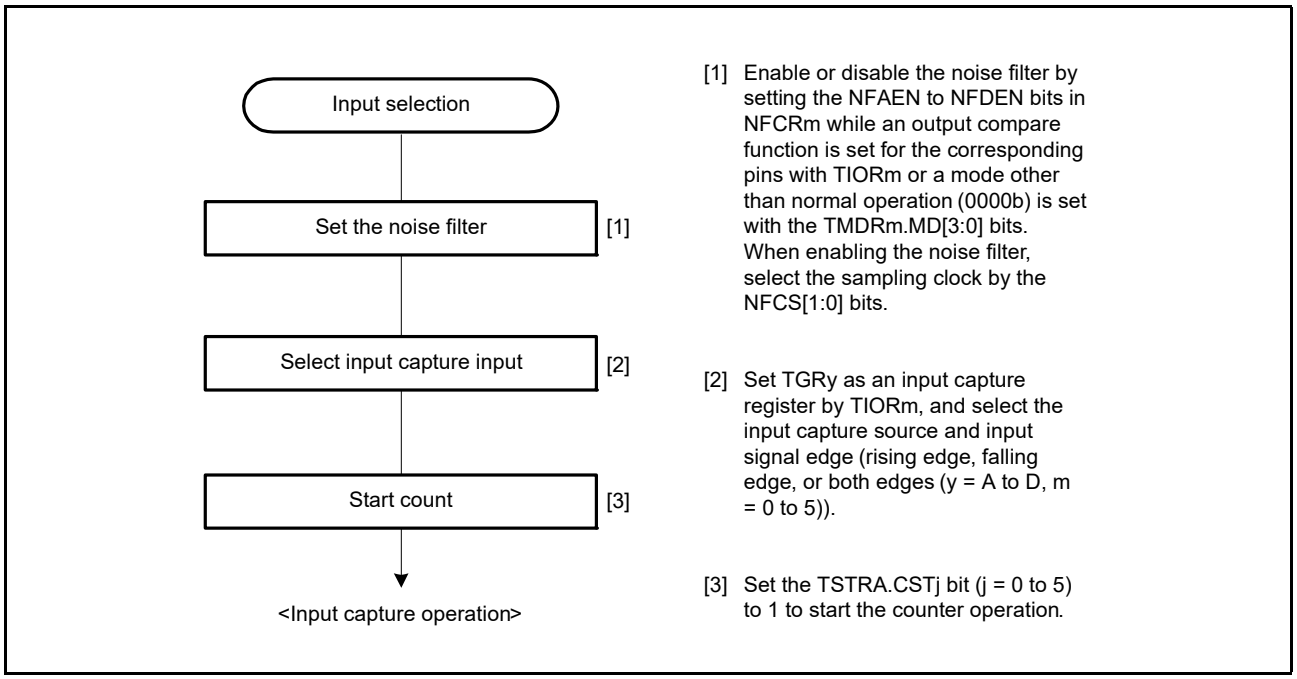


Figure 18.8 Example of Setting Procedure for Input Capture Operation

(b) Example of input capture operation

Figure 18.9 shows an example of input capture operation when the noise filter is stopped.
In this example, both rising and falling edges have been selected as the TIOCAm pin input capture input edge, the falling edge has been selected as the TIOCBm pin input capture input edge, and counter clearing by TGRB input capture has been set for TCNT (m = 0 to 5).

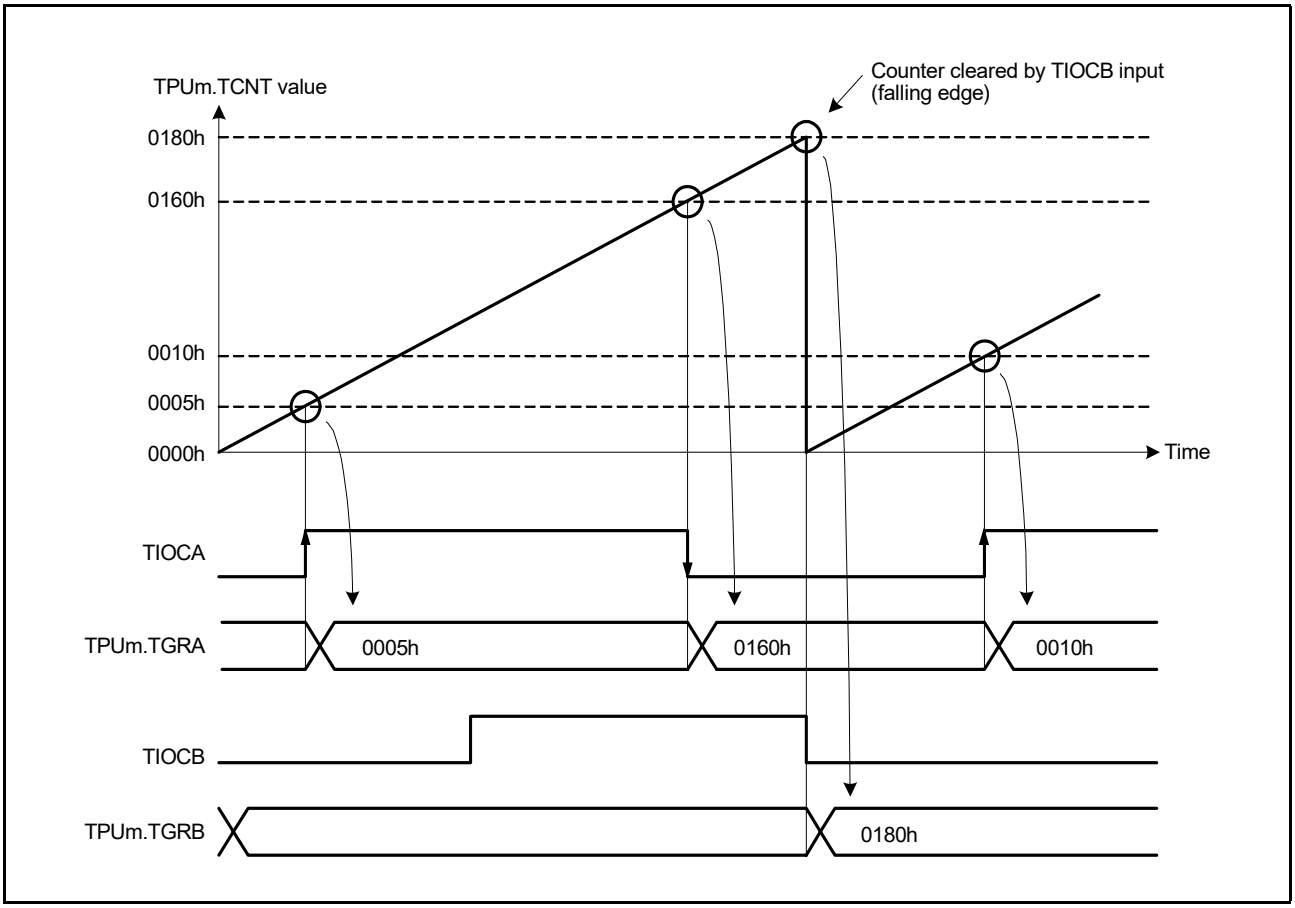


Figure 18.9 Example of Input Capture Operation (with Noise Filter Stopped)

When noise filtering is enabled, see Figure 18.30.

18.3.2 Synchronous Operation

In synchronous operation, the values in multiple TCNT counters can be rewritten simultaneously (synchronous presetting). Also, multiple TCNT counters can be cleared simultaneously (synchronous clearing) by making the appropriate setting in TCR.
Synchronous operation enables TGRy to be incremented with respect to a single time base.
All channels of TPU0 to TPU5 can be set for synchronous operation.

(1) Example of Synchronous Operation Setting Procedure

Figure 18.10 shows an example of the synchronous operation setting procedure.

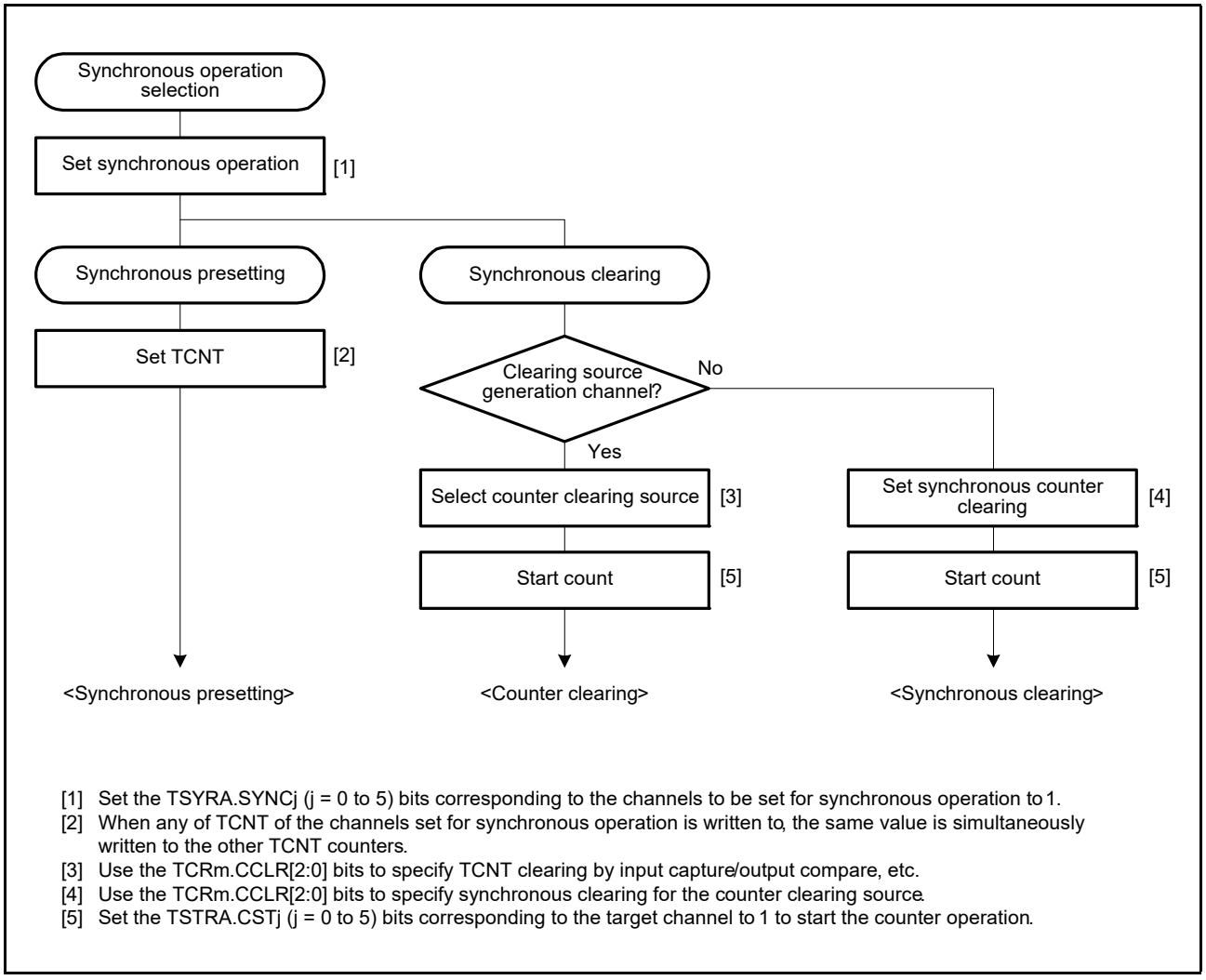


Figure 18.10 Example of Synchronous Operation Setting Procedure

(2) Example of Synchronous Operation

Figure 18.11 shows an example of synchronous operation.
In this example, synchronous operation and PWM mode 1 have been set for TPU0 to TPU2, TPU0.TGRB compare match has been set as the TPU0 counter clearing source, and synchronous clearing has been set for the TPU1 and TPU2 counter clearing source.
Three-phase PWM waveforms are output from pins TIOCA0, TIOCA1, and TIOCA2. At this time, synchronous presetting and synchronous clearing by TPU0.TGRB compare match are performed for TCNT of TPU0 to TPU2, and the data set in TPU0.TGRB is used as the PWM cycle.
For details on PWM modes, see section 18.3.5, PWM Modes.

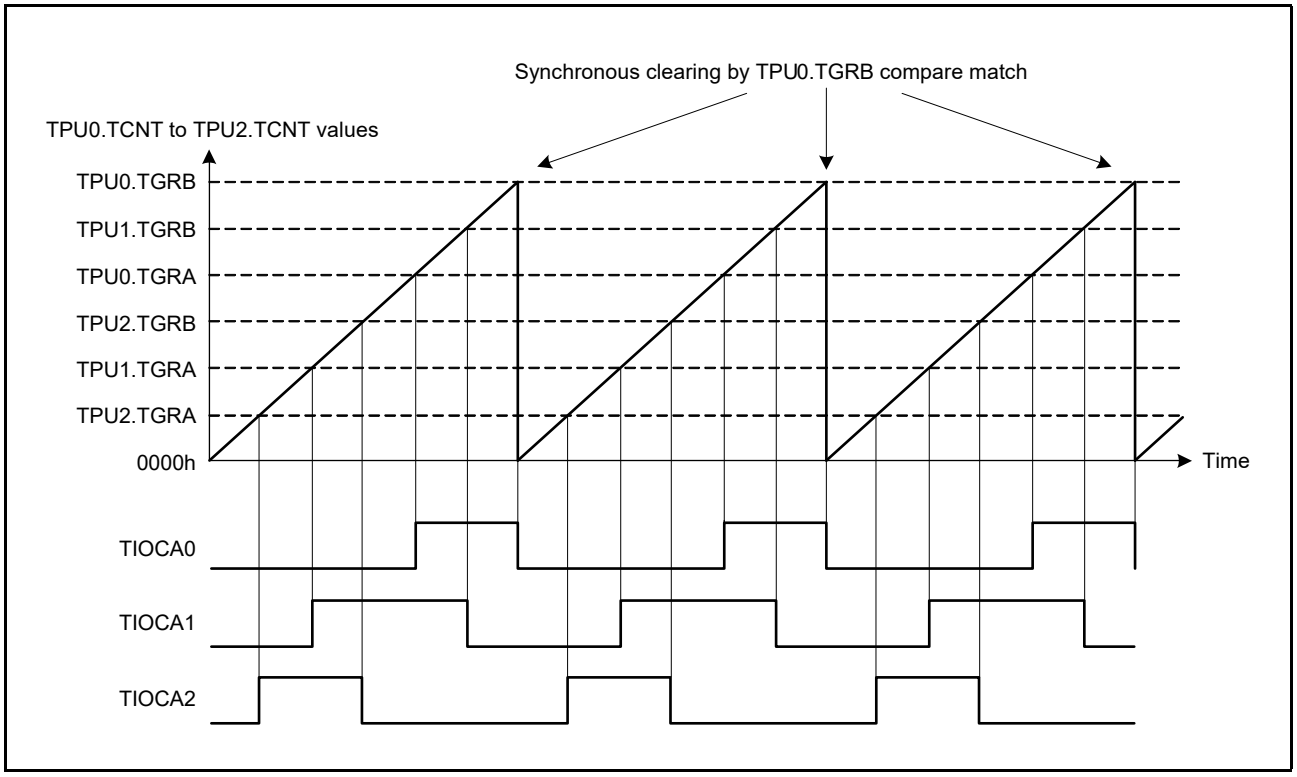


Figure 18.11 Example of Synchronous Operation

18.3.3 Buffer Operation

Buffer operation, provided for TPU0 and TPU3, enables TPUm.TGRC and TPUm.TGRD to be used as buffer registers. Buffer operation differs depending on whether TPUm.TGRy has been set as an input capture register or a compare match register.

Table 18.21 lists the register combinations used in buffer operation.

Table 18.21 Register Combinations in Buffer Operation

Channel	Timer General Register	Buffer Register
TPU0	TPU0.TGRA	TPU0.TGRC
	TPU0.TGRB	TPU0.TGRD
TPU3	TPU3.TGRA	TPU3.TGRC
	TPU3.TGRB	TPU3.TGRD

- When TPUm.TGRy 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 shown in Figure 18.12.

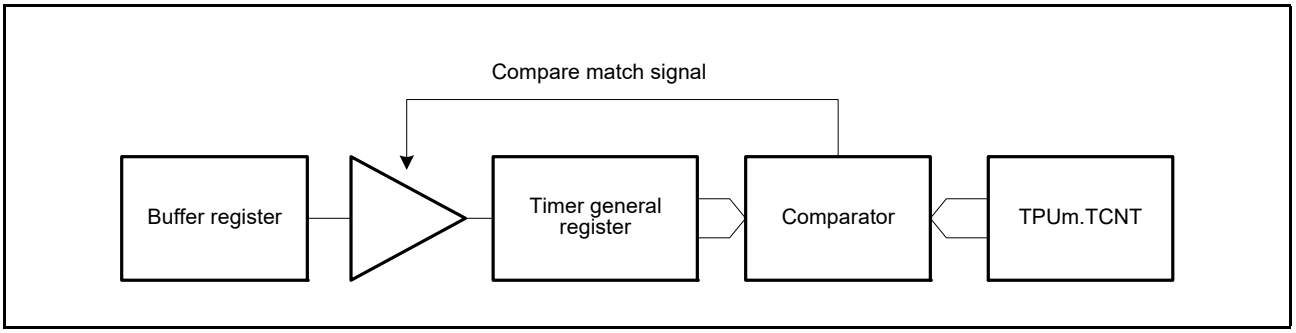


Figure 18.12 Compare Match Buffer Operation

- When TPUm.TGRy is an input capture register

When input capture occurs, the value in TPUm.TCNT is transferred to TGRy and the value previously held in TGRy is simultaneously transferred to the buffer register.

This operation is shown in Figure 18.13.

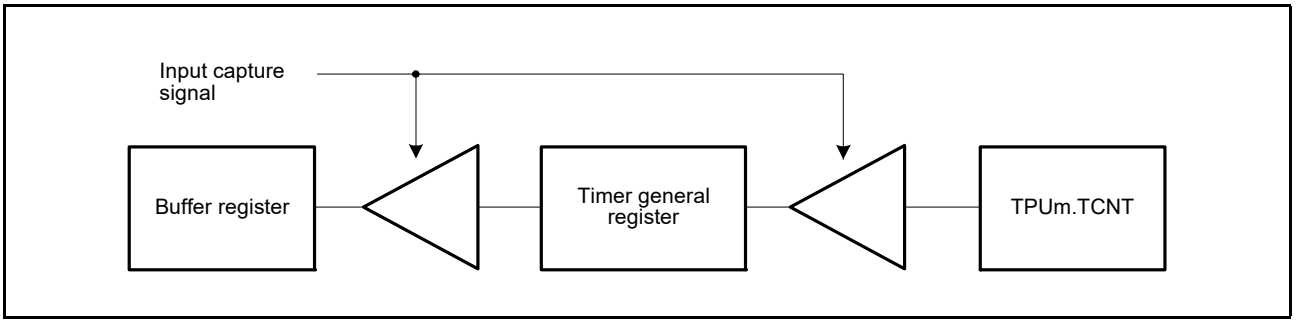


Figure 18.13 Input Capture Buffer Operation

(1) Example of Buffer Operation Setting Procedure

Figure 18.14 shows an example of the buffer operation setting procedure.

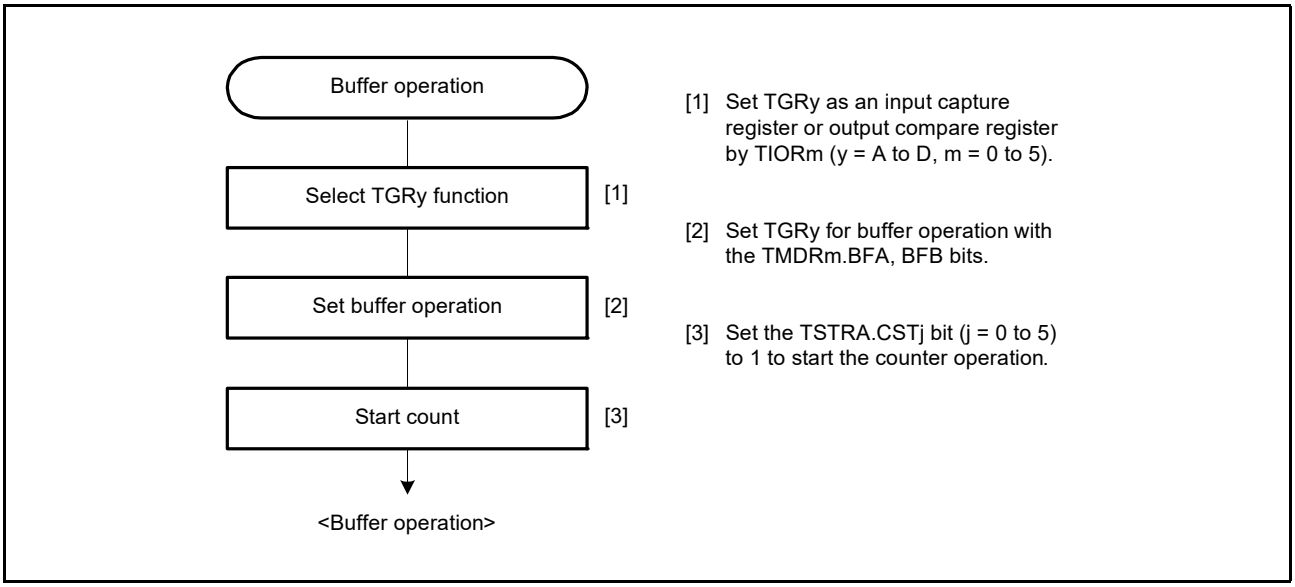


Figure 18.14 Example of Buffer Operation Setting Procedure

(2) Examples of Buffer Operation

(a) When TPUm.TGRy is an output compare register

Figure 18.15 shows an operation example in which PWM mode 1 has been set for TPU0, and buffer operation has been set for TPU0.TGRA and TPU0.TGRC. The settings used in this example are TPU0.TCNT clearing by compare match B, high output at compare match A, and low output at compare match B.
As buffer operation has been set, when compare match A occurs, the output changes and the TPU0.TGRC value is simultaneously transferred to TPU0.TGRA. This operation is repeated each time compare match A occurs.
For details on PWM modes, see section 18.3.5, PWM Modes.

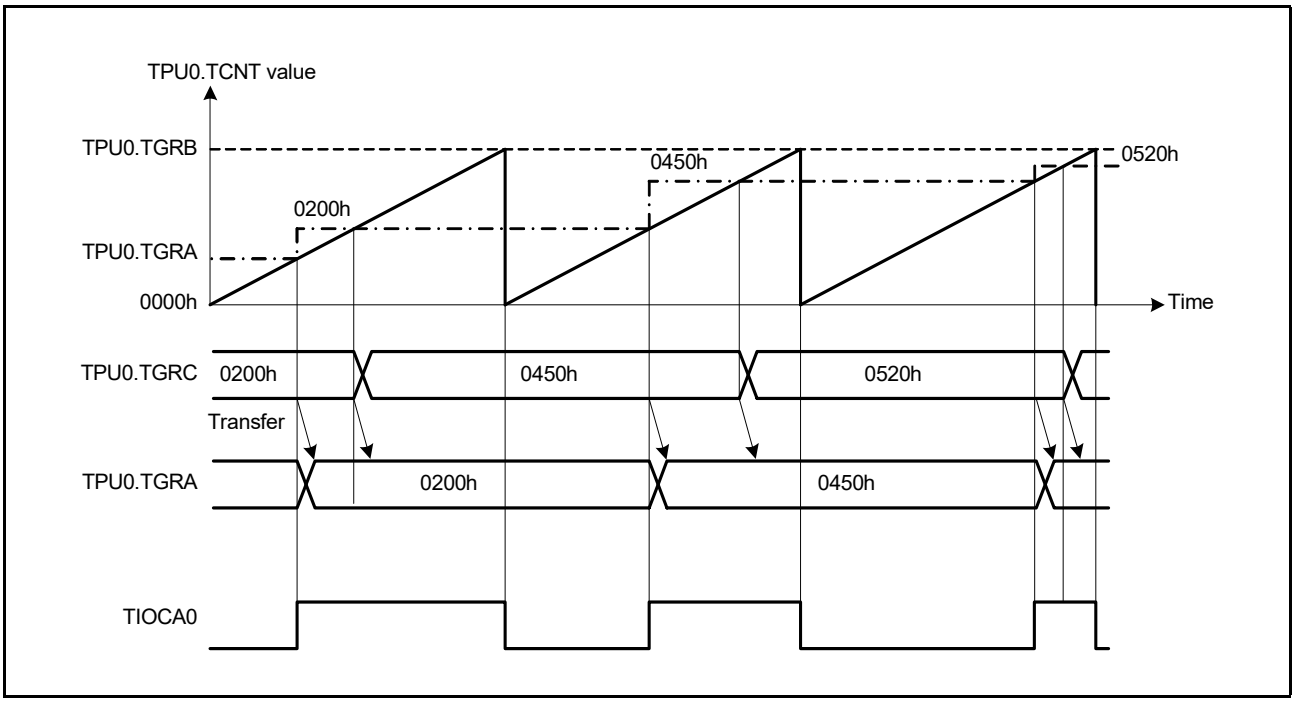


Figure 18.15 Example of Buffer Operation (1)

(b) When TPUm.TGRy is an input capture register

Figure 18.16 shows an operation example in which TPUm.TGRA has been set as an input capture register, and buffer operation has been set for TGRA and TPUm.TGRC.
Counter clearing by TGRA input capture has been set for TPUm.TCNT, and both rising and falling edges have been selected as the TIOCAn pin input capture input edge.
As buffer operation has been set, when the TCNT value is stored in TGRA upon occurrence of input capture A, the value previously stored in TGRA is simultaneously transferred to TGRC (m = 0 to 5).

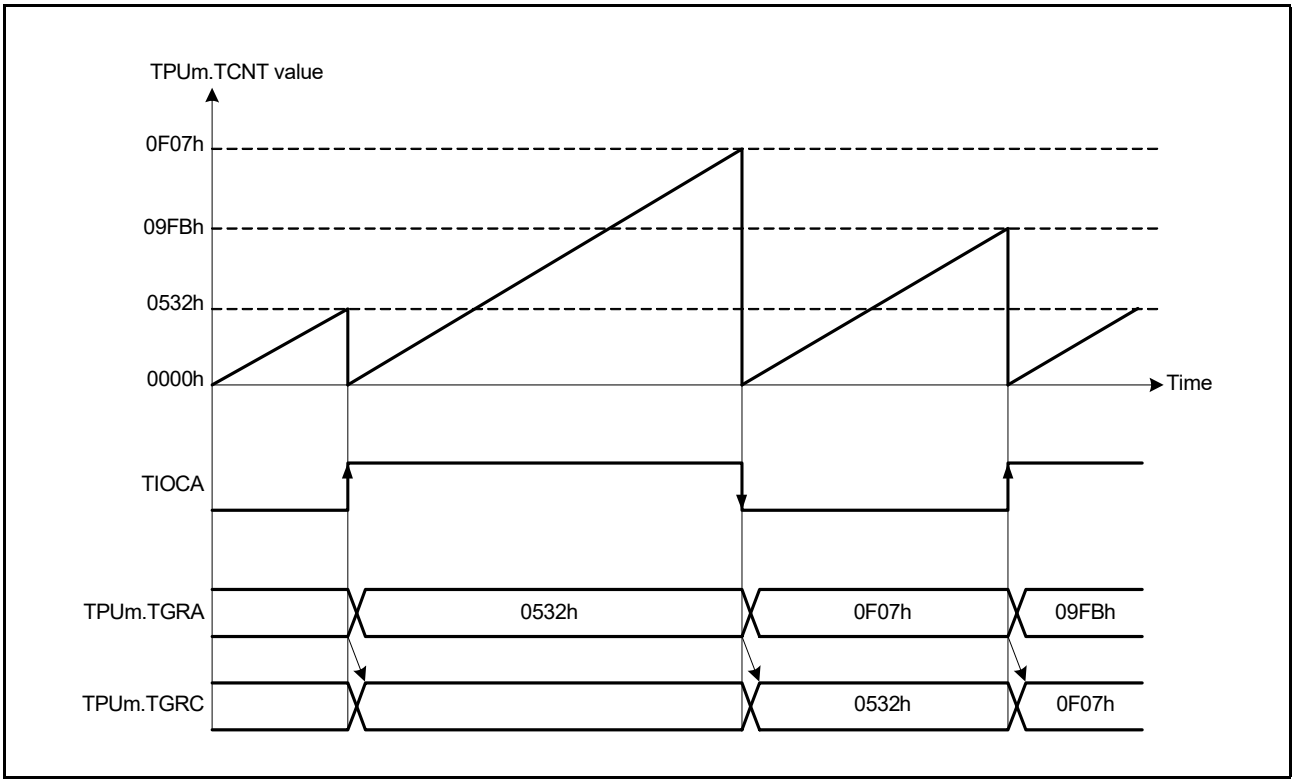


Figure 18.16 Example of Buffer Operation (2)

18.3.4 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 TPUm (m = 1, 4) counter clock at overflow/underflow of TPUn.TCNT (n = 2, 5) as set by the TPUm.TCR.TPSC[2:0] bits (m = 1, 4).
Underflow occurs only when the lower 16-bit TPUn.TCNT is in phase counting mode.
Table 18.22 lists the register combinations used in cascaded operation.

Note 1. When phase counting mode is set for TPU1 or TPU4, the counter clock setting is invalid and the counter operates independently in phase counting mode.
Note 2. For input capture operation with a cascade connection, be sure to see section 18.8.16, Input Capture Operation in Cascaded Operation.
Note 3. For operation in 32-bit phase counting mode, set the lower-order 16 bits of TPUn to phase counting mode and the higher-order 16 bits of TPUm to normal operating mode.

Table 18.22 Cascaded Combinations

Combination	Upper 16 Bits	Lower 16 Bits
TPU1 and TPU2	TPU1.TCNT	TPU2.TCNT
TPU4 and TPU5	TPU4.TCNT	TPU5.TCNT

(1) Example of Cascaded Operation Setting Procedure

Figure 18.17 shows an example of the setting procedure for cascaded operation.

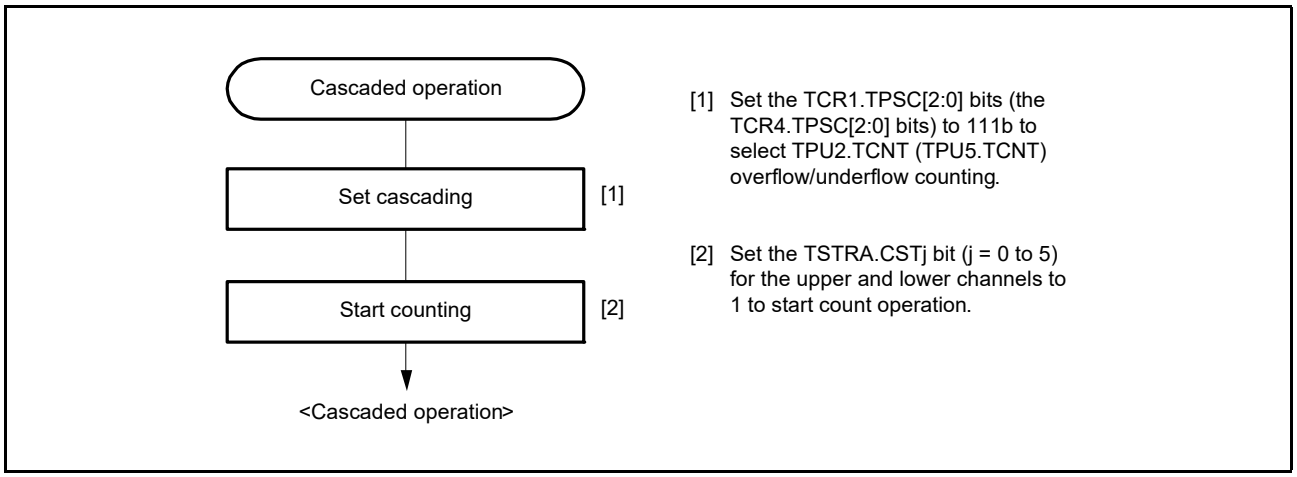


Figure 18.17 Cascaded Operation Setting Procedure

(2) Examples of Cascaded Operation

Figure 18.18 shows the operation when counting upon TPU2.TCNT overflow/underflow has been set for TPU1.TCNT, TPU1.TGRA and TPU2.TGRA have been set as input capture registers, and the rising edge of the TIOCA1 and TIOCA2 pins has been selected.

When a rising edge is simultaneously input to the TIOCA1 and TIOCA2 pins, the 16 higher-order bits and the 16 lower-order bits of the 32-bit value are transferred to the TPU1.TGRA and TPU2.TGRA registers, respectively.

However, even if the rising edge is input to the TIOCA1 and TIOCA2 pins simultaneously, the 16 higher- and 16 lower-order bits may not be captured simultaneously due to delays within the LSI chip, so a value may vary from that at the time. If this is the case, discard the captured value by following the procedure described in section 18.8.16, Input Capture Operation in Cascaded Operation.

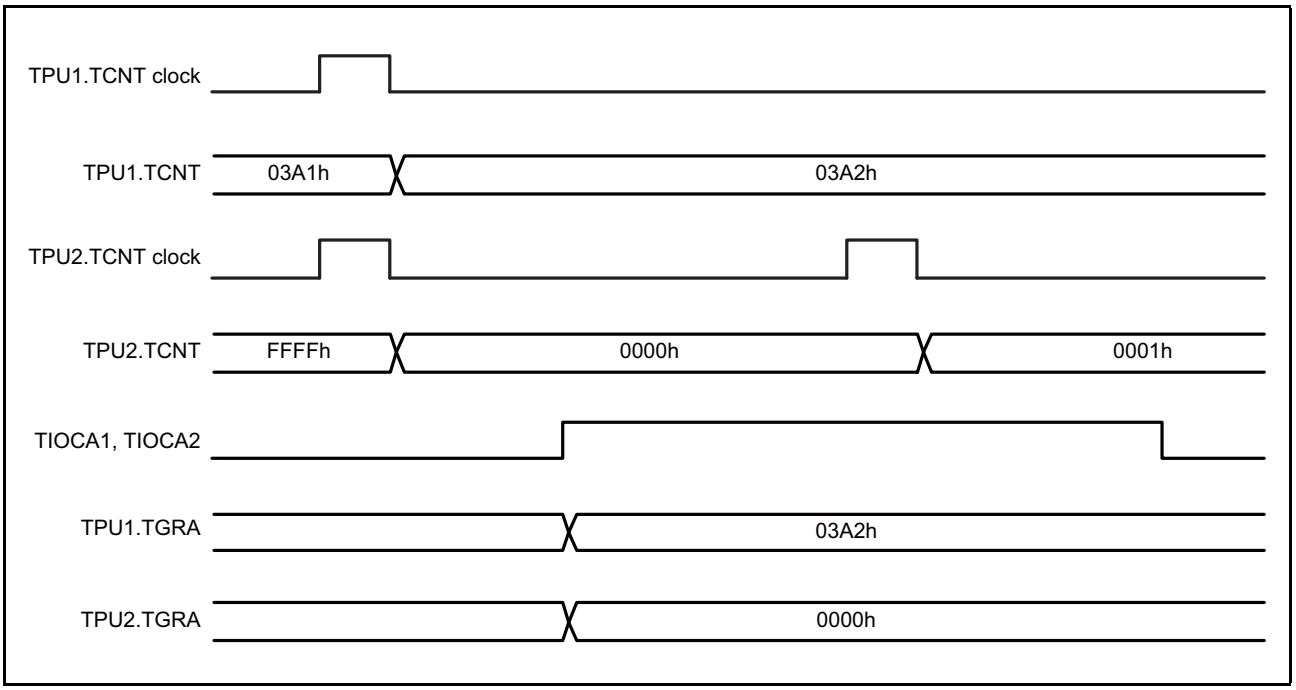


Figure 18.18 Example of Cascaded Operation (1)

Figure 18.19 shows the operation when counting upon TPU2.TCNT overflow/underflow has been set for TPU1.TCNT, and phase counting mode has been specified for TPU2.

TPU1.TCNT is incremented by TPU2.TCNT overflow and decremented by TPU2.TCNT underflow.

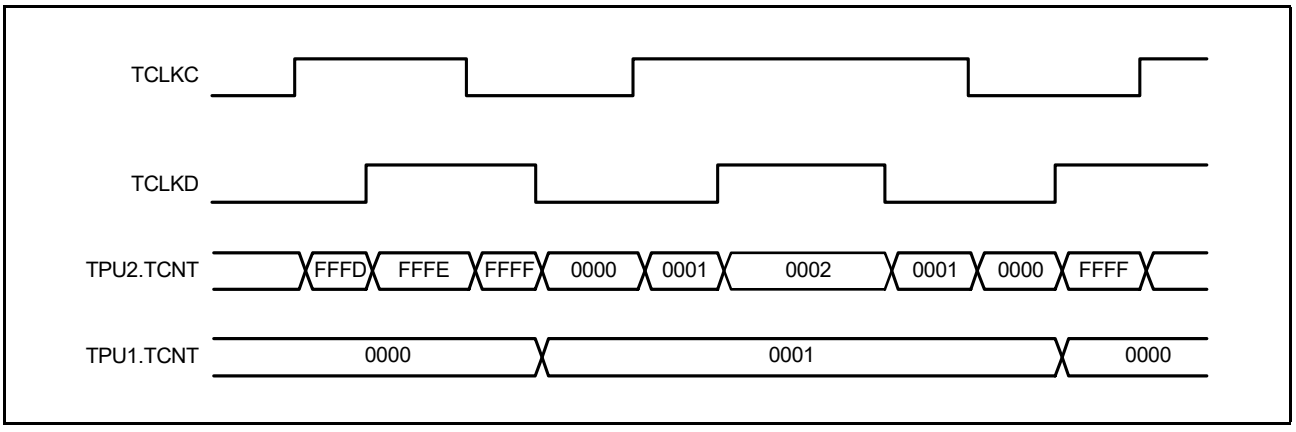


Figure 18.19 Example of Cascaded Operation (2)

18.3.5 PWM Modes

In PWM mode, PWM waveforms are output from the output pins. low-, high-, or toggle-output can be selected as the output level in response to compare match of each TGRy.
Settings of TGRy registers can output a PWM waveform in the range of 0% to 100% duty cycle.
Specifying TGRy compare match as the counter clearing source enables the cycle to be set in that register. All channels can be set for PWM mode independently. Synchronous operation is also possible.
There are two PWM modes, as described below.

1. PWM mode 1
   PWM waveform is generated from the TIOCAn and TIOCCn pins by pairing TPUm.TGRA with TPUm.TGRB and TPUm.TGRC with TPUm.TGRD. The levels specified by the TPUm.TIOR(H).IOA[3:0] bits and TPUm.TIORL.IOC[3:0] bits are output from the TIOCAn and TIOCCn pins at compare matches A and C, respectively. The levels specified by theTPUm.TIOR(H).IOB[3:0] bits and TPUm.TIORL.IOD[3:0] bits are output from the TIOCAn and TIOCCn pins at compare matches B and D, respectively. The initial output value of the TIOCAn or TIOCCn pin is the value set in the TIOR.IOA or TIORL.IOC bit. If the set values of paired TGRy registers are identical, the output value does not change even when a compare match occurs.
   In PWM mode 1, up to eight phases of PWM waveforms can be output.
2. PWM mode 2
   PWM waveform is generated by using one TPUm.TGRy as the cycle register and the others as duty cycle registers. The level specified in TPUm.TIORH, TPUm.TIORL, or TPUm.TIOR is output at compare matches. Upon counter clearing by a cycle register compare match, the output value of each pin is the initial value set in TIORH, TIORL, or TIOR. If the set values of the cycle register and duty cycle register are identical, the output value does not change even when a compare match occurs.
   In PWM mode 2, up to 15 phases of PWM waveforms can be output when using synchronous operation in combination.

(m = 0 to 5 when y = A, B; m = 0, 3 when y = C, D)

The correspondence between PWM output pins and registers is listed in Table 18.23.

Table 18.23 PWM Output Registers and Output Pins

		Output Pin	
Channel	**Register**	**PWM Mode 1**	**PWM Mode 2**
TPU0	TPU0.TGRA	TIOCA0	TIOCA0
	TPU0.TGRB		TIOCB0
	TPU0.TGRC	TIOCC0	TIOCC0
	TPU0.TGRD		TIOCD0
TPU1	TPU1.TGRA	TIOCA1	TIOCA1
	TPU1.TGRB		TIOCB1
TPU2	TPU2.TGRA	TIOCA2	TIOCA2
	TPU2.TGRB		TIOCB2
TPU3	TPU3.TGRA	TIOCA3	TIOCA3
	TPU3.TGRB		TIOCB3
	TPU3.TGRC	TIOCC3	TIOCC3
	TPU3.TGRD		TIOCD3
TPU4	TPU4.TGRA	TIOCA4	TIOCA4
	TPU4.TGRB		TIOCB4
TPU5	TPU5.TGRA	TIOCA5	TIOCA5
	TPU5.TGRB		TIOCB5

Note: In PWM mode 2, PWM waveform output is not possible for the TPUm.TGRy register in which the cycle is set.

(1) Example of PWM Mode Setting Procedure

Figure 18.20 shows an example of the PWM mode setting procedure.

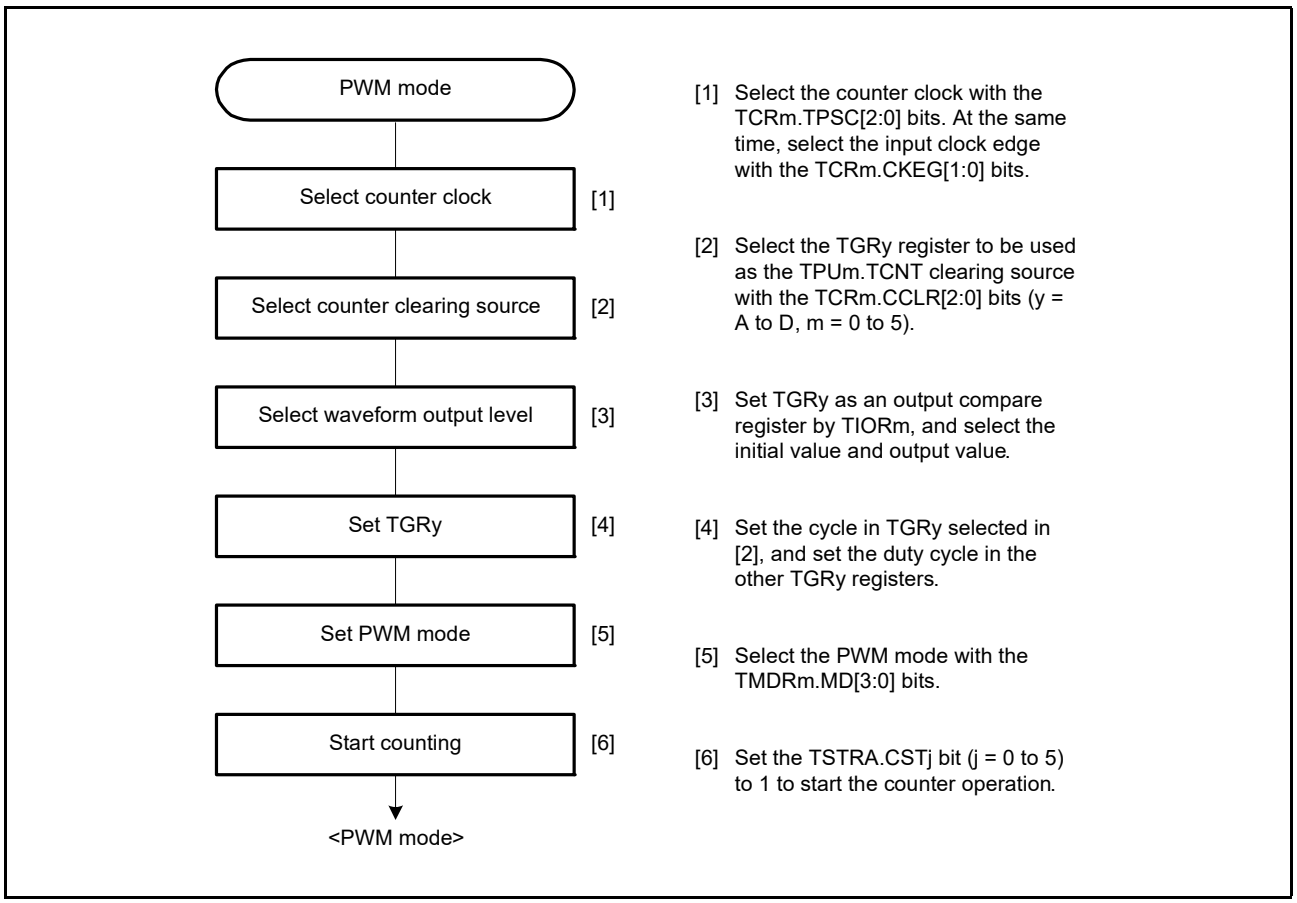


Figure 18.20 Example of PWM Mode Setting Procedure

(2) Examples of PWM Mode Operation

Figure 18.21 shows an example of PWM mode 1 operation.

In this example, TPUm.TGRA compare match is set as the TPUm.TCNT clearing source, 0 is set for the TGRA initial output value and output value, and 1 is set as the TPUm.TGRB output value (m = 0 to 5).

In this case, the value set in TGRA is used as the cycle, and the value set in TGRB is used as the duty cycle.

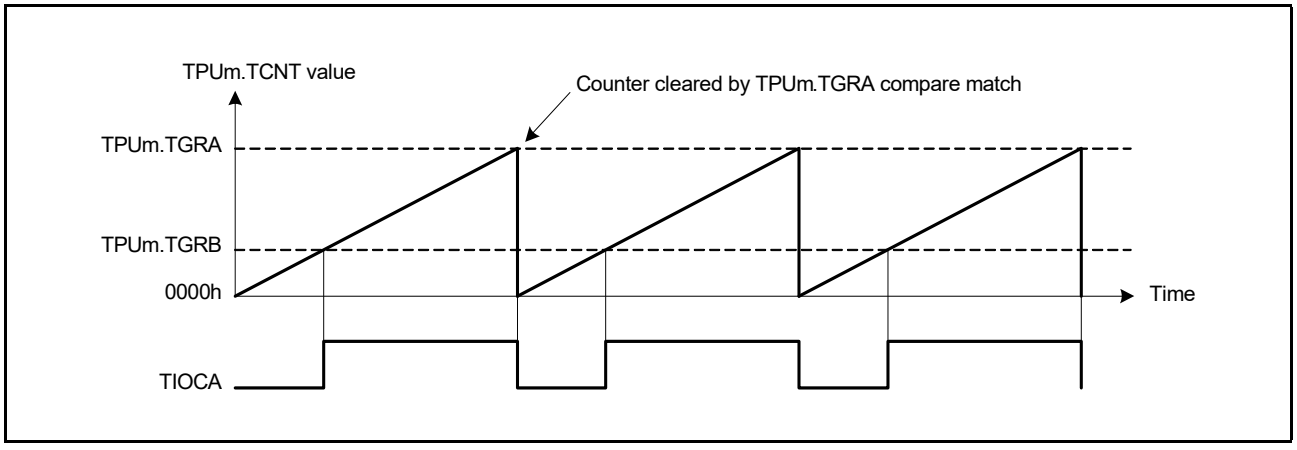


Figure 18.21 Example of PWM Mode Operation (1)

Figure 18.22 shows an example of PWM mode 2 operation.
In this example, synchronous operation is specified for TPU0 and TPU1, TPU1.TGRB compare match is set as TPUm.TCNT (m = 0, 1) clearing source, and 0 is set for the initial output value and 1 for the output value of the other TGRy registers (TPU0.TGRA to TPU0.TGRD and TPU1.TGRA), to output a 5-phase PWM waveform.
In this case, the value set in TPU1.TGRB is used as the cycle, and the values set in the other TGRy registers are used as the duty cycle.

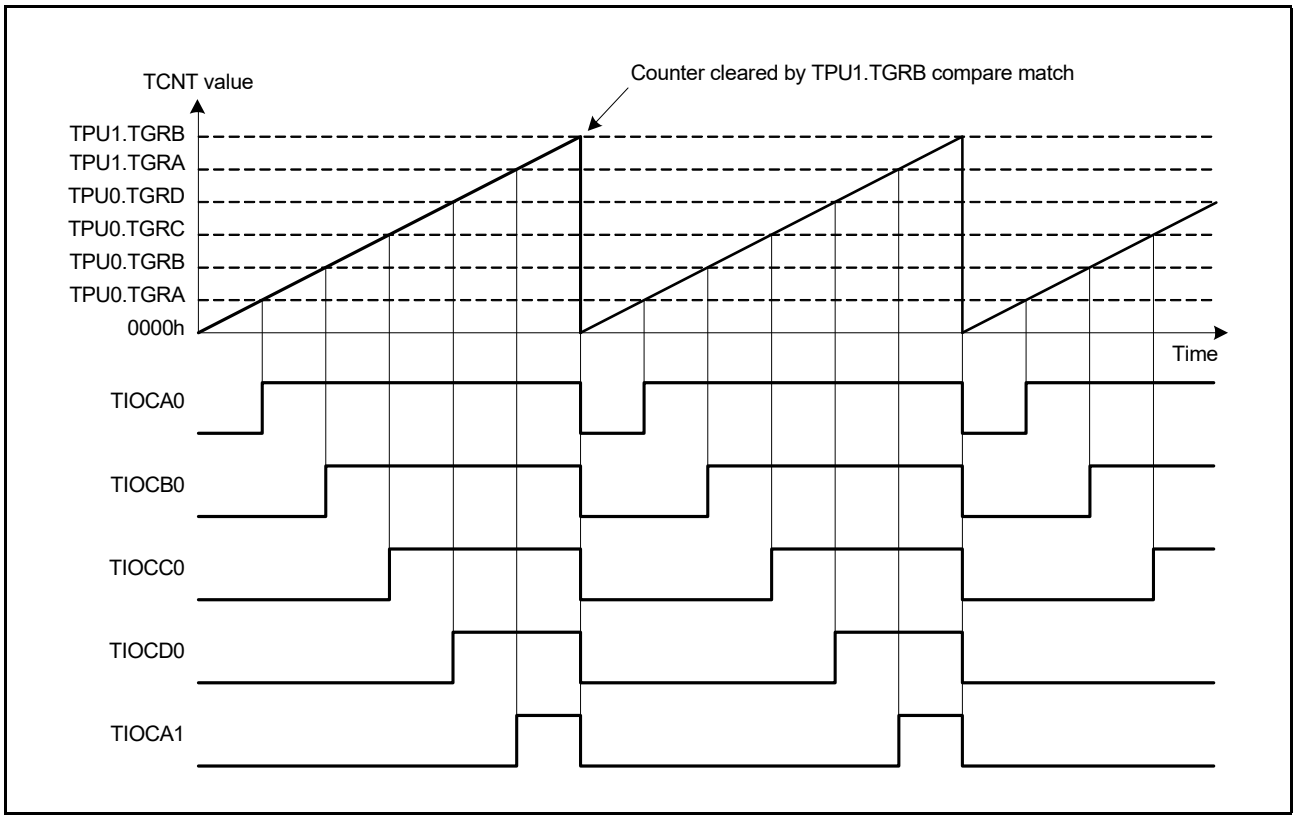


Figure 18.22 Example of PWM Mode Operation (2)

Figure 18.23 shows examples of PWM waveform output with 0% duty cycle and 100% duty cycle in PWM mode.

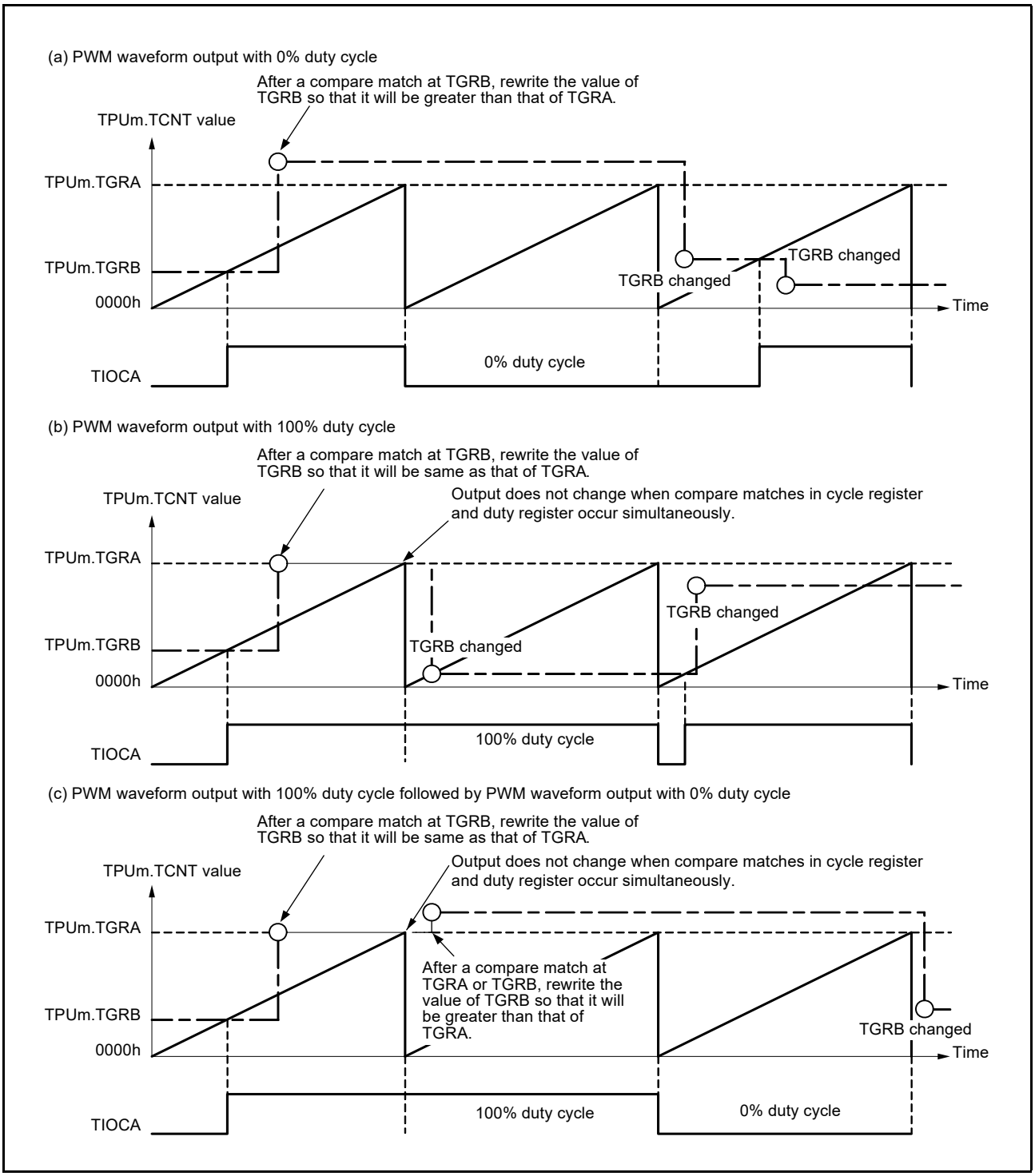


Figure 18.23 Example of PWM Mode Operation (3)

18.3.6 Phase Counting Mode

In phase counting mode, the phase difference between two external clock inputs is detected by the settings for channels 1, 2, 4, and 5 and TPUm.TCNT is incremented/decremented accordingly (m = 1, 2, 4, 5).
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 the TPUm.TCR.TPSC[2:0] and CKEG[1:0] bits. However, the lower 2 bits of the TPUm.TCR.CCLR[2:0] bits and the functions of TPUm.TIORH, TPUm.TIORL, TPUm.TIOR, TPUm.TIER, and TPUm.TGRy are valid, and therefore input capture/compare match and interrupt functions are available.
This can be used for two-phase encoder pulse input.
When an overflow occurs while TCNT is counting up, a TCIV interrupt request is generated; when an underflow occurs while TCNT is counting down, a TCIU interrupt request is generated. The TPUm.TSR.TCFD flag is the count direction flag. Reading the TCFD flag provides an indication of whether TCNT is counting up or down.
Table 18.24 lists the correspondence between external clock pins and channels.

Table 18.24 Clock Input Pins in Phase Counting Mode

	External Clock Pins	
Channel	**A-Phase**	**B-Phase**
When TPU1 or TPU5 is set to phase counting mode	TCLKA	TCLKB
When TPU2 or TPU4 is set to phase counting mode	TCLKC	TCLKD

(1) Example of Phase Counting Mode Setting Procedure

Figure 18.24 shows an example of the phase counting mode setting procedure.

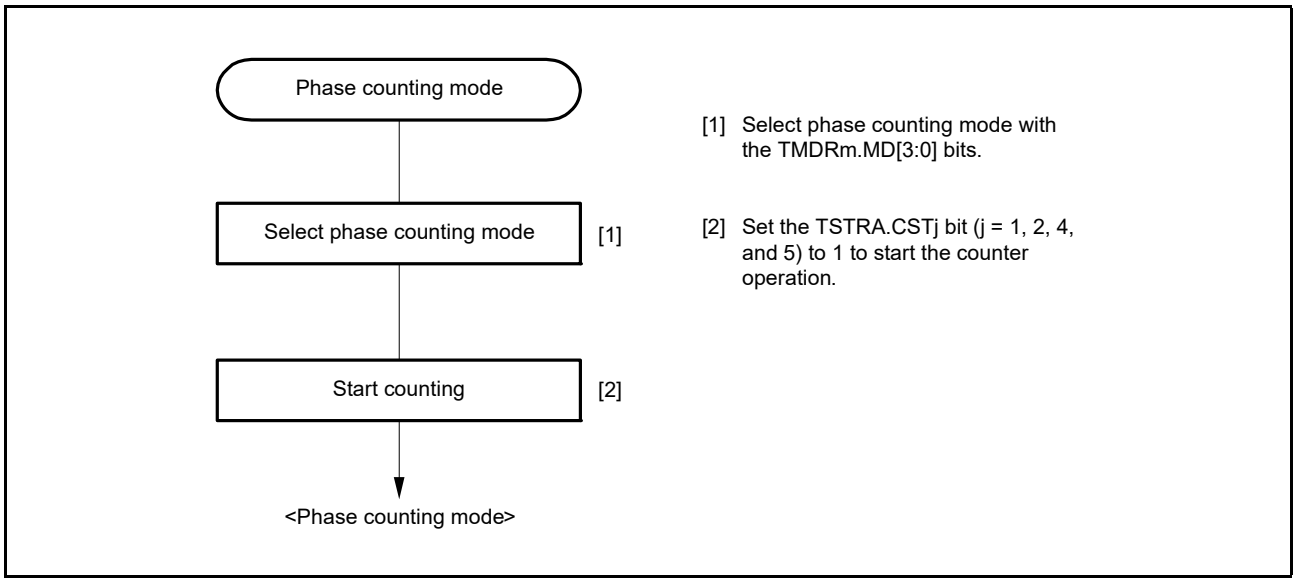


Figure 18.24 Example of Phase Counting Mode Setting Procedure

(2) Examples of Phase Counting Mode Operation

In phase counting mode, TPUm.TCNT (m = 1, 2, 4, 5) 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 18.25 shows an example of phase counting mode 1 operation , and Table 18.25 lists the TPUm.TCNT up-/down-count conditions (m = 1, 2, 4, 5).

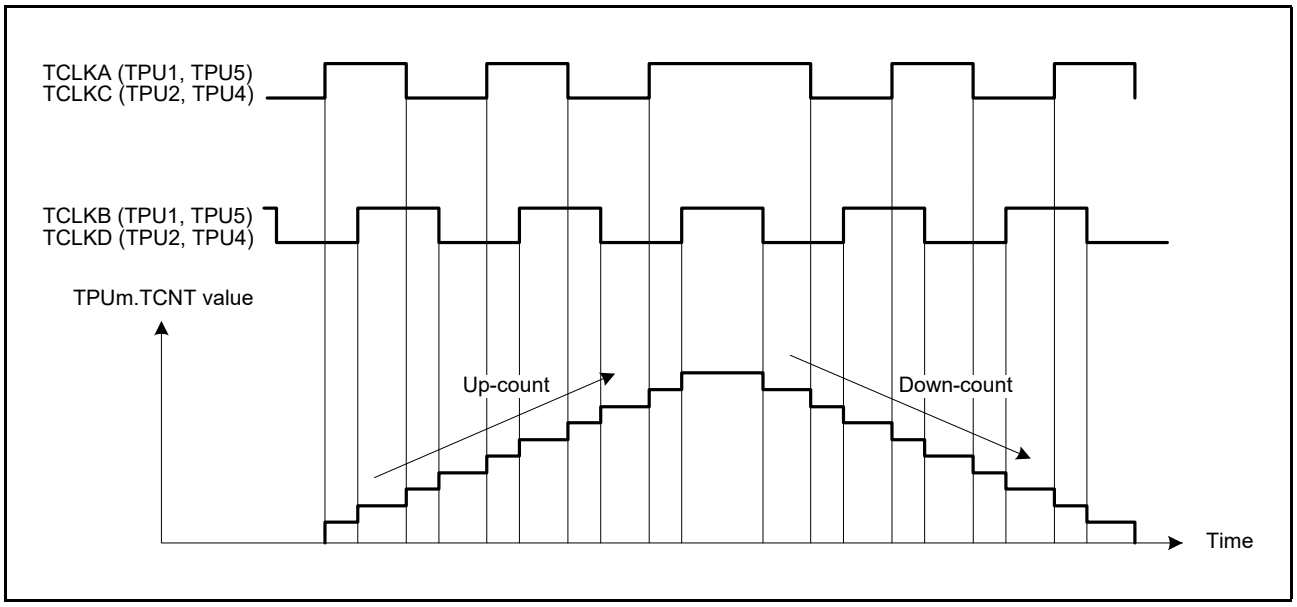


Figure 18.25 Example of Phase Counting Mode 1 Operation

Table 18.25 Up-/Down-Count Conditions in Phase Counting Mode 1

TCLKA (TPU1, TPU5) TCLKC (TPU2, TPU4)	TCLKB (TPU1, TPU5) TCLKD (TPU2, TPU4)	Operation
High	↑	Up-count
Low	↓	
↑	Low	
↓	High	
High	↓	Down-count
Low	↑	
↑	High	
↓	Low	

↑ : Rising edge

↓ : Falling edge

(b) Phase counting mode 2

Figure 18.26 shows an example of phase counting mode 2 operation , and Table 18.26 lists the TPUm.TCNT up-/down-count conditions (m = 1, 2, 4, 5).

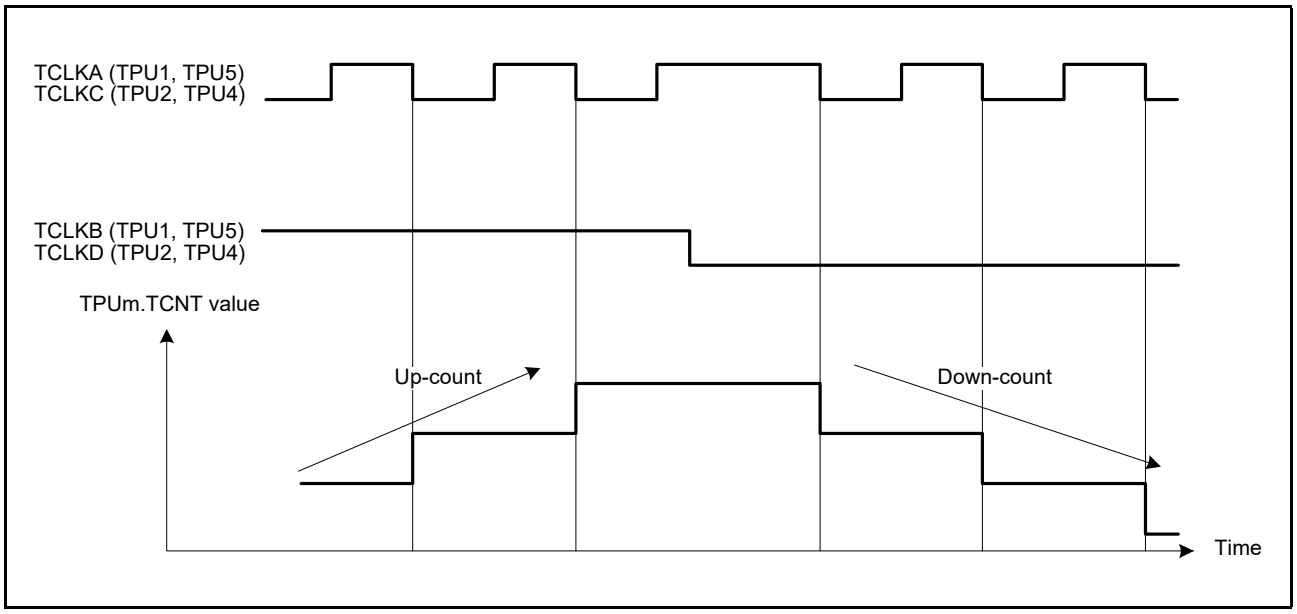


Figure 18.26 Example of Phase Counting Mode 2 Operation

Table 18.26 Up-/Down-Count Conditions in Phase Counting Mode 2

TCLKA (TPU1, TPU5) TCLKC (TPU2, TPU4)	TCLKB (TPU1, TPU5) TCLKD (TPU2, TPU4)	Operation
High	↑	Don't care
Low	↓	Don't care
↑	Low	Don't care
↓	High	Up-count
High	↓	Don't care
Low	↑	Don't care
↑	High	Don't care
↓	Low	Down-count

↑ : Rising edge

↓ : Falling edge

(c) Phase counting mode 3

Figure 18.27 shows an example of phase counting mode 3 operation, and Table 18.27 lists the TPUm.TCNT up-/down-count conditions (m = 1, 2, 4, 5).

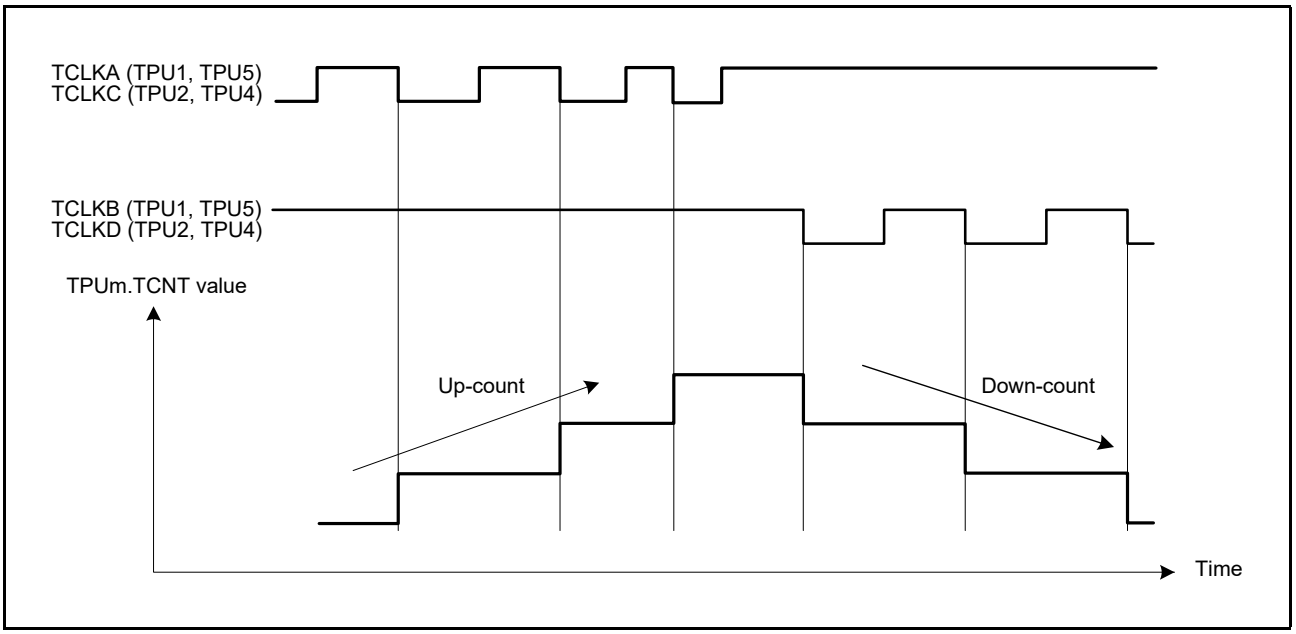


Figure 18.27 Example of Phase Counting Mode 3 Operation

Table 18.27 Up-/Down-Count Conditions in Phase Counting Mode 3

TCLKA (TPU1, TPU5) TCLKC (TPU2, TPU4)	TCLKB (TPU1, TPU5) TCLKD (TPU2, TPU4)	Operation
High	↑	Don't care
Low	↓	Don't care
↑	Low	Don't care
↓	High	Up-count
High	↓	Down-count
Low	↑	Don't care
↑	High	Don't care
↓	Low	Don't care

↑ : Rising edge

↓ : Falling edge

(d) Phase counting mode 4

Figure 18.28 shows an example of phase counting mode 4 operation, and Table 18.28 lists the TPUm.TCNT up-/down-count conditions (m = 1, 2, 4, 5).

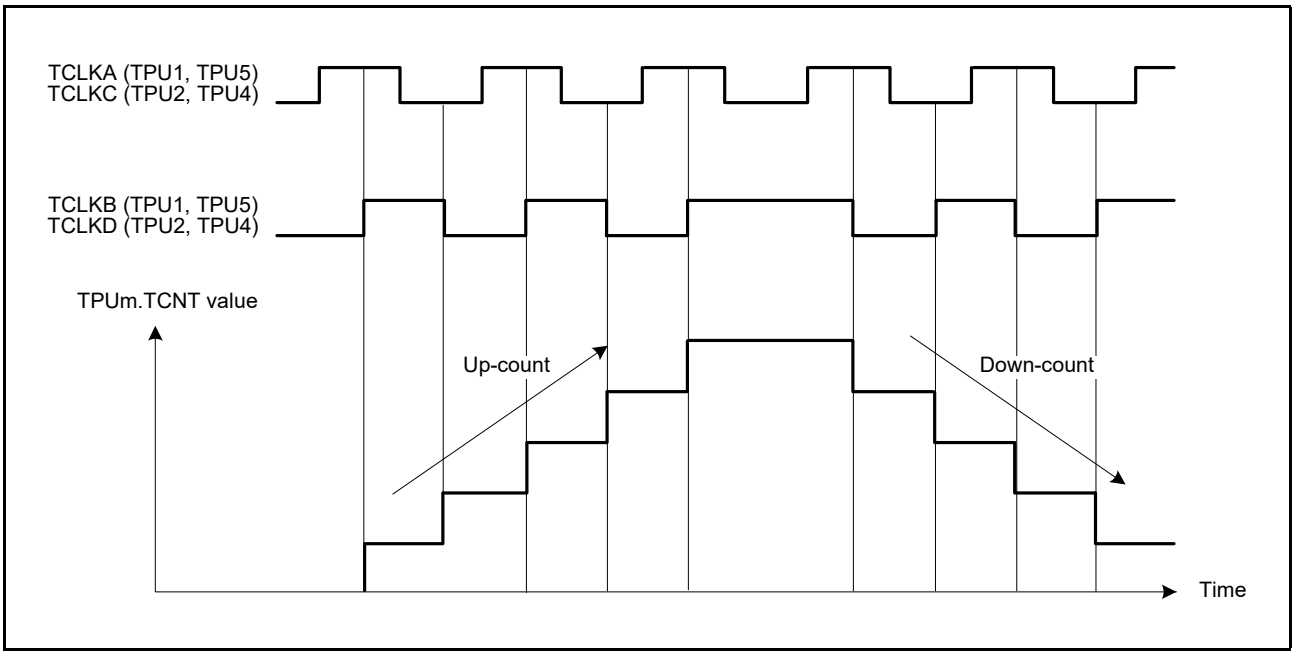


Figure 18.28 Example of Phase Counting Mode 4 Operation

Table 18.28 Up-/Down-Count Conditions in Phase Counting Mode 4

TCLKA (TPU1, TPU5) TCLKC (TPU2, TPU4)	TCLKB (TPU1, TPU5) TCLKD (TPU2, TPU4)	Operation
High	↑	Up-count
Low	↓	
↑	Low	Don't care
↓	High	
High	↓	Down-count
Low	↑	
↑	High	Don't care
↓	Low	

↑ : Rising edge

↓ : Falling edge

18.3.6.1 Phase Counting Mode Application Example

Figure 18.29 shows an example in which phase counting mode is set for TPU1, and TPU1 is coupled with TPU0 to input servo motor 2-phase encoder pulses in order to detect the position or speed.

In this example, TPU1 is set to phase counting mode 1, and the encoder pulse A-phase and B-phase are input to the TCLKA and TCLKB pins.

TPU0 operates with TPU0.TCNT counter clearing by TPU0.TGR compare match; TPU0.TGRA and TPU0.TGRC are used for the compare match function and are set with the speed control cycle and position control cycle. TPU0.TGRB is used for input capture, with TPU0.TGRB and TPU0.TGRD operating in buffer mode. The TPU1 counter input clock is specified as the source to drive input capture in TPU0.TGRB. The TPU0 counter values are input to and captured by TPU0.TGRB in response to counting up or down by the TPU1.TCNT counter (with the previously captured value being transferred to TPU0.TGRD), allowing measurement of edge occurrence (pulse width measurement).

TPU1.TGRA and TPU1.TGRB for TPU1 are specified for input capture, TPU0.TGRA and TPU0.TGRC compare matches are selected as the input capture source, and the up-/down-counter values for the control cycles are stored. This procedure enables accurate position/speed detection to be achieved.

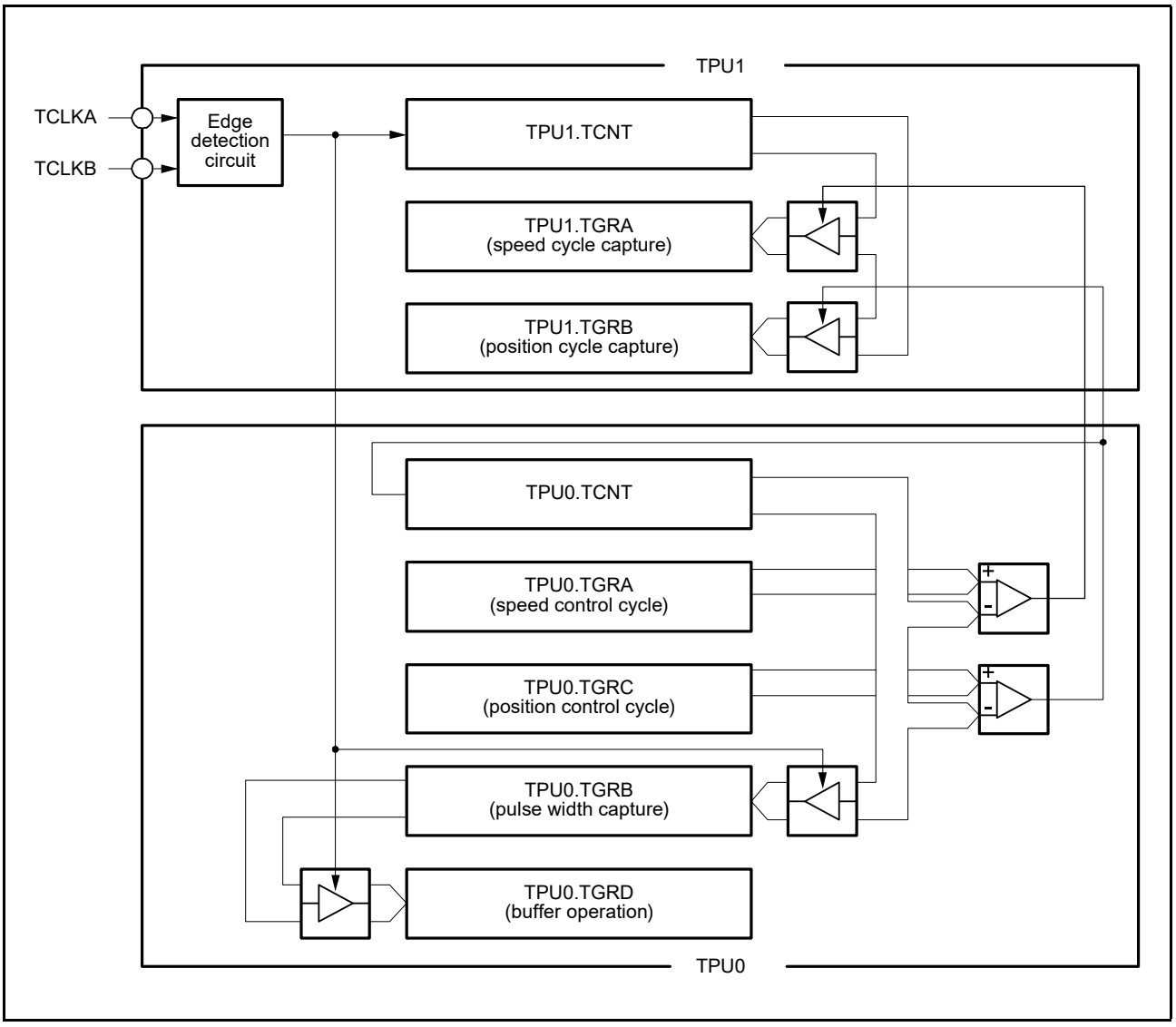


Figure 18.29 Phase Counting Mode Application Example

18.3.7 Noise Filters

Each pin for use in input capture by TPU is equipped with a noise filter. The noise filter samples input signals at the frequency of the sampling clock and pulses with levels that only match once or twice are removed.
The noise filter functionality includes enabling and disabling of the noise filter for each pin and setting of the sampling clock for each channel.
Figure 18.30 is a timing chart for the noise filter.

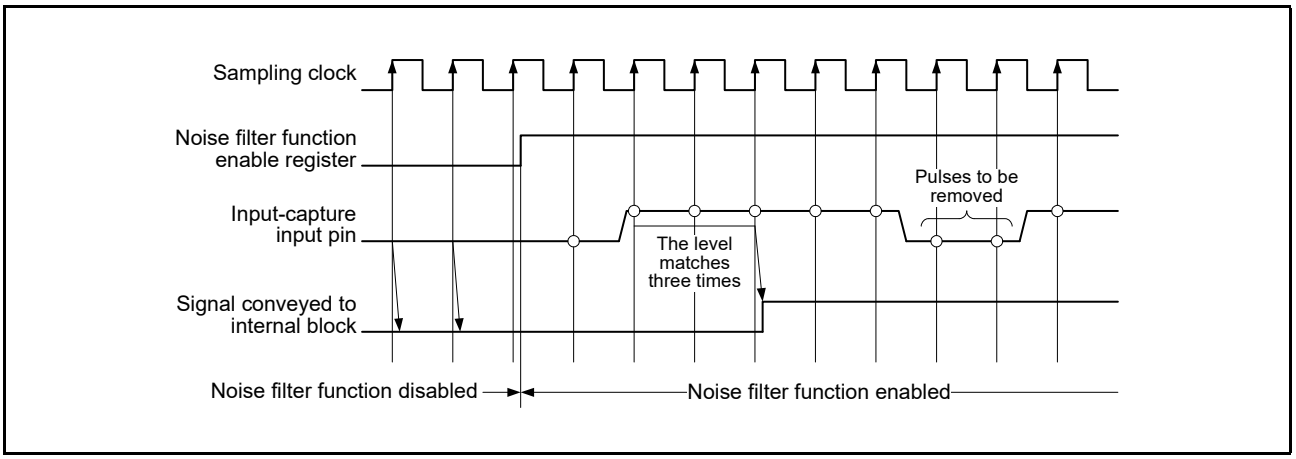


Figure 18.30 Timing Chart for the Noise Filter

If noise filtering is set, input capture operation is performed on the edges of noise-filtered signal after a minimum delay of (sampling interval × 2 + PCLKD) due to noise filtering for the input capture input.

18.4 Interrupt Sources

There are three kinds of TPU interrupt sources: TPUm.TGRy input capture/compare match, TPUm.TCNT overflow, and TPUm.TCNT underflow.

Relative channel priority levels can be changed by the interrupt controller, but the priority within a channel is fixed. For details, see section 12, Interrupt Controller (ICUA).

Table 18.29 lists the TPU interrupt sources.

Table 18.29 TPU Interrupt Sources

Channel	Name	Interrupt Source	DMAC Activation
TPU0	TGI0A	TPU0.TGRA input capture/compare match	Possible
	TGI0B	TPU0.TGRB input capture/compare match	Possible
	TGI0C	TPU0.TGRC input capture/compare match	Not possible
	TGI0D	TPU0.TGRD input capture/compare match	Not possible
	TCI0V	TPU0.TCNT overflow	Not possible
TPU1	TGI1A	TPU1.TGRA input capture/compare match	Possible
	TGI1B	TPU1.TGRB input capture/compare match	Possible
	TCI1V	TPU1.TCNT overflow	Not possible
	TCI1U	TPU1.TCNT underflow	Not possible
TPU2	TGI2A	TPU2.TGRA input capture/compare match	Possible
	TGI2B	TPU2.TGRB input capture/compare match	Possible
	TCI2V	TPU2.TCNT overflow	Not possible
	TCI2U	TPU2.TCNT underflow	Not possible
TPU3	TGI3A	TPU3.TGRA input capture/compare match	Possible
	TGI3B	TPU3.TGRB input capture/compare match	Possible
	TGI3C	TPU3.TGRC input capture/compare match	Not possible
	TGI3D	TPU3.TGRD input capture/compare match	Not possible
	TCI3V	TPU3.TCNT overflow	Not possible
TPU4	TGI4A	TPU4.TGRA input capture/compare match	Possible
	TGI4B	TPU4.TGRB input capture/compare match	Possible
	TCI4V	TPU4.TCNT overflow	Not possible
	TCI4U	TPU4.TCNT underflow	Not possible
TPU5	TGI5A	TPU5.TGRA input capture/compare match	Possible
	TGI5B	TPU5.TGRB input capture/compare match	Possible
	TCI5V	TPU5.TCNT overflow	Not possible
	TCI5U	TPU5.TCNT underflow	Not possible

Note: This table lists the initial state immediately after a reset. The relative channel priority levels can be changed by the interrupt controller.

(1) Input Capture/Compare Match Interrupt

A TGImy interrupt (m = 0 to 5) is requested when the TPUm.TIER.TGIEy bit (y = A, B, C, D) is set to 1 by the occurrence of a TPUm.TGRy input capture/compare match on a channel. The TPU has 16 input capture/compare match interrupts, four each for TPU0 and TPU3, and two each for TPU1, TPU2, TPU4, and TPU5.

(2) Overflow Interrupt

A TCImV interrupt (m = 0 to 5) is requested when the TPUm.TIER.TCIEV bit is set to 1 by the occurrence of a TPUm.TCNT overflow on a channel. The TPU has 6 overflow interrupts, one for each channel.

(3) Underflow Interrupt

A TCImU interrupt (m = 0 to 4) is requested when the TPUm.TIER.TCIEU bit is set to 1 by the occurrence of a TPUm.TCNT underflow on a channel. The TPU has four underflow interrupts, one each for TPU1, TPU2, TPU4, and TPU5.

18.5 DMAC Activation

Input capture/compare match interrupts from the TPUm.TGRA and TPUm.TGRB registers can activate the corresponding DMACs. For details, see section 14, DMA Controller (DMACAa).
A total of twelve TPUm.TGRA, TGRB input capture/compare match interrupts can be used as DMAC activation sources, two for each channel (m = 0 to 5)

18.6 A/D Converter Activation

The TPU can activate the A/D converter by the TPUm.TGRA input capture/compare match for each channel (m = 0 to 4).
When the TPUm.TIER.TTGE bit is set to 1, the TPU requests the A/D converter to start A/D conversion by the occurrence of a TPUm.TGRA input capture/compare match on a particular channel (m = 0 to 4).
For the corresponding unit of A/D converter, see section 30, 12-Bit A/D Converter (S12ADCa).

18.7 Operation Timing

18.7.1 Input/Output Timing

(1) TPUm.TCNT Count Timing

Figure 18.31 shows TPUm.TCNT count timing in internal clock operation, and Figure 18.32 shows TCNT count timing in external clock operation (m = 0 to 5).

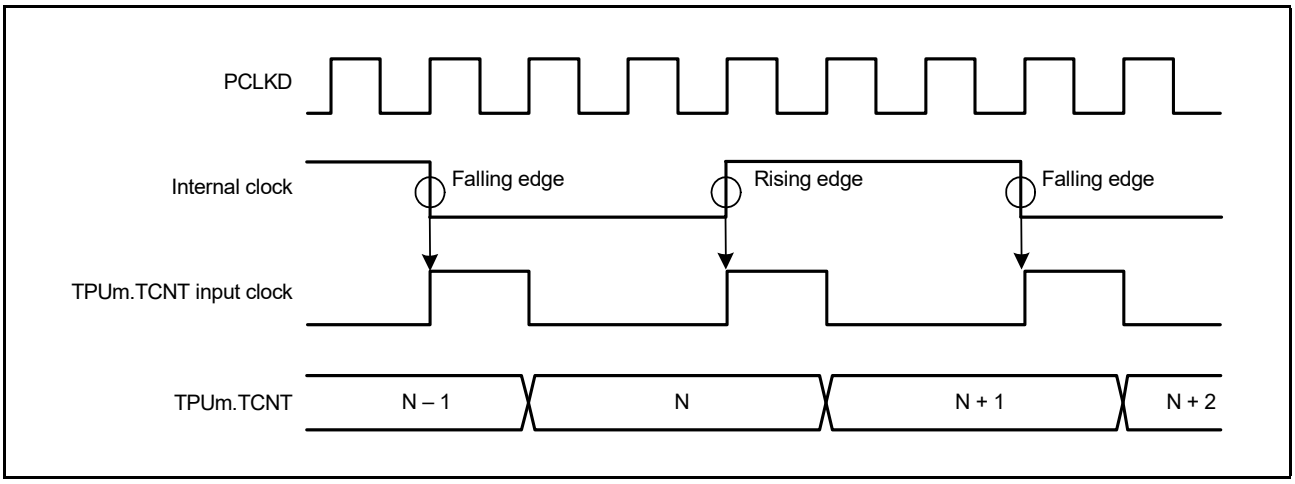


Figure 18.31 Count Timing in Internal Clock Operation

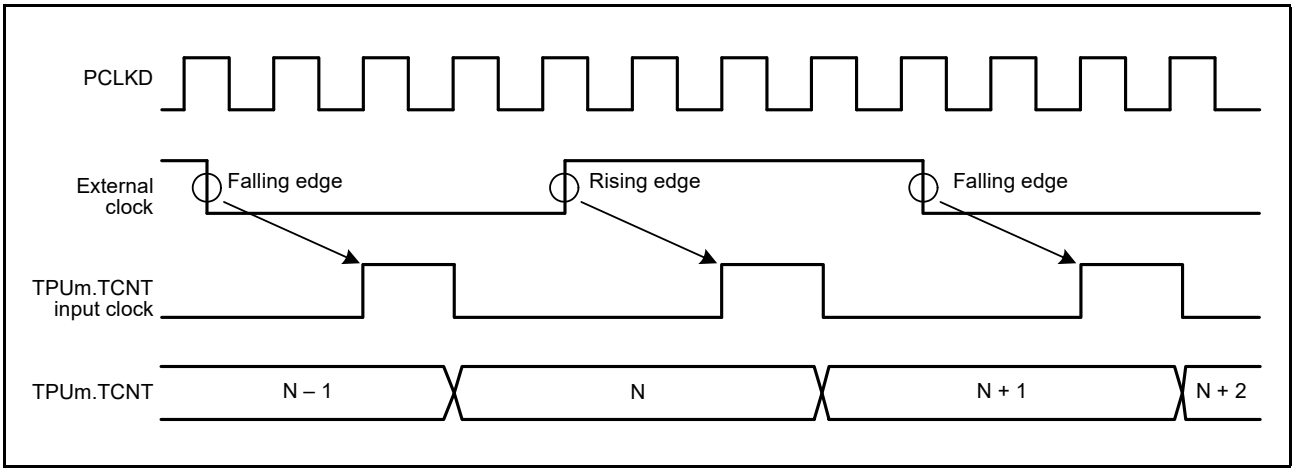


Figure 18.32 Count Timing in External Clock Operation

(2) Output Compare Output Timing

A compare match signal is generated in the final state in which TPUm.TCNT and TPUm.TGRy 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 TPUm.TIORH, TPUm.TIORL, or TPUm.TIOR is output to the output compare output pin TIOCyn (y = A to D when n = 0, 3; y = A, B when n = 1, 2, 4, 5). After a match between TCNT and TGRy, the compare match signal is not generated until the TCNT input clock is generated.

Figure 18.33 shows output compare output timing.

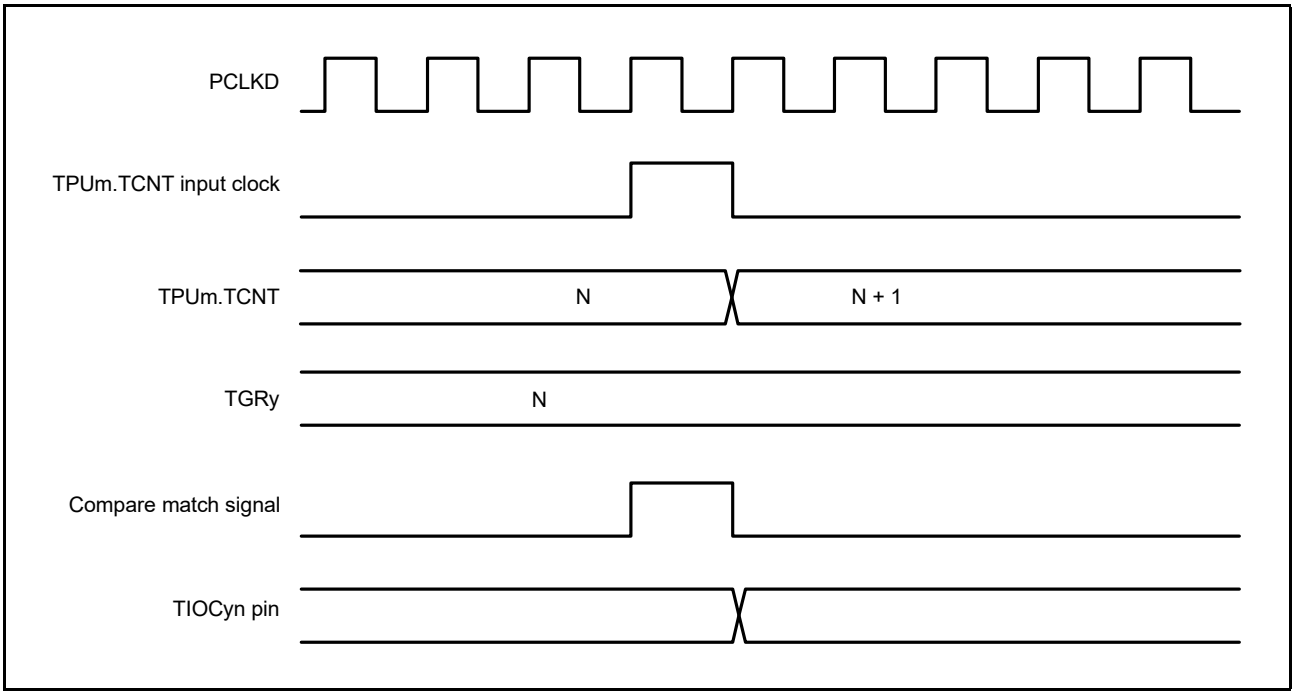


Figure 18.33 Output Compare Output Timing

(3) Input Capture Signal Timing

Figure 18.34 shows input capture signal timing.

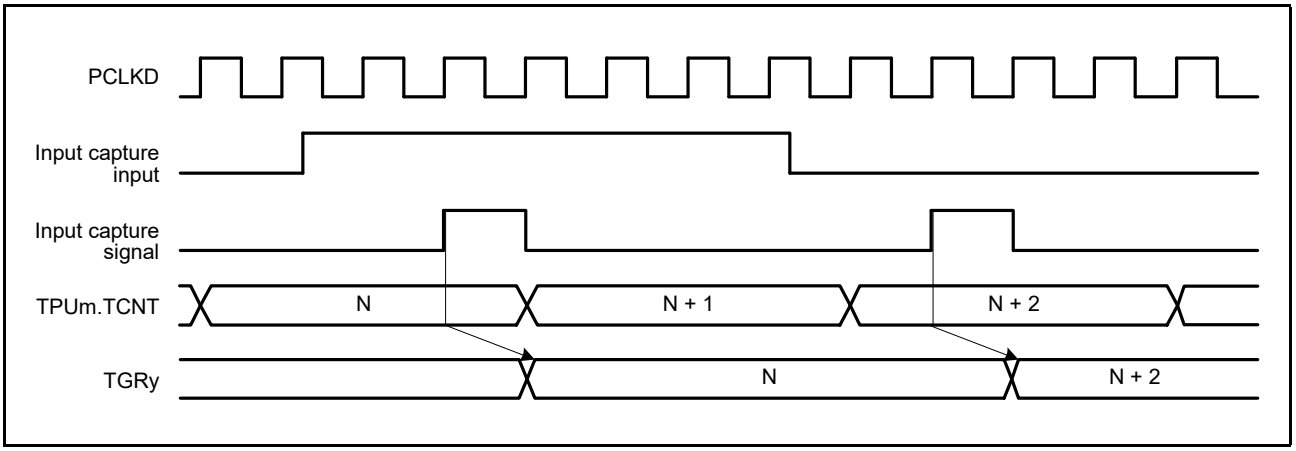


Figure 18.34 Input Capture Signal Timing

(4) Timing for Counter Clearing by Compare Match/Input Capture

Figure 18.35 shows the timing when counter clearing by compare match occurrence is specified, and Figure 18.36 shows the timing when counter clearing by input capture occurrence is specified.

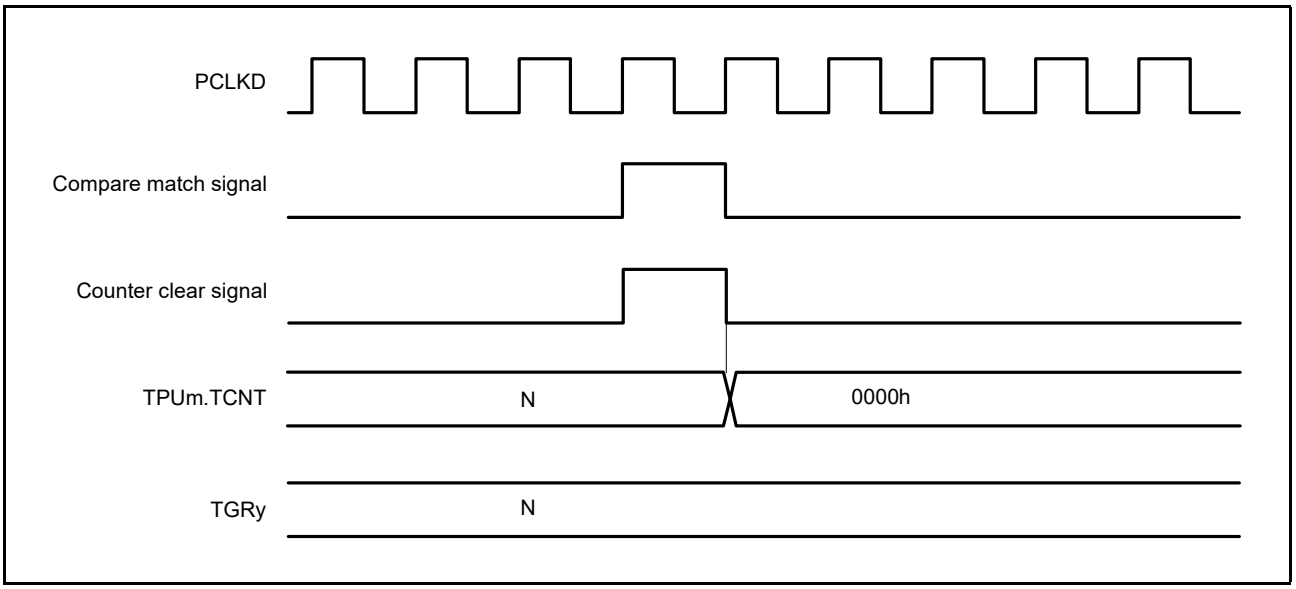


Figure 18.35 Counter Clear Timing (Compare Match)

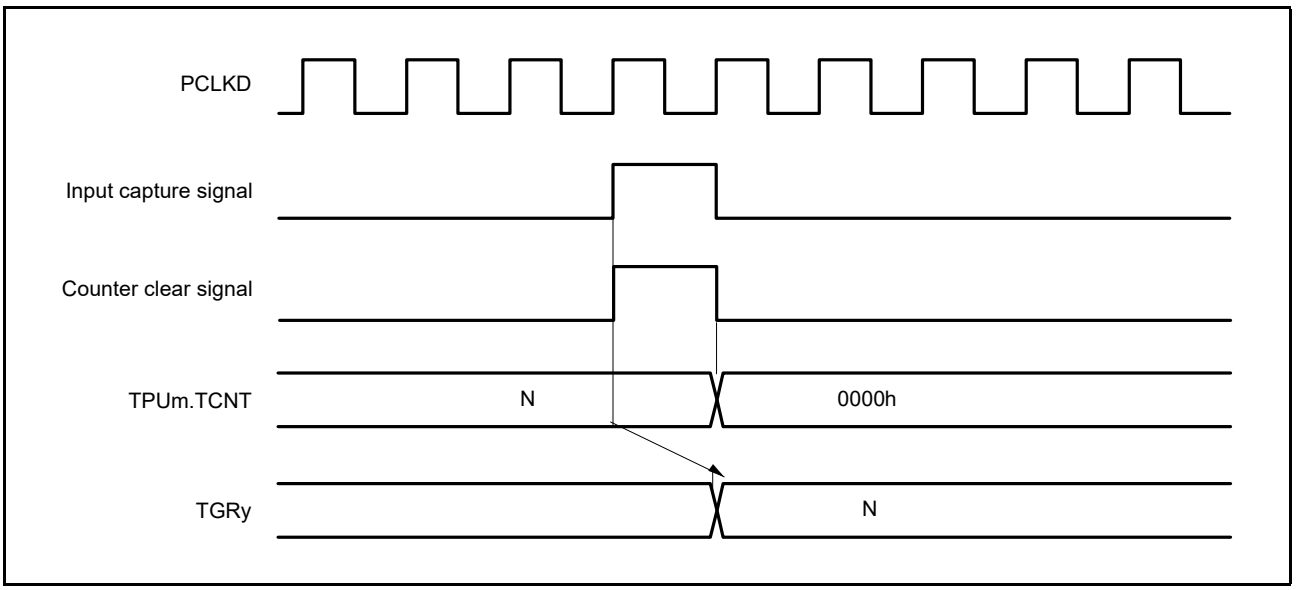


Figure 18.36 Counter Clear Timing (Input Capture)

(5) Buffer Operation Timing

Figure 18.37 and Figure 18.38 show the timings in buffer operation.

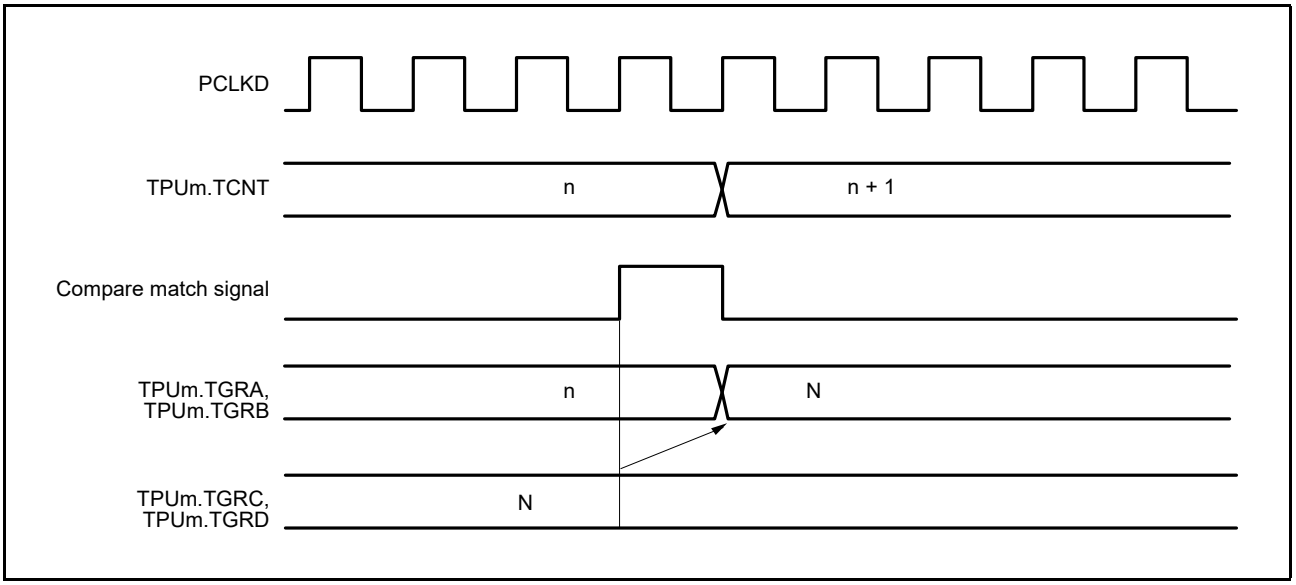


Figure 18.37 Buffer Operation Timing (Compare Match)

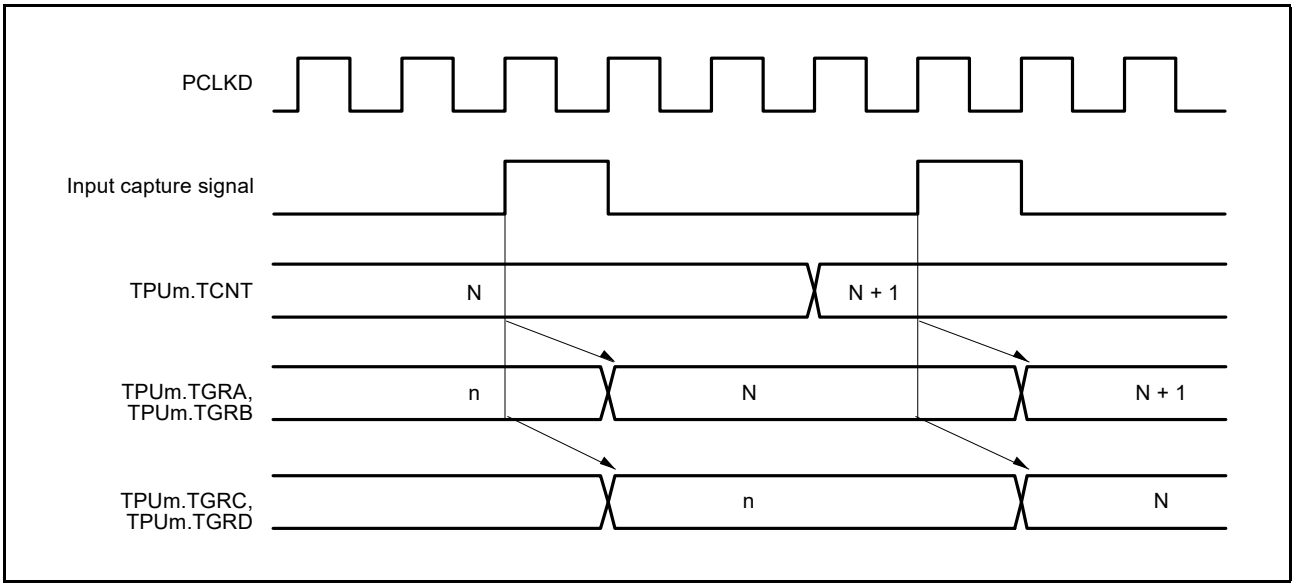


Figure 18.38 Buffer Operation Timing (Input Capture)

18.7.2 Interrupt Signal Timing

(1) Timing of Interrupt Signal Setting on Compare Match

Figure 18.39 shows the timing for setting the TGImy interrupt signal by compare match occurrence (y = A to D when m = 0, 3; y = A, B when m = 1, 2, 4, 5).

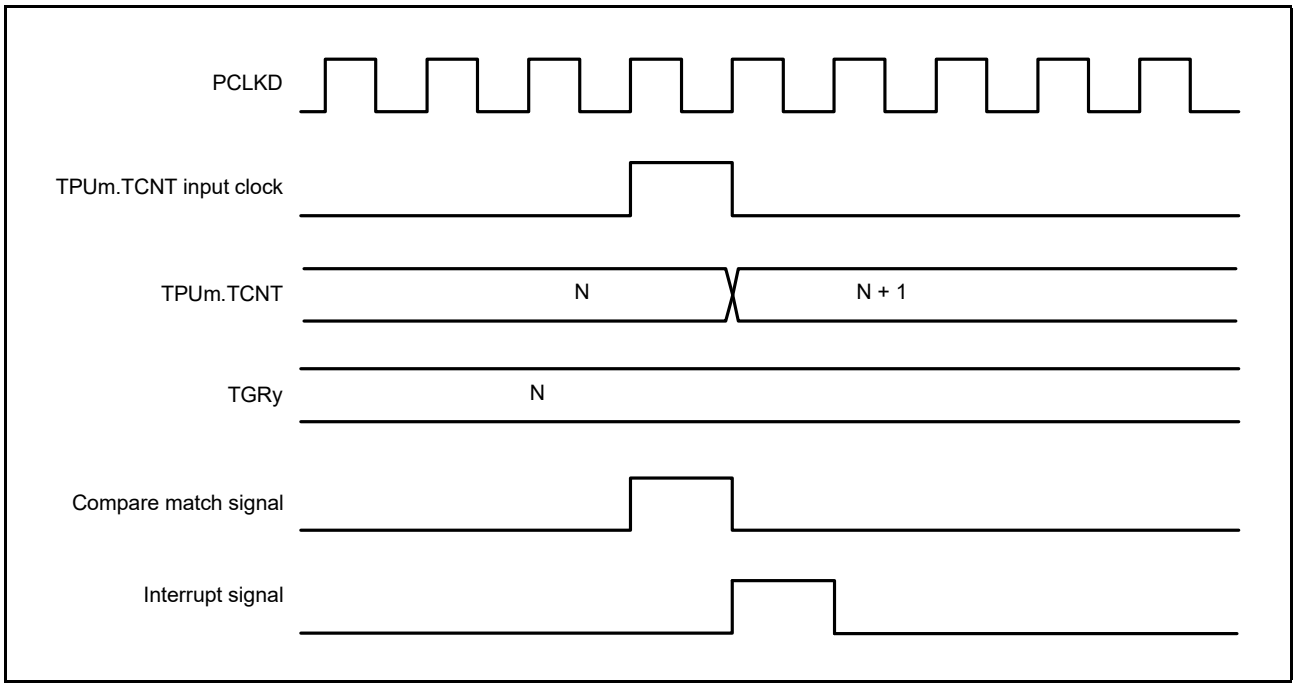


Figure 18.39 TGImy Interrupt Timing (Compare Match)

(2) Timing of Interrupt Signal Setting on Input Capture

Figure 18.40 shows the timing for setting the TGImy interrupt signal by input capture occurrence (y = A to D when m = 0, 3; y = A, B when m = 1, 2, 4, 5).

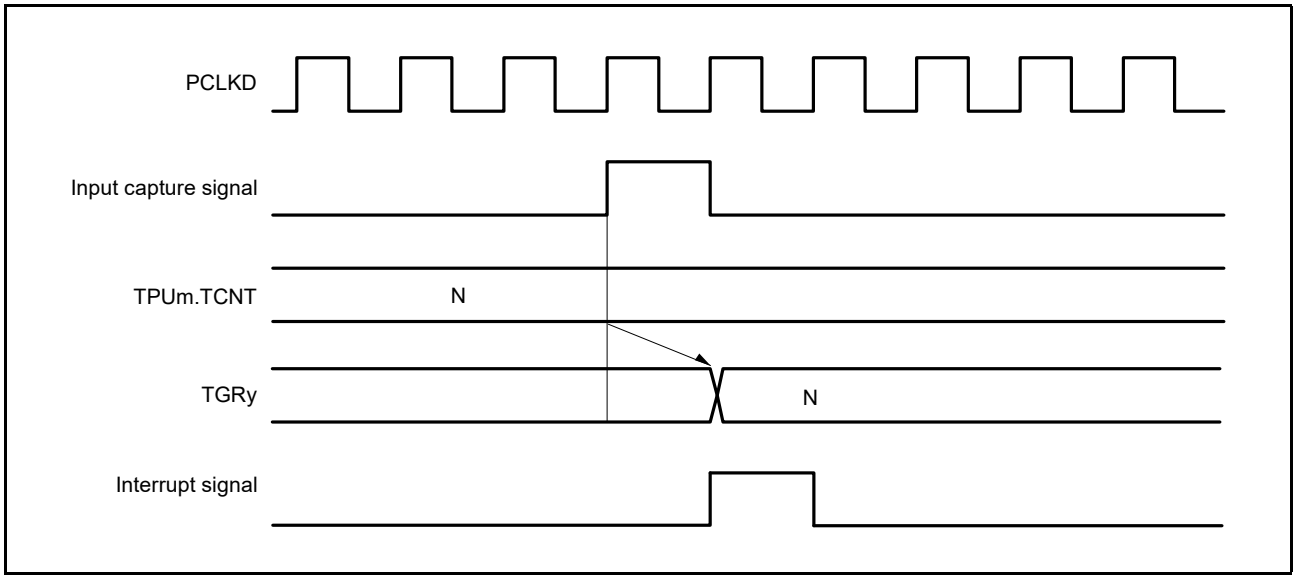


Figure 18.40 TGImy Interrupt Timing (Input Capture)

(3) Timing of TCImV/TCImU Interrupt Signal Setting

Figure 18.41 shows the timing for generating the TCImV interrupt signal by overflow occurrence (m = 0 to 5).
Figure 18.42 shows the timing for generating the TCInU interrupt signal by underflow occurrence (n = 1, 2, 4, 5).

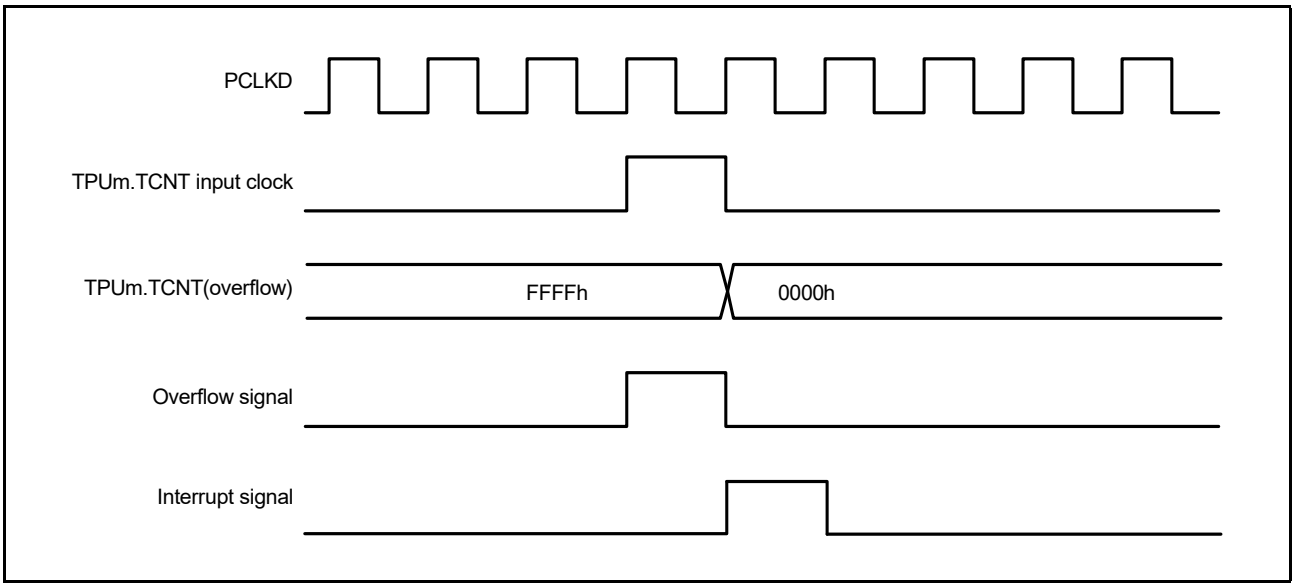


Figure 18.41 TCImV Interrupt Setting Timing

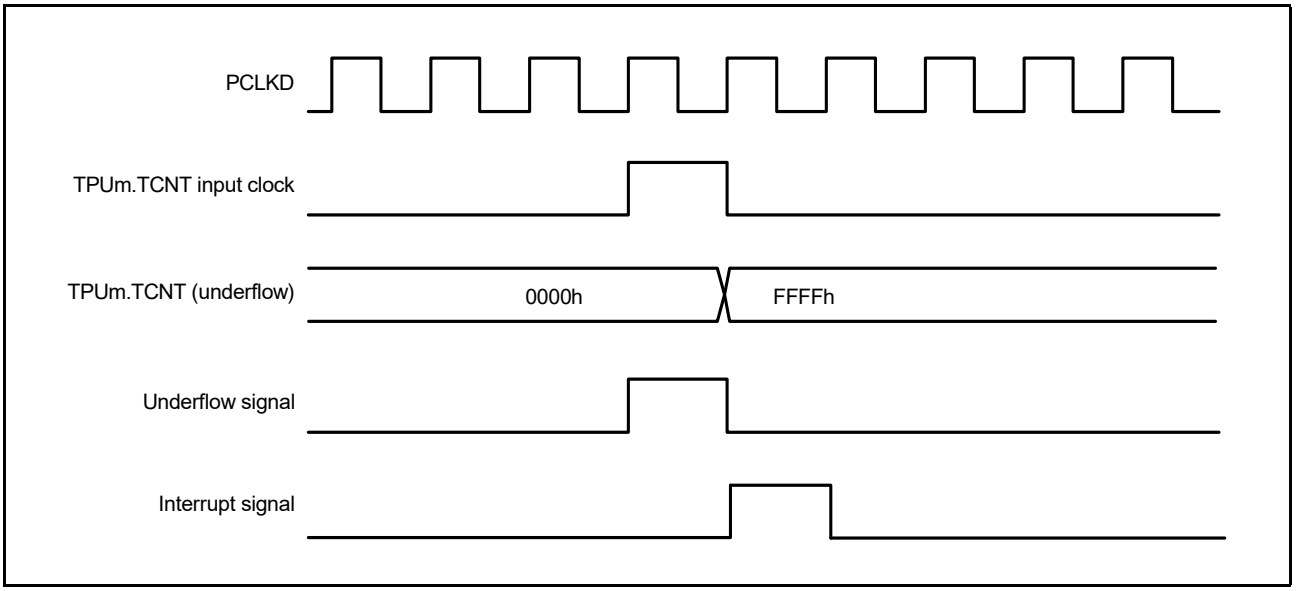


Figure 18.42 TCImU Interrupt Setting Timing

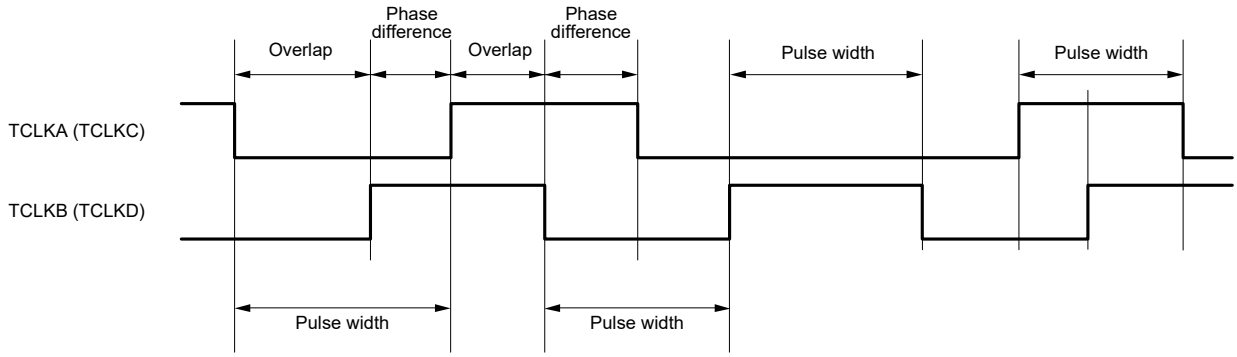
18.8 Usage Notes

18.8.1 Module-Stop Function Setting

Operation of the TPU can be disabled or enabled using the module-stop control register. The TPU does not operate with the initial setting. Register access is enabled by clearing module-stop state. For details, see section 9, Low-Power Consumption Function.

18.8.2 Input Clock Restrictions

The input clock pulse width must be at least 1.5 cycles of PCLKD in the case of single-edge detection, and at least 2.5 cycles of PCLKD in the case of both-edge detection. The TPU will not operate properly with a narrower pulse width. In phase counting mode, the phase difference and overlap between the two input clocks must be at least 1.5 cycles of PCLKD, and the pulse width must be at least 2.5 cycles of PCLKD. Figure 18.43 shows the input clock conditions in phase counting mode.



Note: Phase difference or overlap: At least 1.5 cycles of PCLKD
Pulse width : At least 2.5 cycles of PCLKD

Figure 18.43 Phase Difference, Overlap, and Pulse Width in Phase Counting Mode

18.8.3 Caution on Cycle Setting

When counter clearing by compare match is set, TPUm.TCNT is cleared in the final state in which it matches the TGRy value (the point at which the count value matched by TCNT is updated). Consequently, the actual counter frequency is given by the following formula (y = A to D when m = 0, 3; y = A, B when m = 1, 2, 4, 5).

$$f = \frac{f_{TCNT\_CLK}}{(N+1)}$$

f : Counter frequency
f_{TCNT_CLK} : Counter clock frequency
N : TGRy set value

18.8.4 Conflict between TPUm.TCNT Write and Clear Operations

If the counter clearing signal is generated in a TPUm.TCNT write cycle, TPUm.TCNT clearing takes precedence and the TPUm.TCNT write is not performed (m = 0 to 5). Figure 18.44 shows the timing in this case.

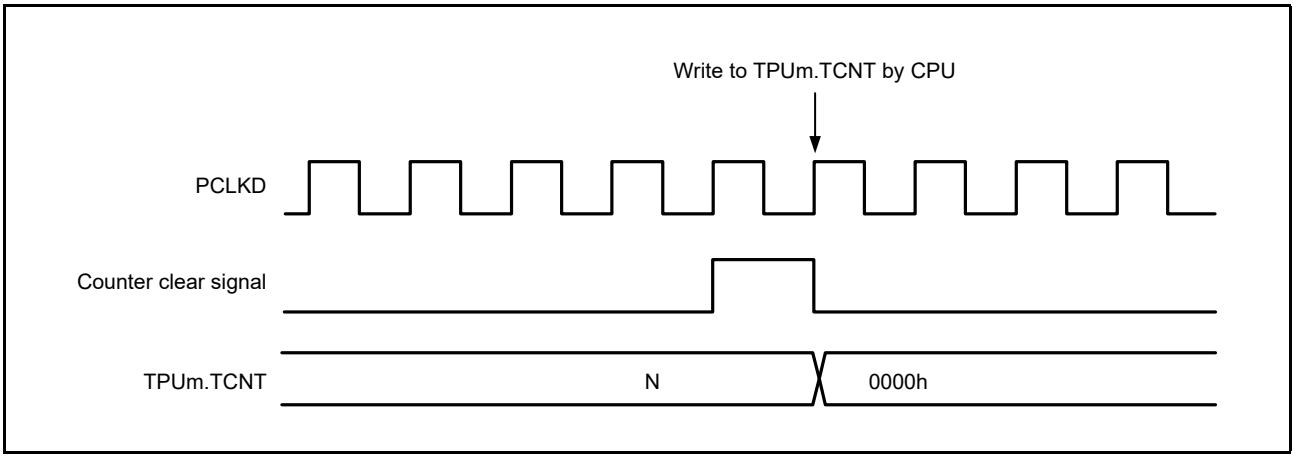


Figure 18.44 Conflict between TPUm.TCNT Write and Clear Operations

18.8.5 Conflict between TPUm.TCNT Write and Increment Operations

If incrementing occurs in a TPUm.TCNT write cycle, the TPUm.TCNT write takes precedence and TPUm.TCNT is not incremented (m = 0 to 5). Figure 18.45 shows the timing in this case.

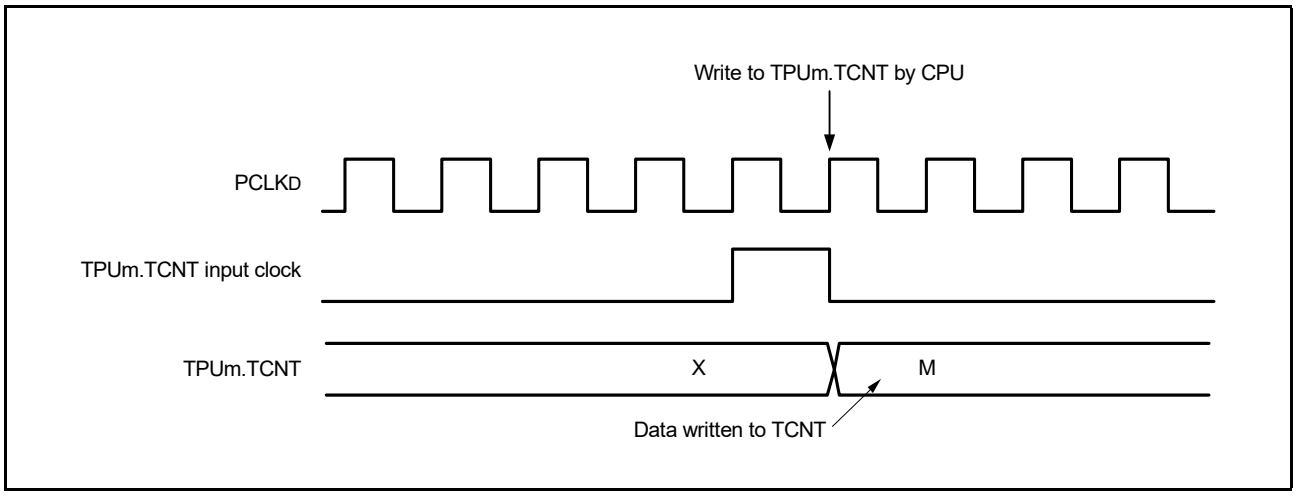


Figure 18.45 Conflict between TPUm.TCNT Write and Increment Operations

18.8.6 Conflict between TPUm.TGRy Write and Compare Match

If a compare match occurs in a TGRy write cycle, the TGRy write takes precedence and the compare match signal is disabled. A compare match also does not occur when the same value as before is written (y = A to D when m = 0, 3; y = A, B when m = 1, 2, 4, 5).
Figure 18.46 shows the timing in this case.

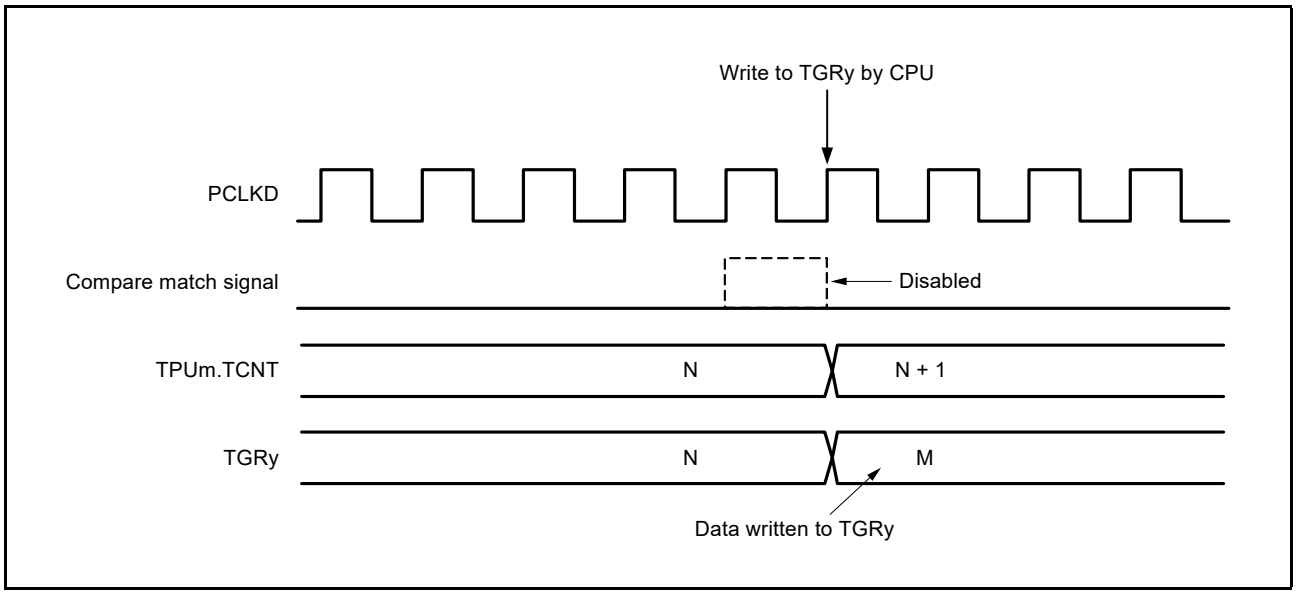


Figure 18.46 Conflict between TGRy Write and Compare Match

18.8.7 Conflict between Buffer Register Write and Compare Match

If a compare match occurs in a TGRy write cycle, the data transferred to TGRy by the buffer operation will be the data before writing (y = A to D when m = 0, 3; y = A, B when m = 1, 2, 4, 5).
Figure 18.47 shows the timing in this case.

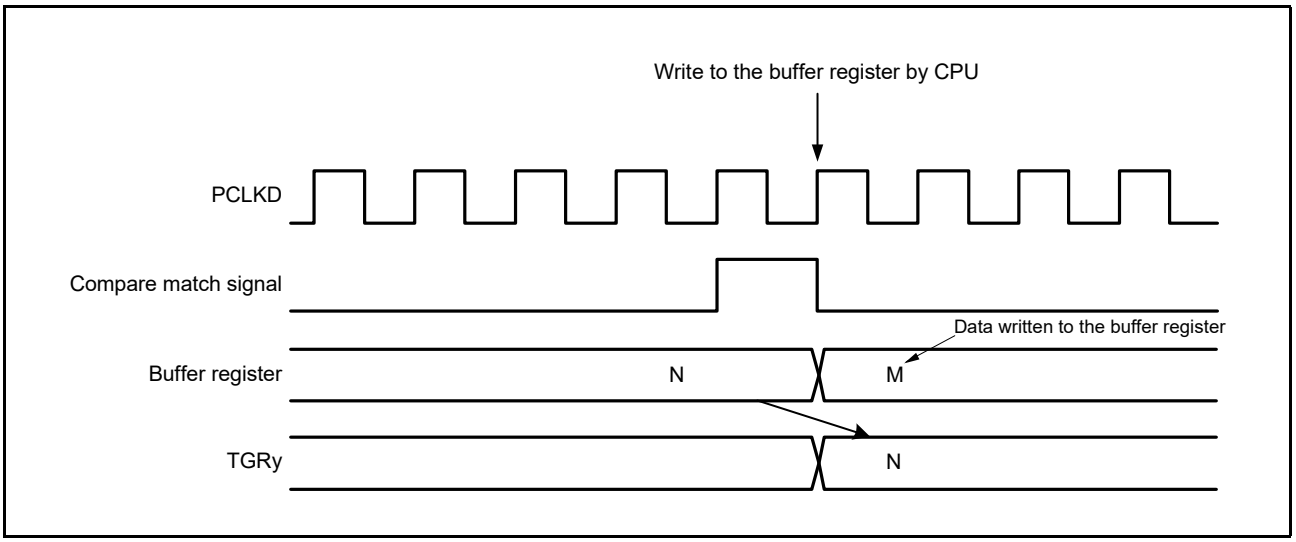


Figure 18.47 Conflict between Buffer Register Write and Compare Match

18.8.8 Conflict between TPUm.TGRy Read and Input Capture

If the input capture signal is generated in a TGRy read cycle, the data that is read will be the data before input capture transfer (y = A to D when m = 0, 3; y = A, B when m = 1, 2, 4, 5).
Figure 18.48 shows the timing in this case.

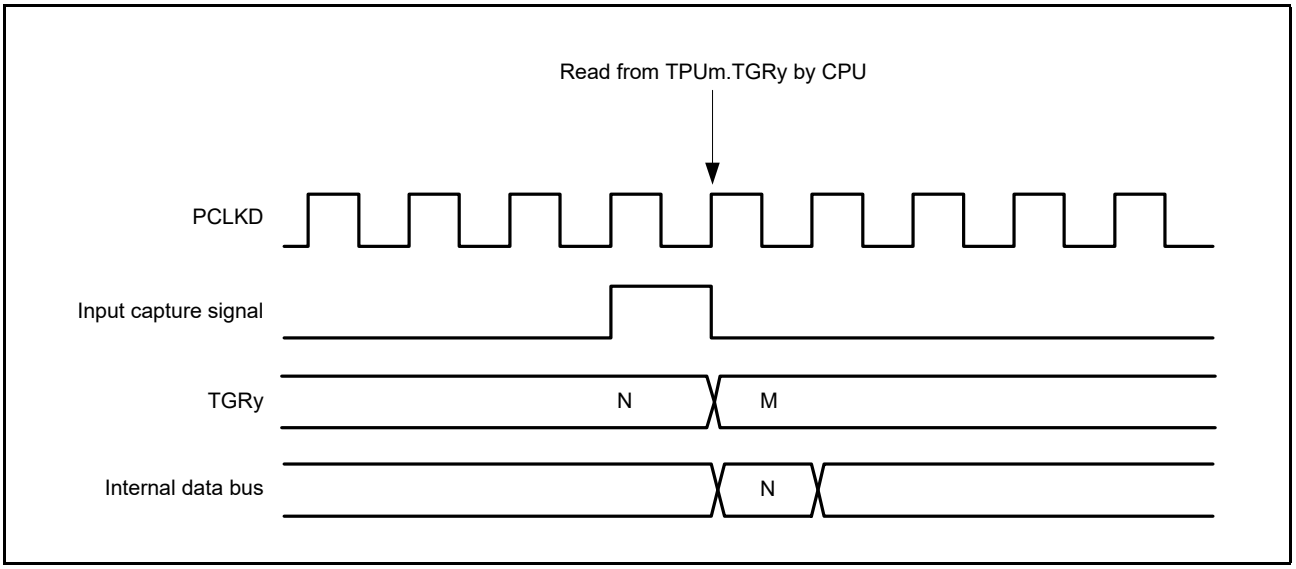


Figure 18.48 Conflict between Reading from the TPUm.TGRy Register and Input Capture

18.8.9 Conflict between TPUm.TGRy Write and Input Capture

If the input capture signal is generated in a TGRy write cycle, the input capture operation takes precedence and the write to TGRy is not performed (y = A to D when m = 0, 3; y = A, B when m = 1, 2, 4, 5). Figure 18.49 shows the timing in this case.

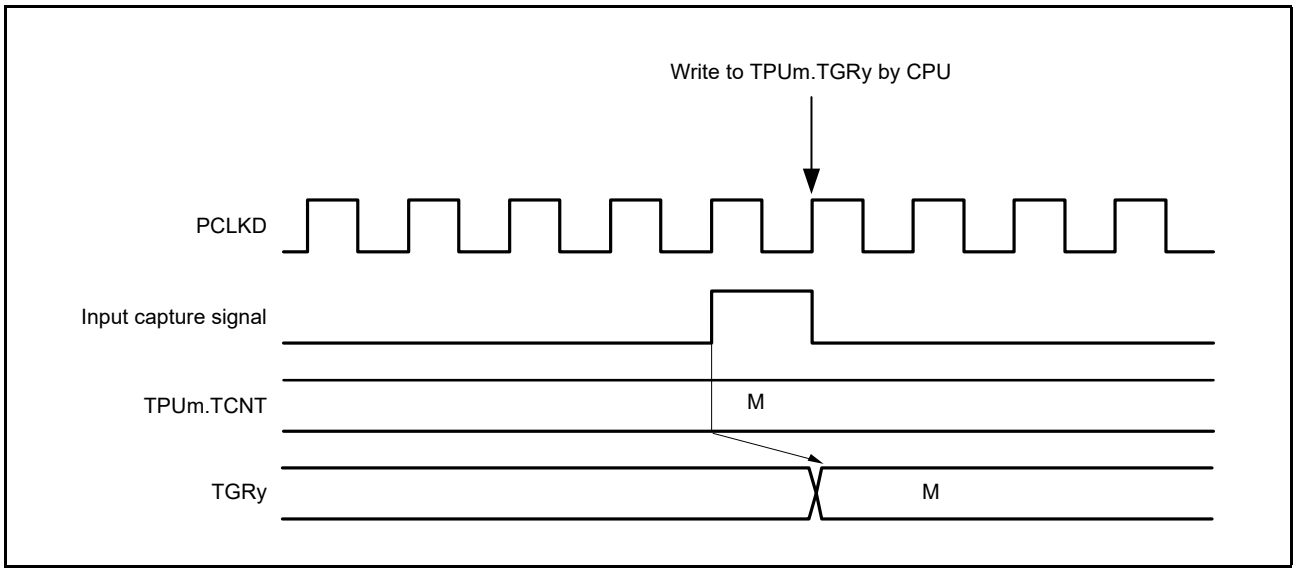


Figure 18.49 Conflict between Writing to the TPUm.TGRy Register and Input Capture

18.8.10 Conflict between Buffer Register Write and Input Capture

If the input capture signal is generated in a buffer register write cycle, the buffer operation takes precedence and the write to the buffer register is not performed (y = A to D when m = 0, 3; y = A, B when m = 1, 2, 4, 5). Figure 18.50 shows the timing in this case.

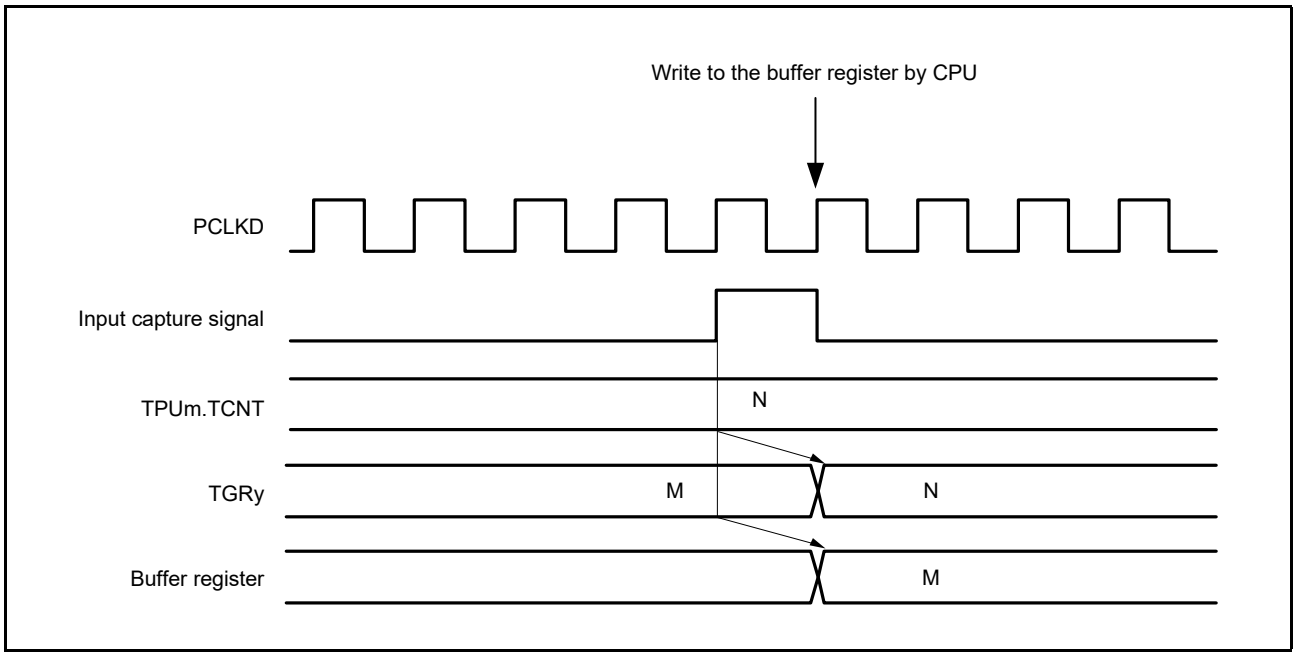


Figure 18.50 Conflict between Buffer Register Write and Input Capture

18.8.11 Conflict between Overflow/Underflow and Counter Clearing

If overflow/underflow and counter clearing*1 occur simultaneously, TPUm.TCNT counter is cleared and an overflow interrupt (TCImV)/underflow interrupt (TCInU) is generated (m = 0 to 5, n = 1, 2, 4, 5).

Figure 18.51 shows the operation timing when a TPUm.TGRy compare match is specified as the clearing source and FFFFh is set in TGRy (y = A to D when m = 0, 3; y = A, B when m = 1, 2, 4, 5).

Note 1. There are four counter clearing sources:
- Compare match
- Input capture
- Synchronous clearing
- Counter clear operation by an event signal

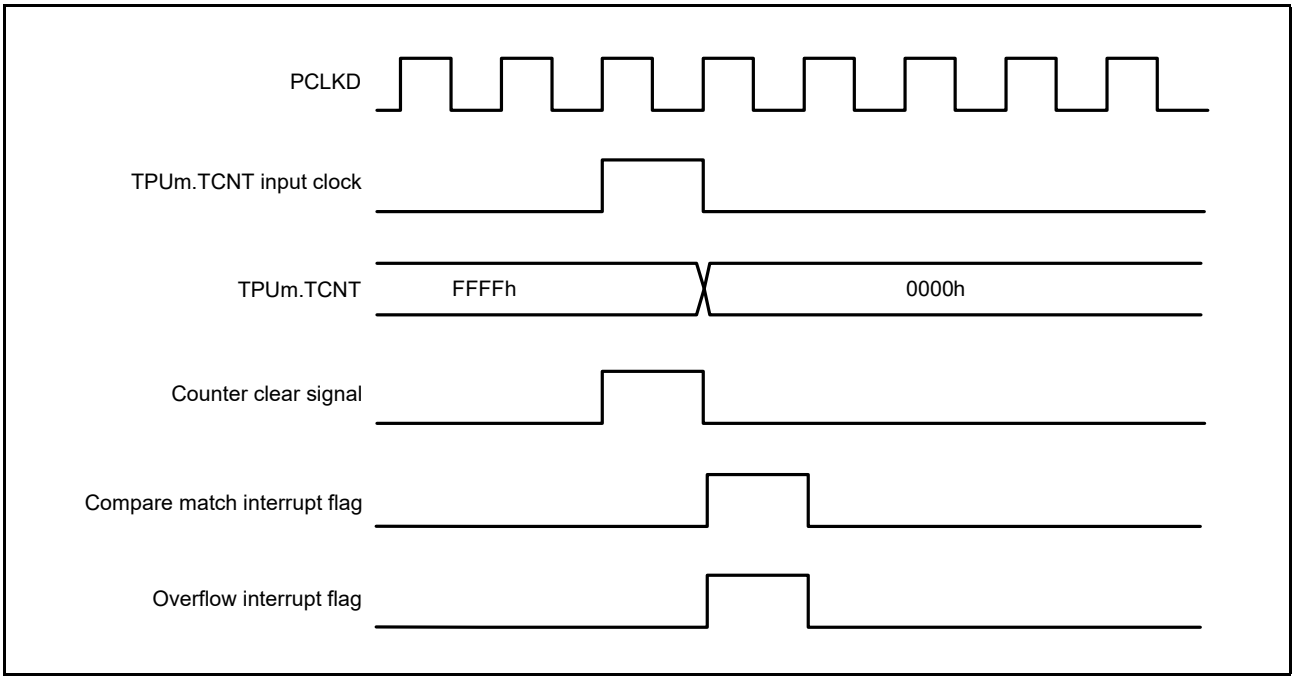


Figure 18.51 Conflict between Overflow and Counter Clearing

18.8.12 Conflict between TPUm.TCNT Write and Overflow/Underflow

If an overflow/underflow occurs due to increment/decrement in a TPUm.TCNT write cycle, the TPUm.TCNT write takes precedence.

Figure 18.52 shows the operation timing when there is conflict between TPUm.TCNT write and overflow (m = 0 to 5).

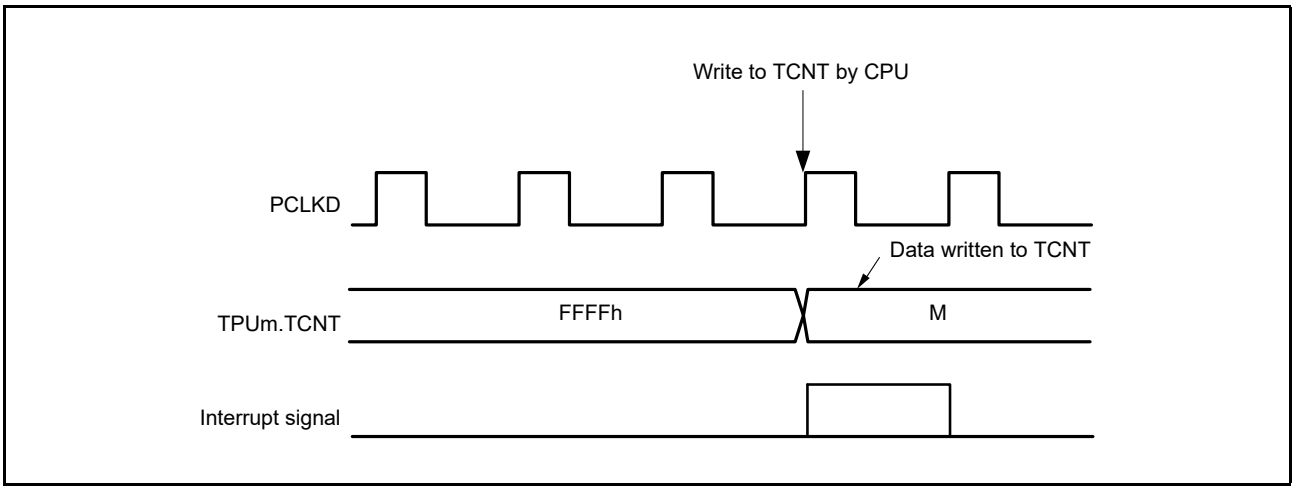


Figure 18.52 Conflict between TCNT Write and Overflow

18.8.13 Continuous Output of Interrupt Signal in Response to Compare Match

When TGRy is set to 0000h, PCLKD/1 is set as the counter clock, and compare match is set as the trigger for clearing of the counter clock, the value of the counter (TCNT) counter remains 0000h, and the interrupt signal will be output continuously (i.e. its level will be flat) rather than output over a single cycle. Consequently, interrupts will not be detected in response to second and subsequent compare matches. (y = A to D)

Figure 18.53 shows the timing for continuous output of the interrupt signal in response to a compare match.

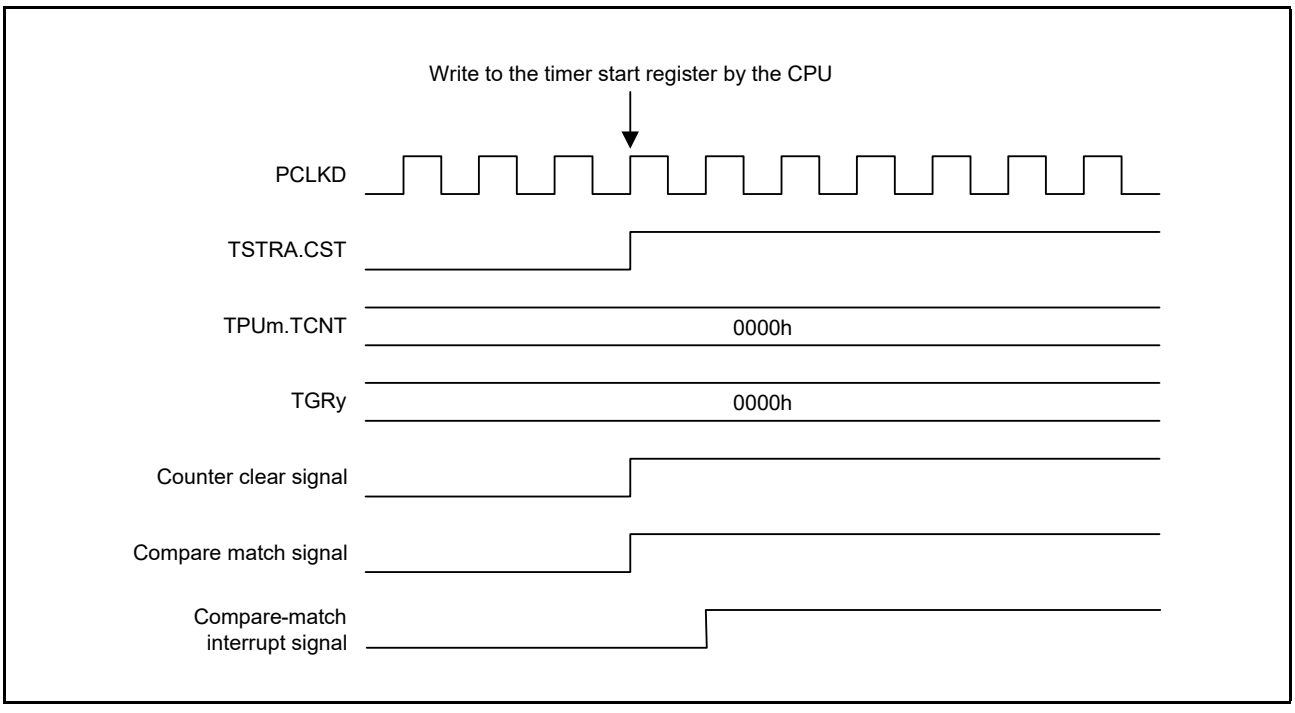


Figure 18.53 Continuous Output of the Interrupt Signal in Response to a Compare Match

18.8.14 Continuous Output of Interrupt Signal in Response to Input Capture

When input-capture signal is set on both edges and when the pulse width of the input-capture input equals to one PCLKD cycle detected by internal sampling, input capture is generated continuously on the rising and falling edges. Therefore, the interrupt signal will be output continuously (i.e. its level will be flat) rather than output over a single cycle. Consequently, interrupts will not be detected in response to second and subsequent input capture.
Figure 18.54 shows the timing for continuous output of the interrupt signal in response to input capture.

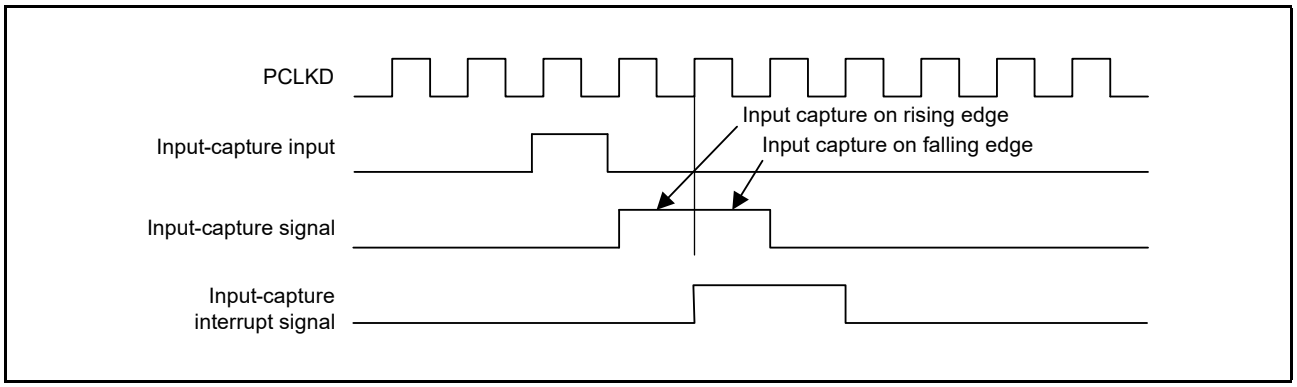


Figure 18.54 Continuous Output of the Interrupt Signal in Response to Input Capture

18.8.15 Continuous Output of Interrupt Signal in Response to Underflow

If two external clock signals' same direction edges to be phase counted are generated within two PCLKD cycles in phase counting mode 1, with TGRy being 0000h, and compare match set as the counter clear source, the TPUm.TCNT counter remains 0000h, and a compare-match interrupt signal and an underflow interrupt signal are output continuously to form a flat signal level (y = A to D, m = 1, 2, 4, 5). Consequently, interrupts will not be detected in response to second and subsequent compare match and underflow.

Figure 18.55 shows the timing for continuous output of the interrupt signal in response to underflow.

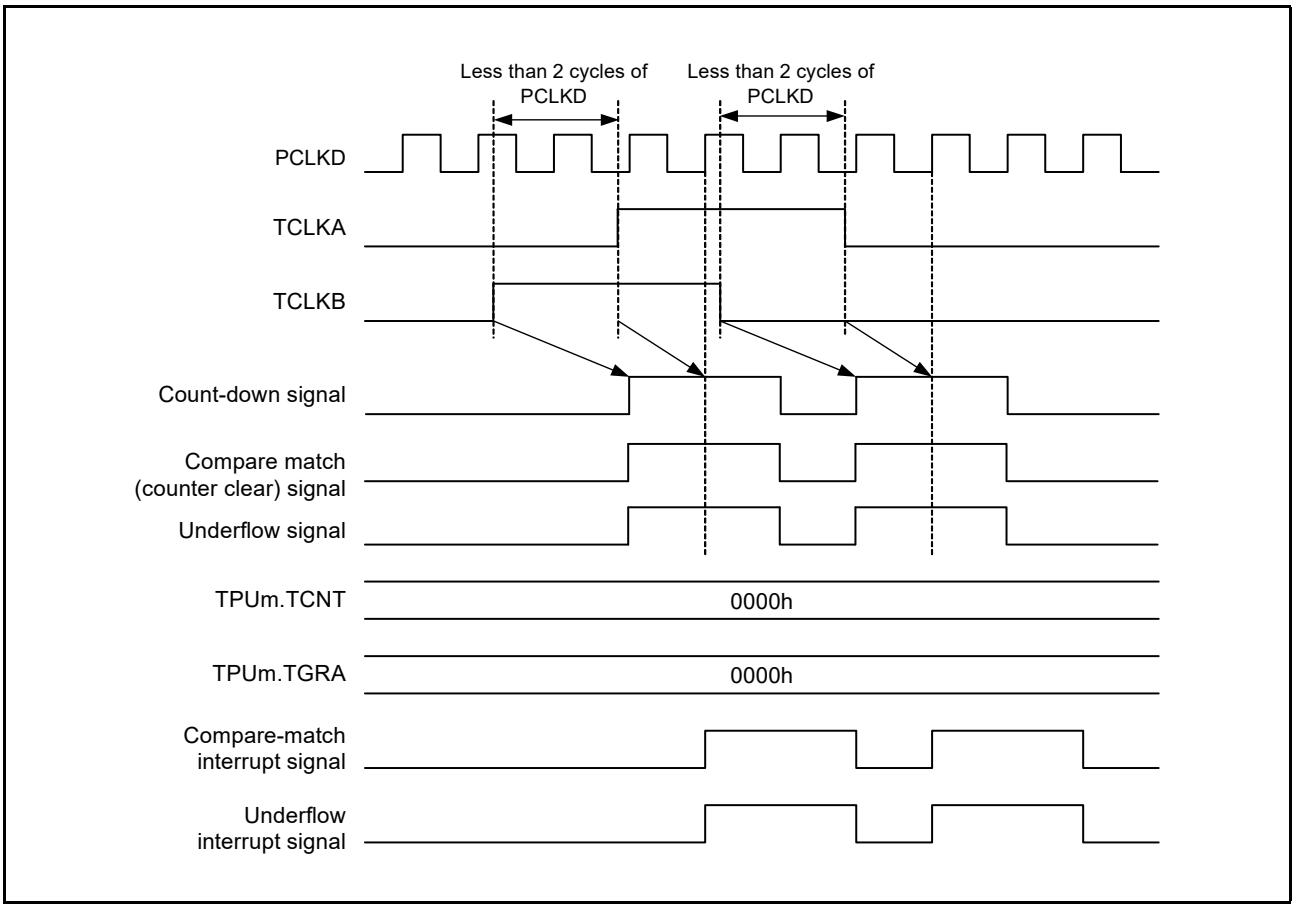


Figure 18.55 Continuous Output of the Interrupt Signal in Response to Underflow

18.8.16 Input Capture Operation in Cascaded Operation

When two 16-bit counters are cascade-connected, with the counter for the 16 higher-order bits (TPUm.TCNT; m = 1, 4) and the counter for the 16 lower-order bits TPUn.TCNT (n = 2, 5) being used together to form a 32-bit counter, if a rising edge is input to the TIOCAm and TIOCAn pins simultaneously, the value of the 16 higher-order bits is transferred to the TPUm.TGRA register and that of the 16 lower-order bits is transferred to the TPUn.TGRA register.

However, even when a rising edge is input to the TIOCA1 and TIOCA2 pins simultaneously, the timing for capturing the 16 higher- and 16 lower-order bits may differ by one clock cycle due to the difference between the delays until they are latched within the LSI chip.

If such a difference coincides with an overflow of the lower-order 16-bit counter so that the value of the 32- bit counter is changing from 0000 FFFFh or 0001 0000h, while the actual value of the counter at the time of capture is 0000 FFFFh or 0001 0000h, the actual captured value may be 0000 0000h or 0001 FFFFh.

Accordingly, with input capture while the counters are in use as a 32-bit counter, if the value of the lower-order 16-bit counter when the values may deviate in the way described is 0000h or FFFFh, take care to handle processing for invalidation, for example, discarding the captured value.

18.9 Link Operation

18.9.1 Transmitting an Event Signal to ELC

The TPU uses the ELC (event link controller) to perform link operation to the previously specified module using the interrupt request signal as the event signal.
The event signal can be output regardless of the setting of the interrupt request enable bits (TGIEA, TGIEB, TGIEC, or TGIED, and TCIEV, or TCIEU) of the corresponding TIER0 register.
Table 18.30 lists the event signals that can be transmitted through the respective channels.

Table 18.30 Ability of Interrupt Sources to Serve as Event Signals for Transmission to the ELC

Channel No.	Compare match A	Compare match B	Compare match C	Compare match D	Overflow	Underflow
Channel 0	√	√	√	√	√	—*1
Channel 1	√	√	—*2	—*2	√	√
Channel 2	√	√	—*2	—*2	√	√
Channel 3	√	√	√	√	√	—*1

√: Possible
—: Not possible
Note 1. Channels 0 and 3 do not have these interrupt request signals since they will never underflow.
Note 2. Channels 1 and 2 do not have these interrupt request signals since they do not have TGRC and TGRD registers.

18.9.2 Receiving an Event Signal from ELC

The TPU can perform any of the following three operations using the event link setting register of the ELC (event link controller).

(1) Start Counting

When an event signal is received while the TPU count start operation is selected by the ELOPF register of the ELC, the CSTn bit in TSTRA (the timer start register) is set to 1 and counting starts.
However, if this event is generated for the channels when the CSTn bit is set to 1, the event is ignored.
Table 18.31 lists the CSTn bits in TSTR used for each channel.
Figure 18.56 shows the timing of the count start operation.

For details on the setting procedure to start counting, see section 18.3.1, (1) Counter Operation.

Table 18.31 Correspondence between Channels and CSTn Bits in TSTRA

Channel No.	CSTn Bits in TSTRA
Channel 0	CST0
Channel 1	CST1
Channel 2	CST2
Channel 3	CST3

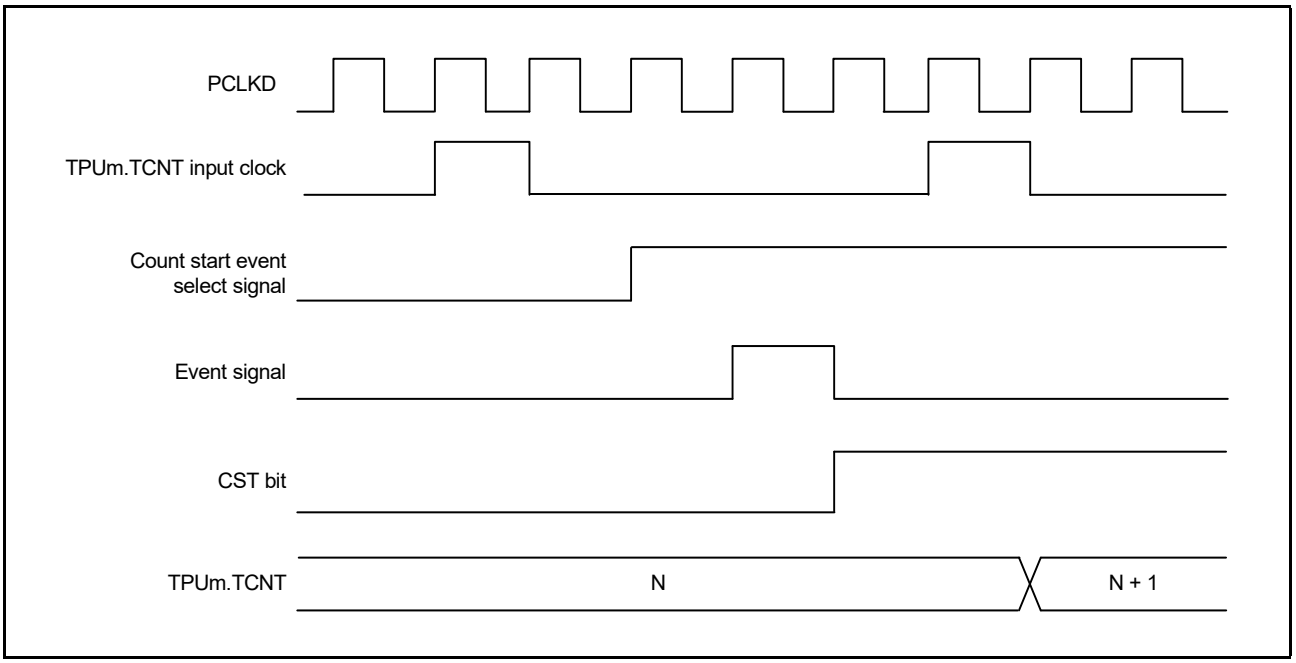


Figure 18.56 Start Counting on Reception of the Event Signal

(2) Clear Counting

Counter clearing operation of a TPU is selected by the ELOPF register of the ELC. When the event specified in the corresponding ELSRn register is generated, the timer counter (TCNT) is returned to its initial value (0000h). Counting continues, however, if the setting of the CSTn bit of the timer start register (TSTRA) is 1 at this time, so counting can be automatically restarted in this way.

Table 18.31 lists the CSTn bits in TSTR used for each channel.

Figure 18.57 shows the timing of the count restart operation.

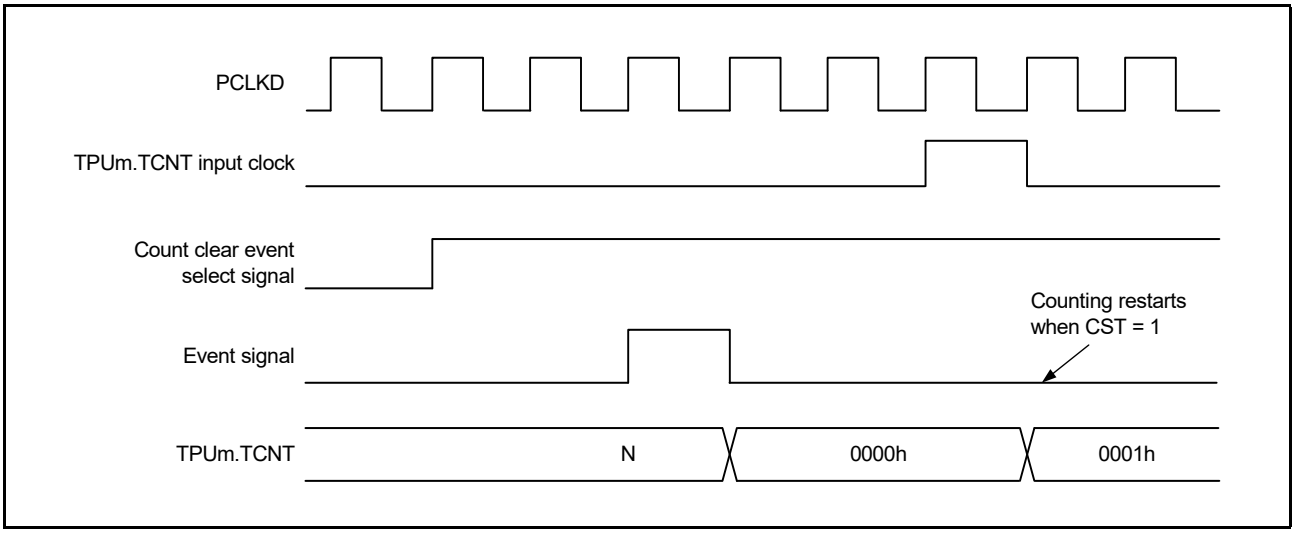


Figure 18.57 Restart Counting on Reception of the Event Signal

(3) Input Capture Operation

When an event signal is received while the TPU input capture operation is selected by the ELOPF register of the ELC, the value of TCNT (the timer counter) of the corresponding channel is captured in TGRA (timer general register A).
When using input capture due to the event link, set the IOA bit in TIOR (the timer I/O control register) to specify input capture, and then set the CSTn bit in TSTRA (the timer start register) to 1 to start counting.
Table 18.32 lists TGRA and TIOR register bits used for each channel. For the CSTn bits in TSTRA used for each channel, see Table 18.31.
Figure 18.58 shows the timing of input capture operation.
When input capture operation due to the event link is selected, the setting of TIOR and the corresponding input capture (the linkage of the TIOCAm pin (input capture pin) input with the specific operation of other channels) are not effective (m = 0 to 5).

For details on the setting procedure for input capture, see section 18.3.1, (3) Input Capture Function.

Table 18.32 TGR and TIOR Used for Input Capture by ELC

Channel No.	Capture Destination Registers	Bits in TIOR
Channel 0	TGRA (channel 0)	IOA[3:0] bits (TIORH0)
Channel 1	TGRA (channel 1)	IOA[3:0] bits (TIOR1)
Channel 2	TGRA (channel 2)	IOA[3:0] bits (TIOR2)
Channel 3	TGRA (channel 3)	IOA[3:0] bits (TIORH3)

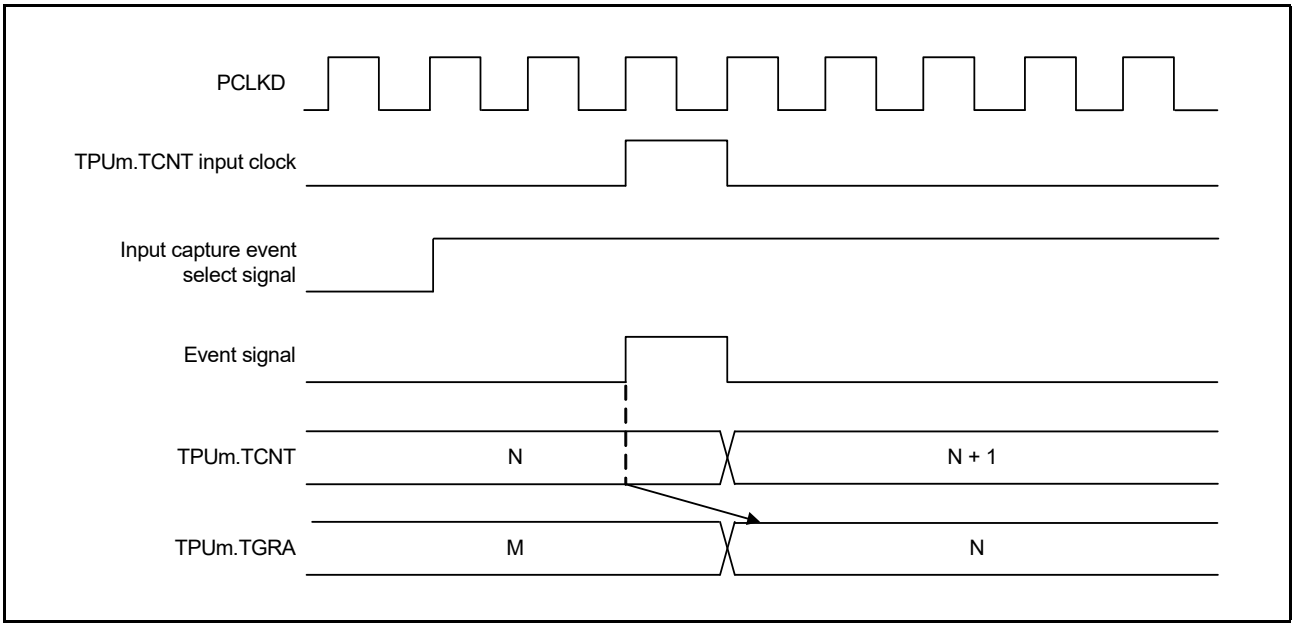


Figure 18.58 Input Capture on Reception of the Event Signal

18.9.3 Usage Notes on Operation on Reception of the Event Link Signal

The followings are the notes on using the TPU for event link operations.

(1) Start Counting

When writing to the CSTn bit in TSTRA (the timer start register) and a counting start are in contention, writing to the CSTn bit does not proceed since setting of the CSTn bit to 1 in response to the event takes priority.
Figure 18.59 shows the timing in this case.
Furthermore, even when a counting start due to the event link is selected, CPU writing to the CSTn bit proceeds if the event signal is at a low level.

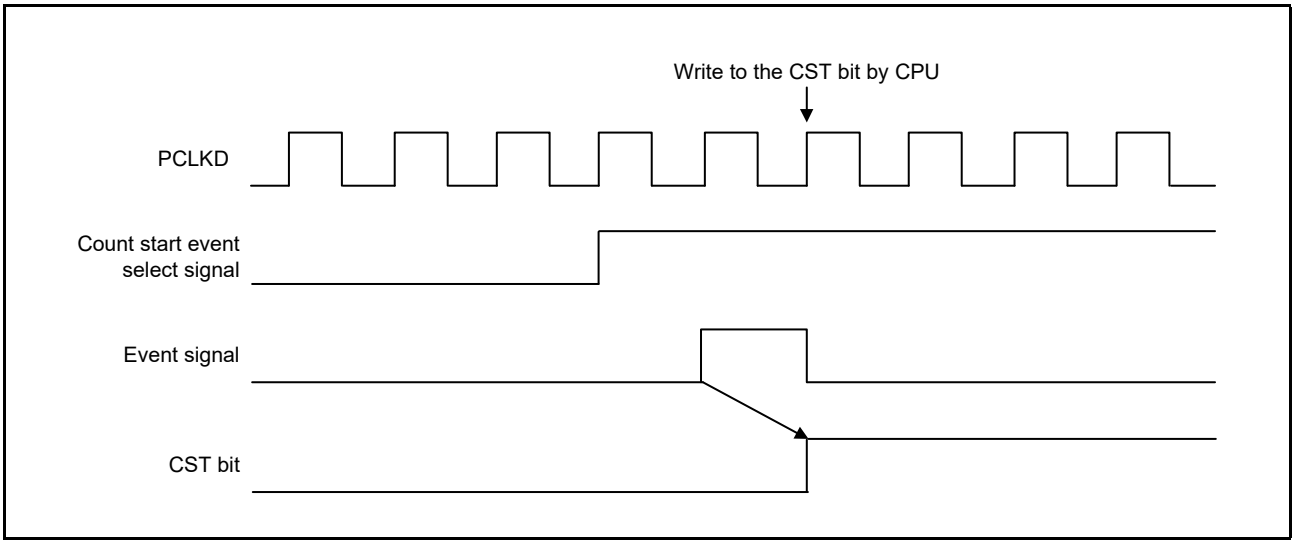


Figure 18.59 Conflict between Writing to the CSTn Bit and Counting Start

(2) Clear Counting

When a TCNT (a timer counter) write cycle and a counting clear are in contention, writing to TCNT does not proceed since the counter value initialization in response to the counting clear takes priority.
Figure 18.60 shows the timing in this case.
Furthermore, even when a counting clear due to the event link is selected, CPU writing to TCNT proceeds if the event signal is at a low level.

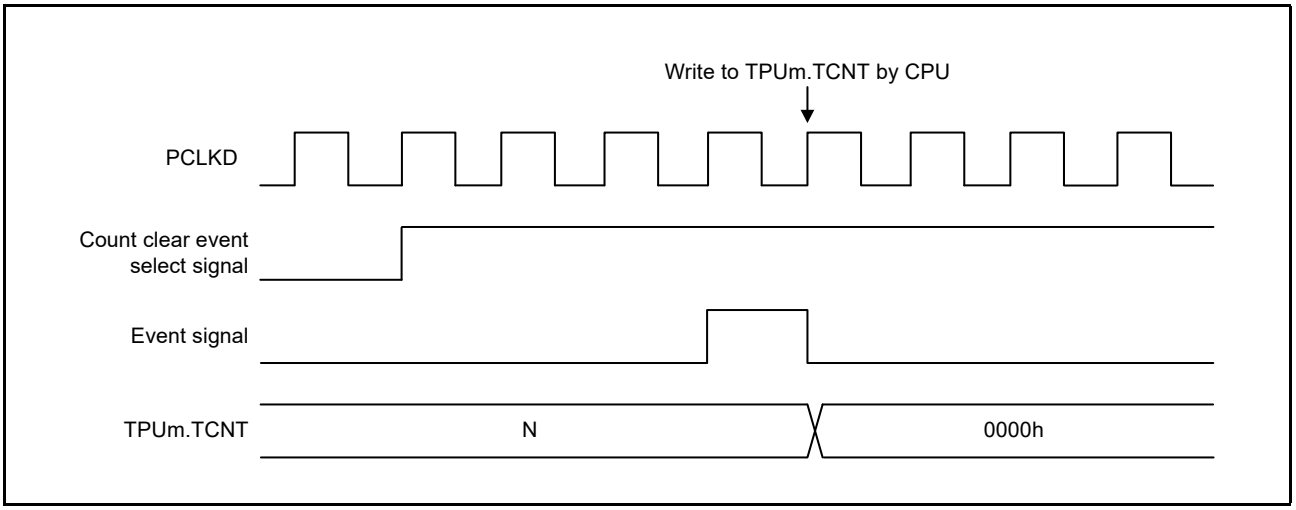


Figure 18.60 Conflict between TCNT Write Cycle and Counting Clear

(3) Input Capture Operation

If a TGRy (the timer general register) read/write cycle and input capture operation are in contention, operation proceeds as follows (y = A to D).

(a) Conflict between TGRy Read Cycle and Input Capture

The internal data bus reads the data before input capture transfer.
Figure 18.61 shows the timing in this case.

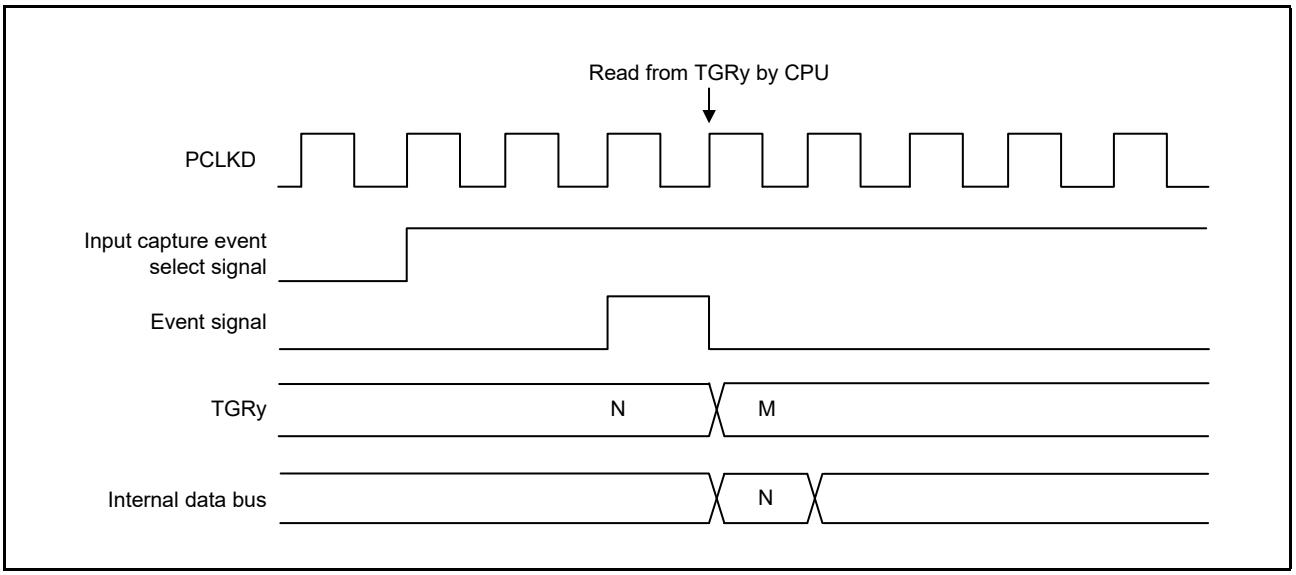


Figure 18.61 Conflict between TGRy Read Cycle and Input Capture Operation

(b) Conflict between TGRy Write Cycle and Input Capture

Writing to TGRy does not proceed since input capture takes priority.
Figure 18.62 shows the timing in this case.
Furthermore, even when input capture operation due to the event link is selected, CPU writing to TGRy proceeds if the event signal is at a low level.

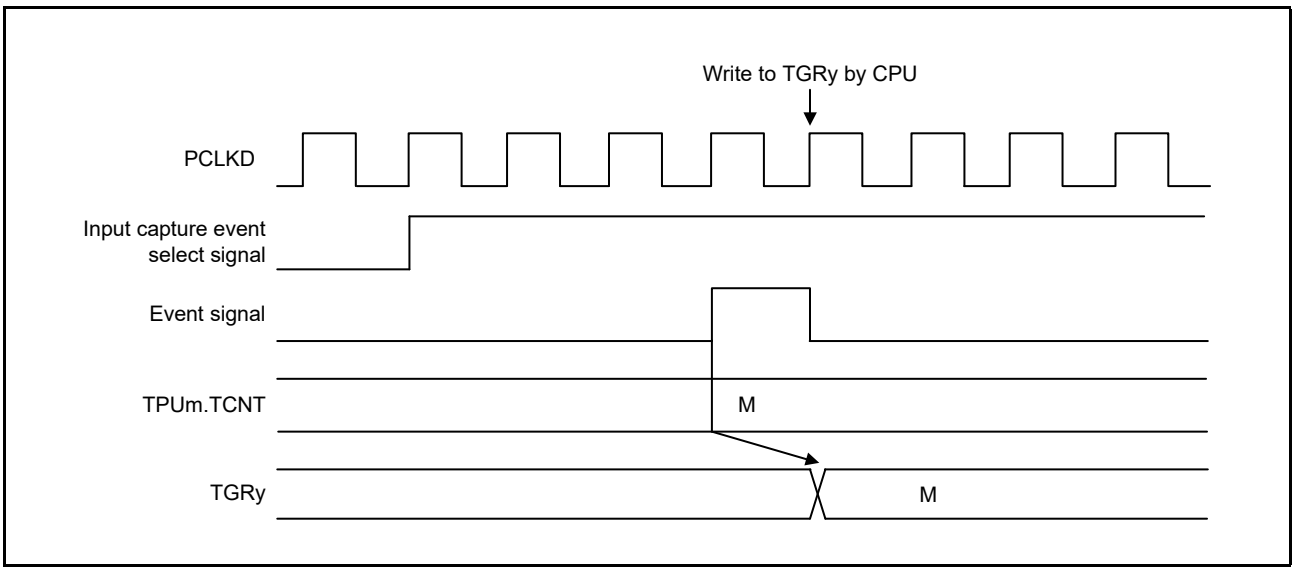


Figure 18.62 Conflict between TGRy Write Cycle and Input Capture Operation

18.9.4 Notes on Transmitting the Event Link Signal

The followings are the notes on transmitting the event link signal.

(1) Transmitting the Compare Match Event Signal

When the TGRy register is set to 0000h, PCLKD/1 is set as the counter clock (TCRn.TPSC[2:0] = 000b), and compare match is set as the trigger for clearing of the counter clock , the value of the TCNT remains 0000h, and the event output signal will be output continuously (i.e. its level will be flat) rather than output over a single cycle (y = A to D).
Figure 18.63 shows the timing for continuous output of the event output signal in response to a compare match.

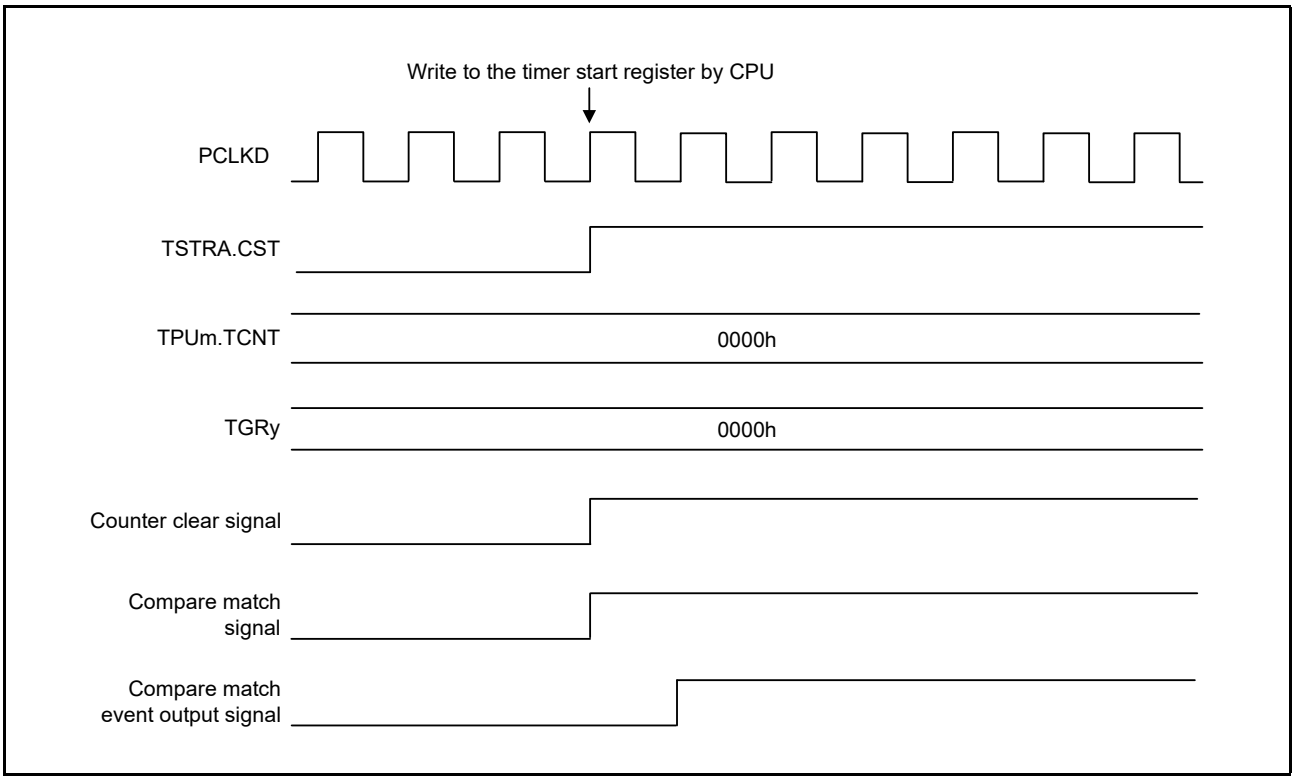


Figure 18.63 Continuous Output of the Compare Match Event Output Signal

(2) Transmitting the Underflow Event Signal

If two external clock signals' same direction edges to be phase counted are generated within two PCLKD cycles in phase counting mode 1, with TGRy being 0000h, and compare match set as the counter clear source, the TCNT counter remains 0000h, and a compare-match event signal and an underflow event signal are output continuously to form a flat signal level (y = A to D).

Figure 18.64 shows the timing for continuous output of the event output signal in response to underflow.

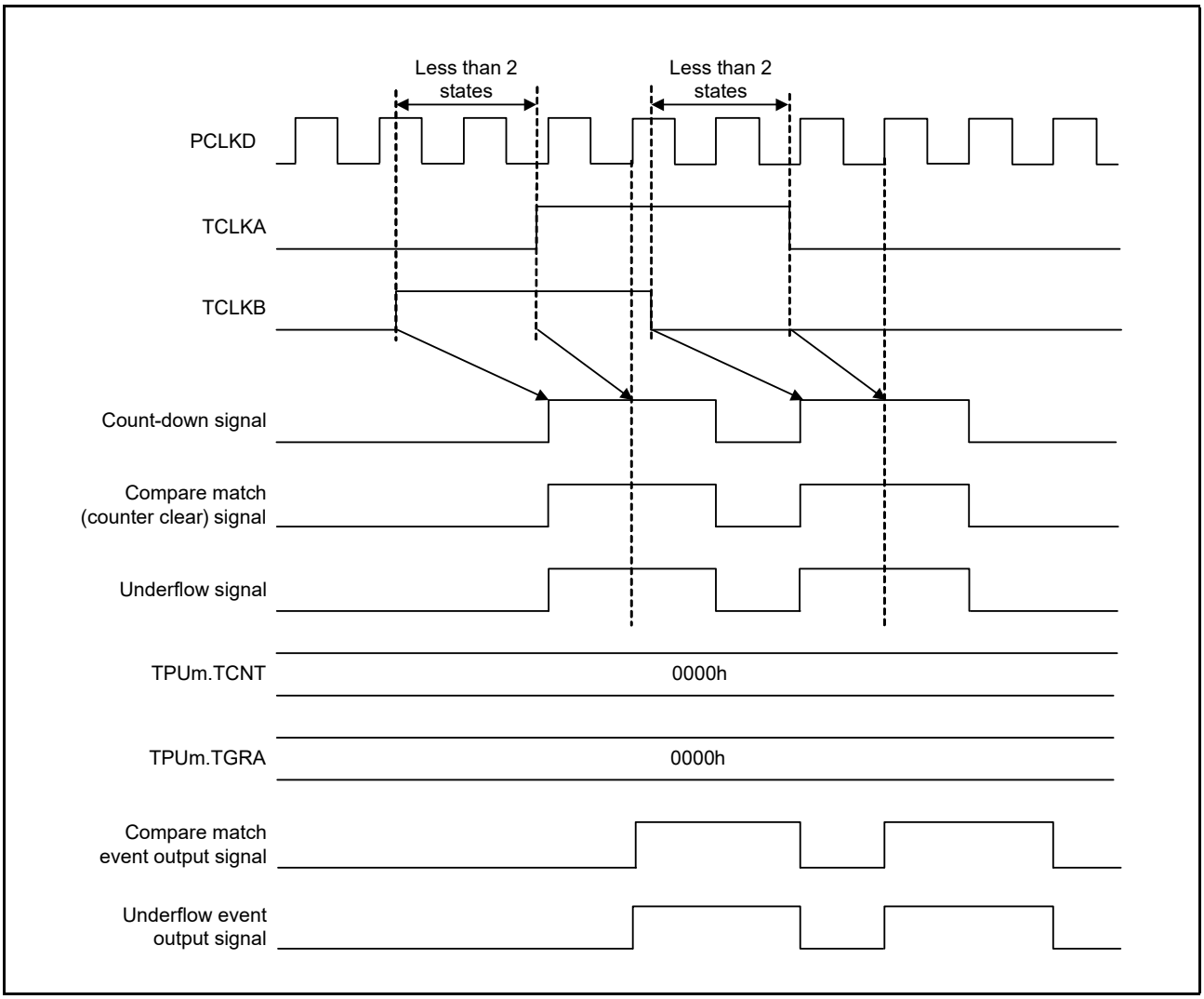


Figure 18.64 Continuous Output of the Underflow Event Output Signal

19. Compare Match Timer (CMT)

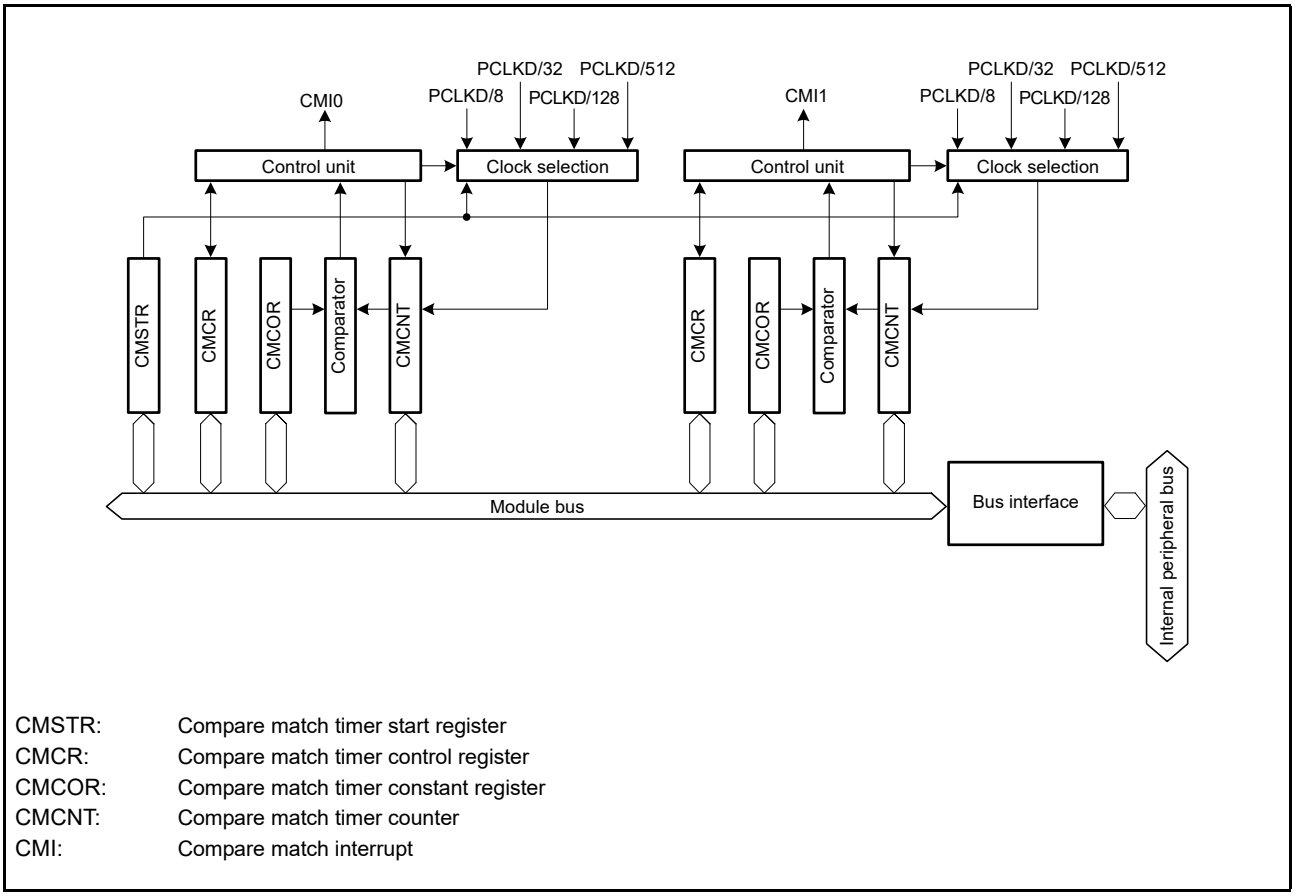
The compare match timer (CMT) consists of a two-channel 16-bit timer, and can generate interrupts at set intervals by using its 16-bit counter.

19.1 Overview

Table 19.1 lists the specifications for the CMT.
Figure 19.1 shows a block diagram of the CMT.

Table 19.1 CMT Specifications

Item	Description
Number of internal channels	Two channels × two units
Timer counter (per channel)	16-bit up counter (Counted according to the count enable signal output by the prescaler.) Returned to 0000h after compare match.
Prescaler (per channel)	9-bit counter (Linked with enabling/disabling of timer counter operation.) • Outputs four types of count enable signals. The type can be selected from PCLKD/8, PCLKD/32, PCLKD/128, and PCLKD/512.
Event link function (only channel 1 of unit 0)	One of the following three operations is possible depending on the received event: • Count start • Event count • Count clear This function can issue a compare match event.
Reset	Asynchronous reset



CMSTR: Compare match timer start register
CMCR: Compare match timer control register
CMCOR: Compare match timer constant register
CMCNT: Compare match timer counter
CMI: Compare match interrupt

Figure 19.1 Block Diagram of CMT (Unit 0, Unit 1)

19.2 Register Descriptions

19.2.1 Compare Match Timer Start Register 0 (CMSTR0)

The CMSTR0 register sets starting or stopping of the CMT0.CMCNT and CMT1.CMCNT counters of unit 0.

Address(es): A008 0000h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STR1	STR0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	STR0	Count Start 0	This bit selects starting or stopping of the CMT0.CMCNT counter operation. The corresponding prescaler starts or stops according to the setting of this bit. 0: The CMT0.CMCNT counter is stopped. 1: The CMT0.CMCNT counter is started.	R/W
b1	STR1	Count Start 1	This bit selects starting or stopping of the CMT1.CMCNT counter operation. The corresponding prescaler starts or stops according to the setting of this bit. 0: The CMT1.CMCNT counter is stopped. 1: The CMT1.CMCNT counter is started.	R/W
b15 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

19.2.2 Compare Match Timer Start Register 1 (CMSTR1)

The CMSTR1 register sets starting or stopping of the CMT2.CMCNT and CMT3.CMCNT counters of unit 1.

Address(es): A008 0020h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STR3	STR2
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	STR2	Counter Start 2	This bit selects starting or stopping of the CMT2.CMCNT counter operation. The corresponding prescaler starts or stops according to the setting of this bit. 0: The CMT2.CMCNT counter is stopped. 1: The CMT2.CMCNT counter is started.	R/W
b1	STR3	Counter Start 3	This bit selects starting or stopping of the CMT3.CMCNT counter operation. The corresponding prescaler starts or stops according to the setting of this bit. 0: The CMT3.CMCNT counter is stopped. 1: The CMT3.CMCNT counter is started.	R/W
b15 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

19.2.3 Compare Match Timer Control Register (CMCR)

The CMCRn register specifies a clock used for count-up operation.

Address(es): CMT0: A008 0002h, CMT1: A008 0008h, CMT2: A008 0022h, CMT3: A008 0028h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	CMIE	—	—	—	—	CKS[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	CKS[1:0]	Clock Select	These bits select a clock to be input to the CMTn.CMCNT counter from the internal clocks obtained by dividing the frequency of the low-speed peripheral module clock (PCLKD). Setting the CMSTRm.STRn bit to 1 starts count-up operation of the corresponding CMCNT counter by using the clock selected in the CKS[1:0] bits (m = 0, 1, n = 0 to 3). b1 b0 0 0: PCLKD/8 0 1: PCLKD/32 1 0: PCLKD/128 1 1: PCLKD/512	R/W
b5 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b6	CMIE	Compare Match Interrupt Enable	This bit selects whether to enable or disable generation of a compare match interrupt (CMIn) when the values in the CMCNT counter and in the CMCOR register match (n = 0 to 3). 0: Compare match interrupt (CMIn) disabled 1: Compare match interrupt (CMIn) enabled	R/W
b7	—	Reserved	When read, the value returned is undefined. The write value should be 0.	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

19.2.4 Compare Match Timer Counter (CMCNT)

The CMCNT counter (the main unit of the compare match timer) is a readable/writable up-counter.
When an internal clock is selected by the CMCR.CKS[1:0] bits and the CMSTRm.STRn (m = 0, 1, n = 0 to 3) bit is set to 1, the CMCNT counter starts counting up using the selected clock.
When the value in the CMCNT counter and the value in the CMCOR register match, the CMCNT counter is cleared to 0000h. At the same time, a compare match interrupt (CMIn) is generated (n = 0 to 3).

Address(es): CMT0: A008 0004h, CMT1: A008 000Ah, CMT2: A008 0024h, CMT3: A008 002Ah

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

19.2.5 Compare Match Timer Constant Register (CMCOR)

The CMCOR register is a readable/writable register to set a cycle for compare match with the CMCNT counter.
The cycle for compare matches is as follows.

Compare-match cycle = (setting of the CMCOR register + 1) × counter-clock cycle*1

Note 1. This is a clock cycle set by the CMCR.CKS[1:0] bits.

Address(es): CMT0: A008 0006h, CMT1: A008 000Ch, CMT2: A008 0026h, CMT3: A008 002Ch

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

19.3 Operation

19.3.1 Periodic Count Operation

When an internal clock is selected by the CMCRn.CKS[1:0] bits and the CMSTRm.STRn (m = 0, 1, n = 0 to 3) bit is set to 1, the CMCNT counter starts counting up using the selected clock.
When the value in the CMCNT counter and the value in the CMCOR register match, the CMCNT counter is cleared to 0000h, and then a compare match interrupt (CMIn) is generated. The CMCNT counter then starts counting up again from 0000h. Figure 19.2 shows the operation of the CMCNT counter.

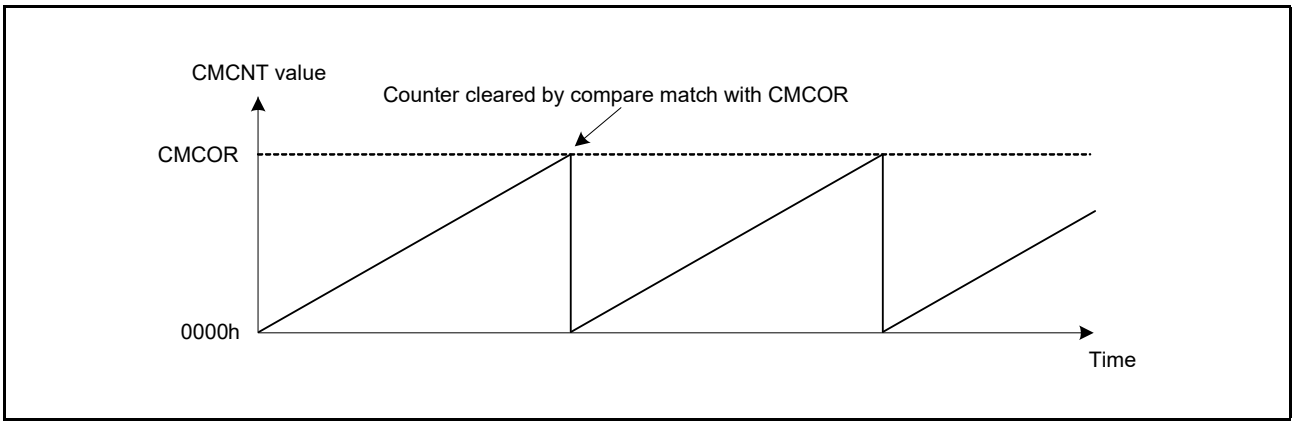


Figure 19.2 Counter Operation

19.3.2 CMCNT Count Timing

As the count clock, one of four internal clocks (PCLKD/8, PCLKD/32, PCLKD/128, and PCLKD/512) obtained by dividing the low-speed peripheral module clock (PCLKD) can be selected with the CMCR.CKS[1:0] bits. Figure 19.3 shows the timing of the CMCNT counter.

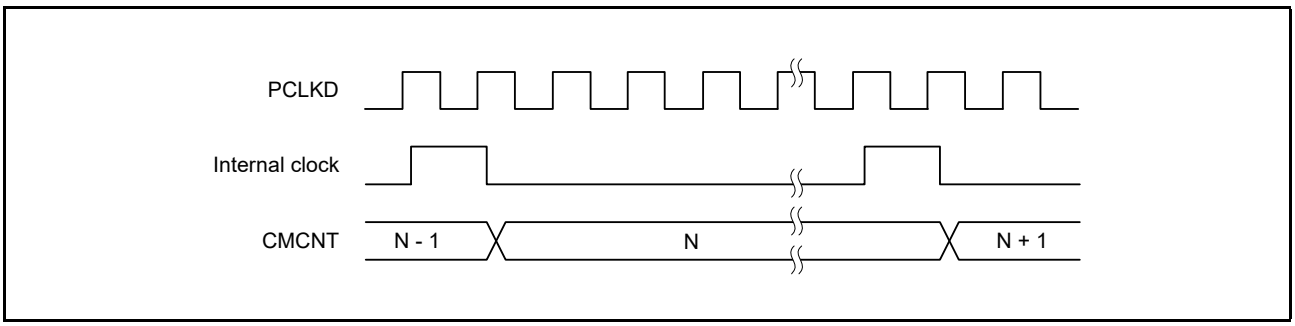


Figure 19.3 CMCNT Count Timing

19.4 Interrupts

19.4.1 Interrupt Sources

The CMT has channels and each of them to which a different vector address is allocated has a compare match interrupt (CMIn). The priority of channels can be changed by the interrupt controller settings.

Table 19.2 CMT Interrupt Sources

Name	Interrupt Sources
CMI0	Compare match between the CMT0.CMCNT counter and CMT0.CMCOR register
CMI1	Compare match between the CMT1.CMCNT counter and CMT1.CMCOR register
CMI2	Compare match between the CMT2.CMCNT counter and CMT2.CMCOR register
CMI3	Compare match between the CMT3.CMCNT counter and CMT3.CMCOR register

19.4.2 Timing of Compare Match Interrupt Generation

When the CMCNT counter and the CMCOR register match, a compare match interrupt (CMIn) is generated.
A compare match signal is generated at the last state in which the values match (the timing when the CMCNT counter updates the matched count value). That is, after a match between the CMCOR register and the CMCNT counter, the compare match signal is not generated until the CMCNT counter input clock is generated (n = 0 to 3).
Figure 19.4 shows the timing of a compare match interrupt.

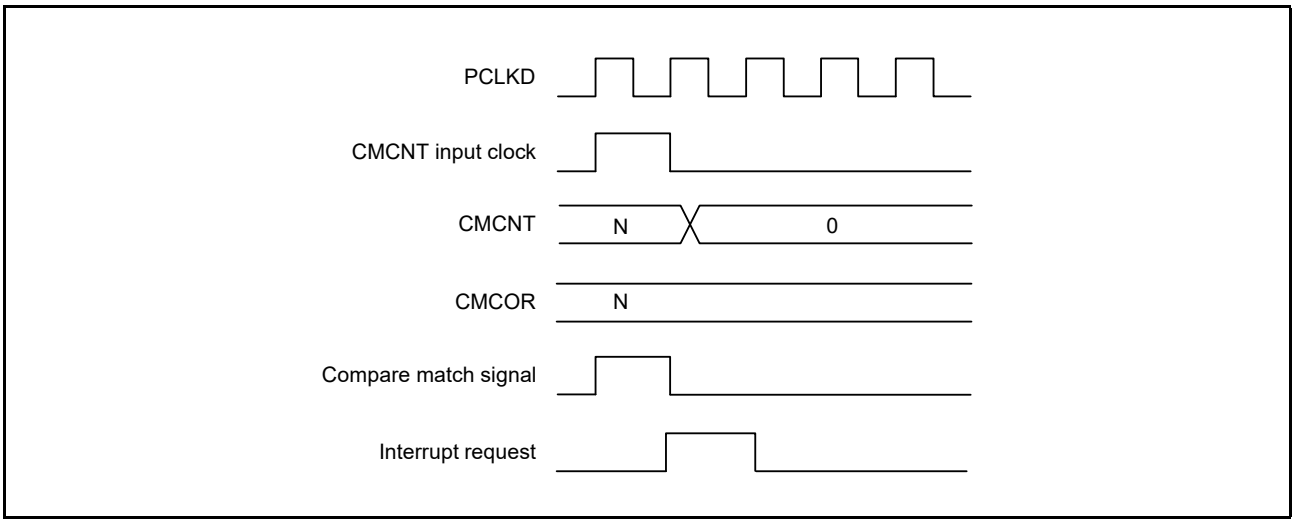


Figure 19.4 Timing a Compare Match Interrupt is Set

19.5 Event Link Operations

19.5.1 Event Issuance to ELC

The CMT issues an interrupt request when compare match occurs. Then, the CMT can use the event link controller (ELC) to perform link operation to a preset module using the interrupt request signal as the event signal.
The event signal can be output regardless of the setting of the corresponding interrupt request enable bit (CMT1.CMCR.CMIE).

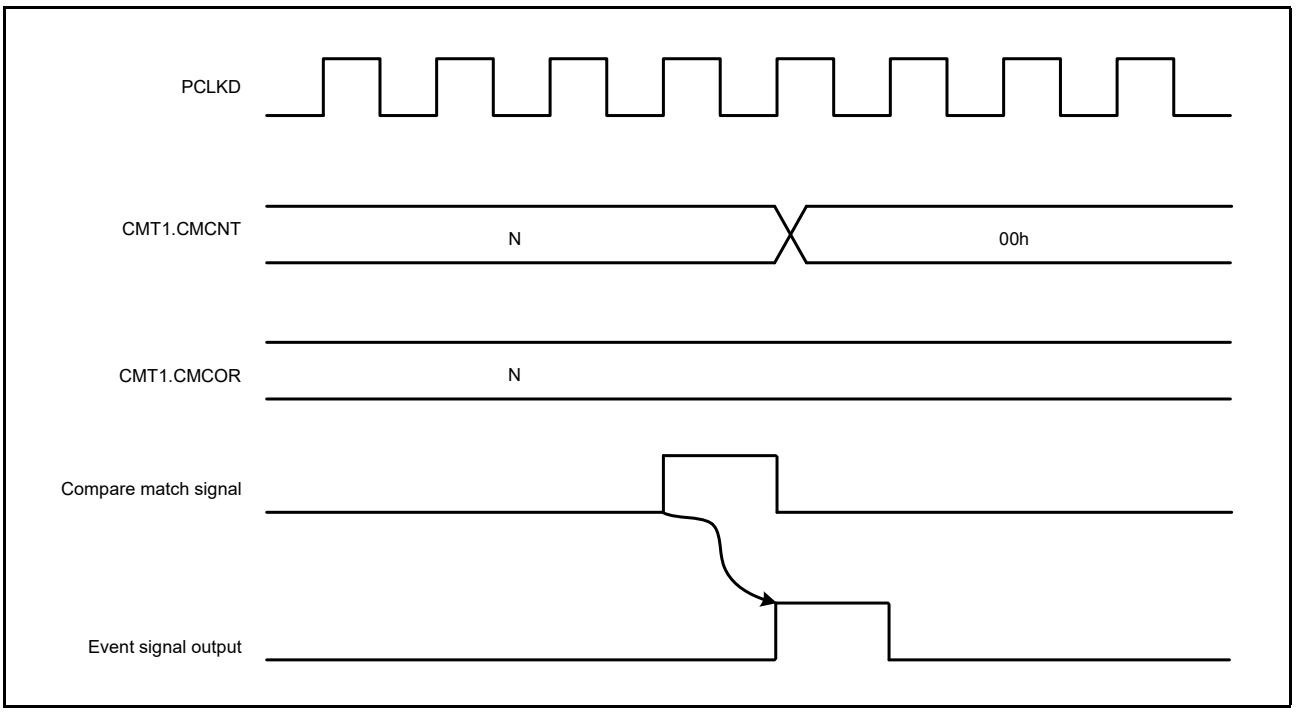


Figure 19.5 Timing of Event Issuance

19.5.2 CMT Operation When Receiving an Event from ELC

The CMT can perform either of the following three operations upon the event preset in the event link controller (ELC).

(1) Count Start

When the CMT count start operation is selected in the ELC and an event is received, the STR1 bit in the corresponding CMSTR0 (compare match timer start register 0) is set to 1, starting the count operation.
However, if the specified event occurs while the CMSTR0.STR1 bit is 1, the event is invalid.

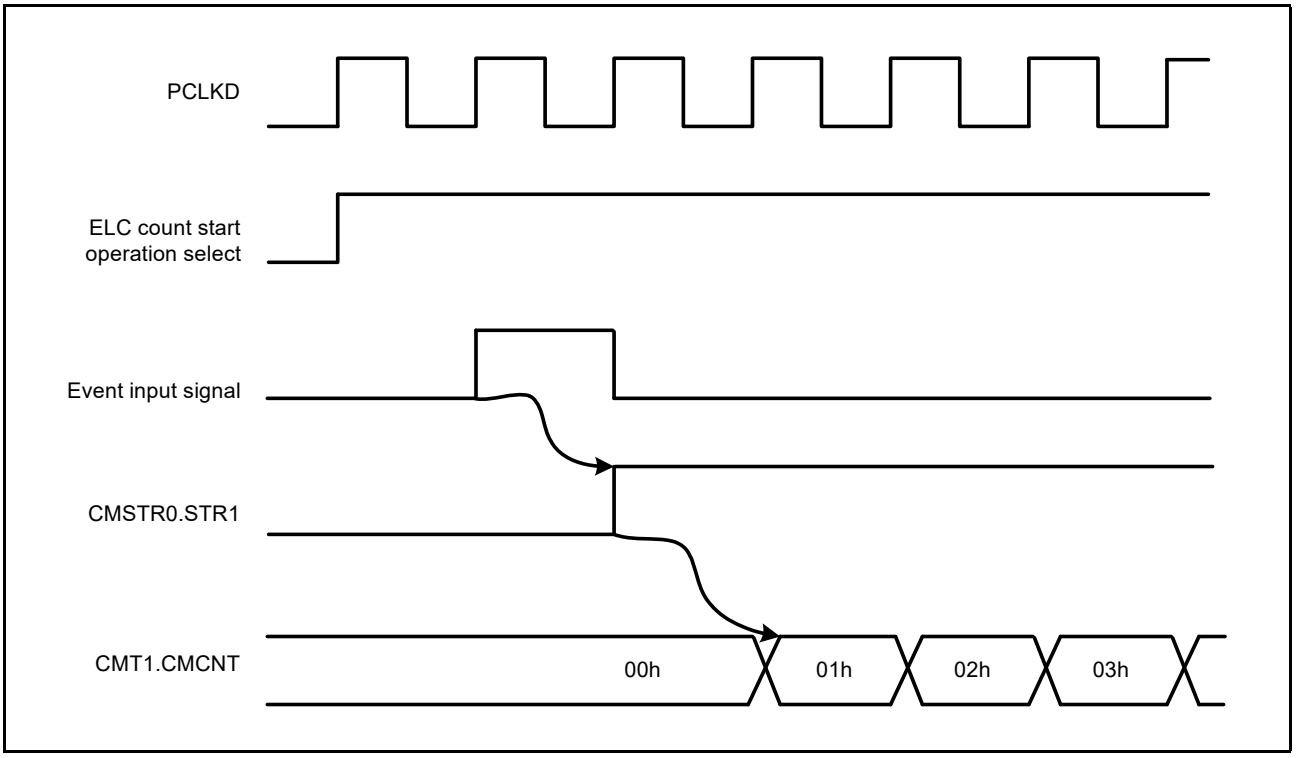


Figure 19.6 Count Start Operation at Reception of an Event

(2) Event Count

When the CMT event count operation is selected in the ELC and an event is received, the CMT1.CMCNT (compare match timer counter) is incremented, regardless of the setting of the CKS[1:0] bits in CMT1.CMCR (compare match timer control register). The STR1 bit in the CMSTR0 (compare match timer start register) must be set to 1 before receiving an event.

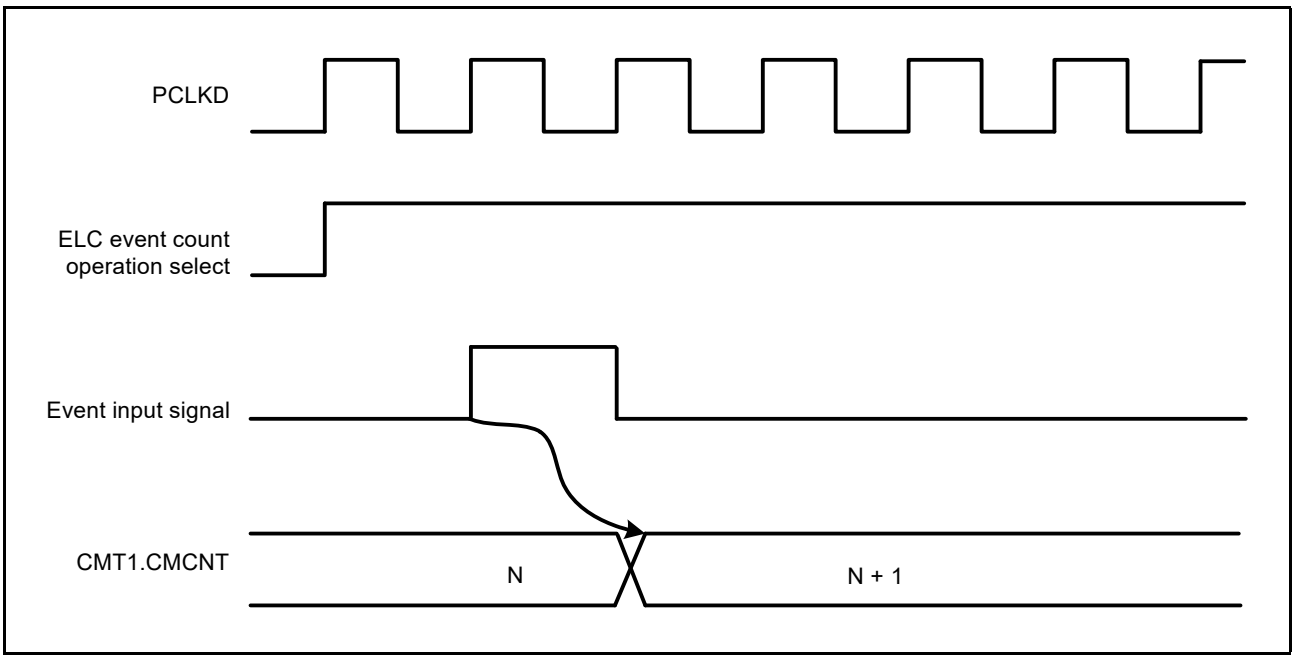


Figure 19.7 Event Count Operation at Reception of an Event

(3) Count Clear

When the CMT count clear operation is selected in the ELC and an event is received, the compare match timer counter (CMT1.CMCNT) is returned to its initial value. Counting continues, however, if the setting of the STR1 bit of compare match timer start register 0 (CMSTR0) is 1 at this time, so counting can be automatically restarted in this way.

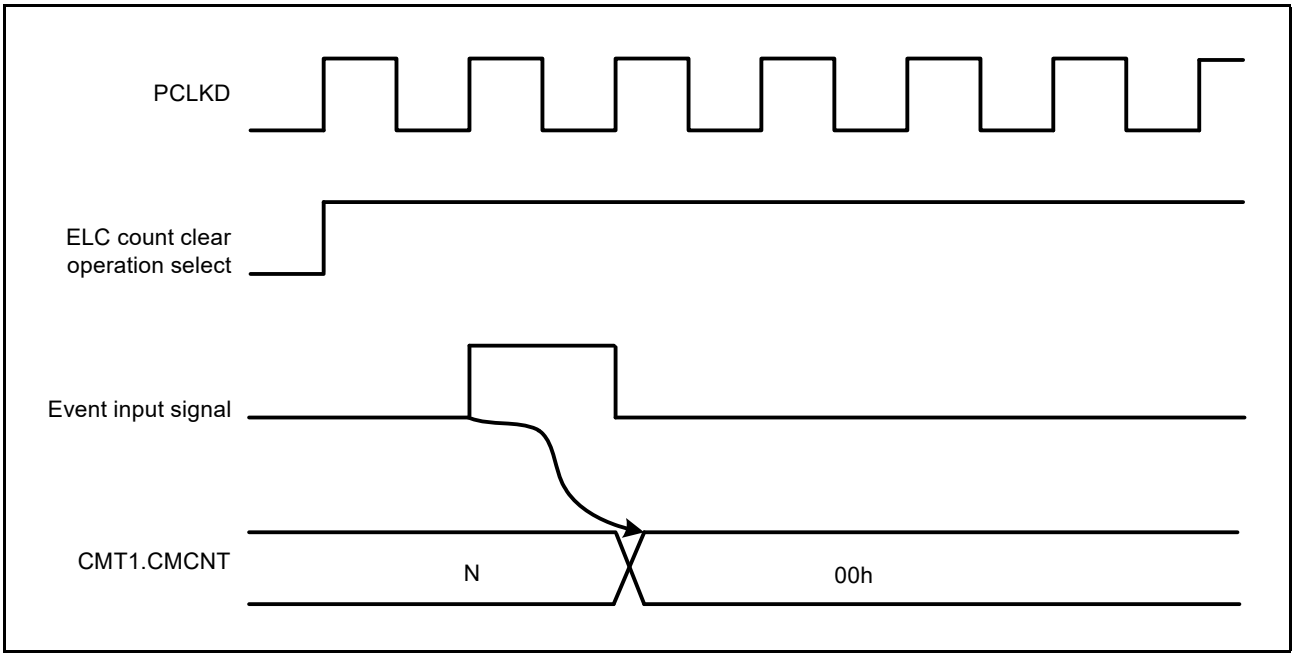


Figure 19.8 Count Clear Operation at Reception of an Event

19.5.3 Notes on CMT Event Link Operation

Note the following when using the CMT with event link operation.

(1) Count Start

When an event occurs during the write access to the STR1 bit in the CMSTR0 (compare match timer start register 0), that write access is not performed, and setting 1 according to the event occurrence takes priority.

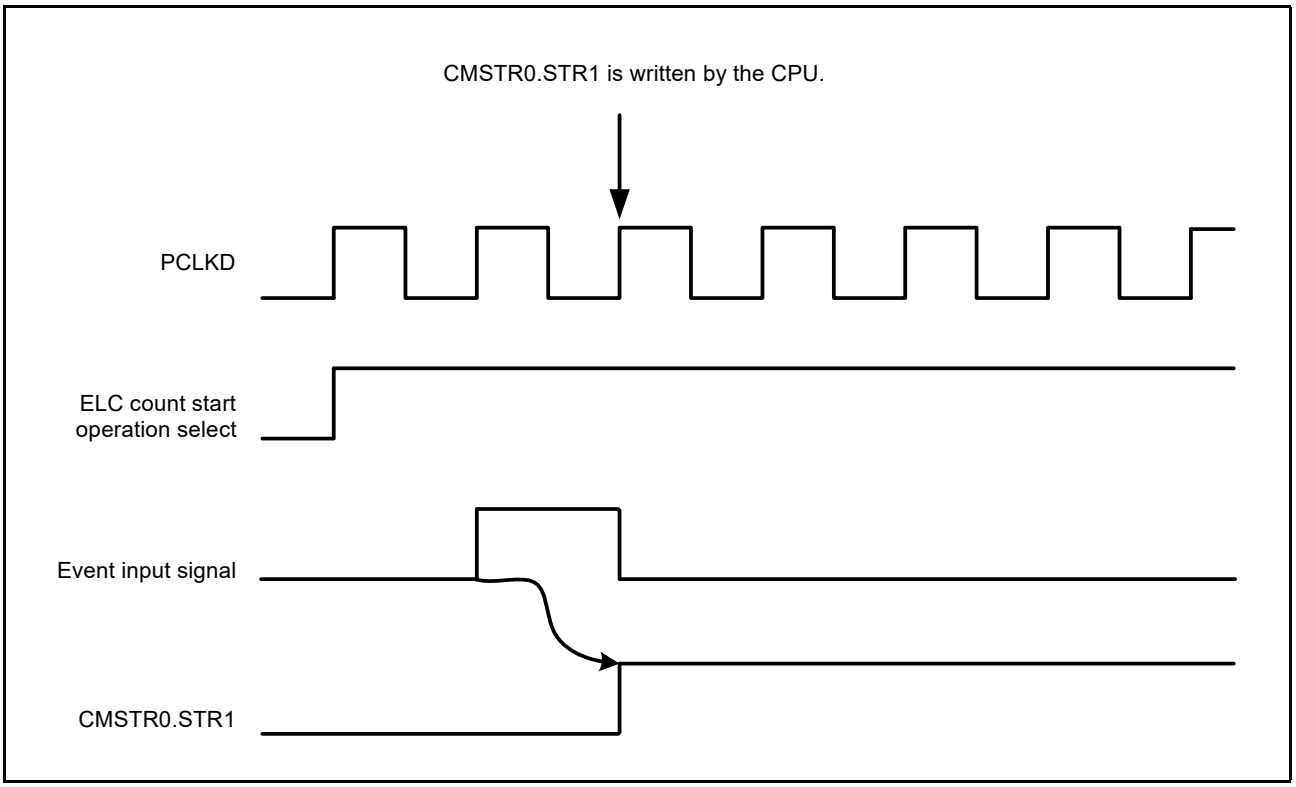


Figure 19.9 Conflict between Event Reception and Register Access at Count Start Operation

(2) Event Count

When an event occurs during the write access to the CMT1.CMCNT (compare match timer counter), that write access is not performed, and event count operation according to the event occurrence takes priority.

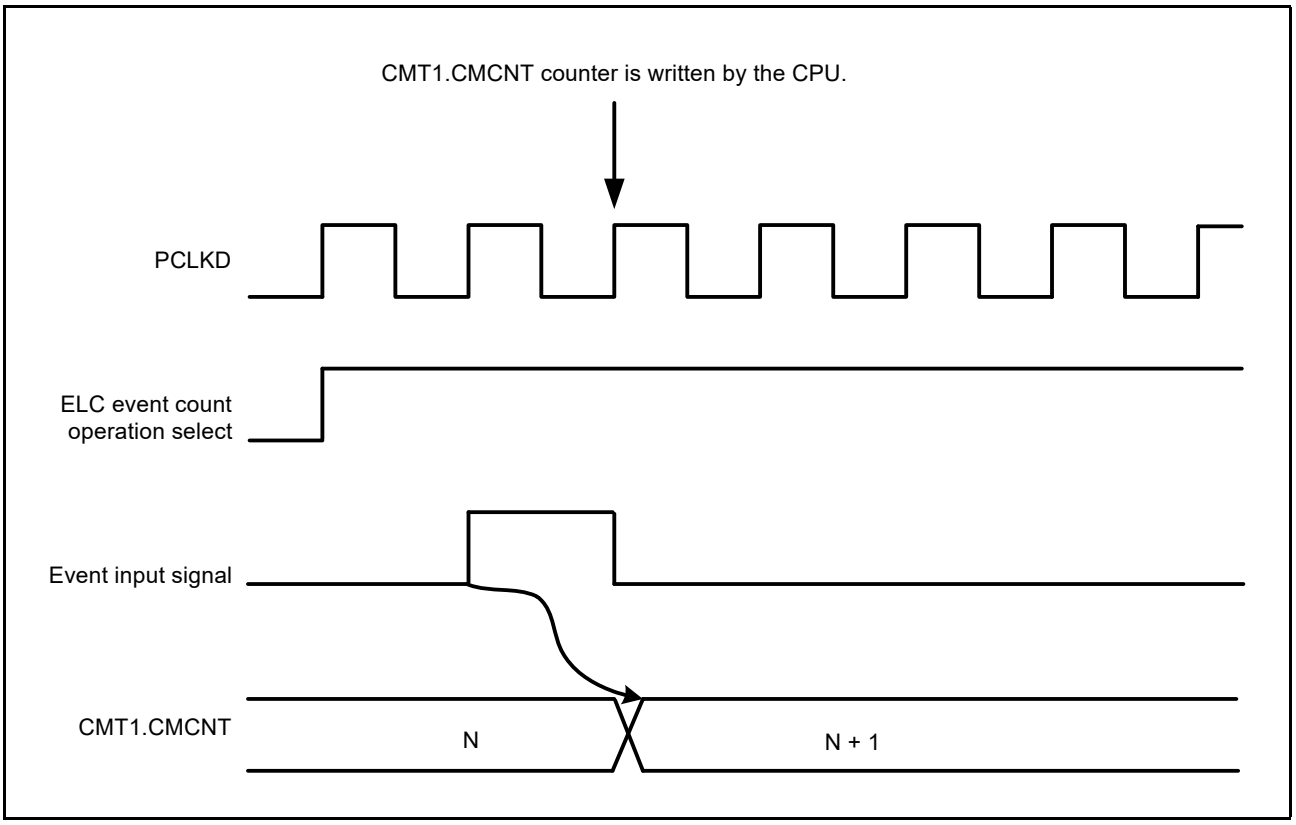


Figure 19.10 Conflict between Event Reception and Register Access at Event Count Operation

(3) Count Clear

When an event occurs during the write access to the CMT1.CMCNT (compare match timer counter), that write access is not performed, and count value initialization according to the event occurrence takes priority.

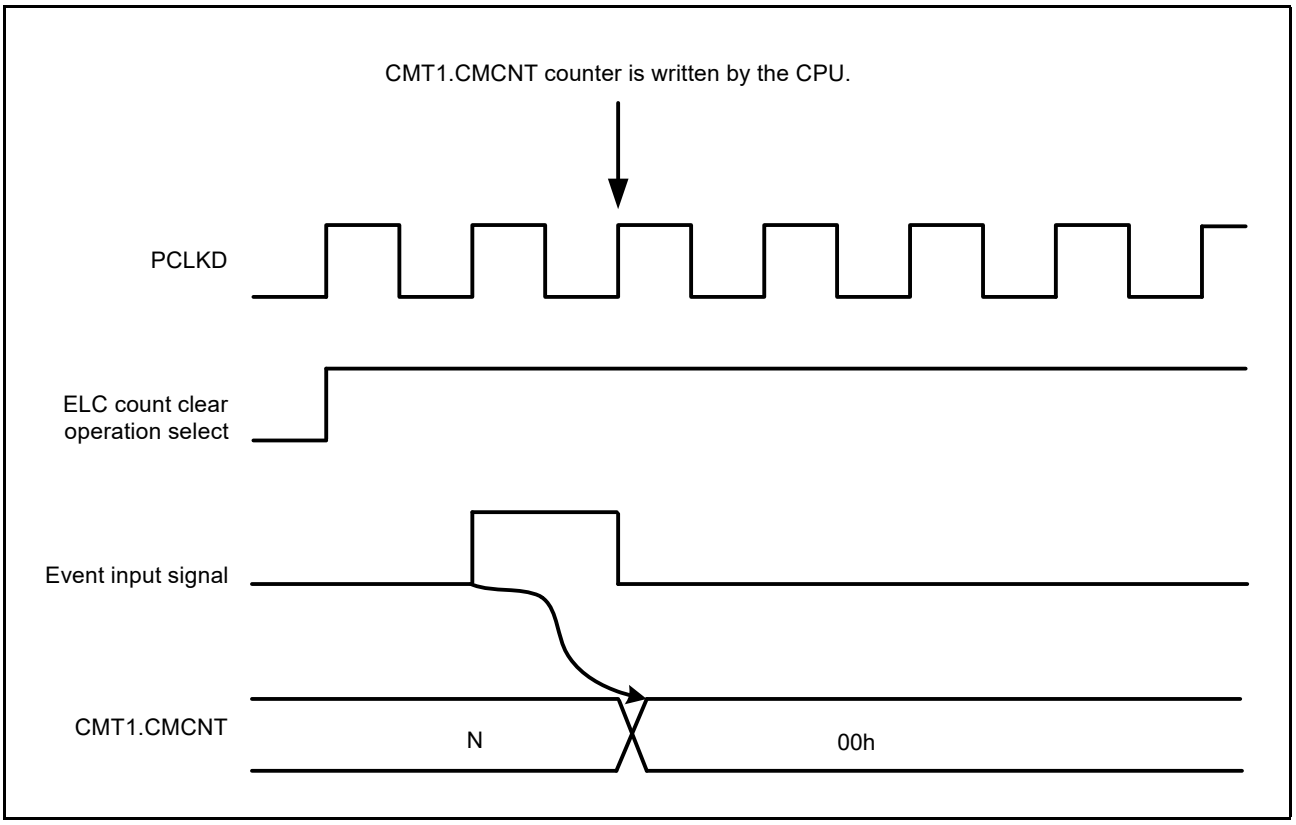


Figure 19.11 Conflict between Event Reception and Register Access at Count Clear Operation

19.6 Usage Notes

19.6.1 Setting the Module Stop Function

The CMT can be enabled or disabled using the module stop control register. After a reset, the CMT is in the module-stop state. The registers can be accessed by canceling the module-stop state. For details, see section 9, Low-Power Consumption Function.

19.6.2 Conflict between Write and Compare-Match Processes of CMCNT

When the compare match signal is generated while writing to the CMCNT counter, clearing the CMCNT counter is given priority over writing to it. In this case, the CMCNT counter is not written to. Figure 19.12 shows the timing to clear the CMCNT counter.

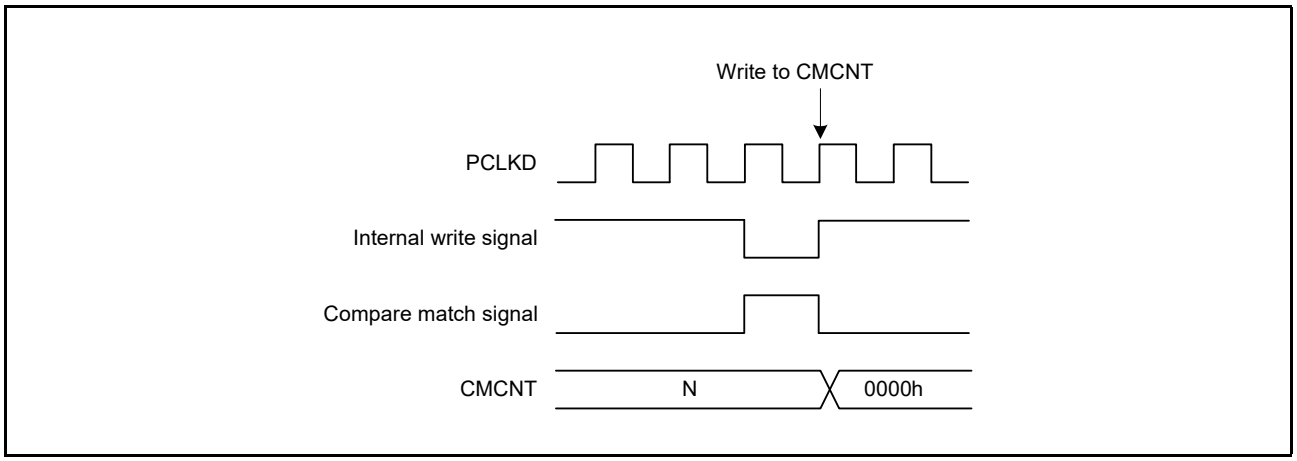


Figure 19.12 Conflict between Write and Compare Match Processes of CMCNT

19.6.3 Conflict between Write and Count-Up Processes of CMCNT

If count-up occurs during the write access to the CMCNT counter, that writing has priority over the count-up. Figure 19.13 shows the timing to write the CMCNT counter.

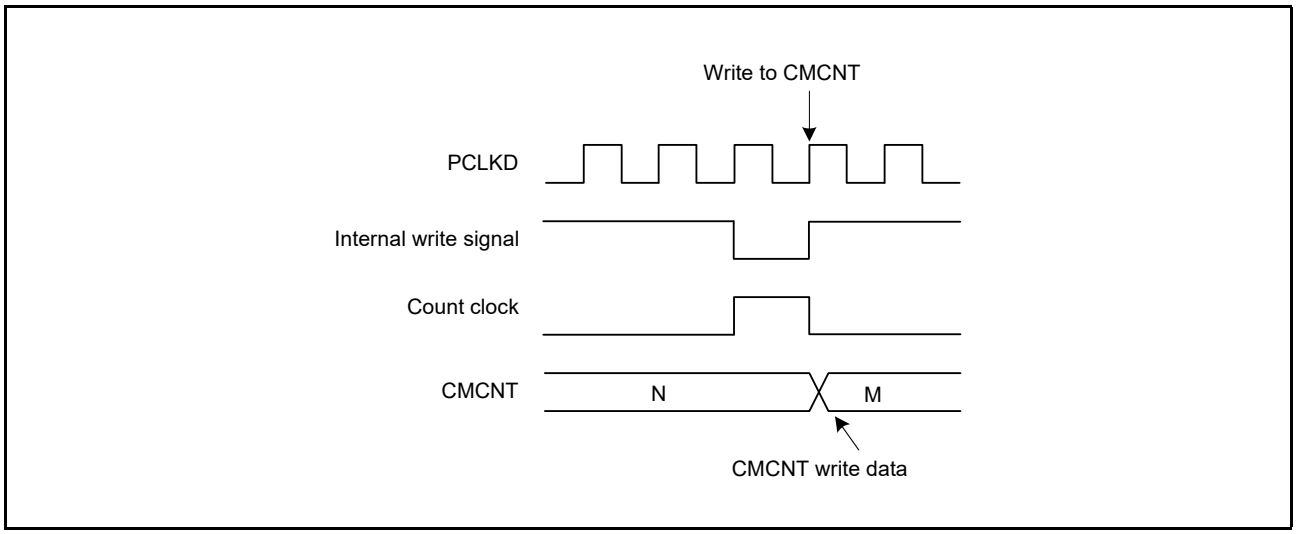


Figure 19.13 Conflict between Write and Count-Up Processes of CMCNT

Table 19.3 Summary of Conflicted Operations among Event Link Operation, Register Access, and Counter Status

Event Link Operation	Register Access	CMCNT Status	Operation to be Performed
Count start	Writing to CMSTR0.STR1	Stopped	Count start
		Compare match	Count start
		Count up	Count start
Event count	Writing to CMCNT	—	Event count
	Writing to CMCOR	Compare match	Compare match
Count clear	Writing to CMCNT	Other than compare match	Count clear
	Writing to CMCNT	Compare match	Compare match
	(No register access)	Compare match	Compare match
(No event)	Writing to CMCNT	Compare match	Compare match
		Count up	Writing to CMCNT
	Writing to CMCOR	Compare match	Compare match

20. Compare Match Timer W (CMTW)

This LSI includes two units with one channel of 32-bit compare match timer W (CMTW).
CMTW has a 32-bit counter and can generate interrupts each time a set period elapses.

20.1 Overview

Table 20.1 shows the specifications of the CMTW. Figure 20.1 shows a block diagram of the CMTW.

Table 20.1 Specifications of CMTW

Item	Function
Number of channel	One channel × two units
Timer counter	16-bit/32-bit selectable up-counter Counting starts after the output of a counting enable signal from the prescaler. The counter returns to 0000 0000h after a compare match.
Prescaler	9-bit counter (operates to enable and disable the timer counter) Outputs four signals to enable counting. Selectable either of PCLKD/8, PCLKD/32, PCLKD/128, and PCLKD/512
Input capture	Up to two input capture input signals available.
Output compare	Up to two output compare output signals available.
Compare match	One compare match available.
Interrupt	• Compare match interrupt • Input capture 0 and 1 interrupts • Output compare 0 and 1 interrupts
Event link	One of the following three operations is enabled after an event signal is received: • Counting start • Event counting • Counting clear Also, the following event signal can be issued: • Compare match event
Low-power consumption function	The module-stop state can be set for each unit.

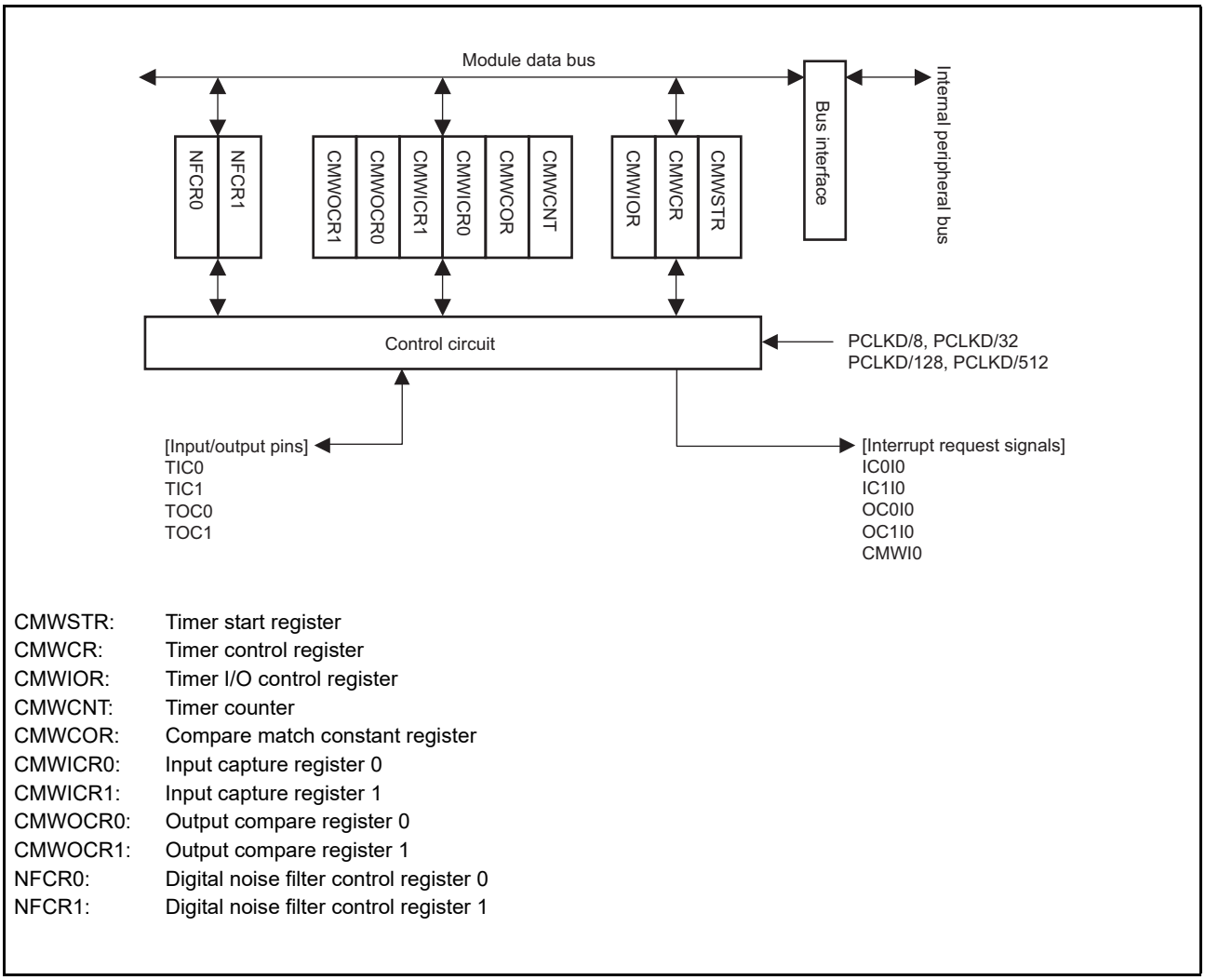


Figure 20.1 Block Diagram of CMTW (Unit 0)

Table 20.2 shows the CMTW pin configuration.

Table 20.2 Input/Output Pins of CMTW

Unit	Pin Name	I/O	Description
CMTW0	TIC0	Input	Input capture input 0
	TIC1	Input	Input capture input 1
	TOC0	Output	Output compare output 0
	TOC1	Output	Output compare output 1
CMTW1	TIC2	Input	Input capture input 2
	TIC3	Input	Input capture input 3
	TOC2	Output	Output compare output 2
	TOC3	Output	Output compare output 3

20.2 Register Descriptions

20.2.1 Timer Start Register (CMWSTR)

The CMWSTR register is used to start or stop the CMWCNT counter.

Address(es): CMTW0.CMWSTR A008 0300h, CMTW1.CMWSTR A008 0380h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STR
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	STR	Counter Start	0: The CMWCNT counter stops counting. (The value immediately before a stop of counting is retained and counting is stopped.) 1: The CMWCNT counter starts counting.	R/W
b15 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

STR Bit (Counter Start)

Specifies whether the timer counter operates or is stopped. The relevant prescaler operates or is stopped according to the settings of STR bit.

20.2.2 Timer Control Register (CMWCR)

The CMWCR register selects the counter clearing source and the counter input clock, and enables or disables interrupts. The CMWCR register should be set while the timer counter (CMWCNT) operation is stopped.

Address(es): CMTW0.CMWCR A008 0304h, CMTW1.CMWCR A008 0384h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	CCLR[2:0]			—	—	—	CMS	—	OC1IE	OC0IE	IC1IE	IC0IE	CMWIE	—	CKS[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	CKS[1:0]	Clock Select	b1 b0 0 0: PCLKD/8 0 1: PCLKD/32 1 0: PCLKD/128 1 1: PCLKD/512	R/W
b2	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b3	CMWIE	Compare Match Interrupt Enable	0: Disables a compare match interrupt request (CMWI). 1: Enables a compare match interrupt request (CMWI).	R/W
b4	IC0IE	Input Capture 0 Interrupt Enable	0: Disables an interrupt request by the input capture 0 bit (IC0I). 1: Enables an interrupt request by the input capture 0 bit (IC0I).	R/W
b5	IC1IE	Input Capture 1 Interrupt Enable	0: Disables an interrupt request by the input capture 1 bit (IC1I). 1: Enables an interrupt request by the input capture 1 bit (IC1I).	R/W
b6	OC0IE	Output Compare 0 Interrupt Enable	0: Disables an interrupt request by the output compare 0 bit (OC0I). 1: Enables an interrupt request by the output compare 0 bit (OC0I).	R/W
b7	OC1IE	Output Compare 1 Interrupt Enable	0: Disables an interrupt request by the output compare 1 bit (OC1I). 1: Enables an interrupt request by the output compare 1 bit (OC1I).	R/W
b8	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b9	CMS	Timer Counter Size	0: 32 bits 1: 16 bits	R/W
b12 to b10	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15 to b13	CCLR[2:0]	Counter Clear	b15 b13 0 0 0: The CMWCNT counter is cleared by CMWCOR register compare match. 0 0 1: The CMWCNT counter is not cleared. 0 1 0: The CMWCNT counter is not cleared. 0 1 1: The CMWCNT counter is not cleared. 1 0 0: The CMWCNT counter is cleared by CMWICR0 register input capture. 1 0 1: The CMWCNT counter is cleared by CMWICR1 register input capture. 1 1 0: The CMWCNT counter is cleared by CMWOCR0 register compare match. 1 1 1: The CMWCNT counter is cleared by CMWOCR1 register compare match.	R/W

CKS[1:0] Bits (Clock Select)

Select the clock to be input to the CMWCNT counter among four internal clocks obtained by dividing the peripheral clock (PCLKD). When the CMWSTR.STR bit is set to 1, the CMWCNT counter starts counting up based on the clock selected with the CMWCNT.CKS[1:0] bits.

CMWIE Bit (Compare Match Interrupt Enable)

Enables or disables compare match interrupt (CMWI) request generation when the CMWCNT counter and CMWCOR register values match.

IC0IE Bit (Input Capture 0 Interrupt Enable)

Enables or disables input capture interrupt 0 (IC0I) request generation when input capture is generated in the CMWICR0 register.

IC1IE Bit (Input Capture 1 Interrupt Enable)

Enables or disables input capture interrupt 1 (IC1I) request generation when input capture is generated in the CMWICR1 register.

OC0IE Bit (Output Compare 0 Interrupt Enable)

Enables or disables compare match interrupt 0 (OC0I) request generation when the CMWCNT counter and CMWOCR0 register values match.

OC1IE Bit (Output Compare 1 Interrupt Enable)

Enables or disables compare match interrupt 1 (OC1I) request generation when the CMWCNT counter and CMWOCR1 register values match.

CMS Bit (Timer Counter Size)

Selects either 16 or 32 bits as the size of the timer counter (CMWCNT). The size selected with the CMS bit is valid in the compare match constant register (CMWCOR), input capture registers (CMWICR0 and CMWICR1), and output compare registers (CMWOCR0 and CMWOCR1).

CCLR[2:0] Bits (Counter Clear)

Selects the CMWCNT counter clearing source.

20.2.3 Timer I/O Control Register (CMWIOR)

The CMWIOR register controls the CMWCOR, CMWICR0, CMWICR1, CMWOCR0, and CMWOCR1 registers. CMWIOR should be set while the timer counter (CMWCNT) operation is stopped.

Address(es): CMTW0.CMWIOR A008 0308h, CMTW1.CMWIOR A008 0388h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	CMWE	—	OC1E	OC0E	OC1[1:0]		OC0[1:0]		—	—	IC1E	IC0E	IC1[1:0]		IC0[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	IC0[1:0]	Input Compare Control 0	b1 b0 0 0: Input capture at the rising edge on the TIC0 pin. 0 1: Input capture at the falling edge on the TIC0 pin. 1 0: Input capture at both edges on the TIC0 pin. 1 1: Setting prohibited	R/W
b3, b2	IC1[1:0]	Input Capture Control 1	b3 b2 0 0: Input capture at the rising edge on the TIC1 pin. 0 1: Input capture at the falling edge on the TIC1 pin. 1 0: Input capture at both edges on the TIC1 pin. 1 1: Setting prohibited	R/W
b4	IC0E	Input Capture Enable 0	0: Disables the input capture operation of the CMWICR0 register. 1: Enables the input capture operation of the CMWICR0 register.	R/W
b5	IC1E	Input Capture Enable 1	0: Disables the input capture operation of the CMWICR1 register. 1: Enables the input capture operation of the CMWICR1 register.	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b9, b8	OC0[1:0]	Output Compare Control 0	b9 b8 0 0: Retains the output value.*1 0 1: Initially outputs 0 and toggles the output value upon compare match. 1 0: Initially outputs 1 and toggles the output value upon compare match. 1 1: Setting prohibited	R/W
b11, b10	OC1[1:0]	Output Compare Control 1	b11 b10 0 0: Retains the output value.*1 0 1: Initially outputs 0 and toggles the output value upon compare match. 1 0: Initially outputs 1 and toggles the output value upon compare match. 1 1: Setting prohibited	R/W
b12	OC0E	Compare Match Enable 0	0: Disables the compare match operation using the CMWOCR0 register. 1: Enables the compare match operation using the CMWOCR0 register.	R/W
b13	OC1E	Compare Match Enable 1	0: Disables the compare match operation using the CMWOCR1 register. 1: Enables the compare match operation using the CMWOCR1 register.	R/W
b14	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b15	CMWE	Compare Match Enable	0: Disables the compare match operation using the CMWCOR register. 1: Enables the compare match operation using the CMWCOR register.	R/W

Note 1. After reset, 0 is output until the CMWIOR register is set.

IC0[1:0] Bits (Input Compare Control 0)

Selects the input capture operation of the CMWICR0 register.

IC1[1:0] Bits (Input Capture Control 1)

Selects the input capture operation of the CMWICR1 register.

IC0E Bit (Input Capture Enable 0)

Enables or disables the input capture operation of the CMWICR0 register.

IC1E Bit (Input Capture Enable 1)

Enables or disables the input capture operation of the CMWICR1 register.

OC0[1:0] Bits (Output Compare Control 0)

Sets the output compare operation using the CMWOCR0 register.

OC1[1:0] Bits (Output Compare Control 1)

Sets the output compare operation using the CMWOCR1 register.

OC0E Bit (Compare Match Enable 0)

Enables or disables the compare match operation using the CMWOCR0 register.

OC1E Bit (Compare Match Enable 1)

Enables or disables the compare match operation using the CMWOCR1 register.

CMWE Bit (Compare Match Enable)

Enables or disables the compare match operation using the CMWCOR register.

20.2.4 Timer Counter (CMWCNT)

The CMWCNT counter is used as a readable/writable up-counter.

Before starting counter operation, the timer control register (CMWCR) should be set. When the 16-bit counter size is selected with the CMS bit in the timer control register (CMWCR), bits 15 to 0 in the CMWCNT counter are valid. Since access to this register is in 32-bit units, when writing, write 32-bit values to it with 0000h in the higher-order bits. If a value other than 0000h is set in the higher-order bits, a value greater than 0000 FFFFh may be read when this register is read.

When the STR bit is set to 1, the CMWCNT counter starts counting. When the STR bit is set to 0, the CMWCNT counter retains the value immediately before a stop of counting and stops counting.

Address(es): CMTW0.CMWCNT A008 0310h, CMTW1.CMWCNT A008 0390h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

20.2.5 Compare Match Constant Register (CMWCOR)

The CMWCOR register is a readable/writable register that specifies the time up to a compare match between the timer counter (CMWCNT) value and CMWCOR value. When the 16-bit counter size is selected with the CMS bit in the timer control register (CMWCR), bits 15 to 0 in this register are valid. Since access to this register is in 32-bit units, when writing, write 32-bit values to it with 0000h in the higher-order bits.
The cycle for compare matches is as follows.

Compare-match cycle = (setting of the CMWCOR register + 1) × counter-clock cycle*1

Note 1. This is a clock cycle set by the CMWCR.CKS[1:0] bits.

Address(es): CMTW0.CMWCOR A008 0314h, CMTW1.CMWCOR A008 0394h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

20.2.6 Input Capture Registers 0 and 1 (CMWICR0 and CMWICR1)

The CMWICR0 and CMWICR1 registers are read-only registers in which the CMWCNT counter value is stored when an input capture is generated.
When the 16-bit counter size is selected with the CMS bit in the timer control register (CMWCR), bits 15 to 0 in these registers are valid. Writing to these registers is invalid.

Address(es): CMTW0.CMWICR0 A008 0318h, CMTW0.CMWICR1 A008 031Ch,
CMTW1.CMWICR0 A008 0398h, CMTW1.CMWICR1 A008 039Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

20.2.7 Output Compare Registers 0 and 1 (CMWOCR0 and CMWOCR1)

The CMWOCR0 and CMWOCR1 registers are readable/writable registers that set the value to be compared when an output compare is generated.

When the 16-bit counter size is selected with the CMS bit in the timer control register (CMWCR), bits 15 to 0 of these registers become valid. Since access to this register is in 32-bit units, when writing, write 32-bit values to it with 0000h in the higher-order bits.

Address(es): CMTW0.CMWOCR0 A008 0320h, CMTW0.CMWOCR1 A008 0324h,
CMTW1.CMWOCR0 A008 03A0h, CMTW1.CMWOCR1 A008 03A4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

20.2.8 Digital Noise Filter Control Register 0 (NFCR0)

The NFCR0 register controls digital noise filters for input capture signals (TICn, n = 0, 1) of CMTW0.
The NFCR0 register should be set while the CMTW0.CMWCNT counter operation is stopped.

Address(es): CMTW.NFCR0 A008 0400h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	NFCS0[1:0]		NF1EN	NF0EN
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	NF0EN	Digital Noise Filter Enable 0	0: The digital noise filter for the TIC0 pin is disabled. 1: The digital noise filter for the TIC0 pin is enabled.	R/W
b1	NF1EN	Digital Noise Filter Enable 1	0: The digital noise filter for the TIC1 pin is disabled. 1: The digital noise filter for the TIC1 pin is enabled.	R/W
b3, b2	NFCS0 [1:0]	Digital Noise Filter Clock Select 0	b3 b2 0 0: PCLKD/1 0 1: PCLKD/8 1 0: PCLKD/32 1 1: PCLKD/64	R/W
b31 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

NF0EN Bit (Digital Noise Filter Enable 0)

This bit enables or disables the noise filter for the TIC0 pin. Before changing the value of this bit, select the output compare function for the relevant pin in the timer I/O control register. Changing the value of the bit when the output compare function is not selected may lead to the internal generation of an unexpected edge.

NF1EN Bit (Digital Noise Filter Enable 1)

This bit enables or disables the noise filter for the TIC1 pin. Before changing the value of this bit, select the output compare function for the relevant pin in the timer I/O control register. Changing the value of the bit while the output compare function is not selected may lead to the internal generation of an unexpected edge.

NFCS0[1:0] Bits (Digital Noise Filter Clock Select 0)

These bits select the sampling clock for the digital noise filter of CMTW0.
The input-capture signal is sampled on rising edges of the selected clock signal. If the sampled levels match three times in a row, the given level is regarded as the input-capture signal. If the levels do not match, the existing value is retained.
After setting these bits, wait for two selected sampling periods before selecting the input capture function.

20.2.9 Digital Noise Filter Control Register 1 (NFCR1)

The NFCR1 register controls digital noise filters for input capture signals (TICn, n = 2, 3) of CMTW1.
The NFCR1 register should be set while the CMTW1.CMWCNT counter operation is stopped.

Address(es): CMTW.NFCR1 A008 0404h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	NFCS1[1:0]		NF3EN	NF2EN
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	NF2EN	Digital Noise Filter Enable 2	0: The digital noise filter for the TIC2 pin is disabled. 1: The digital noise filter for the TIC2 pin is enabled.	R/W
b1	NF3EN	Digital Noise Filter Enable 3	0: The digital noise filter for the TIC3 pin is disabled. 1: The digital noise filter for the TIC3 pin is enabled.	R/W
b3, b2	NFCS1 [1:0]	Digital Noise Filter Clock Select 1	b3 b2 0 0: PCLKD/1 0 1: PCLKD/8 1 0: PCLKD/32 1 1: PCLKD/64	R/W
b31 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

NF2EN Bit (Digital Noise Filter Enable 2)

This bit enables or disables the noise filter for the TIC2 pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, select the output compare function for the relevant pin in the timer I/O control register before doing so.

NF3EN Bit (Digital Noise Filter Enable 3)

This bit enables or disables the noise filter for the TIC3 pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, select the output compare function for the relevant pin in the timer I/O control register before doing so.

NFCS1[1:0] Bits (Digital Noise Filter Clock Select 1)

These bits select the sampling clock for the digital noise filter of CMTW1.
The input-capture signal is sampled on rising edges of the selected clock signal. If the sampled levels match three times in a row, the given level is regarded as the input-capture signal. If the levels do not match, the existing value is retained.
After setting these bits, wait for two selected sampling periods before selecting the input capture function.

20.3 Operation

When the CMWCR register is set and then the STR bit in CMWSTR is set to 1, the CMTW starts counter operation. Setting the CMWSTR.STR bit to 0 enables the CMWCNT counter to retain the value immediately before a stop of counting and stop counting. Setting CMWIOR register enables using the compare match function, input capture input function, and output compare output function.

20.3.1 Period Counting Operation

When the internal clock is selected by using the CMWCNT.CKS[1:0] bits and the CMWSTR.STR bit is set to 1, the CMWCNT counter starts counting up cycles of the selected clock. When the CCLR[2:0] bits in CMWCR are set so that CMWCNT should be cleared by a specific counter clearing source and the counter clearing source is generated, the CMWCNT counter is cleared to 0000 0000h and continues incrementing. When the CCLR[2:0] bits are set so that CMWCNT should not be cleared by any specific counter clearing source, the CMWCNT counter is cleared to 0000 0000h only when an overflow is generated (FFFF FFFFh → 0000 0000h (when the counter size is 32 bits) or 0000 FFFFh → 0000 0000h (when the counter size is 16 bits)) and continues incrementing.

20.3.2 Compare Match Function

When the values of the CMWCNT counter and CMWCOR register match, a compare match interrupt (CMWI) is generated. The CMWCNT counter operates as follows according to the setting of CMWCR.CCLR[1:0].

1. When CMWCR.CCLR[2:0] = 000b
   When the values of the CMWCNT counter and CMWCOR register match, the CMWCNT counter is cleared to 0000 0000h. The CMWCNT counter then restarts counting up from 0000 0000h.
2. When CMWCR.CCLR[2:0] ≠ 000b
   Even when the values of the CMWCNT counter and CMWCOR register match, the CMWCNT counter is not cleared to 0000 0000h but continues counting up until the clearing condition set in CMWCR.CCLR[1:0] is satisfied or the value of the counter reaches FFFF FFFFh (when the size of the counter is 32 bits) or ****FFFFh (when the size of the counter is 16 bits).
   The CMWCNT counter is then cleared to 0000 0000h and restarts counting up from 0000 0000h.

Figure 20.2 shows an example of procedure for setting compare match operation.

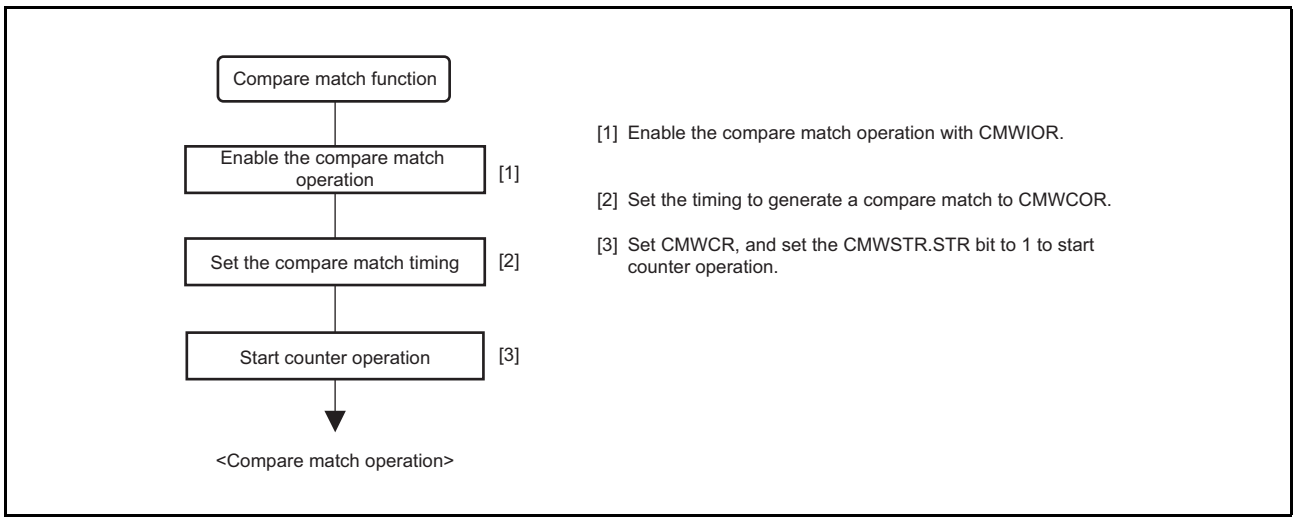


Figure 20.2 Procedure for Setting Compare Match Operation

Figure 20.3 shows an example when compare match with CMWCOR is set as a counter clearing source.

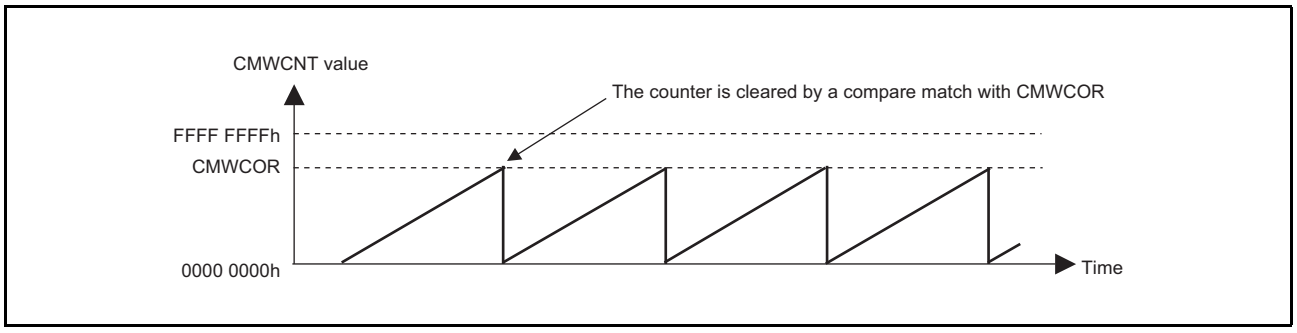


Figure 20.3 Example of Compare Match Operation

Figure 20.4 shows an example when CMWCOR is set to FFFF FFFFh and an overflow is detected.

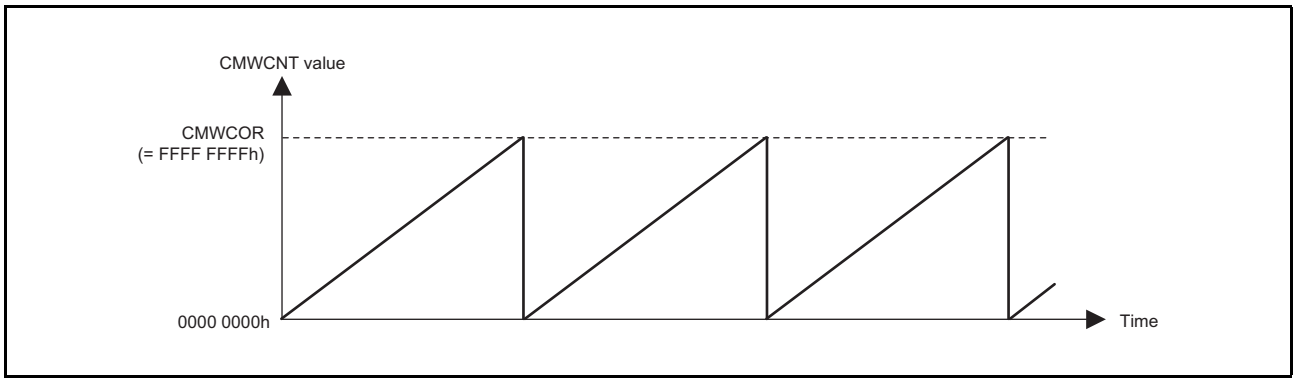


Figure 20.4 Example of Compare Match Operation (Overflow Detected)

20.3.3 Output Compare Function

Using the output compare function, toggle output from the relevant output pins can be provided. When the CMWCNT counter value matches either of the values of CMWOCR0 or CMWOCR1 register, the output compare interrupt (OC0I or OC1I) is generated. The CMWCNT counter operates as follows according to the setting of CMWCR.CCLR[2:0].

1. When CMWCR.CCLR[2:0] = 110b
   When the values of the CMWCNT counter and CMWOCR0 register match, the CMWCNT counter is cleared to 0000 0000h. The CMWCNT counter then restarts counting up from 0000 0000h.
2. When CMWCR.CCLR[2:0] = 111b
   When the values of the CMWCNT counter and CMWOCR1 register match, the CMWCNT counter is cleared to 0000 0000h. The CMWCNT counter then restarts counting up from 0000 0000h.
3. When CMWCR.CCLR[2:0] ≠ 110b or 111b
   Even when the values of the CMWCNT counter and CMWOCR0 or CMWOCR1 register match, the CMWCNT counter is not cleared to 0000 0000h but continues counting up until the clearing condition set in CMWCR.CCLR[2:0] is satisfied or the value of the counter reaches FFFF FFFFh (when the size of the counter is 32 bits) or ****FFFFh (when the size of the counter is 16 bits). The CMWCNT counter is then cleared to 0000 0000h and restarts counting up from 0000 0000h.

Figure 20.5 shows an example of procedure for setting output compare operation.

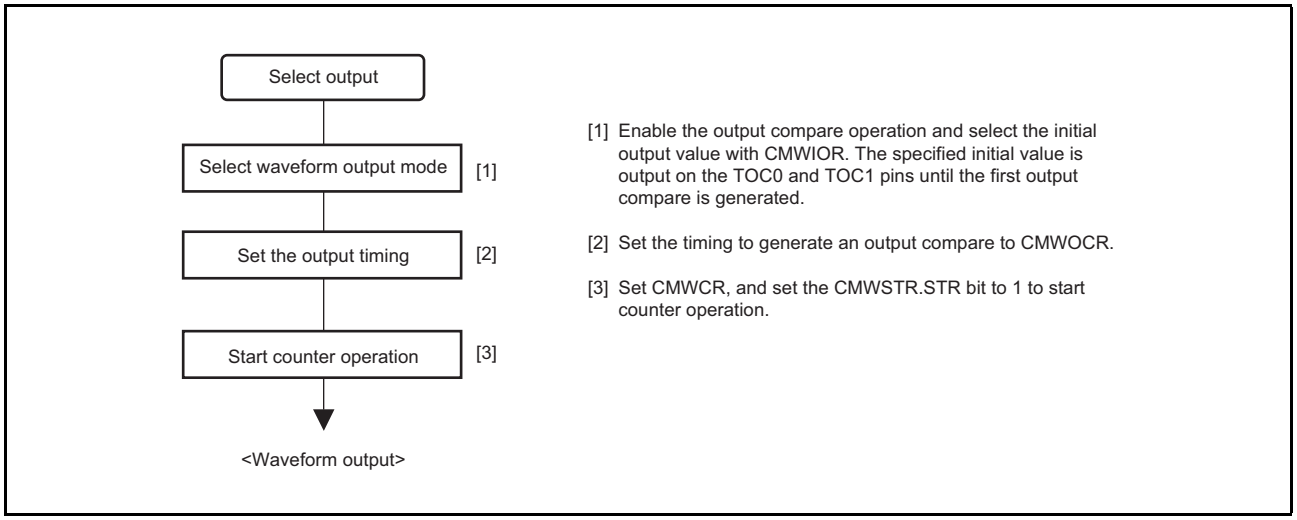


Figure 20.5 Procedure for Setting Output Compare Operation

Figure 20.6 shows an example when the counter is cleared upon compare match with CMWOCR1 register and toggle outputs are provided from the TOC0 and TOC1 pins.

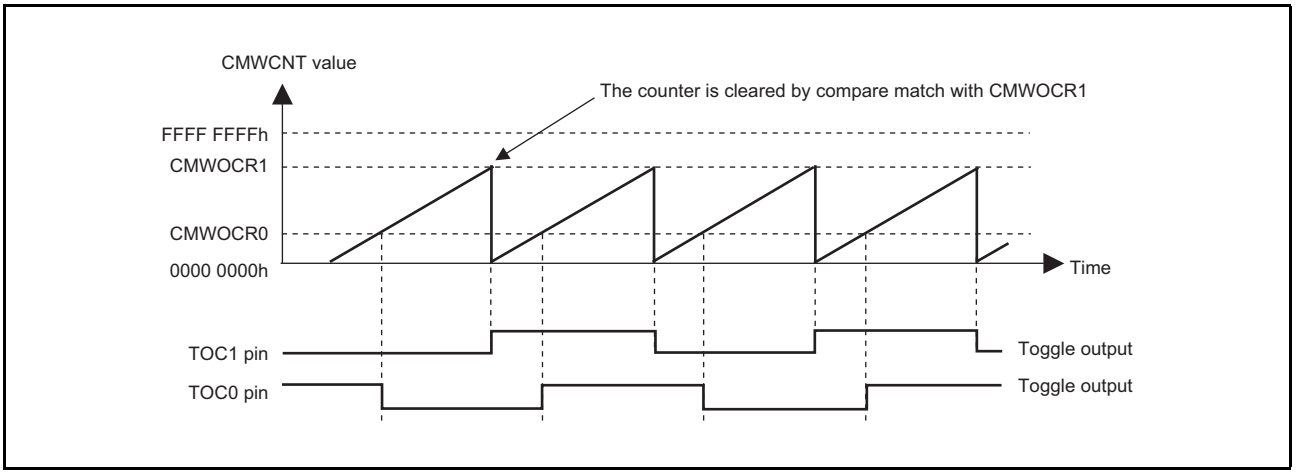


Figure 20.6 Example of Output Compare Operation

20.3.4 Input Capture Function

Through detecting the edge on the TIC0 and TIC1 pin input, the CMWCNT counter value can be transferred to CMWICR0 and CMWICR1 registers, respectively. The edges to be detected can be selected from among the rising edge alone, falling edge alone, and both the rising and falling edges. When the CMWCNT counter value is transferred to CMWICR0 or CMWICR1 register using the input capture function, an input compare interrupt (IC0I or IC1I) is generated. The CMWCNT counter operates as follows according to the setting of CMWCR.CCLR[2:0].

1. When CMWCR.CCLR[2:0] = 100b
   When the CMWCNT counter value is transferred to CMWICR0 using the input capture operation, the CMWCNT counter is cleared to 0000 0000h.
   The CMWCNT counter then restarts counting up from 0000 0000h.
2. When CMWCR.CCLR[2:0] = 101b
   When the CMWCNT counter value is transferred to CMWICR1 using the input capture operation, the CMWCNT counter is cleared to 0000 0000h.
   The CMWCNT counter then restarts counting up from 0000 0000h.
3. When CMWCR.CCLR[2:0] ≠ 100b or 101b
   Even when the CMWCNT counter value is transferred to CMWICR0 or CMWICR1 using the input capture operation, the CMWCNT counter is not cleared to 0000 0000h but continues counting up until the clearing condition set in CMWCR.CCLR[2:0] is satisfied or the value of the counter reaches FFFF FFFFh (when the size of the counter is 32 bits) or ****FFFFh (when the size of the counter is 16 bits). The CMWCNT counter is then cleared to 0000 0000h and restarts counting up from 0000 0000h.

Figure 20.7 shows an example of procedure for setting input capture operation.

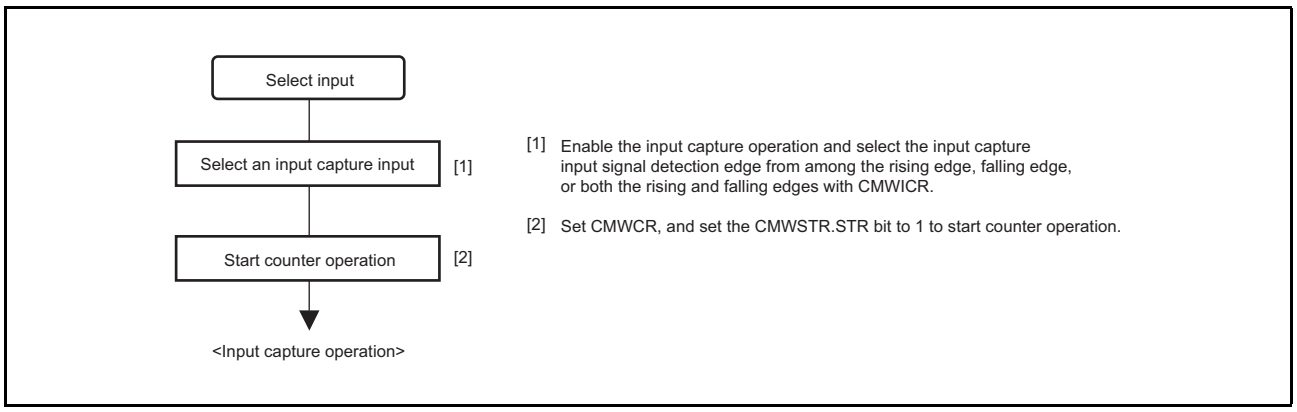


Figure 20.7 Procedure for Setting Input Capture Operation

Figure 20.8 shows an example in which both the rising and falling edges are selected for the TIC0 pin input capture input edge and the falling edge for the TIC1 pin, and the CMWCNT counter is cleared by a CMWICR1 register input capture.

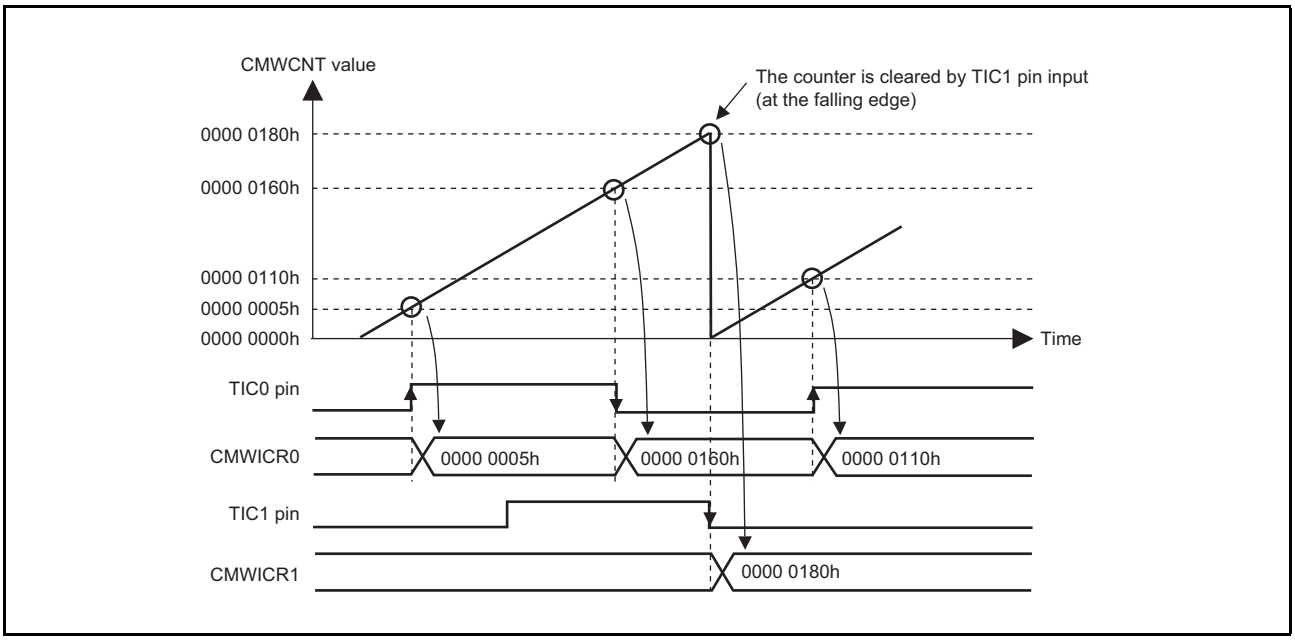


Figure 20.8 Example of Input Capture Operation

20.3.5 Counter Size

With the CMTW, either 16 or 32 bits can be selected as the counter size by using the CMWSTR.CMS bit. When the counter is used as a 32-bit counter, set the CMWCOR, CMWOCR0, or CMWOCR1 register to the desired values in 32-bit units. In reading, all 32 bits of CMWICR0 and CMWICR1 are valid. When the counter is used as a 16-bit counter, a 32-bit value should be set to the CMWCOR register with 0000h in the higher-order bits. Similarly, a 32-bit value should be set to CMWOCR0 and CMWOCR1 registers with 0000h in the higher-order bits.
A 32-bit value with 0000h in the higher-order bits is read from CMWICR0 and CMWICR1 registers.

20.3.6 Count Timing based on CMWCNT

One of four clocks (PCLKD/8, PCLKD/32, PCLKD/128, and PCLKD/512) obtained by dividing the peripheral clock (PCLKD) can be selected with the CMWCR.CKS[1:0] bits. Figure 20.9 shows the timing.

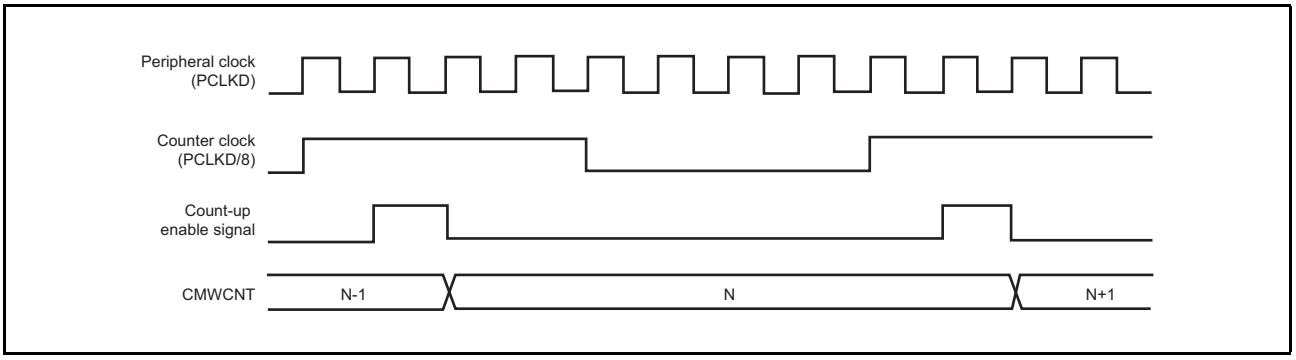


Figure 20.9 Count Timing (PCLKD/8)

20.3.7 Output Compare Output Timing

A compare match signal is generated in the last state in which the CMWOCR register and CMWCNT counter values match (the CMWCNT counter value is updated immediately after the state). That is, the compare match signal is not generated if the CMWCNT counter clock is not input after a match between the CMWOCR register and CMWCNT counter values. When a compare match signal is generated, the output compare output pin (TOC) changes in accord with the setting of the OC0 or OC1 bit in the CMWIOR register. Figure 20.10 shows output compare output timing.

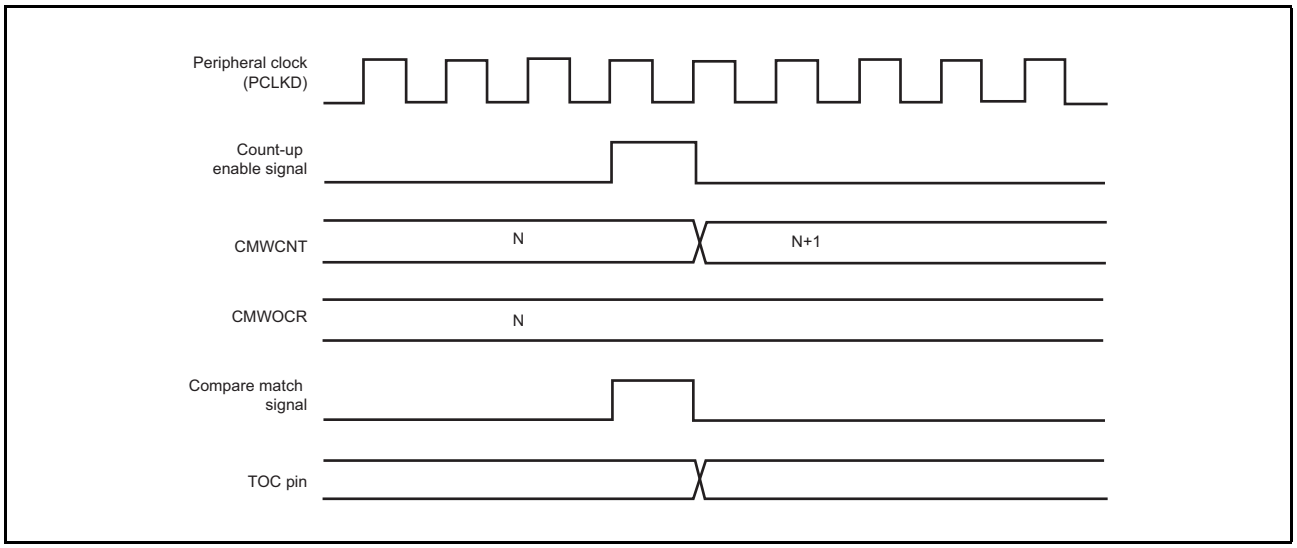


Figure 20.10 Output Compare Output Timing

20.3.8 Input Capture Signal Timing

Figure 20.11 shows the input capture timing.

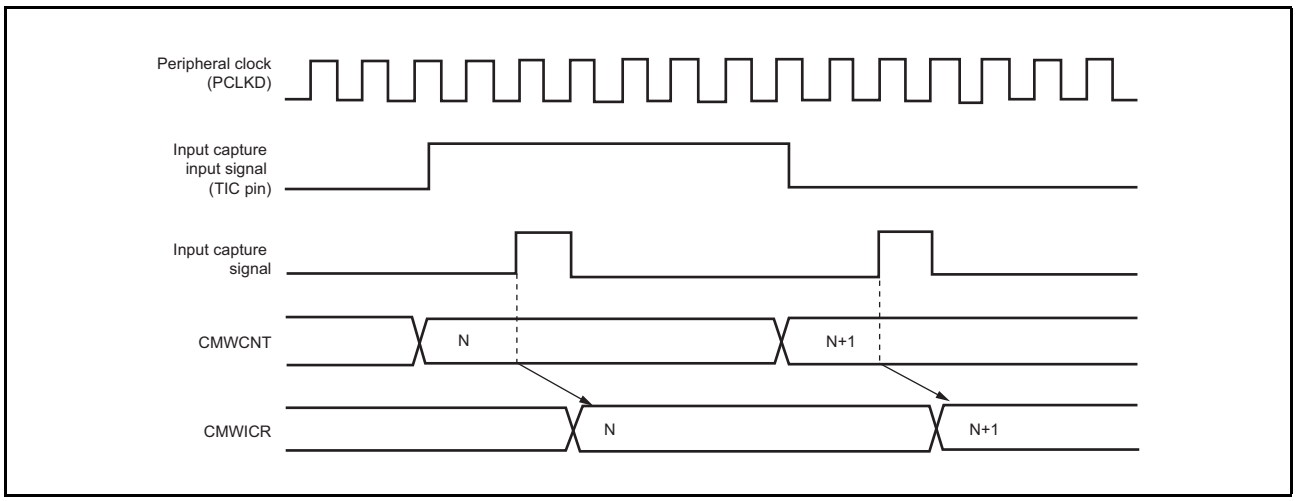


Figure 20.11 Input Capture Input Signal Timing

20.3.9 Digital Noise Filtering

The noise filter samples CMTW input-capture signals at the frequency of the sampling clock and the pulses with levels that only match once or twice are removed.
The digital noise filtering functionality includes enabling and disabling of the noise filtering for each pin. Figure 20.12 shows the timing of digital noise filtering.

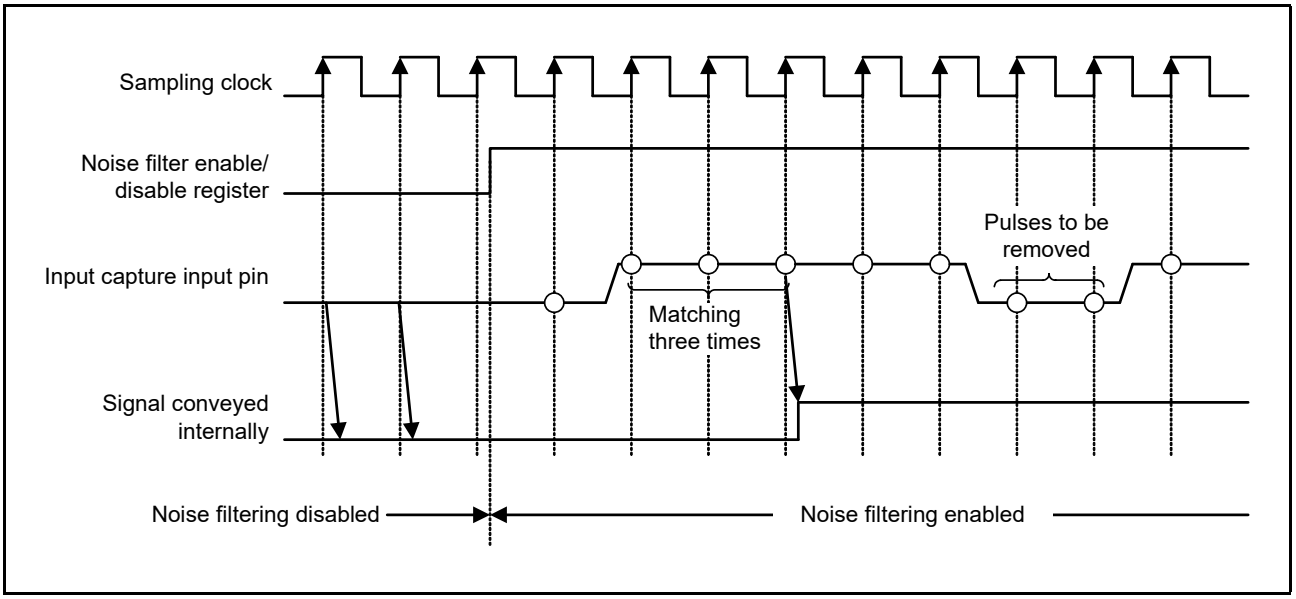


Figure 20.12 Timing of Digital Noise Filtering

If digital noise filtering is enabled, input capture proceeds on edges of the noise filtered signal after a delay of (minimum sampling interval × 2 + PCLKD) due to noise filtering of the input capture input.

20.4 Interrupts

20.4.1 CMTW Interrupt Sources and DMAC Transfer Requests

The CMTW has five interrupt sources: two input capture interrupt requests (IC0In and IC1In), two output compare interrupt requests (OC0In and OC1In), and a compare match interrupt request (CMWIn) (n = 0, 1).
Table 20.3 shows the interrupt sources and priority. The interrupt sources can be enabled or disabled using the IC0IE, IC1IE, OC0IE, OC1IE, and CMWIE bits in CMWCR and are separately issued to the interrupt controller.

Table 20.3 CMTW Interrupt Sources

Interrupt Source	Interrupt	Interrupt Enable Bit	DMAC Activation	Priority
CMWIn	Interrupt caused by compare match	CMWIE	Possible	High
IC0In	Interrupt caused by input capture 0	IC0IE	Possible	
IC1In	Interrupt caused by input capture 1	IC1IE	Possible	
OC0In	Interrupt caused by output compare 0	OC0IE	Possible	
OC1In	Interrupt caused by output compare 1	OC1IE	Possible	Low

20.4.2 Timing of Compare Match Interrupt Generation

When the values of the CMWCNT counter and CMWCOR register match, a compare match interrupt (CMWI) is generated. The compare match signal is generated at the end of the cycle where the values matched (i.e. when the CMWCNT counter is updated from the matching counter value). The compare match signal, therefore, is not generated until a further cycle of the input clock (PCLKD/8, PCLKD/32, PCLKD/128, or PCLKD/512) for the CMWCNT counter arrives after the values of the CMWCNT counter and CMWCOR register have matched. Figure 20.13 shows the timing of compare match interrupt generation.

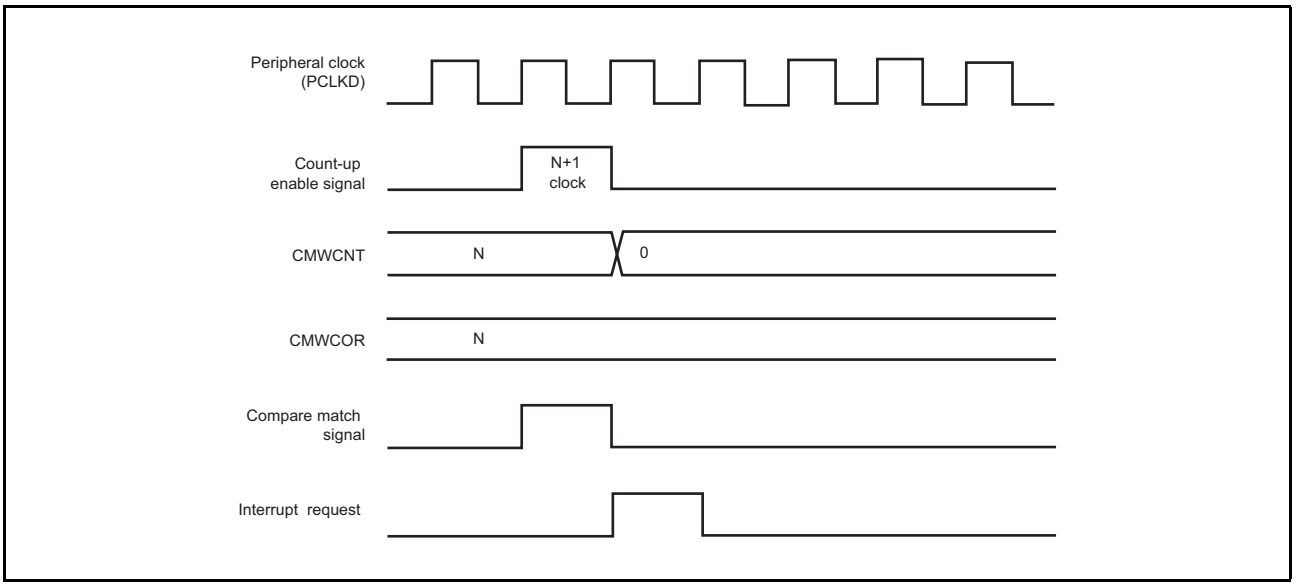


Figure 20.13 Timing of Compare Match Interrupt Generation

20.4.3 Timing of Output Compare Interrupt Generation

Figure 20.14 shows the timing of output compare interrupt generation.

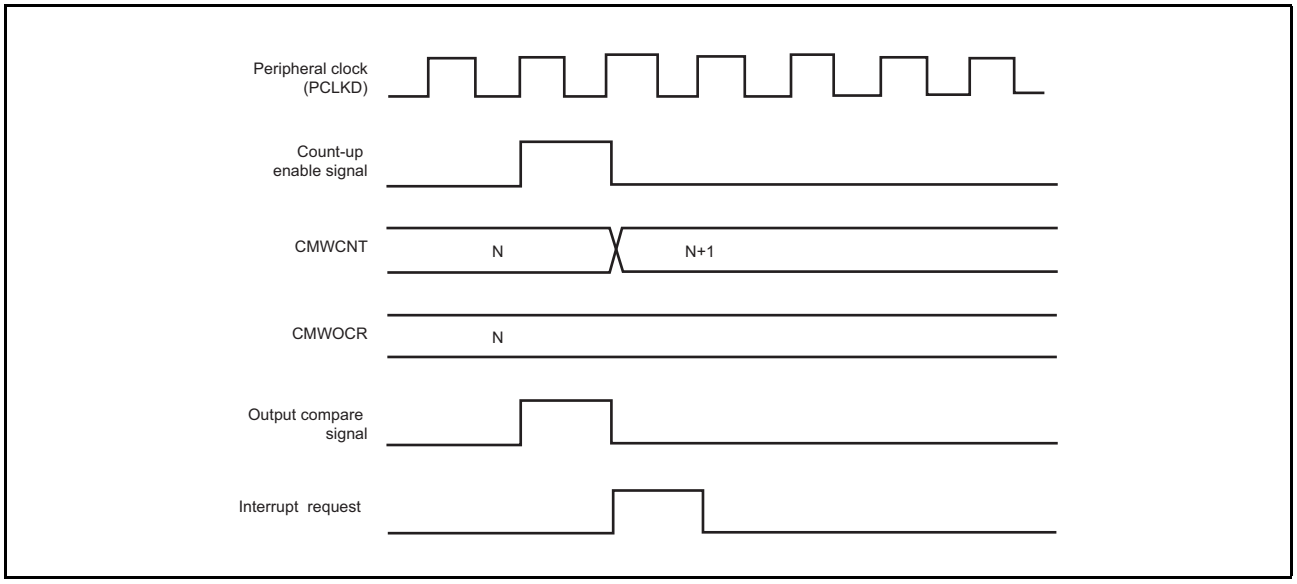


Figure 20.14 Timing of Output Compare Interrupt Generation

20.4.4 Timing of Input Capture Interrupt Generation

Figure 20.15 shows the timing of input capture interrupt generation.

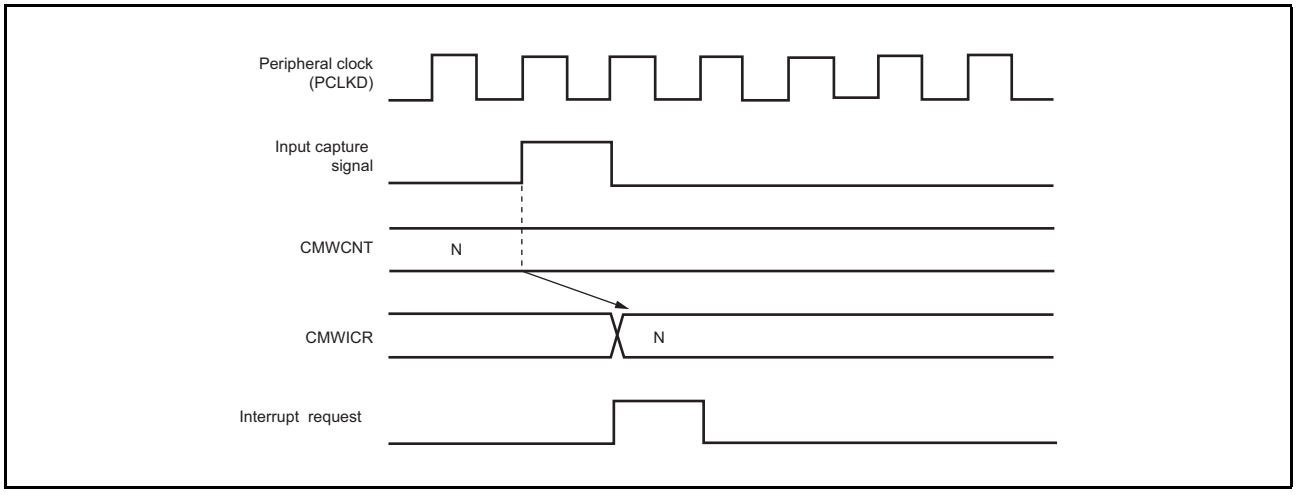


Figure 20.15 Timing of Input Capture Interrupt Generation

20.5 Event Link Operation

20.5.1 Issuing Events to the ELC

The CMTW can issue event signals to the event link controller (ELC) in response to the following events.

Compare Match Event

In response to a compare match, the CMTW simultaneously issues an interrupt request and a compare match event signal to the ELC. The event signal is issued regardless of the settings of the corresponding interrupt request enable bit (CMWCR.CMWIE bit).

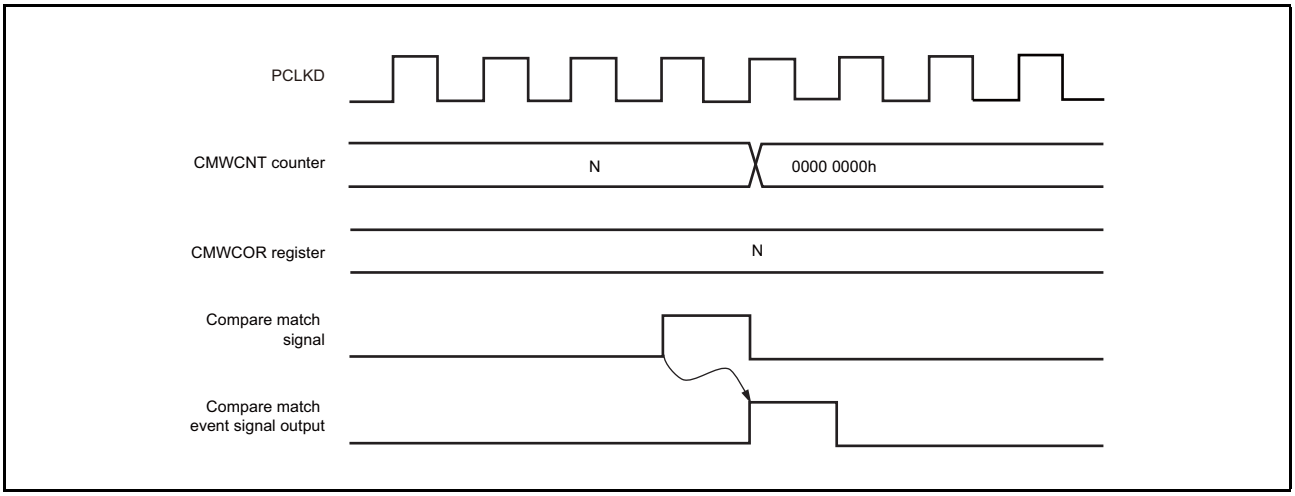


Figure 20.16 Timing of Issuing a Compare Match Event Signal

20.5.2 Actions on Acceptance of Event Signals from ELC

CMTW can respond with any of the following three actions when the action is set in the event link controller (ELC) and CMTW accepts the event signal.

(1) Start Counting

When an event signal is received while starting to count is the selected action, the STR bit in CMWSTR (the timer start register) is set to 1 and counting starts.

However, the event signal is ignored if it is accepted when the CMWSTR.STR bit has already been set to 1.

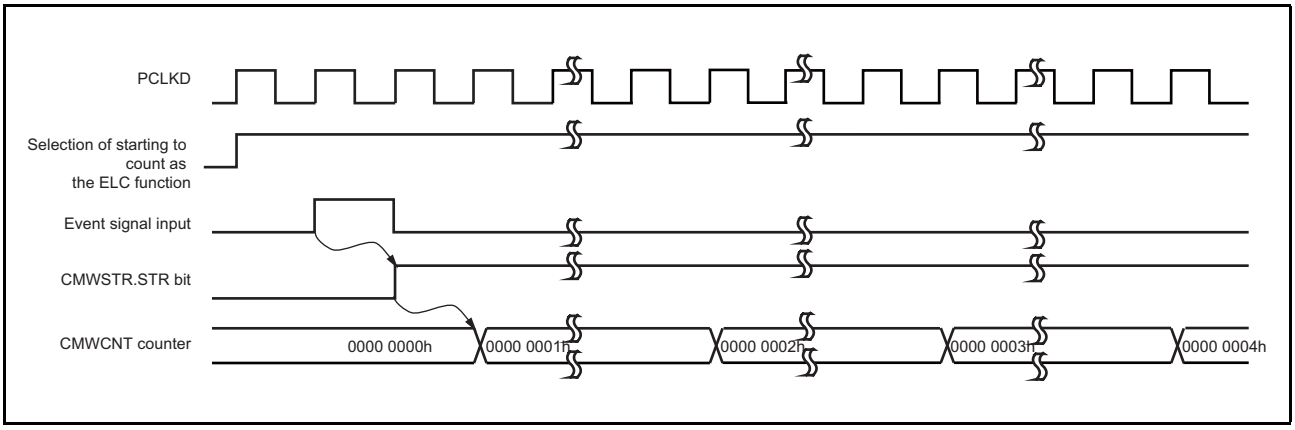


Figure 20.17 Starting to Count on Acceptance of the Event Signal

(2) Event Counting

When an event signal is received while counting events is the selected action, CMWCNT (the timer counter) is incremented. In this case, however, the STR bit in CMWSTR (the timer start register) must be set to 1 before the event signal is received.

In event counting, the setting of the CKS[1:0] bit in CMWCR (the timer control register) is not effective.

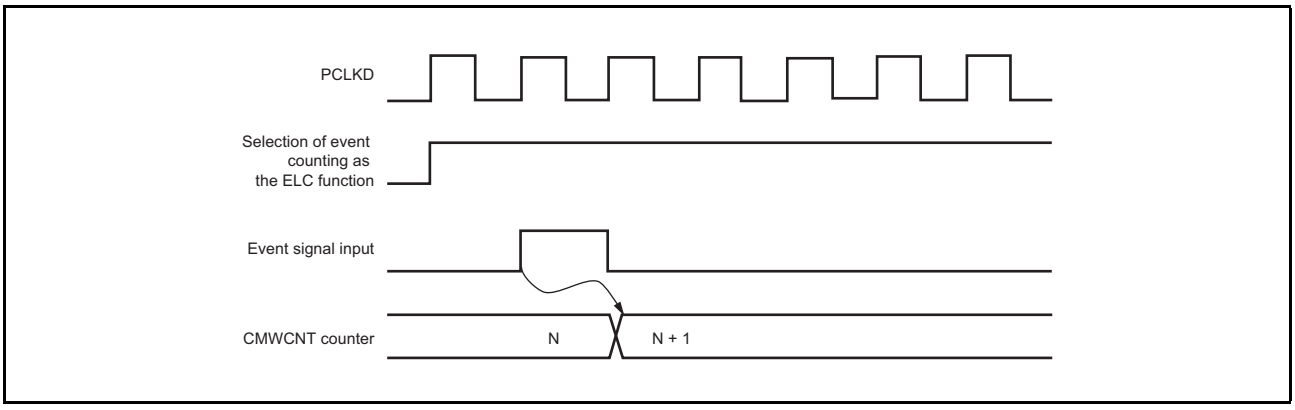


Figure 20.18 Counting an Event on Acceptance of the Event Signal

(3) Clear a Counter

Clearing of the CMTW counter is selected in the ELOPH register of the ELC. When the event specified in the corresponding ELSRn register is generated, the timer counter (CMWCNT) is returned to its initial value (0000h). Counting continues, however, if the setting of the STR bit of the timer start register (CMWSTR) is 1 at this time, so counting can be automatically restarted in this way. A timing chart of restarting the counter is shown in Figure 20.19

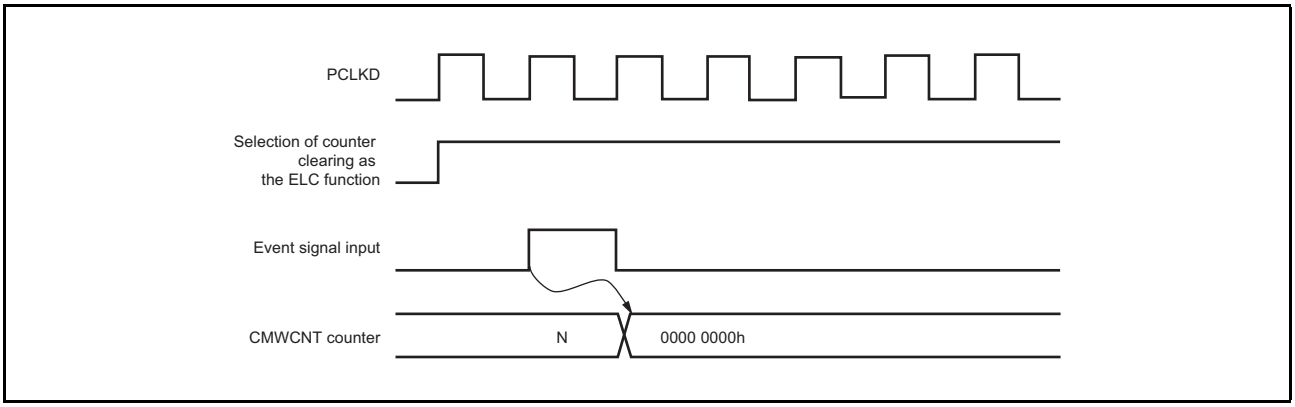


Figure 20.19 Restarting Counting on Acceptance of the Event Signal

20.6 Usage Notes

20.6.1 Module-Stop Function

The CMTW operation can be enabled or disabled by using the module-stop control register (MSTPCRA). In the initial setting, the CMTW is in the module-stop state. Register access is enabled by releasing the CMTW from the module-stop state. For details, refer to section 9, Low-Power Consumption Function.

20.6.2 Contention between CMWCNT Counter Writing and Compare Match

If the compare match signal is generated during CPU writing to the CMWCNT counter, the compare match request is output but the counter is not cleared since the CPU writing to the counter is given priority.
Figure 20.20 shows the timing of contention between CMWCNT counter writing and compare match.

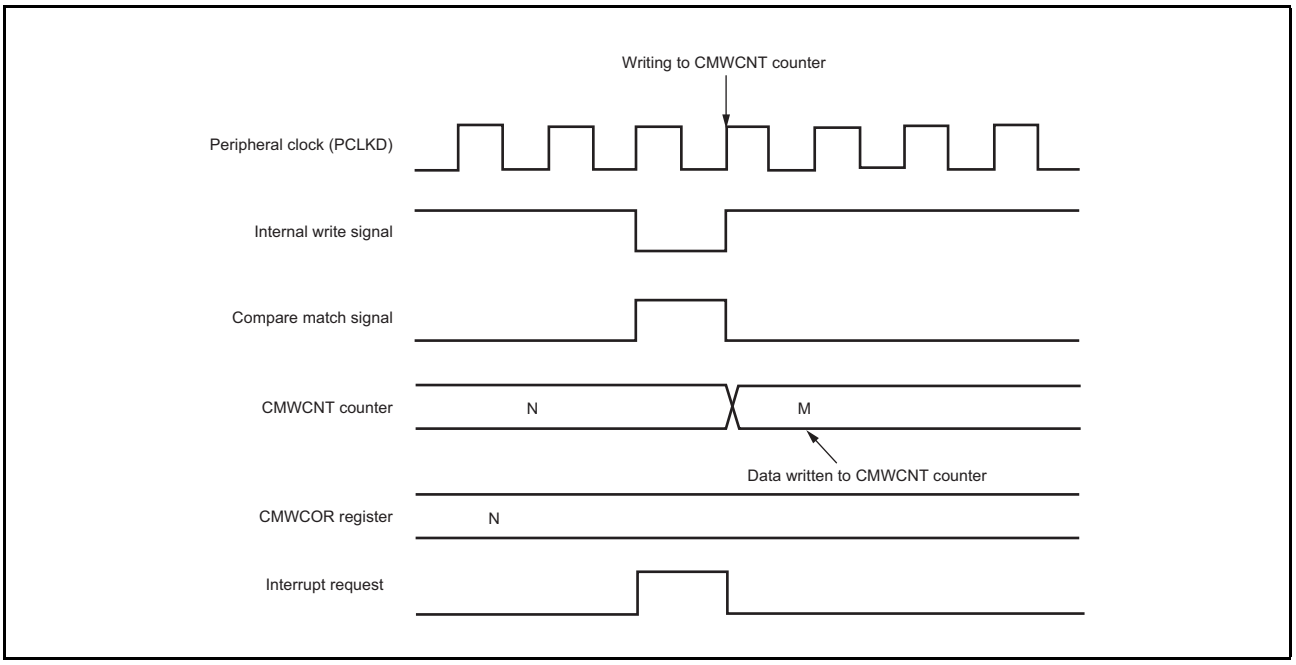


Figure 20.20 Contention between CMWCNT Counter Writing and Compare Match

20.6.3 Contention between CMWCNT Counter Writing and Incrementing or Clearing

In case of contention between incrementation or clearing of the CMWCNT counter and CPU writing to the counter, the counter is not actually incremented or cleared since the CPU writing to the CMWCNT counter is given priority.
Figure 20.21 shows the timing in the case of contention between writing to the CMWCNT counter and incrementation or clearing.

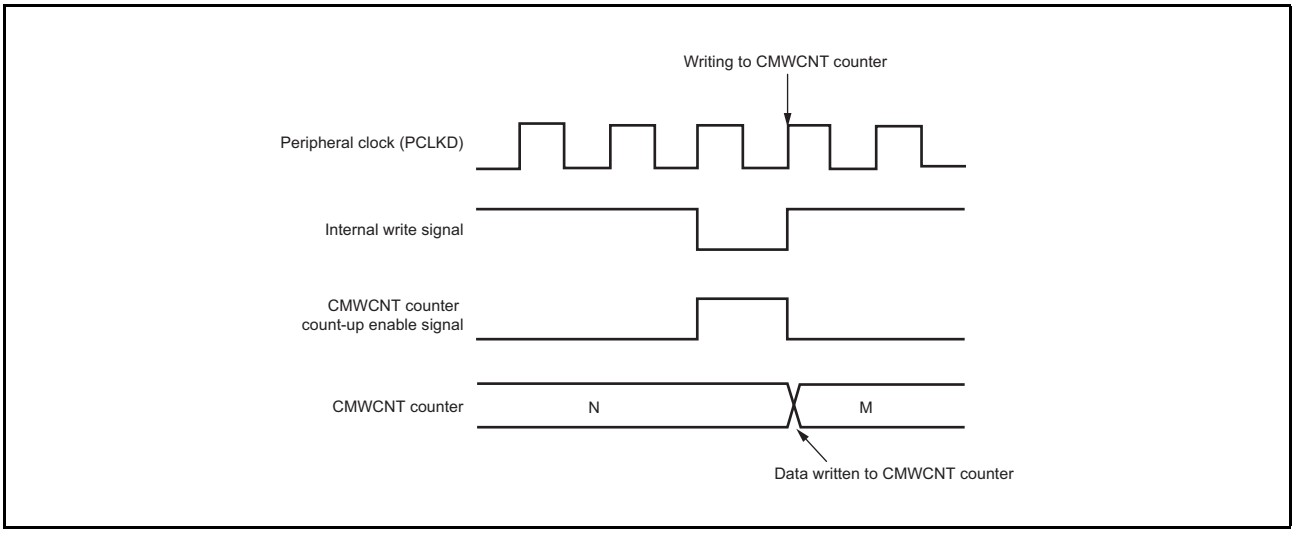


Figure 20.21 Contention between CMWCNT Counter Writing and Incrementing

20.6.4 Contention between CMWCOR Register Writing and Compare Match

If the compare match is generated during CPU writing to the CMWCOR register, the CPU writing to the CMWCOR register proceeds and also the compare match signal is output.
Figure 20.22 shows the timing of contention between CMWCOR register writing and compare match.

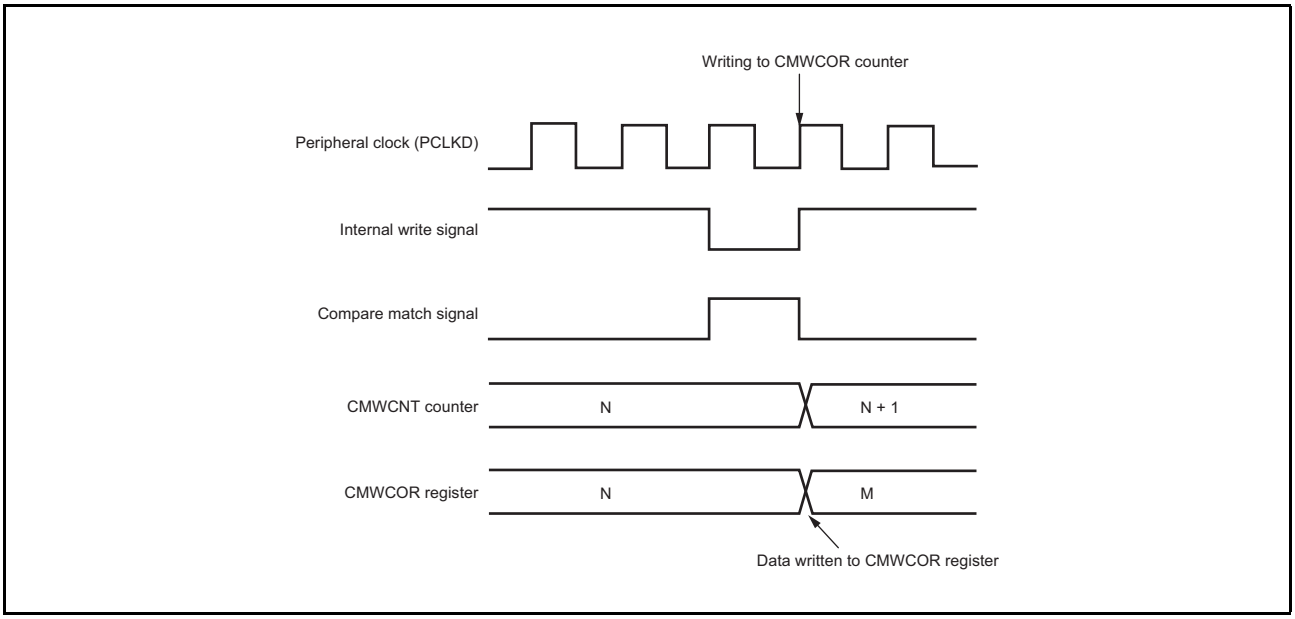


Figure 20.22 Contention between CMWCOR Register Writing and Compare Match

20.6.5 Contention between CMWOCR Register Writing and Compare Match

If the compare match is generated during CPU writing to the CMWOCR register, the CPU writing to the CMWOCR register proceeds and also the compare match signal is output.
Figure 20.23 shows the timing of contention between CMWOCR register writing and compare match.

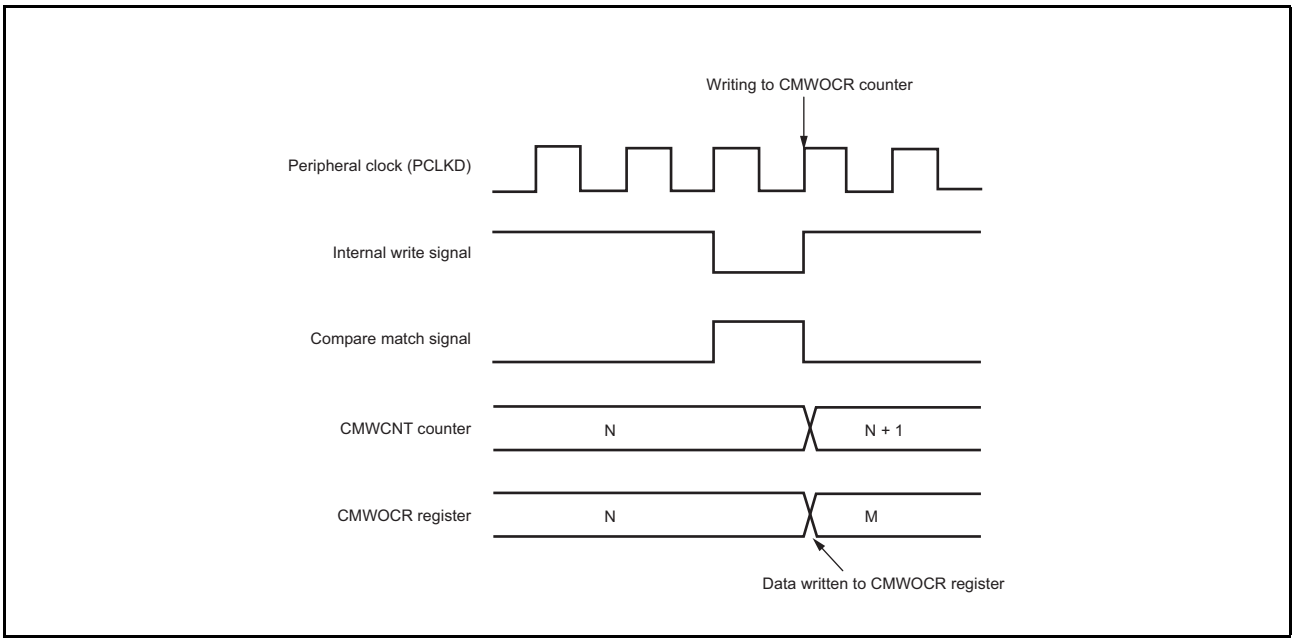


Figure 20.23 Contention between CMWOCR Register Writing and Compare Match

20.6.6 Contention between CMWCNT Counter Reading and Incrementing or Clearing

If the CMWCNT counter incrementing or clearing process occurs at the same time that the data of the CMWCNT counter is read, the value having been in the CMWCNT counter before incremented or cleared is read.
Figure 20.24 shows the timing of contention between the CMWCNT counter reading and incrementing.

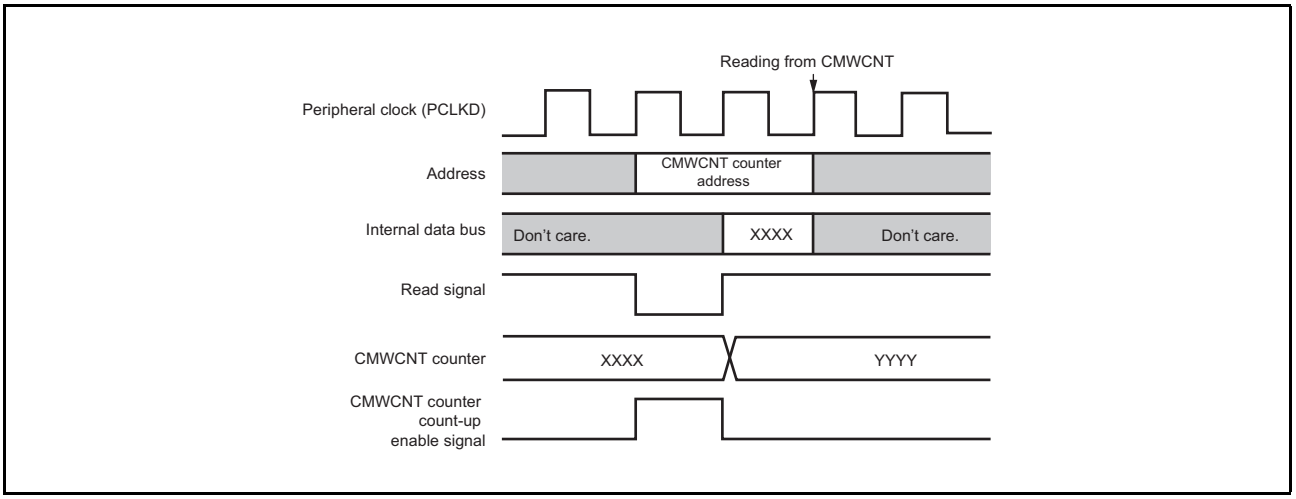


Figure 20.24 Contention between CMWCNT Counter Reading and Incrementing (When the Data Reading and Incrementing Process Occur Simultaneously)

20.6.7 Contention between CMWICR Register Reading and Input Capture

If the input capture signal is generated at the same time that the data of CMWICR register is read, the value having been in CMWICR register before updated by input capture transfer is read.

Figure 20.25 shows the timing of contention between CMWICR register reading and input capture.

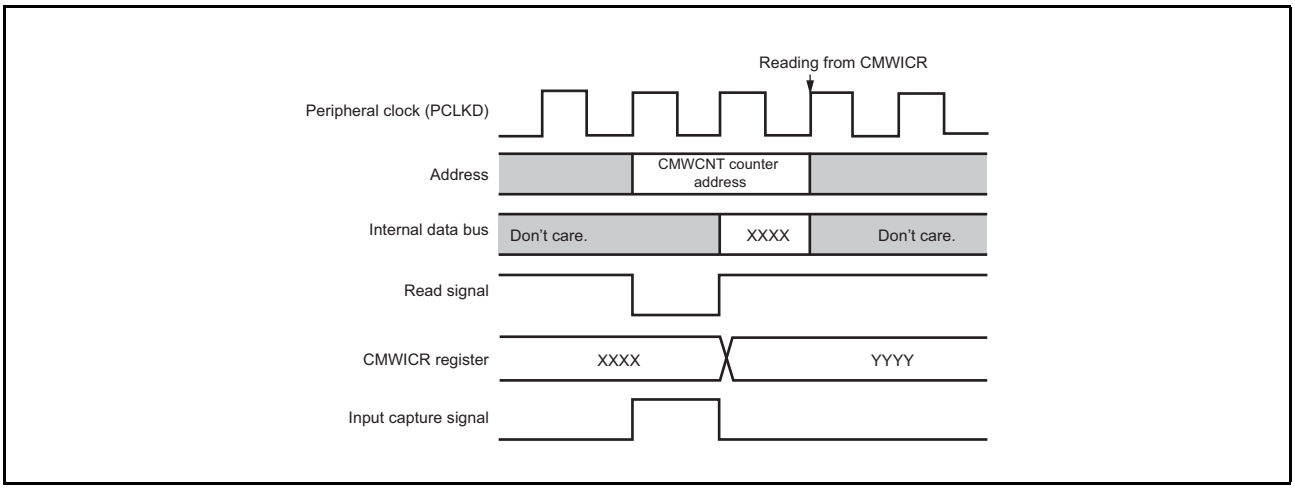


Figure 20.25 Contention between CMWICR Register Reading and Input Capture (When the Input Capture Signal and Read Signal are Generated Simultaneously)

20.6.8 Contention between Event Link Operation and Register Access

The followings are the notes on using CMTW for event link operations.
Table 20.4 summarizes contention between operations due to the event link, access to registers, and changes to the counter's state.

(1) Start Counting

When writing to the STR bit in the CMWSTR register and acceptance of the event signal are in contention, the CPU writing to the STR bit is ignored since setting of the STR bit to 1 in response to the event is given priority.

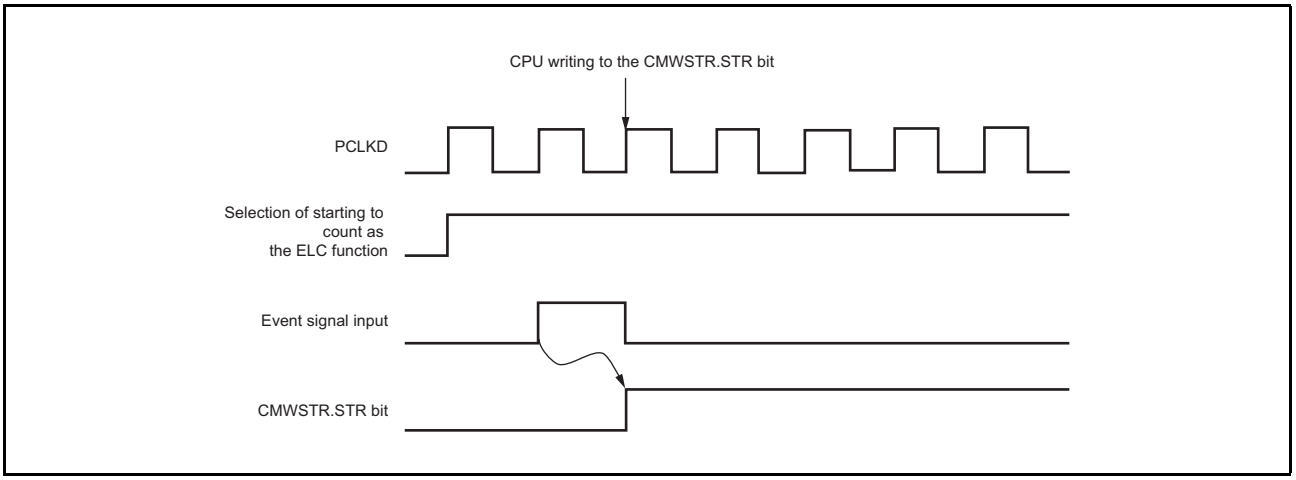


Figure 20.26 Contention between Event Acceptance and Register Access in Counting Start Operation

(2) Event Counting

When writing to CMWCNT (the timer counter) and acceptance of the event signal are in contention, the CPU writing to the CMWCNT counter is ignored since the counting operation in response to the event is given priority.

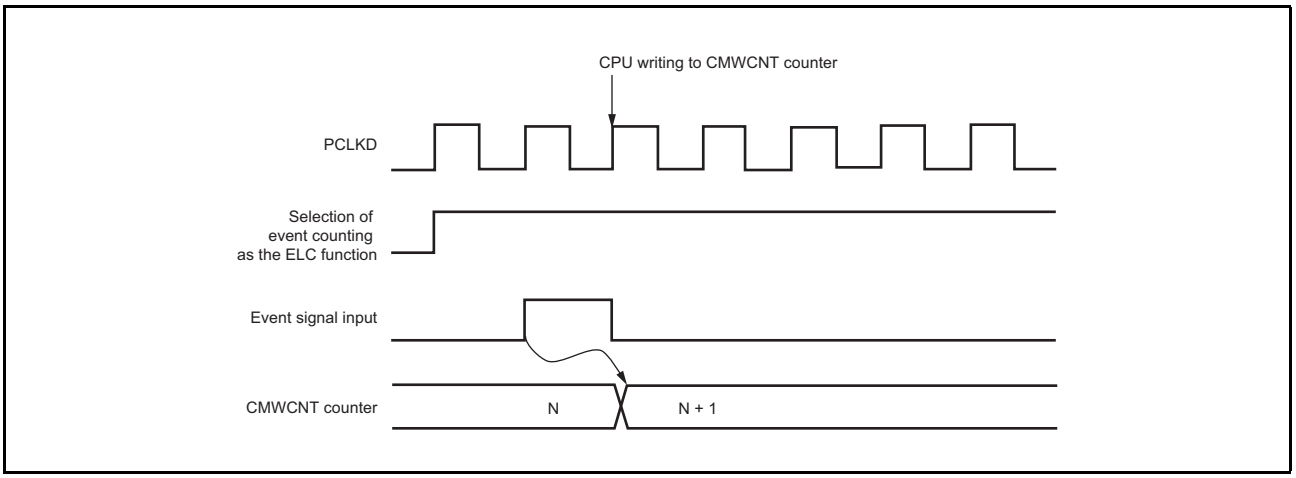


Figure 20.27 Contention between Event Acceptance and Register Access in Event Counting Operation

(3) Clear a Counter

When writing to CMWCNT (the timer counter) and acceptance of the event signal are in contention, the CPU writing to the CMWCNT counter is ignored since the counter value initialization in response to the event occurrence is given priority.

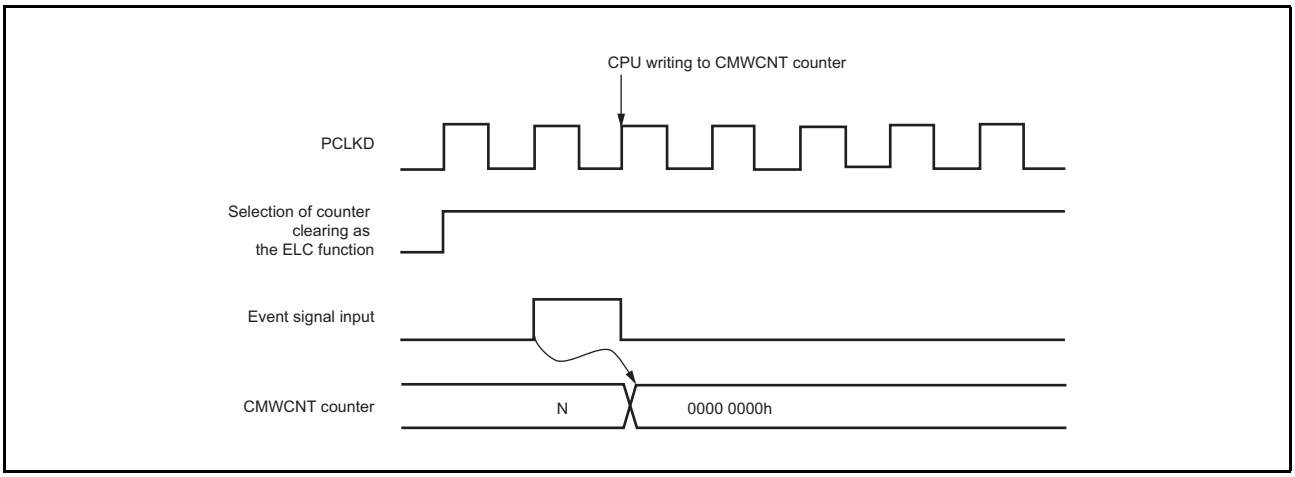


Figure 20.28 Contention between Event Acceptance and Register Access in Counting Clear Operation

Table 20.4 Summary of Contention between Operations Due to the Event Link, Access to Registers, and Changes to the Counter's State

Event Link Operation	Register Access	CMWCNT State	CMWICR0/1 State	Operation to be Performed
Counting start	Writing to CMWSTR.STR	Stopped state	—	Counting start
		Operating	—	Counting start
		Compare match	—	Counting start (CMWSTR.STR retains 1) and compare match
		Counting up	—	Counting start (CMWSTR.STR retains 1) and counting up
Event counting	Writing to CMWCNT	—	—	Event counting
	Writing to CMWCOR	Compare match	—	Compare match
Counting clear	Writing to CMWCNT	Other than compare match	—	Counting clear
	Writing to CMWCNT	Compare match	—	Compare match and counter clear
	(No access to registers)	Compare match	—	Compare match and counter clear
(No events)	Writing to CMWCNT	Compare match	—	Output of compare match interrupt request / Writing to CMWCNT
		Counting up	—	Writing to CMWCNT
	Writing to CMWCOR	Compare match	—	Compare match
	Writing to CMWOCR0	Output compare 0	—	Output compare 0
	Writing to CMWOCR1	Output compare 1	—	Output compare 1
	Reading from CMWCNT	Counting up	—	Counting up and reading of the previous value
	Reading from CMWICR0	—	Input capture 0	Input capture 0 and reading of the value before transfer
	Reading from CMWICR1	—	Input capture 1	Input capture 1 and reading of the value before transfer

21. Watchdog Timer (WDTA)

The watchdog timer (WDT) contains a 14-bit down-counter. If the counter underflows, an error notification to the error control module (ECM) is generated. The count value of the down-counter can be refreshed to the value after reset to enable counting to start again. Counter refreshing can be preformed during a period (the refresh-permitted period) you specify. If you perform refresh (register writing) outside of the permitted period, an error notification is generated and sent to the ECM. This enables detection of runaway of the program, taking the refresh interval into consideration. If underflow occurs or if refresh is performed outside of the refresh-permitted period, the WDT stops counting. Counting restarts after refresh is performed. (For details on refresh operation, see section 21.3.3, Refresh Operation.)
For details on the error control module (ECM), see section 29, Error Control Module (ECM).

21.1 Overview

WDT starts counting when refresh (register writing) is performed after reset is released.
Before starting the count, it is necessary to set the clock division ratio, the window start/end positions, and the timeout period in the WDT control register (WDTCR).
Table 21.1 lists the specifications of the WDT and Figure 21.1 shows a block diagram of the WDT.

Table 21.1 WDT Specifications

Item	Specifications
Number of internal channels	One channel
Count source	Peripheral clock (PCLKE)
Clock division ratio	Divide by 4, 64, 128, 512, 2048, or 8192
Counter operation	Counting down using a 14-bit down-counter
Conditions for starting the counter	Refresh (writing 00h and then FFh to the WDTRR register)
Conditions for stopping the counter	• Reset (the down-counter and other registers return to the values after reset) • A counter underflows or a refresh error occurs
Window function	Window start and end positions can be specified (refresh-permitted period)
Sources of sending an error notification to ECM	• Down-counter underflows • Refreshing outside the refresh-permitted period (refresh error)
Reading the counter value	The down-counter value can be read by the WDTSR register.
WDT register control	• Selection of clock division ratio after refresh operation (WDTCR.CKS[3:0] bits) • Selection of timeout period of the watchdog timer (WDTCR.TOPS[1:0] bits) • Selection of window start position of the watchdog timer (WDTCR.RPSS[1:0] bits) • Selection of window end position of the watchdog timer (WDTCR.RPES[1:0] bits)

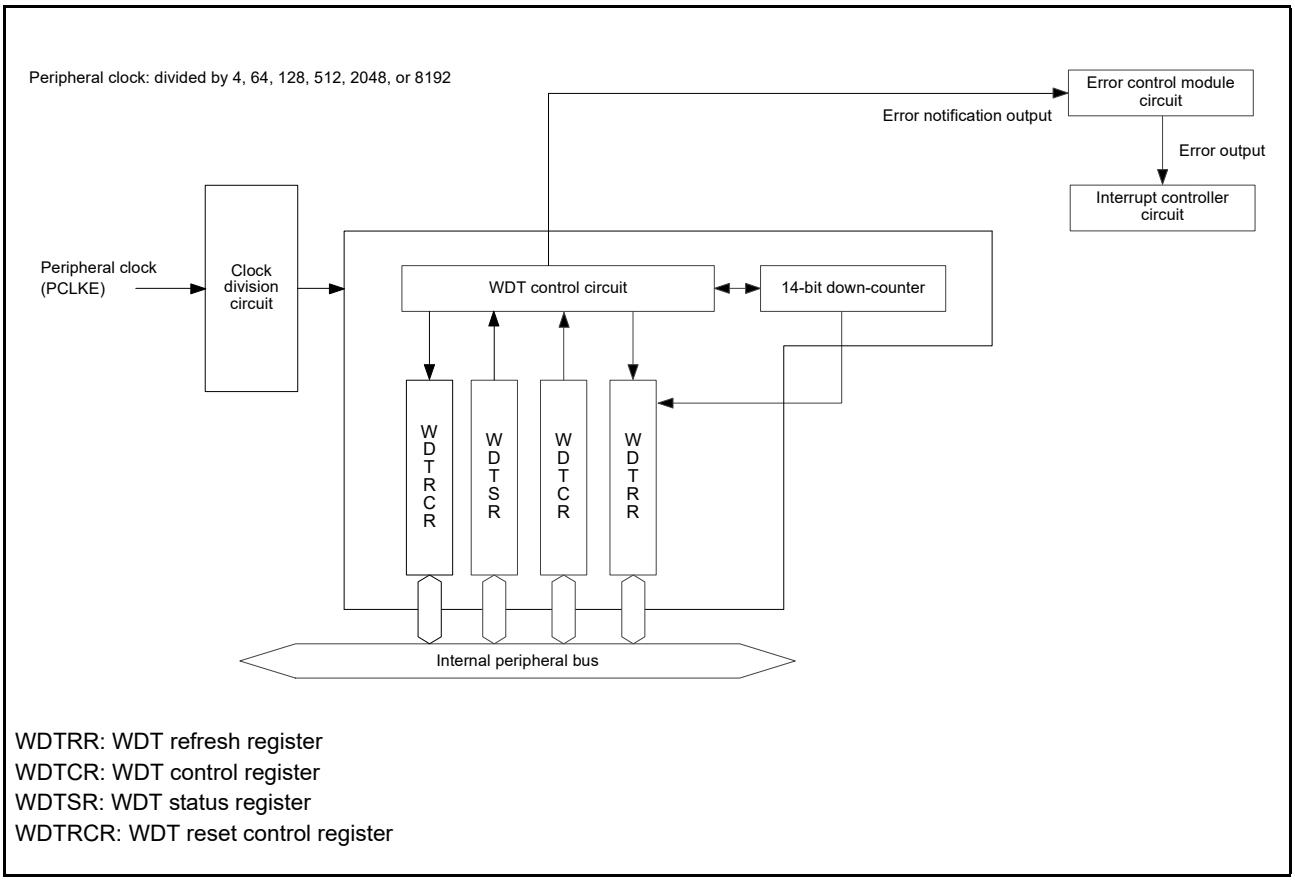


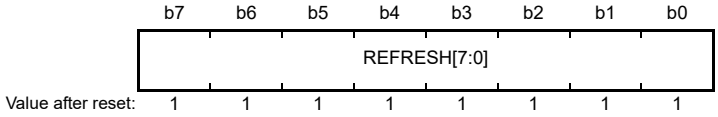
Figure 21.1 WDT Block Diagram

21.2 Register Descriptions

21.2.1 WDT Refresh Register (WDTRR)

The WDTRR register refreshes the down-counter of the WDT. To refresh the down-counter of the WDT, write 00h and then FFh (refresh operation) to the WDTRR register in the refresh-permitted period. After being refreshed, the down-counter starts counting down from the value specified with the WDTCR.TOPS[1:0] bits.
When 00h is written, the read value is 00h. When a value other than 00h is written, the read value is always FFh.
For details on the refresh operation, see section 21.3.3, Refresh Operation.

Address(es): WDT0.WDTRR A008 0600h



Bit	Symbol	Bit Name	Description	R/W
b7 to b0	REFRESH [7:0]	Refresh Register	The down-counter is refreshed by writing 00h and then writing FFh to this register.	R/W

21.2.2 WDT Control Register (WDTCR)

The WDTCR register allows you to select a timeout period before the down-counter underflows, a clock division ratio, and a window start/end position for refreshing. There are some restrictions on writing to this register. For details, see section 21.3.2, Control over Writing to the WDTCR and WDTRCR Registers.

Address(es): WDT0.WDTCR A008 0602h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	RPSS[1:0]		—	—	RPES[1:0]		CKS[3:0]				—	—	TOPS[1:0]	
Value after reset:	0	0	1	1	0	0	1	1	1	1	1	1	0	0	1	1

Bit	Symbol	Bit Name	Description	R/W
b1, b0	TOPS[1:0]	Timeout Period Selection	b1 b0 0 0: 1,024 cycles (03FFh) 0 1: 4,096 cycles (0FFFh) 1 0: 8,192 cycles (1FFFh) 1 1: 16,384 cycles (3FFFh) Values within parentheses are down-count start values.	R/W
b3, b2	—	Reserved	These bits are read as 0 and cannot be modified.	R/W
b7 to b4	CKS[3:0]	Clock Division Ratio Selection	b7 b4 0 0 0 1: PCLKE/4 0 1 0 0: PCLKE/64 1 1 1 1: PCLKE/128 0 1 1 0: PCLKE/512 0 1 1 1: PCLKE/2048 1 0 0 0: PCLKE/8192 Other settings are prohibited.	R/W
b9, b8	RPES[1:0]	Window End Position Selection	b9 b8 0 0: 75% 0 1: 50% 1 0: 25% 1 1: 0% (window end position is not specified)	R/W
b11, b10	—	Reserved	These bits are read as 0 and cannot be modified.	R/W
b13, b12	RPSS[1:0]	Window Start Position Selection	These bits allow selection of a window start position of the down-counter from among 100%, 75%, 50%, and 25% of the counting period (assuming that the count start position as 100% and underflow occurrence as 0%). The period from the window start position to the window end position is the refresh-permitted period, and any other periods are the refresh-prohibited periods. Figure 21.2 shows the relation between the settings of the RPSS[1:0] and RPES[1:0] bits and the refresh-permitted and refresh-prohibited periods. b13 b12 0 0: 25% 0 1: 50% 1 0: 75% 1 1: 100% (window start position is not specified)	R/W
b15, b14	—	Reserved	These bits are read as 0 and cannot be modified.	R/W

TOPS[1:0] Bits (Timeout Period Selection)

These bits allow you to select a timeout period (period before the down-counter underflows) from among 1024, 4096, 8192, and 16384 cycles, taking the divided clock specified by the CKS[3:0] bits as one cycle. After the down-counter is refreshed, the combination of the CKS[3:0] and TOPS[1:0] bits determines the time (the number of peripheral clock (PCLKE) cycles) before the counter underflows.

Table 21.2 lists relations between the CKS[3:0] and TOPS[1:0] bit settings, the timeout period, and the number of PCLKE cycles.

CKS[3:0] Bits (Clock Division Ratio Selection)

These bits allow you to select a division ratio from among the peripheral clocks (PCLKE) divided by 4, 64, 128, 512, 2048, and 8192. Combined with the TOPS[1:0] bit setting, a counting period between 4,096 and 134,217,728 cycles of the peripheral clock (PCLKE) can be selected for the WDT.

RPES[1:0] Bits (Window End Position Selection)

These bits allow selection of a window end position from among 75%, 50%, 25%, and 0% of the counting period. The selected window end position should be a value smaller than the window start position (window start position > window end position). If the window end position is greater than the window start position, only the window start position setting is valid.

The counter values of the window start and end positions specified by the RPES[1:0] and RPSS[1:0] bit settings vary according to the TOPS[1:0] bit setting.

Table 21.3 lists the counter values of the window start and end positions corresponding to the TOPS[1:0] bit setting.

RPSS[1:0] Bits (Window Start Position Selection)

These bits allow selection of a window start position of the down-counter from among 100%, 75%, 50%, and 25% of the counting period (assuming that the count start position as 100% and underflow occurrence as 0%). The period from the window start position to the window end position is the refresh-permitted period, and any other periods are the refresh-prohibited periods.

Figure 21.2 shows the relation between the settings of the RPSS[1:0] and RPES[1:0] bits and the refresh-permitted and refresh-prohibited periods.

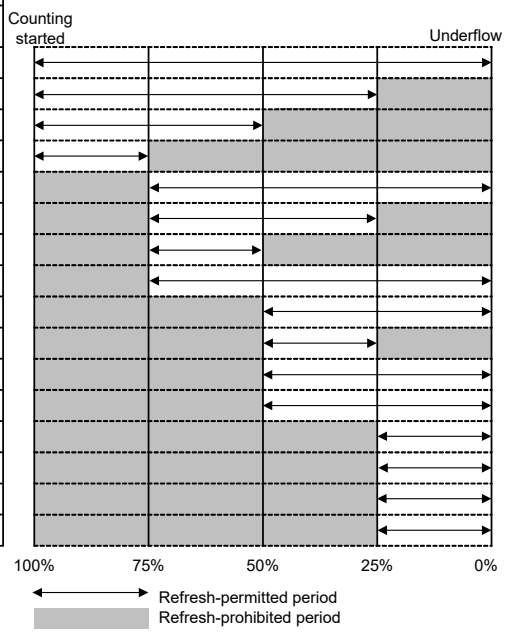
Table 21.2 Timeout Period Settings

CKS[3:0] Bits				TOPS[1:0] Bits				
b7	b6	b5	b4	b1	b0	Clock Division Ratio	Timeout Period (Number of Cycles)	No. of Peripheral Clock (PCLKE) Cycles @Count clock = PCLKE
0	0	0	1	0	0	Count clock/4	1024	4096
				0	1		4096	16384
				1	0		8192	32768
				1	1		16384	65536
0	1	0	0	0	0	Count clock/64	1024	65536
				0	1		4096	262144
				1	0		8192	524288
				1	1		16384	1048576
1	1	1	1	0	0	Count clock/128	1024	131072
				0	1		4096	524288
				1	0		8192	1048576
				1	1		16384	2097152
0	1	1	0	0	0	Count clock/512	1024	524288
				0	1		4096	2097152
				1	0		8192	4194304
				1	1		16384	8388608
0	1	1	1	0	0	Count clock/2048	1024	2097152
				0	1		4096	8388608
				1	0		8192	16777216
				1	1		16384	33554432
1	0	0	0	0	0	Count clock/8192	1024	8388608
				0	1		4096	33554432
				1	0		8192	67108864
				1	1		16384	134217728

Table 21.3 Relationship between Timeout Period and Window Start/End Counter Values

TOPS[1:0] Bits		Timeout Period		Window Start/End Counter Value			
b1	b0	Cycles	Counter Value	100%	75%	50%	25%
0	0	1024	03FFh	03FFh	02FFh	01FFh	00FFh
0	1	4096	0FFFh	0FFFh	0BFFh	07FFh	03FFh
1	0	8192	1FFFh	1FFFh	17FFh	0FFFh	07FFh
1	1	16384	3FFFh	3FFFh	2FFFh	1FFFh	0FFFh

RPSS[1:0] Bits		RPES[1:0] Bits		Window	
b13	b12	b9	b8	Start (%)	End (%)
1	1	1	1	100	0
		1	0		25
		0	1		50
		0	0		75
1	0	1	1	75	0
		1	0		25
		0	1		50
		0	0		75
0	1	1	1	50	0
		1	0		25
		0	1		50
		0	0		75
0	0	1	1	25	0
		1	0		25
		0	1		50
		0	0		75



Note: If window end setting ≥ window start setting, the window end setting is set to 0%.

Figure 21.2 RPSS[1:0] and RPES[1:0] Bit Settings and the Refresh-Permitted and Refresh-Prohibited Periods

21.2.3 WDT Status Register (WDTSR)

The WDTSR register shows the counter value of the down-counter and whether an underflow or refresh error has occurred.

Address(es): WDT0.WDTSR A008 0604h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	REFEF	UNDFF	CNTVAL[13:0]													
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b13 to b0	CNTVAL[13:0]	Down-Counter Value	Value counted by the down-counter	R
b14	UNDFF	Underflow Flag	0: No underflow occurred 1: Underflow occurred	R/(W)
b15	REFEF	Refresh Error Flag	0: No refresh error occurred 1: Refresh error occurred	R/(W)

CNTVAL[13:0] Bits (Down-Counter Value)

Read these bits to confirm the value of the down-counter, but note that the read value may differ from the actual count by a value of one count.

UNDFF Flag (Underflow Flag)

Read this bit to confirm whether an underflow has occurred in the down-counter.

The value 1 indicates that the down-counter has underflowed. The value 0 indicates that the down-counter has not underflowed.

Write 0 to the UNDFF flag to set the value to 0. Writing 1 has no effect.

REFEF Flag (Refresh Error Flag)

Read this bit to confirm whether a refresh error (performing a refresh operation during a refresh-prohibited period) has occurred.

The value 1 indicates that a refresh error has occurred. The value 0 indicates that no refresh error has occurred.

Write 0 to the REFEF flag to set the value to 0. Writing 1 has no effect.

21.2.4 WDT Reset Control Register (WDTRCR)

The WDTRCR register controls whether to send an error notification to the error control module (ECM) when underflow occurs in the down-counter of WDT.

There are some restrictions on writing to this register. For details, see section 21.3.2, Control over Writing to the WDTCR and WDTRCR Registers.

Address(es): WDT0.WDTRCR A008 0606h

	b7	b6	b5	b4	b3	b2	b1	b0
	RSTIRQS	—	—	—	—	—	—	—
Value after reset:	1	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b6 to b0	—	Reserved	These bits are read as 0 and cannot be modified.	R/W
b7	RSTIRQS	Reset Interrupt Request Selection	0: Error notification to ECM is permitted 1: Error notification to ECM is not performed	R/W

RSTIRQS Bit (Reset Interrupt Request Selection)

This bit specifies whether an error notification should be sent to the error control module (ECM) when an underflow or refresh error occurs in the down-counter.

21.3 Operation

21.3.1 Count Operation in Start Mode

Counting starts by a refresh operation of the WDT refresh register (WDTRR) when the WDT control register (WDTCR) and WDT reset control register (WDTRCR) are set.

21.3.1.1 Register Setting

After the reset state is released, set the clock division ratio, window start and end positions, and timeout period in the WDTCR register, and also set the error notification to the error control module (ECM) in the WDTRCR register. Then, the value specified by the WDTCR.TOPS[1:0] bits is set in the down-counter by a refresh operation to start counting down.
Thereafter, the value in the counter is re-set at each refresh operation and count-down continues if the program runs normally and the counter is refreshed in the refresh-permitted period. The WDT does not output an error notification to ECM as long as the count-down continues. However, if the down-counter underflows because the down-counter cannot be refreshed due to a program runaway, or if a refresh error occurs because the counter is refreshed outside of the refresh-permitted period, the WDT outputs an error notification to the ECM.

Figure 21.3 shows an example of operation under the following conditions.

- Reset interrupt request bit (RSTIRQS): 0b (Error notification to ECM is permitted.)
- Window start position selection bits (RPSS[1:0]): 10b (75%)
- Window end position selection bits (RPES[1:0]): 10b (25%)

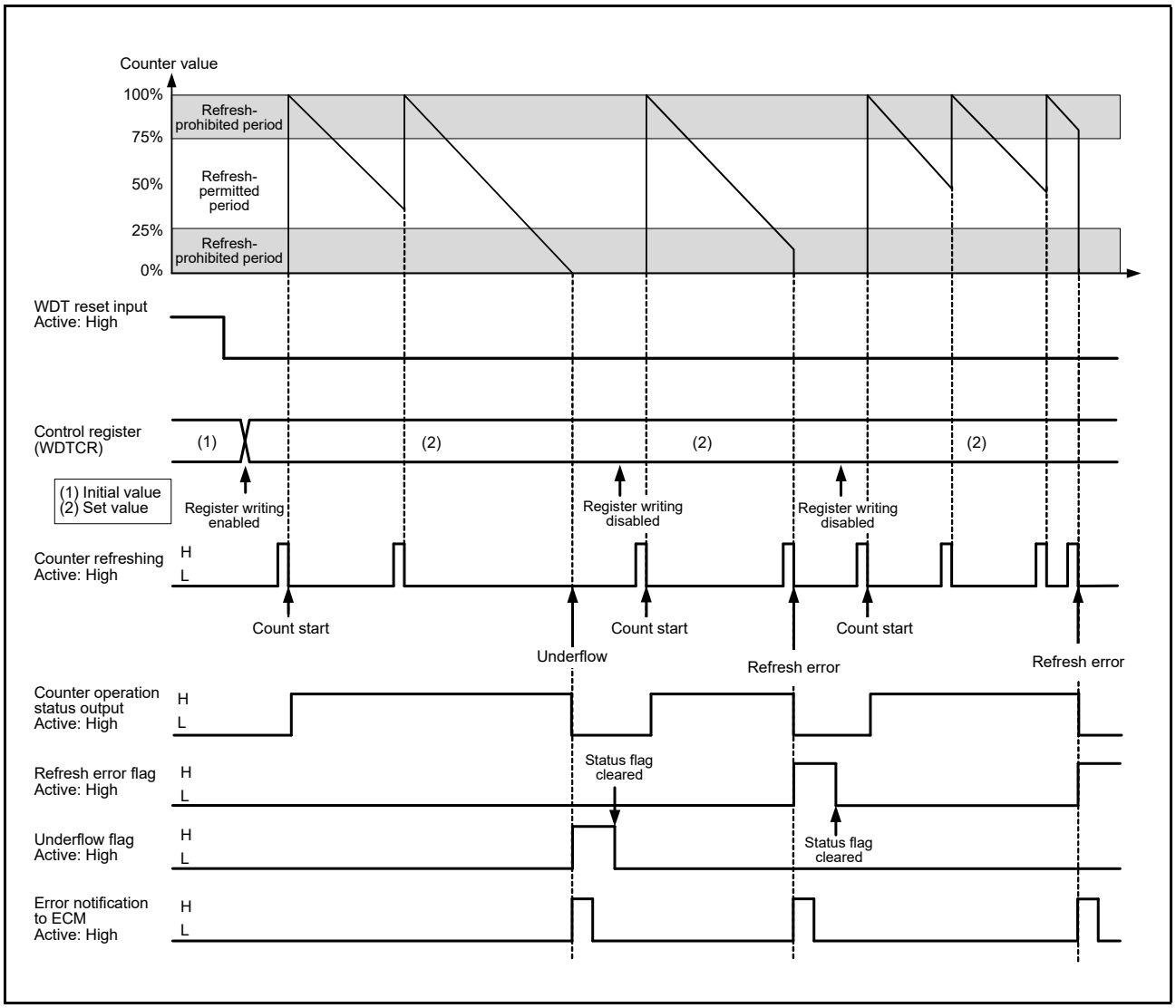


Figure 21.3 Operation Example in Register Start Mode

21.3.2 Control over Writing to the WDTCR and WDTRCR Registers

Writing to the WDT control register (WDTCR) is possible only once between the release from the reset state and the first refresh operation.

After a refresh operation (counting starts) or writing to the WDTCR register, the protection signal in the WDT becomes 1 and WDTCR is protected from subsequent attempts of writing.

Writing to the WDT reset control register (WDTRCR) is also controlled similarly.

This protection is released by the reset source for the WDT. Any other reset sources cannot release the protection.

Figure 21.4 shows control waveforms produced in response to writing to the WDTCR.

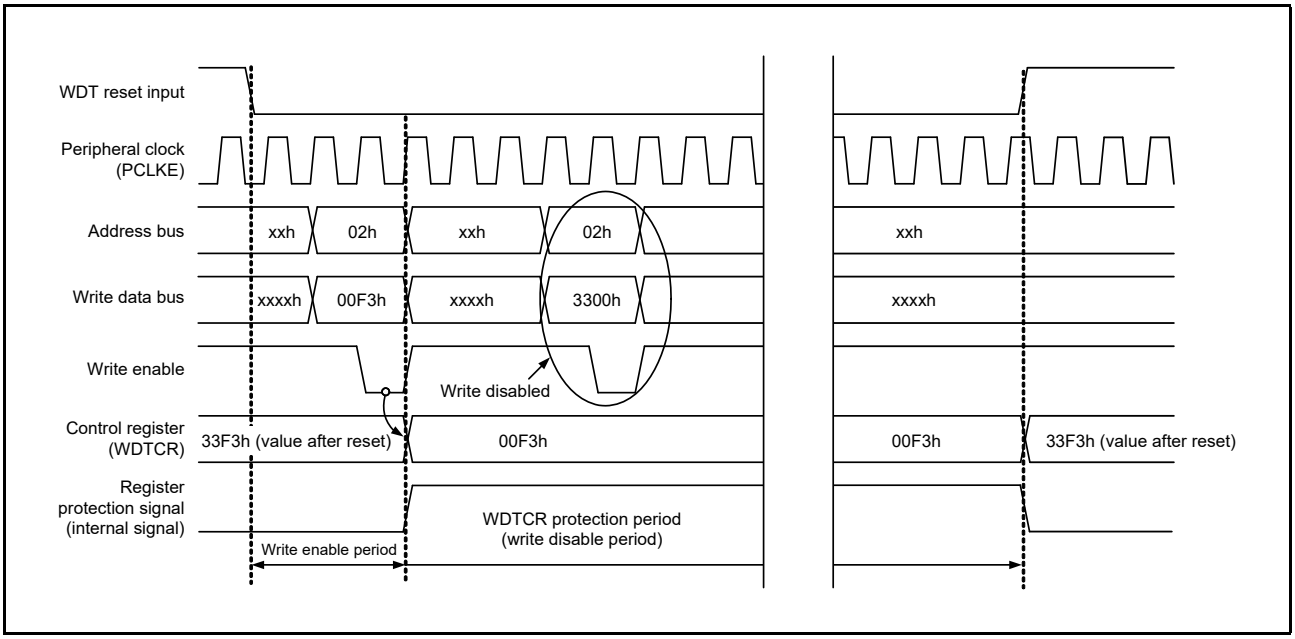


Figure 21.4 Control Waveforms Produced in Response to Writing to the WDTCR Register

21.3.3 Refresh Operation

To refresh the down-counter and to start the down-counter operation (count start due to refreshing), write the values 00h and then FFh to the WDT refresh register (WDTRR). If a value other than FFh is written after 00h, the down-counter is not refreshed. To perform refreshing after such invalid writing, write 00h and then FFh again to the WDTRR register.
If 00h is written twice in succession, writing FFh after the second 00h refreshes the down-counter because the 00h→FFh condition is satisfied. The writing sequence of 00h (n-1th)→00h (nth)→FFh also satisfies the refreshing condition and thus refreshes the down-counter.
Even if a register other than WDTRR is accessed or WDTRR is read between writing 00h and writing FFh to WDTRR, correct refreshing will be done.

[Sample sequences of writing that are valid for refreshing the counter]

- 00h→FFh
- 00h (n–1-th time)→00h (nth time)→FFh
- 00h→access to another register or read from WDTRR→FFh

[Sample sequences of writing that are not valid for refreshing the counter]

- 23h (a value other than 00h)→FFh
- 00h→54h (a value other than FFh)
- 00h→AAh (values other than 00h and FFh)→FFh

If FFh is written to the WDTRR register during the refresh-permitted period after 00h is written outside of the period, write operation is acknowledged and down-counter is refreshed. (Whether writing is made within the refresh-permitted period is determined by when FFh is written.)

After FFh is written to the WDT refresh register (WDTRR), refreshing the down-counter requires up to four counting cycles. (The number of peripheral clock (PCLKE) cycles in a single counting cycle differs depending on the setting of the clock division ratio selection bits (WDTCR.CKS[3:0]).) Therefore, writing FFh to the WDTRR register should be completed within four count-cycles before the refresh-permitted period end position or before the down-counter underflows. Confirm the down-counter value with the down-counter value bits (WDTSR.CNTVAL[13:0]).

[Sample of refresh operation timing]

- When the window start position is 1FFFh, if 00h is written to the WDTRR register before 1FFFh (for example, 2002h), the down-counter can be refreshed by writing FFh to the WDTRR register after the value of WDTSR.CNTVAL[13:0] becomes 1FFFh.
- When the window end position is 1FFFh, if the value of WDTSR.CNTVAL[13:0] is 2003h (four counts before 1FFFh) or more immediately after 00h and FFh are written to the WDTRR register, the down-counter will be refreshed.
- If the refresh-permitted period ranges down to 0000h, refreshing is possible until just before underflow. In this case, if the value of WDTSR.CNTVAL[13:0] is 0003h (four counts before underflow) or more immediately after 00h and FFh are written to the WDTRR register, underflow does not occur and the down-counter is refreshed.

Figure 21.5 shows the WDT refresh-operation waveforms when the clock division ratio = PCLKE/64.

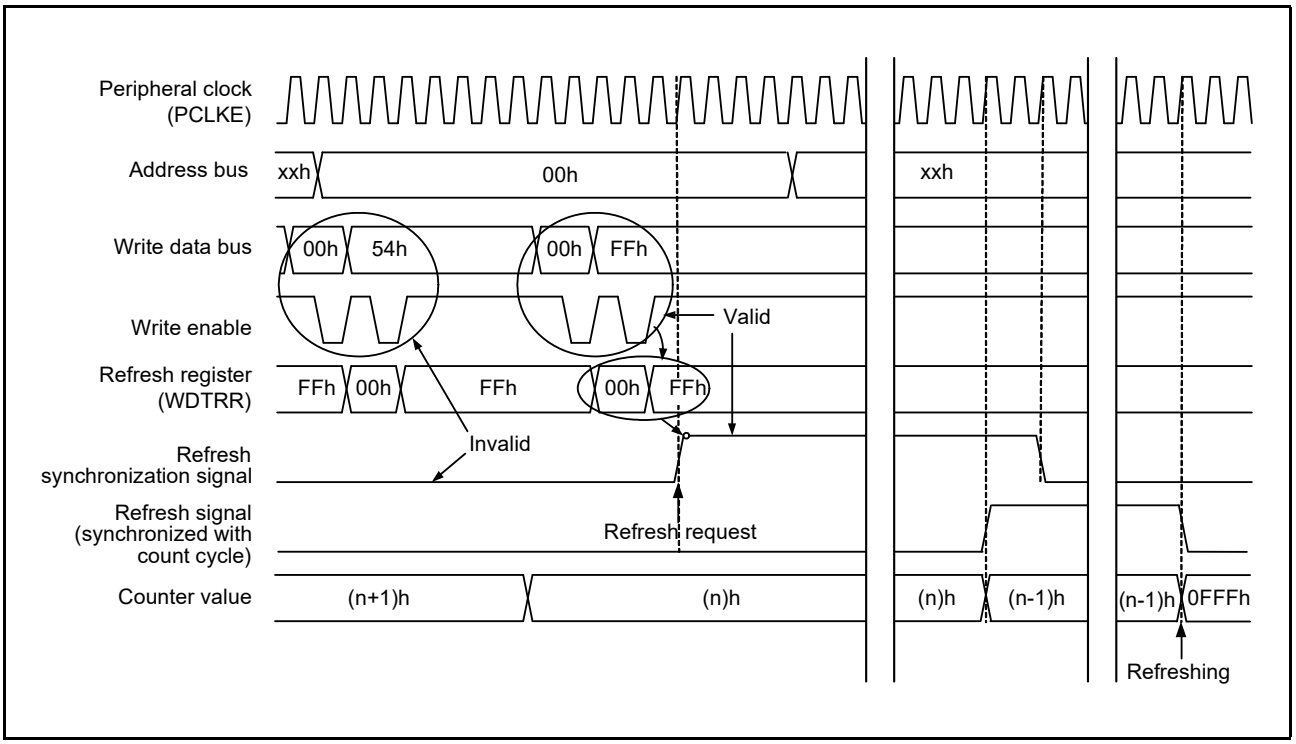


Figure 21.5 WDT Refresh Operation Waveforms (WDTCR.CKS[3:0] = 0100b, WDTCR.TOPS[1:0] = 01b)

21.3.4 Status Flag

The refresh error flag (WDTSR.REFEF) and the underflow flag (WDTSR.UNDFF) retain the error causes after error notifications were output to the error control module (ECM) of WDT.
Occurrence state of error notifications to ECM can be confirmed by reading the WDTSR.REFEF flag or the WDTSR.UNDFF flag after reset is released or upon occurrence of an error notification to ECM.
To clear these flags, write 0. Writing 1 is ignored.
If these flags are not cleared, it will give no effect to the operation. Upon occurrence of the next error notification to ECM, the previous error notification is automatically cleared and a new notification to ECM is written.

21.3.5 Error Notification to the Error Control Module (ECM)

When the reset interrupt selection bit (WDTRCR.RSTIRQS) is set to 0, underflow or refresh error of the down-counter causes an error notification to ECM to be generated during one count cycle.

21.3.6 Reading the Down-Counter Value

The WDT stores the counter value in the down-counter value bits (WDTSR.CNTVAL[13:0]) of the WDT status register. Thus, the counter value can be checked through the WDTSR.CNTVAL[13:0] bits.

Reading requires up to four peripheral clock (PCLKE) cycles. Therefore, the read value may differ from the actual down-counter value by one count.

Figure 21.6 shows the processing for reading the WDT down-counter value when the clock division ratio = PCLKE/64.

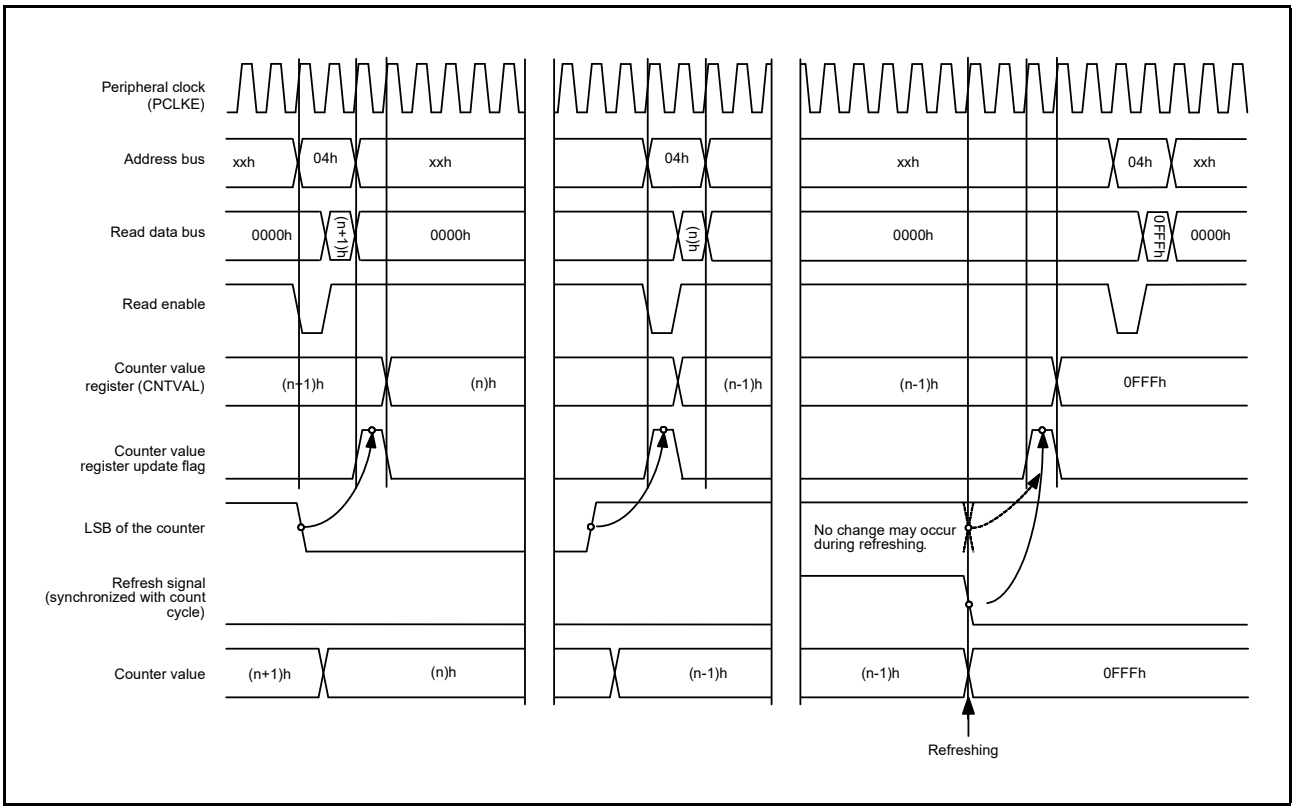


Figure 21.6 Processing for Reading WDT Down-Counter Value (WDTCR.CKS[3:0] = 0100b, WDTCR.TOPS[1:0] = 01b)

21.4 Low-Power Consumption Control

21.4.1 Watchdog Timer Operations in Low-Power Consumption Mode Transition

Clock supply to the WDT can be controlled during transition to the standby mode of Cortex-R4 while the down-counter of WDT is operating.

Table 21.4 lists the WDT operations during transition to the low-power consumption mode.

Table 21.4 WDT Operations during Transition to Low-Power Consumption Mode <in Low-Power Consumption Mode Transition>

Low-Power Consumption Mode	WDT0 Clock Supply	WDT0 Operation
Cortex-R4 standby	√	√

√: Operating

22. Independent Watchdog Timer (IWDTa)

The independent watchdog timer (IWDT) contains a 14-bit down-counter. If the counter underflows, an error notification to the error control module (ECM) is generated. The count value of the down-counter can be refreshed to the value after reset to enable counting to start again. Counter refreshing can be preformed during a period (the refresh-permitted period) you specify. If you perform refresh (register writing) outside of the permitted period, an error notification is generated and sent to the ECM. This enables detection of runaway of the program, taking the refresh interval into consideration. If underflow occurs or if refresh is performed outside of the refresh-permitted period, the IWDT stops counting. Counting restarts after refresh is performed. (For details on refresh operation, see section 22.3.3, Refresh Operation.)

For details on the error control module (ECM), see section 29, Error Control Module (ECM).

22.1 Overview

The IWDT starts counting when refresh (register writing) is performed after reset is released.

Before starting the count, it is necessary to set the clock division ratio, the window start/end positions, and the timeout period in the IWDT control register (IWDTCR).

Table 22.1 lists the specifications of the IWDT.

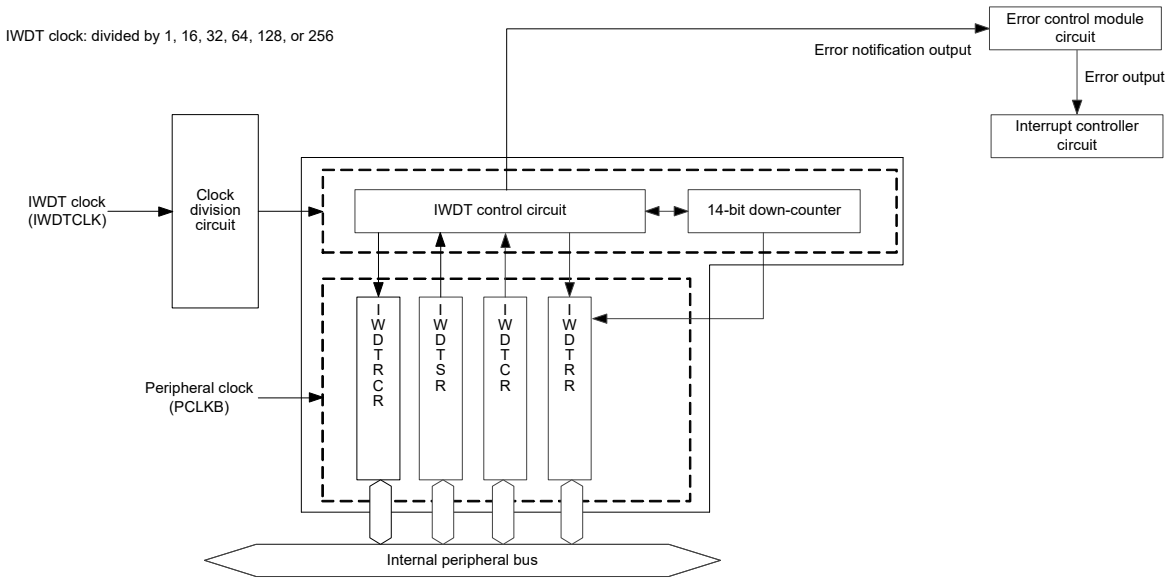
Table 22.1 IWDT Specifications

Item	Description
Count source	IWDT clock (IWDTCLK)
Clock division ratio	Division by 1, 16, 32, 64, 128, or 256
IWDT clock (IWDTCLK) oscillation enable	IWDT clock oscillation starts by a refresh operation.
Counter operation	Counting down using a 14-bit down-counter
Conditions for starting the counter	Counting starts by refreshing the counter (writing 00h and then FFh to the IWDTRR register).
Conditions for stopping the counter	• Reset (the down-counter and other registers return to the values after reset) • A counter underflows or a refresh error is generated
Window function	Window start and end positions can be specified (refresh-permitted period)
Sources for the output of error notification to ECM	• When the down-counter underflows • When refreshing is done outside the refresh-permitted period (refresh error)
Reading the counter value	The down-counter value can be read by reading the IWDTSR register.
IWDT register control	• Selecting the clock frequency division ratio after refreshing (IWDTCR.CKS[3:0] bits) • Selecting the timeout period of the independent watchdog timer (IWDTCR.TOPS[1:0] bits) • Selecting the window start position in the independent watchdog timer (IWDTCR.RPSS[1:0] bits) • Selecting the window end position in the independent watchdog timer (IWDTCR.RPES[1:0] bits)

For operation to continue even if the peripheral clock (PCLKB) stops unexpectedly, counting by the IWDT is driven by two clock signals, the peripheral clock (PCLKB) and the IWDT clock (IWDTCLK). The peripheral clock (PCLKB) provides the timing for the bus interface and registers, and the IWDT clock (IWDTCLK) provides the timing for the 14-bit down-counter and control circuits.

Signals between the peripheral clock (PCLKB) operation blocks and the IWDT clock (IWDTCLK) operation blocks are connected via a synchronization circuit.

Figure 22.1 is a block diagram of the IWDT.



IWDTRR: IWDT refresh register
IWDTCR: IWDT control register
IWDTSR: IWDT status register
IWDTRCR: IWDT reset control register

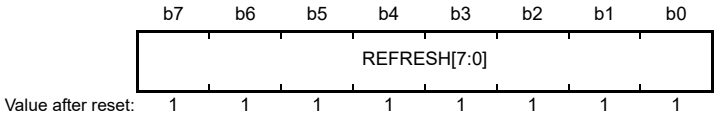
Figure 22.1 IWDT Block Diagram

22.2 Register Descriptions

22.2.1 IWDT Refresh Register (IWDTRR)

The IWDTRR register refreshes the down-counter of the IWDT.
To refresh the down-counter of the IWDT, write 00h and then FFh (refresh operation) to the IWDTRR register in the refresh-permitted period. After being refreshed, the down-counter starts counting down from the value specified with the IWDTCR.TOPS[1:0] bits.
When 00h is written, the read value is 00h. When a value other than 00h is written, the read value is always FFh. For details on the refresh operation, see section 22.3.3, Refresh Operation.

Address(es): A008 0700h



Bit	Symbol	Bit Name	Description	R/W
b7 to b0	REFRESH [7:0]	Refresh Register	The down-counter is refreshed by writing 00h and then writing FFh to this register.	R/W

22.2.2 IWDT Control Register (IWDTCR)

The IWDTCR register allows you to select a timeout period before the down-counter underflows, a clock division ratio, and a window start/end position for refreshing.

There are some restrictions on writing to this register. For details, see section 22.3.2, Control Over Writing to the IWDTCR and IWDTRCR Registers.

Address(es): A008 0702h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	RPSS[1:0]		—	—	RPES[1:0]		CKS[3:0]				—	—	TOPS[1:0]	
Value after reset:	0	0	1	1	0	0	1	1	1	1	1	1	0	0	1	1

Bit	Symbol	Bit Name	Description	R/W
b1, b0	TOPS[1:0]	Timeout Period Selection	b1 b0 0 0: 1,024 cycles (03FFh) 0 1: 4,096 cycles (0FFFh) 1 0: 8,192 cycles (1FFFh) 1 1: 16,384 cycles (3FFFh) Values within parentheses are down-count start values.	R/W
b3, b2	—	Reserved	These bits are read as 0. Writing to these bits has no effect.	R/W
b7 to b4	CKS[3:0]	Clock Division Ratio Selection	b7 b4 0 0 0 0: IWDTCLK 0 0 1 0: IWDTCLK/16 0 0 1 1: IWDTCLK/32 0 1 0 0: IWDTCLK/64 1 1 1 1: IWDTCLK/128 0 1 0 1: IWDTCLK/256 Other settings are prohibited.	R/W
b9, b8	RPES[1:0]	Window End Position Selection	b9 b8 0 0: 75% 0 1: 50% 1 0: 25% 1 1: 0% (window end position is not specified.)	R/W
b11, b10	—	Reserved	These bits are read as 0. Writing to these bits has no effect.	R/W
b13, b12	RPSS[1:0]	Window Start Position Selection	b13 b12 0 0: 25% 0 1: 50% 1 0: 75% 1 1: 100% (window start position is not specified.)	R/W
b15, b14	—	Reserved	These bits are read as 0. Writing to these bits has no effect.	R/W

TOPS[1:0] Bits (Timeout Period Selection)

These bits allow you to select a timeout period (a period before the down-counter underflows) from among 1024, 4096, 8192, and 16384 cycles, taking the divided clock specified by the CKS[3:0] bits as one cycle. After the down-counter is refreshed, the combination of the CKS[3:0] and TOPS[1:0] bits determines the time (the number of IWDT clock (IWDTCLK) cycles) before the counter underflows.

Table 22.2 lists relations between the CKS[3:0] and TOPS[1:0] bit settings, the timeout period, and the number of IWDT clock (IWDTCLK) cycles.

CKS[3:0] Bits (Clock Division Ratio Selection)

These bits allow you to select the IWDT clock (IWDTCLK) division ratio from among division by 1, 16, 32, 64, 128, and 256. Combined with the TOPS[1:0] bit setting, a counting period between 1024 and 4194304 cycles of the IWDT clock (IWDTCLK) can be selected for the IWDT. The down-counter value may not be read correctly depending on the relation between the lowest peripheral clock (PCLKB) frequency and the highest IWDT clock (IWDTCLK) frequency.

RPES[1:0] Bits (Window End Position Selection)

These bits allow selection of a window end position from among 75%, 50%, 25%, and 0% of the counting period. The selected window end position should be a value smaller than the window start position (window start position > window end position). If the window end position is greater than the window start position, only the window start position setting is valid.

The counter values of the window start and end positions specified by the RPES[1:0] and RPSS[1:0] bit settings vary according to the TOPS[1:0] bit setting.

Table 22.3 lists the counter values of the window start and end positions corresponding to the TOPS[1:0] bit setting.

RPSS[1:0] Bits (Window Start Position Selection)

These bits allow selection of a window start position of the down-counter from among 100%, 75%, 50%, and 25% of the counting period (assuming that the count start position as 100% and underflow occurrence as 0%). The period from the window start position to the window end position is the refresh-permitted period, and any other periods are the refresh-prohibited periods.

Figure 22.2 shows the relation between the settings of the RPSS[1:0] and RPES[1:0] bits and the refresh-permitted and refresh-prohibited periods.

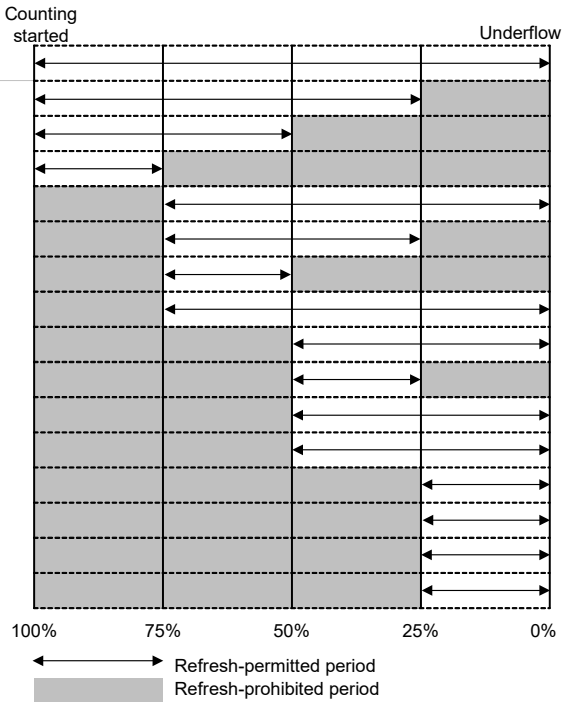
Table 22.2 Settings and Timeout Periods

CKS[3:0] Bits				TOPS[1:0] Bits		Clock Division Ratio	Timeout Period (Number of Cycles)	No. of IWDT Clock (IWDTCLK) Cycles
b7	b6	b5	b4	b1	b0			
0	0	0	0	0	0	IWDTCLK	1024	1024
				0	1		4096	4096
				1	0		8192	8192
				1	1		16384	16384
0	0	1	0	0	0	IWDTCLK/16	1024	16384
				0	1		4096	65536
				1	0		8192	131072
				1	1		16384	262144
0	0	1	1	0	0	IWDTCLK/32	1024	32768
				0	1		4096	131072
				1	0		8192	262144
				1	1		16384	524288
0	1	0	0	0	0	IWDTCLK/64	1024	65536
				0	1		4096	262144
				1	0		8192	524288
				1	1		16384	1048576
1	1	1	1	0	0	IWDTCLK/128	1024	131072
				0	1		4096	524288
				1	0		8192	1048576
				1	1		16384	2097152
0	1	0	1	0	0	IWDTCLK/256	1024	262144
				0	1		4096	1048576
				1	0		8192	2097152
				1	1		16384	4194304

Table 22.3 Relationship between Timeout Period and Window Start and End Counter Values

TOPS[1:0] Bits		Timeout Period		Window Start and End Counter Value			
b1	b0	Cycles	Counter Value	100%	75%	50%	25%
0	0	1024	03FFh	03FFh	02FFh	01FFh	00FFh
0	1	4096	0FFFh	0FFFh	0BFFh	07FFh	03FFh
1	0	8192	1FFFh	1FFFh	17FFh	0FFFh	07FFh
1	1	16384	3FFFh	3FFFh	2FFFh	1FFFh	0FFFh

RPSS[1:0] Bits		RPES[1:0] Bits		Window	
b13	b12	b9	b8	Start (%)	End (%)
1	1	1	1	100	0
		1	0		25
		0	1		50
		0	0		75
1	0	1	1	75	0
		1	0		25
		0	1		50
		0	0		75
0	1	1	1	50	0
		1	0		25
		0	1		50
		0	0		75
0	0	1	1	25	0
		1	0		25
		0	1		50
		0	0		75



Note: If window end setting ≥ window start setting, the window end setting is set to 0%.

Figure 22.2 RPSS[1:0] and RPES[1:0] Bit Settings and the Refresh-Permitted Period

22.2.3 IWDT Status Register (IWDTSR)

The IWDTSR register shows the counter value of the down-counter and whether an underflow or refresh error has occurred.

Address(es): A008 0704h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	REFEF	UNDFF	CNTVAL[13:0]													
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b13 to b0	CNTVAL[13:0]	Counter Value	Value counted by the counter	R
b14	UNDFF	Underflow Flag	0: No underflow occurred 1: Underflow occurred	R/(W)
b15	REFEF	Refresh Error Flag	0: No refresh error occurred 1: Refresh error occurred	R/(W)

CNTVAL[13:0] Bits (Counter Value)

Read these bits to confirm the value of the down-counter, but note that the read value may differ from the actual count by a value of one count.

UNDFF Flag (Underflow Flag)

Read this bit to confirm whether an underflow has occurred in the down-counter.

The value 1 indicates that the down-counter has underflowed. The value 0 indicates that the down-counter has not underflowed.

Write 0 to the UNDFF flag to set the value to 0. Writing 1 has no effect.

REFEF Flag (Refresh Error Flag)

Read this bit to confirm whether a refresh error (performing a refresh operation during a refresh-prohibited period) has occurred.

The value 1 indicates that a refresh error has occurred. The value 0 indicates that no refresh error has occurred.

Write 0 to the REFEF flag to set the value to 0. Writing 1 has no effect.

22.2.4 IWDT Reset Control Register (IWDTRCR)

The IWDTRCR register controls whether to send an error notification to the error control module (ECM) when underflow occurs in the down-counter of IWDT.

There are some restrictions on writing to this register. For details, see section 22.3.2, Control Over Writing to the IWDTCR and IWDTRCR Registers.

Address(es): A008 0706h

	b7	b6	b5	b4	b3	b2	b1	b0
	RSTIRQS	—	—	—	—	—	—	—
Value after reset:	1	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b6 to b0	—	Reserved	These bits are read as 0. Writing to these bits has no effect.	R/W
b7	RSTIRQS	Reset Interrupt Request Selection	0: Error notification to ECM is permitted. 1: Error notification to ECM is not performed.	R/W

RSTIRQS Bit (Reset Interrupt Request Selection)

This bit specifies whether an error notification should be sent to the error control module (ECM) when an underflow or refresh error occurs in the down-counter.

22.3 Operation

22.3.1 Count Operation in Start Mode

Counting starts by a refresh operation of the IWDT refresh register (IWDTRR) if the IWDT control register (IWDTCR) and IWDT reset control register (IWDTRCR) are set.

22.3.1.1 Register Setting

After the reset state is released, set the clock division ratio, window start and end positions, and timeout period in the IWDTCR register, and also set whether to send an error notification to the error control module (ECM) in the IWDTRCR register. Then, the value specified by the timeout period selection bits (IWDTCR.TOPS[1:0]) is set in the down-counter by a refresh operation to start counting down.
Thereafter, the value in the counter is re-set at each refresh operation and count-down continues if the program runs normally and the counter is refreshed in the refresh-permitted period. The IWDT does not output an error notification to ECM as long as the count-down continues.
However, if the down-counter underflows because the down-counter cannot be refreshed due to a program runaway, or if a refresh error occurs because the counter is refreshed outside of the refresh-permitted period, the IWDT outputs an error notification to the ECM.

Figure 22.3 shows an example of operation under the following conditions.

- Reset interrupt request bit (IWDTCR.RSTIRQS): 0b (Error notification to ECM is permitted.)
- Window start position selection bits (IWDTCR.RPSS[1:0]): 10b (75%)
- Window end position selection bits (IWDTCR.RPES[1:0]): 10b (25%)

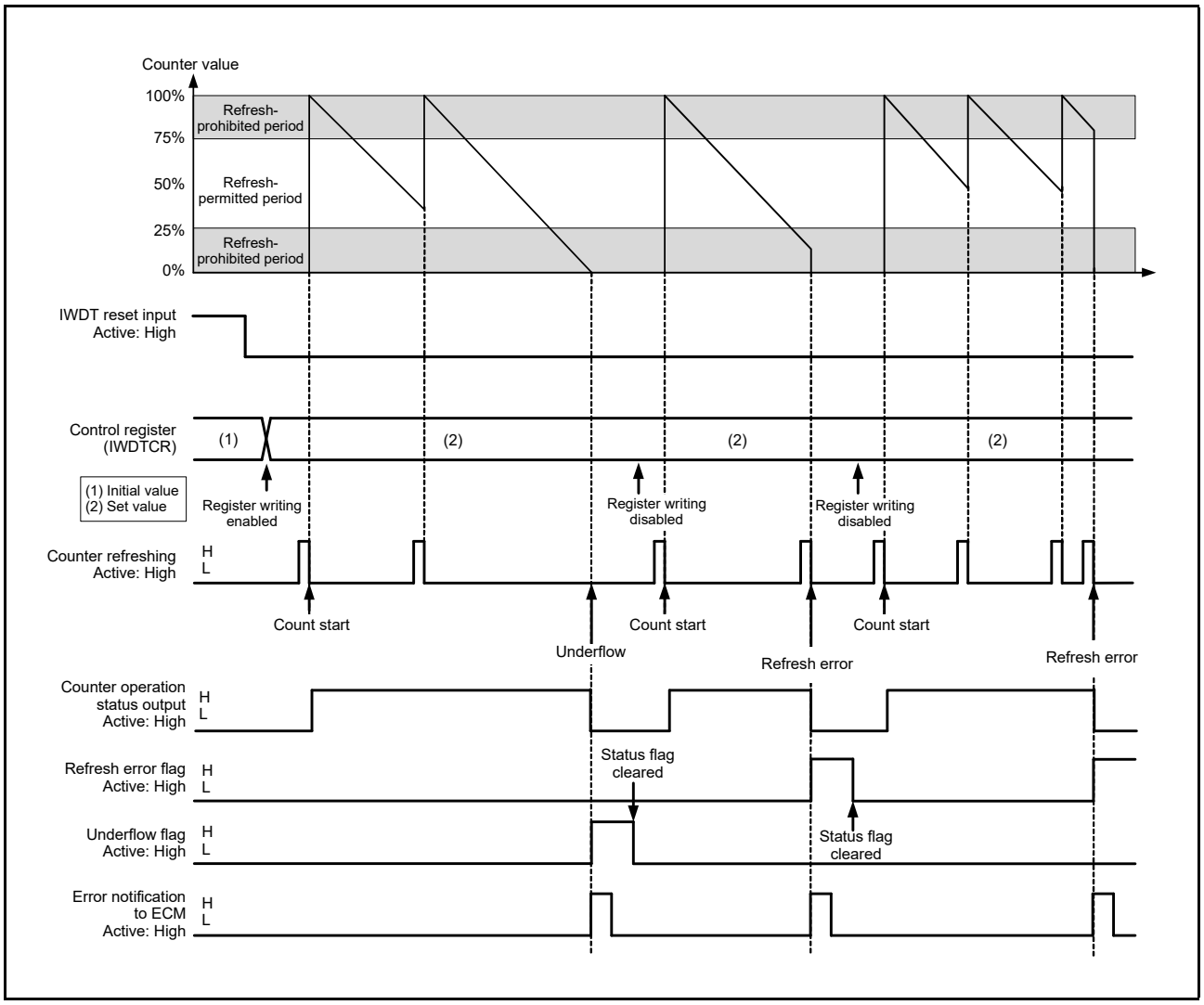


Figure 22.3 Operation Example in Register Start Mode

22.3.2 Control Over Writing to the IWDTCR and IWDTRCR Registers

Writing to the IWDT control register (IWDTCR) is possible only once between the release from the reset state and the first refresh operation.

After a refresh operation (counting starts) or writing to the IWDTCR register, the protection signal in the IWDT becomes 1 to protect IWDTCR from subsequent attempts of writing.

Writing to the IWDT reset control register (IWDTRCR) is also controlled similarly.

This protection is released by the reset source for the IWDT. With other reset sources, the protection is not released.

Figure 22.4 shows control waveforms produced in response to writing to the IWDTCR register.

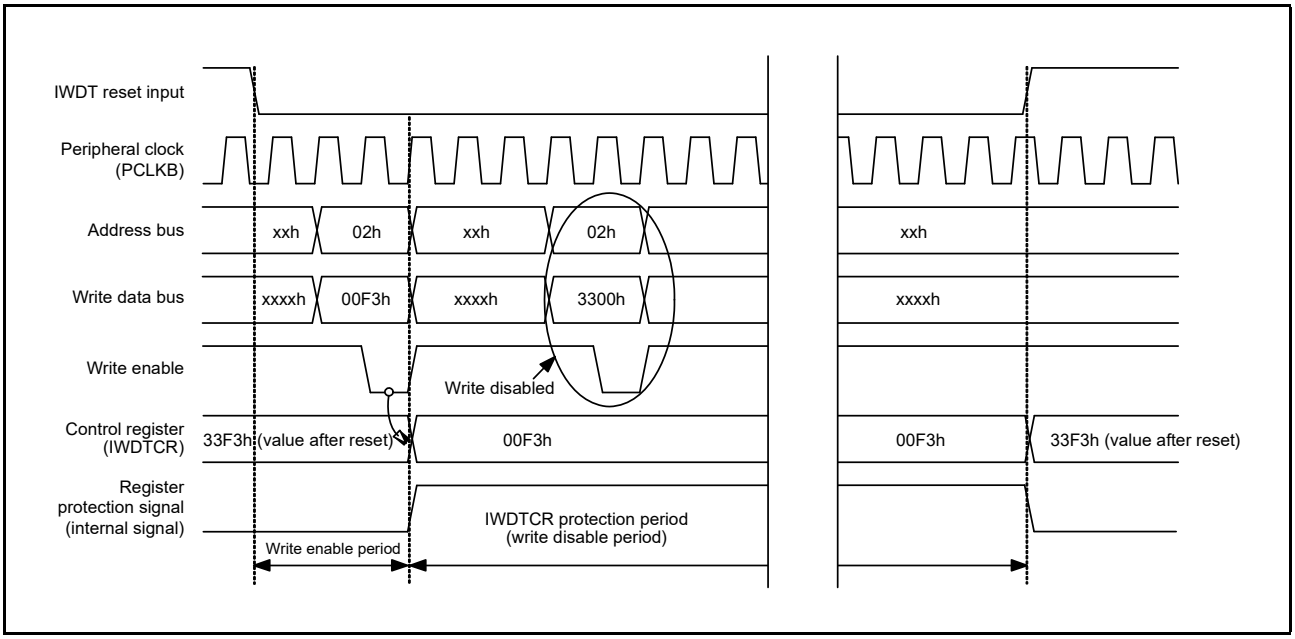


Figure 22.4 Control Waveforms Produced in Response to Writing to the IWDTCR Register

22.3.3 Refresh Operation

To refresh the counter and to start the counter operation (counting is started by refreshing), write the values 00h and then FFh to the IWDT refresh register (IWDTRR). If a value other than FFh is written after 00h, the counter is not refreshed. To perform refreshing after such invalid writing, write 00h and FFh again to the IWDT refresh register (IWDTRR).
When writing is done in the order of 00h (first time) → 00h (second time), and if FFh is written after that, the writing order 00h → FFh is satisfied; writing 00h (n–1-th time) → 00h (nth time) → FFh is valid and correct refreshing will be done. Moreover, even if a register other than IWDTRR is accessed or IWDTRR is read between writing 00h and writing FFh to IWDTRR, correct refreshing will be done.

[Sample sequences of writing that are valid for refreshing the counter]

- 00h → FFh
- 00h (n–1-th time) → 00h (nth time) → FFh
- 00h → access to another register or read from IWDTRR → FFh

[Sample sequences of writing that are not valid for refreshing the counter]

- 23h (a value other than 00h) → FFh
- 00h → 54h (a value other than FFh)
- 00h → AAh (values other than 00h and FFh) → FFh

Even when 00h is written to IWDTRR outside the refresh-permitted period, if FFh is written to IWDTRR in the refresh-permitted period, the writing sequence is valid and refreshing will be done. (Whether writing is made within the refresh-permitted period is determined by when FFh is written.)
After FFh is written to the IWDTRR register, refreshing the counter requires up to four cycles of the signal for counting (the clock division ratio selection bits (IWDTCR.CKS[3:0]) determine how many cycles of the IWDT clock (IWDTCLK) make up one cycle for counting). Therefore, writing FFh to the IWDTRR should be completed four-count cycles before the end position of the refresh-permitted period or a counter underflow. The value of the down-counter can be checked by the down-counter bits (IWDTSR.CNTVAL[13:0]).

[Sample refreshing timings]

- When the window start position is set to 1FFFh, even if 00h is written to IWDTRR before 1FFFh is reached (2002h, for example), refreshing is done if FFh is written to IWDTRR after the value of the IWDTSR.CNTVAL[13:0] bits has reached 1FFFh.
- When the window end position is set to 1FFFh, refreshing is done if 2003h (four-count cycles before 1FFFh) or a greater value is read from the IWDTSR.CNTVAL[13:0] bits immediately after writing 00h → FFh to IWDTRR.
- When the refresh-permitted period continues until count 0000h, refreshing can be done immediately before an underflow. In this case, if 0003h (four-count cycles before an underflow) or a greater value is read from the IWDTSR.CNTVAL[13:0] bits immediately after writing 00h → FFh to IWDTRR, no underflow occurs and refreshing is done.

Figure 22.5 shows the IWDT refresh-operation waveforms when PCLKB > IWDTCLK and clock division ratio = IWDTCLK. Figure 22.6 shows the IWDT refresh-operation waveforms when PCLKB < IWDTCLK and clock division ratio = IWDTCLK/16.

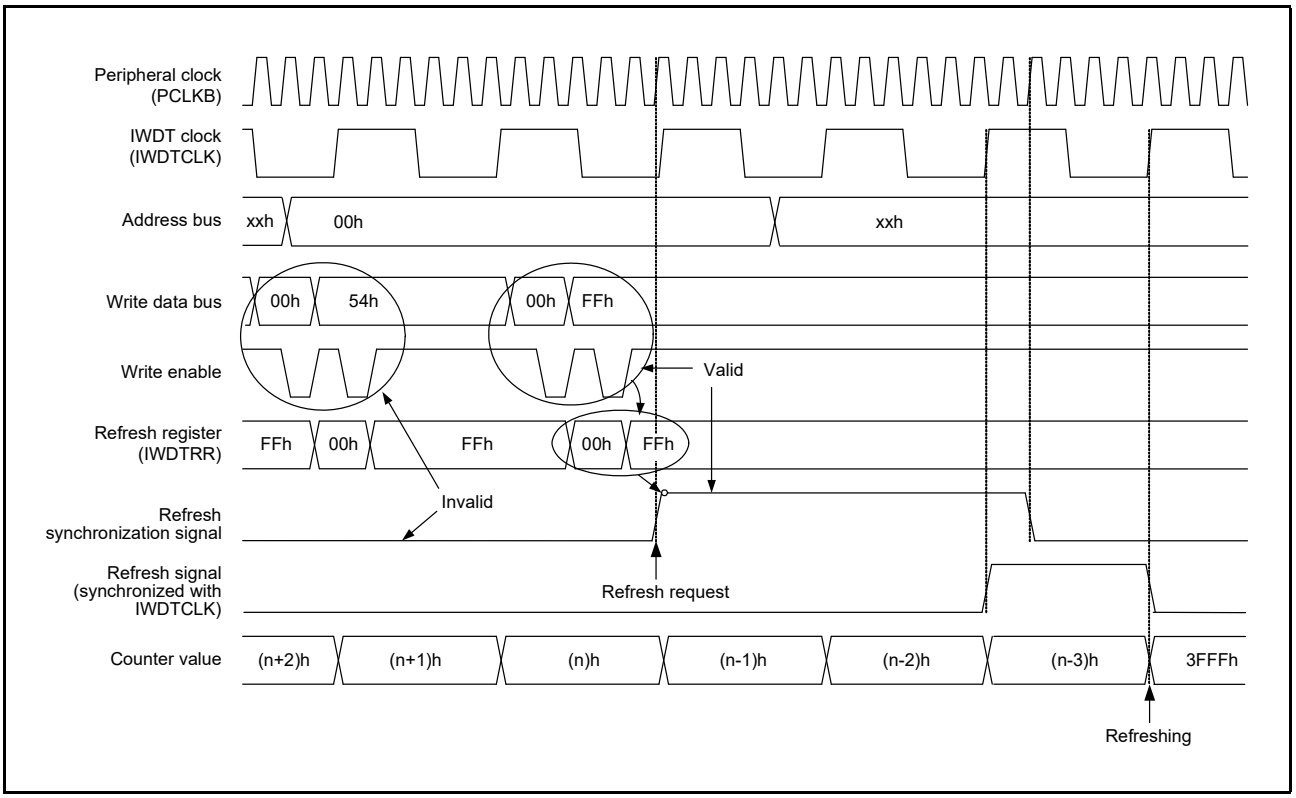


Figure 22.5 IWDT Refresh Operation Waveforms (IWDTCR.CKS[3:0] = 0000b, IWDTCR.TOPS[1:0] = 11b)

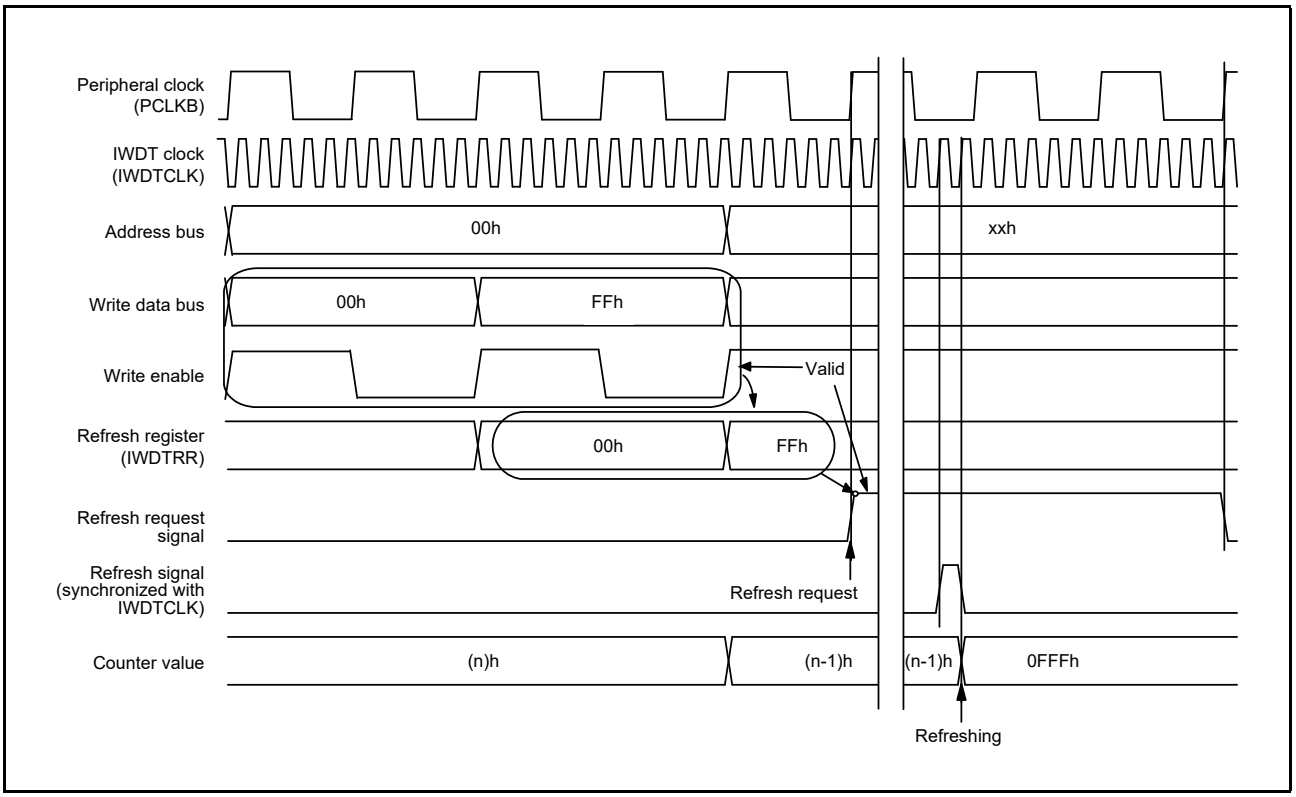


Figure 22.6 IWDT Refresh Operation Waveforms (IWDTCR.CKS[3:0] = 0010b, IWDTCR.TOPS[1:0] = 01b)

22.3.4 Status Flags

The refresh error flag (IWDTSR.REFEF) and the underflow flag (IWDTSR.UNDFF) retain the error causes when error notifications are output to the error control module (ECM) of IWDT.
Occurrence state of error notifications to ECM can be confirmed by reading the IWDTSR.REFEF flag or the IWDTSR.UNDFF flag after reset is released or upon occurrence of an error notification to ECM.
To clear these flags to 0, write 0. Writing 1 is ignored.
If these flags are not cleared, it will give no effect to the operation. Upon occurrence of the next error notification to ECM, the previous error notification is automatically cleared and a new notification to ECM is written.

22.3.5 Error Notification to the Error Control Module (ECM)

When the reset interrupt selection bit (IWDTRCR.RSTIRQS) is set to 0, underflow or refresh error of the down-counter causes an error notification to ECM to be generated during one count cycle.

22.3.6 Reading the Down-Counter Value

As the counter in the IWDT operates with the IWDT clock (IWDTCLK), the IWDT cannot read the counter value directly. Therefore, the IWDT synchronizes the counter value with the peripheral clock (PCLKB) and stores it in the counter value bits (IWDTSR.CNTVAL[13:0]) of the IWDT status register. Thus, the counter value can be checked indirectly through the IWDTSR.CNTVAL[13:0] bits.

Reading the counter value requires multiple PCLKB clock cycles (up to four clock cycles), and the read counter value may differ from the actual counter value by a value of one count.

Figure 22.7 shows the processing for reading the IWDT down-counter value.

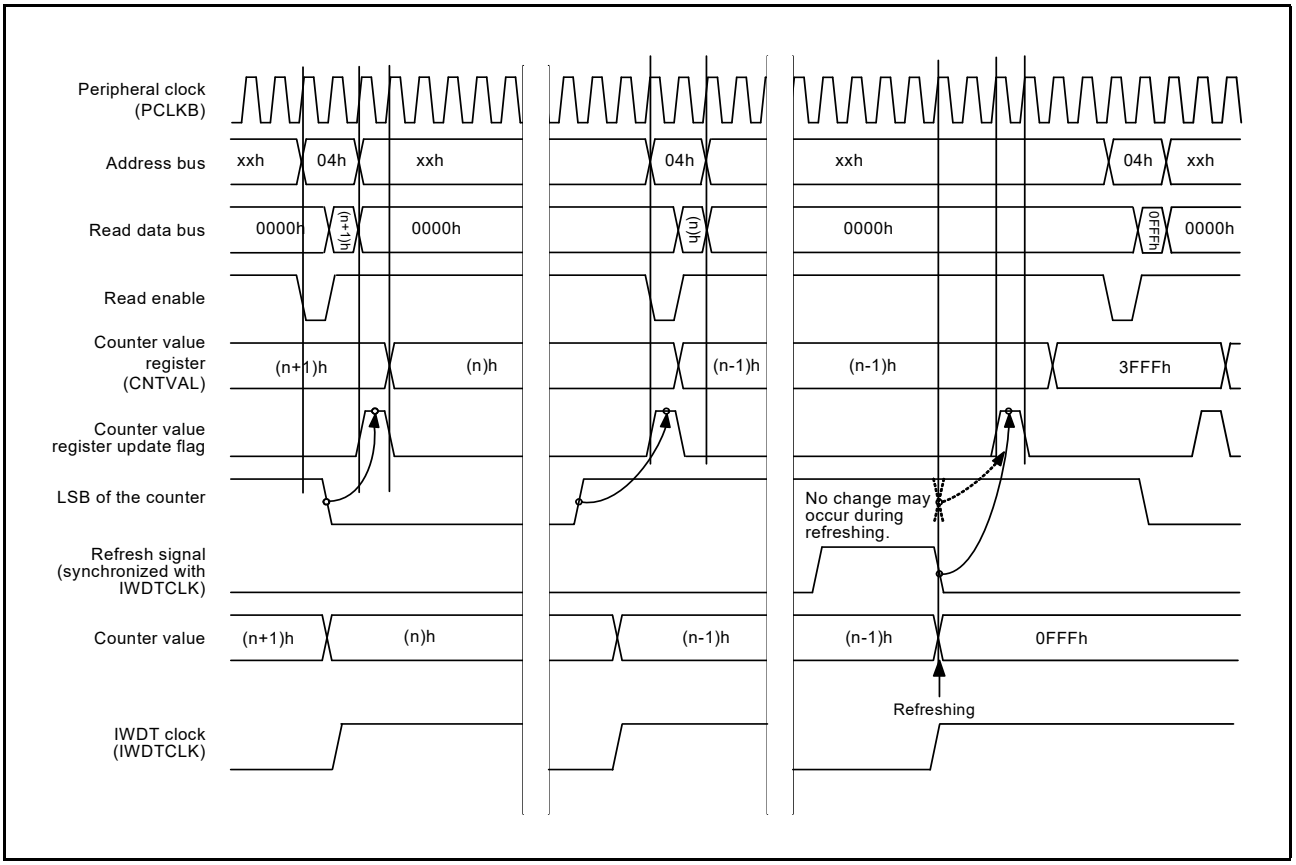


Figure 22.7 Processing for Reading IWDT Counter Value (IWDTCR.CKS[3:0] = 0000b, IWDTCR.TOPS[1:0] = 11b)

22.4 Low-Power Consumption Control

22.4.1 Watchdog Timer Operations in Low-Power Consumption Mode Transition

Clock supply to the IWDT can be controlled during transition to the standby mode of Cortex-R4 while the downcounter of IWDT is operating.

Table 22.4 lists the IWDT operations during transition to the low-power consumption mode.

Table 22.4 IWDT Operations during Transition to Low-Power Consumption Mode <in Low-Power Consumption Mode Transition>

Low-Power Consumption Mode	IWDT Clock Supply	IWDT Operation
Cortex-R4 standby	√	√

√: Operating

23. Management Data Input/Output Interface (MDIO)

This LSI chip has two management data input-output interfaces (MDIO slave and MDIO master).
The MDIO slave and master can be used at the same time.

23.1 Management Data Input/Output Interface (MDIO Slave)

This microcontroller has a single management data input/output interface (MDIO).

23.1.1 Overview

The MDIO slave in this microcontroller is one of the interfaces built in to the optical transceiver module, which is compliant with the CFP MSA specification. The MDIO slave is used for slave operations in data transfer to and from a host device. Some functions of the MDIO (control of the CFP registers and the transmission of state information through GPIO pins) are controlled by the CPU of this microcontroller.
For technical details regarding the MDIO slave, refer to CFP MSA Management Interface Specification Version 2.6 r03a September 26, 2016

Figure 23.1 is a block diagram of the MDIO slave.

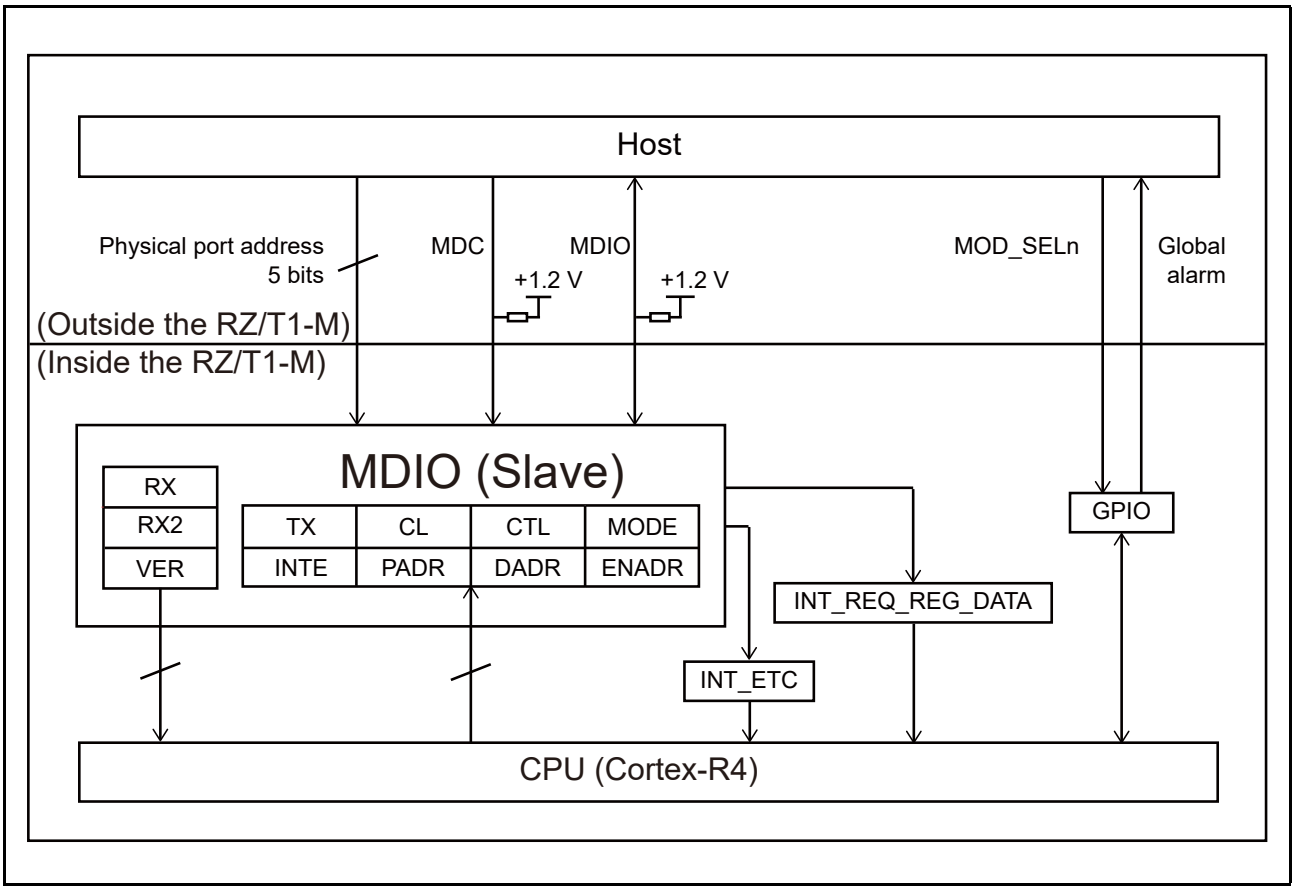


Figure 23.1 Block Diagram of the MDIO Slave

23.1.1.1 I/O Pins

Table 23.1 shows the pin configuration of the MDIO slave. For both input and output, the MDIO slave signals are 1.2-V LVCMOS.

Table 23.1 MDIO I/O Pins

Name	Pin Name	I/O	Function
Management data clock	MDC	Input	MDIO clock input pin for slave operation (up to 4 MHz)
Management data I/O	MDIO	I/O	MDIO data input/output pin for slave operation*1
MDIO physical port address 0	PRTADR0	Input	Optical transceiver module select input pin
MDIO physical port address 1	PRTADR1	Input	Optical transceiver module select input pin
MDIO physical port address 2	PRTADR2	Input	Optical transceiver module select input pin
MDIO physical port address 3	PRTADR3	Input	Optical transceiver module select input pin*2
MDIO physical port address 4	PRTADR4	Input	Optical transceiver module select input pin*2

Note 1. When using the MDIO pin, set B5[1:0] of the driving ability control register (DSCR) for PORT5 to 11 (1.2-V driving output).
Note 2. When using with CFP2 and CFP4, set B0[1:0] and B3[1:0] of the port direction register (PDR) for PORT5 to 00 (non-use) and set B0[1:0] and B3[1:0] of the pull-up/pull-down resistor control register (PCR) for PORT5 to 01 (enables an input pull-down resistor).

23.1.1.2 Interrupt

Table 23.2 lists the interrupt sources.

Table 23.2 Interrupt Sources

Interrupt Source		Conveying of the INT_REQ_REG_DATA Interrupt Signal	Conveying of the INT_ETC Interrupt Signal
Normal source	Completion of data transmission or reception	√	×
	Reception of DEVADD being complete and matching the expected value	×	√
	Reception of DEVADD being complete but not matching the expected value	×	√
	Reception of PHYADR being complete and matching the expected value	×	√
	Reception of PHYADR being complete but not matching the expected value	×	√
	Completion of OP reception	×	√
Error source	Generation of transmission error	×	√

Details of interrupts are as follows:

- There are two interrupt signals for the CPU.
- When the interrupt source is the completion of transmission or reception, the INT_REQ_REG_DATA interrupt signal is conveyed.
- If the interrupt source is other than the above, the INT_ETC interrupt signal is conveyed.
- The INT_ETC interrupt signal is conveyed when any of the interrupt source flags is set. The interrupt notification state continues until all interrupt source bits are cleared.
- In the case of the non-error interrupt, processing for data transmission or reception continues even if the interrupt source bits are not cleared.
- After an error occurs, the MDIO informs the host device that the register value is 0000h, and then stops.
- Since the MDIO has already informed the host device that the register value is 0000h after an error, control by the

CPU to convey the error signal to the host device by GPIO is required.

Interrupt control functions include those listed below for enabling and disabling interrupts in response to the above conditions being satisfied.

- Control over whether or not to convey the interrupt signal when the interrupt source condition is the completion of data transmission or reception even if the address of the CFP register is out of the range of valid values is available.
- Control over whether or not to convey the interrupt signal when the interrupt source condition is the reception of DEVADD being complete and matching the expected value is available.
- Control over whether or not to convey the interrupt signal when the interrupt source condition is the reception of DEVADD being complete but not matching the expected value is available.
- Control over whether or not to convey the interrupt signal when the interrupt source condition is the reception of PHYADR being complete and matching the expected value is available.
- Control over whether or not to convey the interrupt signal when the interrupt source condition is the reception of PHYADR being complete but not matching the expected value is available.
- Control over whether or not to convey the interrupt signal when the interrupt source condition is the completion of OP reception is available.
- Control over whether or not to convey the interrupt signal when the interrupt source condition is the generation of a transmission error is available.
- Perform the procedure described in (2) IRQ Interrupt (Level interrupt) in section 12.4.4.3, Detecting Interrupts by reading the RX register.

23.1.1.3 Restrictions

- In read operations, the CPU must write data read from the CFP register to the MDIO up to output of the read data following TA.
- When referring to operation code (OP) and device address (DEVADD) values received by the CPU, the received OP and DEVADD values must be acquired before the reception of the OP and DEVADD values of the next frame after the transmission/reception complete interrupt.
- For CFP2, if devices having 5-bit PHYADR including CFP exists, set the PCM bit of the mode register to 0.
- When using with CFP2, pull-down the PRTADR3 and PRTADR4 pins (fixed to 0).
- When using with CFP4, set B0[1:0] and B3[1:0] of the port direction register (PDR) for PORT5 to 00 (non-use) and set B0[1:0] and B3[1:0] of the pull-up/pull-down resistor control register (PCR) for PORT5 to 01 (Enables an input pull-down resistor).

23.1.2 Description of Registers

23.1.2.1 Input Registers

(1) Transmission Register (TX)

The TX register handles data for transmission.

Address(es): B011 C100h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	TXD[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	TXD[15:0]	Transmit Data	Data for transmission are set in these bits. On completion of the transmission or reception of the data, the CPU must set the value of the field or register stated below. Which of the latter is to be set depends on the value of the RXOP[1:0] bits of the RX register. If RXOP[1:0] is 01, the value of the RXD[15:0] bits in the RX register If RXOP[1:0] is a value other than 01, the value of the register at the address held in the RXD[15:0] bits of the RX register Setting these bits also clears the source flag for the INT_REQ_REG_DATA data transmission/reception complete interrupt.	W

(2) Interrupt Clear Register (CL)

The CL register handles clearing of the interrupt source flags.

Address(es): B011 C102h

	b7	b6	b5	b4	b3	b2	b1	b0
	CL PHY	—	CL DEV	CL OP	—	CLPHY NM	CLDEV NM	CL TXE
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CLTXE	Transmit Error Flag Clear	This bit is used to clear the transmission error source flag for the INT_ETC interrupt. 0: The transmission error flag is not cleared. 1: The SSTXE bit of the RX register is cleared to 0 to clear the transmission error flag.	W
b1	CLDEVNM	DEVADD Reception Completed but not Matching the Expected Value Flag Clearing	This bit clears the flag for the reception of DEVADD being complete but not matching the expected value, which is selectable as a source for the INT_ETC interrupt. 0: The flag for the reception of DEVADD being complete but not matching the expected value is not cleared. 1: The flag for the reception of DEVADD being complete but not matching the expected value (the SSDEVNM bit of the RX register) is cleared to 0.	W
b2	CLPHYNM	PHYADR Reception Completed but not Matching the Expected Value Flag Clearing	This bit clears the flag for the reception of PHYADR being complete but not matching the expected value, which is selectable as a source for the INT_ETC interrupt. 0: The flag for the reception of PHYADR being complete but not matching the expected value is not cleared. 1: The flag for the reception of PHYADR being complete but not matching the expected value (the SSPHYNM bit of the RX register) is cleared to 0.	W
b3	—	Reserved	The write value should be 0.	W
b4	CLOP	OP Reception Complete Flag Clear	This bit is used to clear the OP reception complete source flag for the INT_ETC interrupt. 0: The OP reception complete flag is not cleared. 1: The SSOP bit of the RX register is cleared to 0 to clear the OP reception complete flag.	W
b5	CLDEV	DEVADD Reception Completed and Matching the Expected Value Flag Clearing	This bit clears the flag for the reception of DEVADD being complete and matching the expected value, which is selectable as a source for the INT_ETC interrupt. 0: The flag for the reception of DEVADD being complete and matching the expected value is not cleared. 1: The flag for the reception of DEVADD being complete and matching the expected value (the SSDEV bit of the RX register) is cleared to 0.	W
b6	—	Reserved	The write value should be 0.	W
b7	CLPHY	PHYADR Reception Completed and Matching the Expected Value Flag Clearing	This bit clears the flag for the reception of PHYADR being complete and matching the expected value, which is selectable as a source for the INT_ETC interrupt. 0: The flag for the reception of PHYADR being complete and matching the expected value is not cleared. 1: The flag for the reception of PHYADR being complete and matching the expected value (the SSPHY bit of the RX register) is cleared to 0.	W

(3) Control Register (CTL)

The CTL register sets how the MDIO slave operates.

Address(es): B011 C104h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	ST MDIO
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	STMDIO	MDIO Slave Operation Control	This bit sets how the MDIO slave operates. 0: The MDIO stops after the completion of transmission/reception if it is operating and OP reception has started. Otherwise, the MDIO stops immediately. 1: The MDIO starts when the SSTXE bit of the RX register is set to 0. If the SSTXE bit of the RX register is set to 1 (indicating an error in transmission) after the start of operations, it continues operating until the transmission of dummy register data (0000h) is complete. Note: Frames are not analyzed while the MDIO interface is stopped. Since the address before it has been stopped will still be in effect when it is re-started, re-start operations from the frame with the given address.	W
b7 to b1	—	Reserved	The write value should be 0.	W

(4) Mode Register (MODE)

The MODE register is used for setting the operational modes of the MDIO slave.

Address(es): B011 C105h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	PCM	OM
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	OM	Output Mode Control	This bit sets how to output data in read-related operations. 0: Push-pull output 1: Open-drain output	W
b1	PCM	PHYADR Checking Mode Control	This bit sets the range of bits in a received PHYADR value that is used for checking whether the PHYADR value matches the expected value. 0: Compare all 5 bits of received 5-bit PHYADR values against the expected value to see if they match. 1: Compare the 3 lower-order bits of received 5-bit PHYADR values against the expected value to see if they match. When using a CFP2 module with a 5-bit PHYADR and in all cases with CFP and CFP8 modules, set this bit to 0. For CFP2 modules with only a 3-bit PHYADR and in all cases with CFP4 modules, set this bit to 1.	W
b7 to b2	—	Reserved	The write value should be 0.	W

(5) Interrupt Enable Register (INTE)

The INTE register is used to write the interrupt control information.

Address(es): B011 C106h

	b7	b6	b5	b4	b3	b2	b1	b0
	PHY	END	DEV	OP	—	PHY NM	DEV NM	TXE
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	TXE	Transmit Error Interrupt Enable	This bit enables or disables INT_ETC interrupts in response to setting of a source flag for this interrupt due to an error in transmission. 0: The INT_ETC interrupt signal is not conveyed even when the setting of the SSTXE bit of the RX register is 1 (indicating an error in transmission). 1: The INT_ETC interrupt signal is conveyed if the setting of the SSTXE bit of the RX register is 1 (indicating an error in transmission). Note: Since this value does not affect the operation in response to errors in transmission (transmission of a dummy value followed by operation being stopped until the SSTXE bit of the RX register is cleared to 0), we recommend checking the SSTXE bit of the RX register regularly if the setting of this bit is 0.	W
b1	DEVNM	DEVADD Reception Completed but not Matching the Expected Value Interrupt Enable	This bit enables or disables the interrupt on the reception of DEVADD being complete but not matching the expected value as a source for the INT_ETC interrupt. 0: The INT_ETC interrupt signal is not conveyed even when the setting of the SSDEVNM bit of the RX register is 1 (receive completion of DEVADD, which does not match with the expected value). 1: The INT_ETC interrupt signal is conveyed if the setting of the SSDEVNM bit of the RX register is 1 (receive completion of DEVADD, which does not match with the expected value).	W
b2	PHYNM	PHYADR Reception Completed but not Matching the Expected Value Interrupt Enable	This bit enables or disables the interrupt on the reception of PHYADR being complete but not matching the expected value as a source for the INT_ETC interrupt. 0: The INT_ETC interrupt signal is not conveyed even when the setting of the SSPHYNM bit of the RX register is 1 (receive completion of PHYADR, which does not match with the expected value). 1: The INT_ETC interrupt signal is conveyed if the setting of the SSPHYNM bit of the RX register is 1 (receive completion of PHYADR, which does not match with the expected value).	W
b3	—	Reserved	The write value should be 0.	W
b4	OP	OP Reception Interrupt Enable	This bit enables or disables INT_ETC interrupts in response to setting of the OP reception complete source flag for this interrupt. 0: The INT_ETC interrupt signal is not conveyed even when the setting of the SSOP bit of the RX register is 1 (indicating the completion of OP reception). 1: The INT_ETC interrupt signal is conveyed if the setting of the SSOP bit of the RX register is 1 (indicating completion of OP reception).	W
b5	DEV	DEVADD Reception Completed and Matching the Expected Value Interrupt Enable	This bit enables or disables the interrupt on the reception of DEVADD being complete and matching the expected value as a source for the INT_ETC interrupt. 0: The INT_ETC interrupt signal is not conveyed even when the setting of the SSDEV bit of the RX register is 1 (receive completion of DEVADD, which matches with the expected value). 1: The INT_ETC interrupt signal is conveyed if the setting of the SSDEV bit of the RX register is 1 (receive completion of DEVADD, which matches with the expected value).	W

Bit	Symbol	Bit Name	Description	R/W
b6	END	Transmission/ Reception Complete Interrupt Enable	This bit enables or disables the interrupt on the completion of data transmission or reception as a source for the INT_REQ_REG_DATA interrupt. 0: The INT_REQ_REG_DATA interrupt signal is conveyed if the setting of the SSEND bit of the RX register is 1 (completion of the transmission or reception of data) and the register address to be accessed is within the range of valid values. 1: The INT_REQ_REG_DATA interrupt signal is conveyed if the setting of the SSEND bit of the RX register is 1 (completion of the transmission or reception of data) regardless of the register address to be accessed.	W
b7	PHY	PHYADR Reception Completed and Matching the Expected Value Interrupt Enable	This bit enables or disables the interrupt on the reception of PHYADR being complete and matching the expected value as a source for the INT_ETC interrupt. 0: The INT_ETC interrupt signal is not conveyed even when the setting of the SSPHY bit of the RX register is 1 (receive completion of PHYADR, which matches with the expected value). 1: The INT_ETC interrupt signal is conveyed if the setting of the SSPHY bit of the RX register is 1 (receive completion of PHYADR, which matches with the expected value).	W

(6) Physical Port Address Register (PADR)

The PADR register sets the method of checking and, if checking against the value in this register is selected, a value for checking against received PHYADR values.

Address(es): B011 C108h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	PHYC[4:0]					—	—	—	EXPPHY[4:0]				
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b4 to b0	EXPPHY [4:0]	PHYADR Expected Value Data	These bits set the expected value for checking for matches against received PHYADR values if checking against this register is selected. For CFP2 modules, set the 2 higher-order bits to 0.	W
b7 to b5	—	Reserved	The write value should be 0.	W
b12 to b8	PHYC [4:0]	PHYADR Checking Method Selection	These bits set the method of checking for matches between received PHYADR values and the expected value. The method can be set bit by bit. 0: The PRTADR bit of the RX2 register is used as the expected value in checking for matches with received PHYADR values. 1: The EXPPHY[4:0] bit in this register is used as the expected value in checking for matches with received PHYADR values. For CFP2 or CFP8 modules, set these bits to 11000 or 11111, respectively.	W
b15 to b13	—	Reserved	The write value should be 0.	W

(7) Device Address Register (DADR)

The DADR register sets the data for checking against received DEVADD values.

Address(es): B011 C10Ah

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	EXPDEV[4:0]				
Value after reset:	0	0	0	0	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b4 to b0	EXPDEV [4:0]	DEVADD Expected Value Data	These bits set the expected value for checking for matches against received DEVADD values.	W
b7 to b5	—	Reserved	The write value should be 0.	W

(8) Enable Address Register (ENADR)

The ENADR register sets the range of valid values for register addresses.

Address(es): B011 C10Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	UADR[15:0]															
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	LADR[15:0]															
Value after reset:	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	LADR [15:0]	Register Address Lower Limit	These bits set the lower limit on the range of valid values for register addresses.	W
b31 to b16	UADR [15:0] *1	Register Address Upper Limit	These bits set the upper limit on the range of valid values for register addresses.	W

Note 1. If these bits are set to the same value as the LADR[15:0] bits, there is no range of valid values for register addresses but only a single setting. Also, set these bits to the same value as or a greater value than that of the LADR[15:0] bits.

23.1.2.2 Output Registers

(1) Reception Register (RX)

The RX register handles received data and indicates the states of interrupt source conditions.

Address(es): B051 C700h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	SS PHY	SS END	SS DEV	SS OP	—	SSPHY NM	SSDEV NM	SS TXE	RXOP[1:0]		—	RXDEV[4:0]				
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RXD[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	RXD [15:0]	Register Access Data	These bits hold data required for access to the CFP registers. The value is updated on completion of the transmission or reception of data and the location for storage depends on the contents of the RXOP[1:0] bits of the RX register. RXOP[1:0] = 00: Received register address RXOP[1:0] = 01: Received register data. The address to which data are to be written is held by the CPU, which is the RXD on completion of the previous transmission or reception. RXOP[1:0] = 11: Register address that was stored on completion of the previous transmission or reception RXOP[1:0] = 10: The following value is obtained from the value of the register address that was stored on completion of the previous transmission or reception. Address out of the range of valid values: Address on completion of the previous transmission or reception Address within the range of valid values but other than FFFFh: Address on completion of the previous transmission or reception + 1 FFFFh: Address on completion of the previous transmission or reception	R
b20 to b16	RXDEV [4:0]	Receive DEVADD Data	These bits hold the received DEVADD data. The value is updated on completion of DEVADD reception.	R
b21	—	Reserved	This bit is read as 0.	R
b23, b22	RXOP[1:0]	Receive OP Data	These bits hold the received OP data. The value is updated on completion of OP reception.	R
b24	SSTXE	Transmit Error	This bit indicates the state of the error in transmission source condition of the INT_ETC interrupt. 0: No error in transmission occurred. 1: An error in transmission occurred. If the setting of the RXOP[1] bit of the RX register is 1, an error in transmission is detected when the SSEND bit of the RX register is not cleared to 0 by the time of the switch from reception to transmission. After an error in transmission, the MDIO transmits a dummy value (0x0000) and then stops until this bit is cleared to 0. This bit is cleared by writing 1 to the CLTXE bit of the CL register.	R
b25	SSDEVNM	DEVADD Reception Completed but not Matching the Expected Value	This bit indicates the state for the reception of DEVADD being complete but not matching the expected value, which is selectable as a source for the INT_ETC interrupt. 0: DEVADD being received does not match the expected value and reception is not completed. 1: On completion of reception, the received DEVADD does not match the expected value. This bit is cleared by writing 1 to the CLDEVNM bit of the CL register.	R

Bit	Symbol	Bit Name	Description	R/W
b26	SSPHYNM	PHYADR Reception Completed but not Matching the Expected Value	This bit indicates the state for the reception of PHYADR being complete but not matching the expected value, which is selectable as a source for the INT_ETC interrupt. 0: PHYADR being received does not match the expected value and reception is not completed. 1: On completion of reception, the received PHYADR does not match the expected value. This bit is cleared by writing 1 to the CLPHYNM bit of the CL register.	R
b27	—	Reserved	This bit is read as 0.	R
b28	SSOP	OP Reception Complete	This bit indicates the state of the OP reception complete source condition of the INT_ETC interrupt. 0: OP not received 1: OP received This bit is cleared by writing 1 to the CLOP bit of the CL register.	R
b29	SSDEV	DEVADD Reception Completed and Matching the Expected Value	This bit indicates the state for the reception of DEVADD being complete and matching the expected value, which is selectable as a source for the INT_ETC interrupt. 0: DEVADD being received matches the expected value, but reception is not completed. 1: On completion of reception, the received DEVADD matches the expected value. This bit is cleared by writing 1 to the CLDEV bit of the CL register.	R
b30	SSEND	Transmission/ Reception Complete	This bit indicates the state of the data transmission/reception complete source condition of the INT_REQ_REG_DATA interrupt. 0: Data transmission or reception not complete 1: Data transmission or reception complete This bit is cleared by writing the desired value to the TXD[15:0] bits of the TX register.	R
b31	SSPHY	PHYADR Reception Completed and Matching the Expected Value	This bit indicates the state for the reception of PHYADR being complete and matching the expected value, which is selectable as a source for the INT_ETC interrupt. 0: PHYADR being received matches the expected value, but reception is not completed. 1: On completion of reception, the received PHYADR matches the expected value. This bit is cleared by writing 1 to the CLPHY bit of the CL register.	R

(2) Reception Register 2 (RX2)

The RX2 register handles the received data and the information on input pins.

Address(es): B051 C704h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	PRTADR[4:0]					—	—	—	RXPHY[4:0]				
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b4 to b0	RXPHY [4:0]	Received PHYADR Data	These bits hold received PHYADR values. The value is updated each time the reception of PHYADR is completed.	R
b7 to b5	—	Reserved	These bits are read as 0.	R
b12 to b8	PRTADR [4:0]	PRTADR Pin Monitor	These bits store the results if monitoring the PRTADR4 to PRTADR0 pins. The results from the PRTADR4, PRTADR3, PRTADR2, PRTADR1, and PRTADR0 pins are stored from the most significant bit in that order. The result is continually updated while the MDIO slave is enabled. 0: The PRTADR pin is low. 1: The PRTADR pin is high.	R
b15 to b13	—	Reserved	These bits are read as 0.	R

(3) Version Register (VER)

The VER register indicates the version number of the configuration data.

Address(es): VER B051 C70Eh

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	MAJR[7:0]								MINR[7:0]							
Value after reset:	0	0	0	0	0	0	1*1	0	0	0	0	0	0	0	0	0

Note 1. All bits are read as 0 until after firmware writing.

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	MINR[7:0]	Minor Part of Version Number	Indicates the minor part of the version number of the configuration data.	R
b15 to b8	MAJR[7:0]	Major Part of Version Number	Indicates the major part of the version number of the configuration data.	R

23.1.3 Outline of Operation

23.1.3.1 MDIO Slave Startup Sequence

Be sure to use the startup sequence of the MDIO slave driver when initializing the MDIO slave. Using the MIDO slave also requires the writing of firmware for the MDIO slave at the time of startup. For details of the specific MDIO slave startup sequence and how to write the firmware, refer to the *RZ/T1-M Group MDIO Slave Application Note.*

23.1.3.2 Flowchart of MDIO Slave Processing

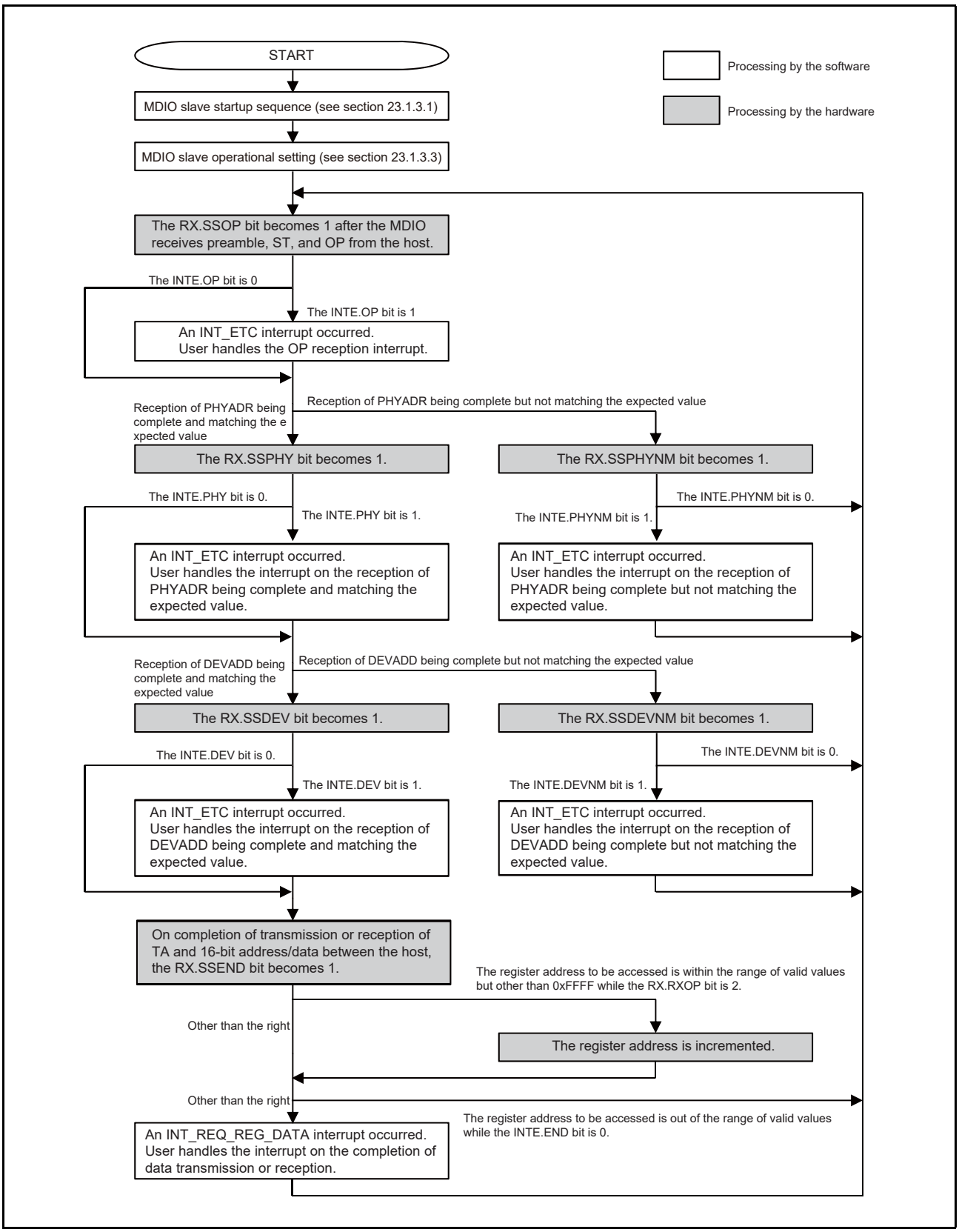


Figure 23.2 Flowchart of MDIO Slave Processing

23.1.3.3 Settings for MDIO Slave Operation

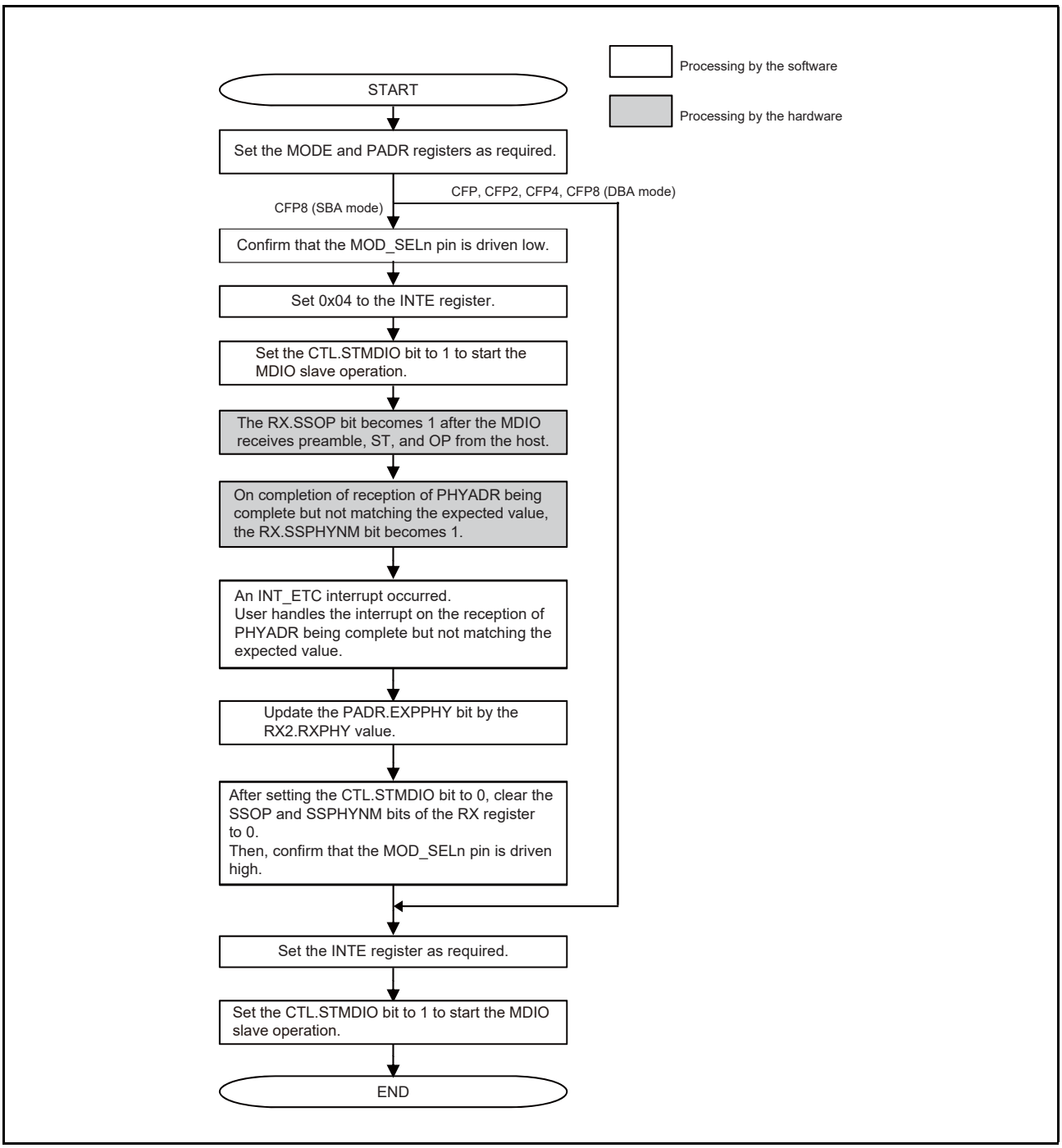


Figure 23.3 Flowchart of MDIO Slave Operation Setting

23.1.4 Usage Notes

- In read operations, the CPU must write data read from the CFP register to the MDIO slave up to output of the read data following TA.
- When referring to OP, PHYADR, and DEVADD values received by the CPU, the received OP, PHYADR, and DEVADD values must be acquired before the reception of the OP, PHYADR, and DEVADD values of the next frame.
- For CFP2, if devices having 5-bit PHYADR including CFP exists, set the PCM bit of the mode register to 0.
- When using with CFP2 and CFP4, set B0[1:0] and B3[1:0] of the port direction register (PDR) for PORT5 to 00 (non-use) and set B0[1:0] and B3[1:0] of the pull-up/pull-down resistor control register (PCR) for PORT5 to 01 (enables an input pull-down resistor).

23.2 Management Data Input/Output Interface (MDIO Master)

23.2.1 Overview

The management data input/output interface (MDIO master) of this LSI chip is an interface for master operation in conformance with the IEEE 802.3 clause 45 specification. It handles transfer to and from slave devices as an MDIO master.

For technical details on MDIO master operation according to clause 45, see *IEEE 802.3, Clause45*.

Table 23.3 Specification of the Management Data Input/Output Interface (MDIO Master)

Item	Description
Number of channels	One channel for 3.3-V operation
Protocol	• IEEE 802.3 Clause 45
Clock frequency	• Up to 10 MHz
Interrupt sources	• Data transmission/reception completed interrupt
Function	• Allows serial communications by using the MDC (Management Data Clock) and MDIO (Management Data I/O) signals. • Only for push-pull operation
Detected errors	• Errors in reception of the second bit of TA data • Transmission overflow errors

Figure 23.4 is a block diagram of the MDIO master.

MMDC is the clock output pin and MMDIO is the data I/O pin. INT_END is an interrupt signal to be generated by the MDIO master.

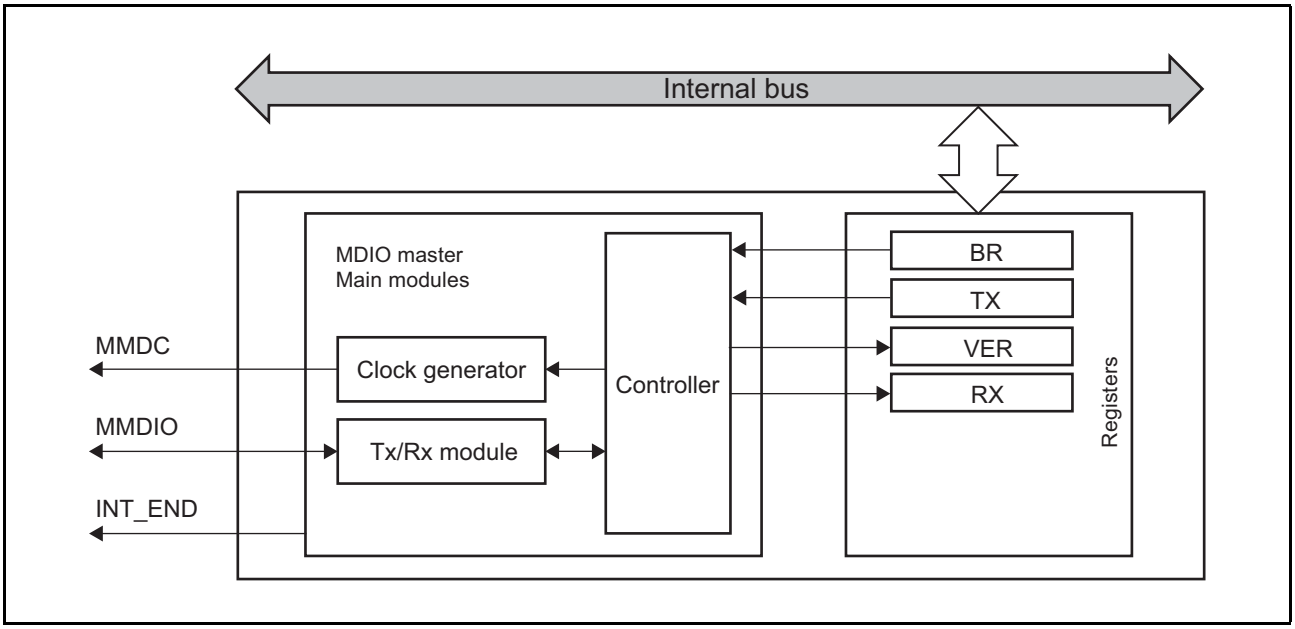


Figure 23.4 Block Diagram of the MDIO Master

23.2.1.1 I/O Pins

Table 23.4 lists the pins of the MDIO master. The input and output levels of the signals of the MDIOM1 channel are 3.3-V CMOS.

Table 23.4 I/O Pins of the MDIO Master

Channel	Pin Name	I/O	Function
MDIOM1	MMDC1	Output	MDIO clock output pin (up to 10 MHz)
	MMDIO1	I/O	MDIO data I/O pin

23.2.1.2 Correspondence between I/O Pins and I/O Ports

Table 23.5 lists the I/O pins and the corresponding I/O ports.

Table 23.5 I/O Pins and the Corresponding I/O Ports

Channel	Pin Name	I/O Port
MDIOM1	MMDC1	PA3
	MMDIO1	PA5

23.2.2 Description of the Registers

Table 23.6 lists the registers for use by the MDIO master. The value after a reset can be read after writing it to firmware. Access to the areas not listed in this table is prohibited. The value after a reset can be read after writing it to firmware.

Table 23.6 List of I/O Registers

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
B0A3 0012h	MDIOM1	Bit rate register	BR	16	16
B0B1 C100h	MDIOM1	Transmission register	TX	32	32
B0B1 C500h	MDIOM1	Version register	VER	16	16
B0B1 CD04h	MDIOM1	Reception register	RX	32	32

23.2.2.1 Bit Rate Register (BR)

The BR register is for setting the clock frequency for MDIO transfer. Set a value in this register before starting MDIO transfer.

Address(es): MDIOM1.BR B0A3 0012h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	CKDIV[7:0]							
Value after reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	CKDIV[7:0]	Clock Division Ratio Setting	The clock frequency for MDIO transfer is the frequency obtained by dividing 200 MHz by the setting of these bits + 1. Allowable settings are only the odd-numbers from 19 (13h) to 199 (C7h). The value of these bits after a reset is 49 (31h).	W
b15 to b8	—	Reserved	The write value should be 0.	W

CKDIV[7:0] Bits (Clock Division Ratio Setting)

These bits set the clock frequency for MDIO transfer.

See Table 23.7 for the correspondence between the settings of these bits and the clock frequencies for MDIO transfer.

Table 23.7 Settings of the CKDIV[7:0] Bits and Clock Frequencies for MDIO Transfer

CKDIV[7:0] Bits	Clock Frequency for MDIO Transfer
19	10 MHz
49	4 MHz
199	1 MHz

23.2.2.2 Transmission Register (TX)

The TX register is for setting data for transmission. The set data are transmitted MSB-first.
Writing to this register leads to MDIO transfer starting in accord with the settings in this register. Additionally, writing to this register during MDIO transfer allows consecutive MDIO transfer.
However, in cases of writing to this register twice or more during the same MDIO transfer, settings made after the first have no effect and the RX.TXOVFERR bit is set to 1 after completion of the MDIO transfer that was in progress.

Address(es): MDIOM1.TX B0B1 C100h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ST[1:0]		OP[1:0]		PHY[4:0]					DEV[4:0]					TA[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ADRDAT[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	ADRDAT [15:0]	16-Bit Address Data Setting	These bits set the address to be transmitted in an address operation and the data to be transmitted in a write operation, following the TA value in the MDIO transfer. Allowable settings are from 0 to 65535 (FFFFh).	R/W
b17, b16	TA[1:0]	TA Data Setting	These bits set the TA value to be transmitted following the DEVADD value in the MDIO transfer. An allowable setting is only 2.	R/W
b22 to b18	DEV[4:0]	DEVADD Data Setting	These bits set the DEVADD value to be transmitted following the PHYADR value in the MDIO transfer. Allowable settings are from 0 to 31 (1Fh).	R/W
b27 to b23	PHY[4:0]	PHYADR Data Setting	These bits set the PHYADR value to be transmitted following the OP value in the MDIO transfer. Allowable settings are from 0 to 31 (1Fh).	R/W
b29, b28	OP[1:0]	OP Data Setting	These bits set the OP value to be transmitted following the ST value in the MDIO transfer. The corresponding operation from the list below proceeds in accord with the setting of these bits. 0: Address operation 1: Write operation 2: Post-read-increment-address operation 3: Read operation	R/W
b31, b30	ST[1:0]	ST Data Setting	These bits set the ST value to be transmitted following the preamble value in the MDIO transfer. An allowable setting is only 0.	R/W

ADRDAT[15:0] Bits (16-Bit Address Data Setting)

These bits set the address to be transmitted in an address operation and the data to be transmitted in a write operation, following the TA value in the MDIO transfer.

DEVADD and subsequent data are not transmitted in read and post-read-increment-address operations, so the setting of these bits is not used in these operations.

TA[1:0] Bits (TA Data Setting)

These bits set the TA value to be transmitted following the DEVADD value in the MDIO transfer.

DEVADD and subsequent data are not transmitted in read and post-read-increment-address operations, so the setting of these bits is not used in these operations.

The setting of these bits should always be 2.

DEV[4:0] Bits (DEVADD Data Setting)

These bits set the DEVADD value to be transmitted following the PHYADR value in the MDIO transfer.

PHY[4:0] Bits (PHYADR Data Setting)

These bits set the PHYADR value to be transmitted following the OP value in the MDIO transfer.

OP[1:0] Bits (OP Data Setting)

These bits set the OP value to be transmitted following the ST value in the MDIO transfer.

See Table 23.8 for the correspondences between the settings of these bits and the types of operation.

Table 23.8 Settings of the OP[1:0] Bits and Types of Operation

OP[1:0] Bits	Type of Operation
0	Address operation
1	Write operation
2	Post-read-increment-address operation
3	Read operation

ST[1:0] Bits (ST Data Setting)

These bits set the ST value to be transmitted following the preamble value in the MDIO transfer.

The setting of these bits should always be 0.

23.2.2.3 Version Register (VER)

The VER register indicates the version number of the configuration data.

Address(es): MDIOM1.VER B0B1 C500h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	MAJR[7:0]								MINR[7:0]							
Value after reset:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	MINR[7:0]	Minor Version	These bits indicate the minor part of the version number of the configuration data.	R
b15 to b8	MAJR[7:0]	Major Version	These bits indicate the major part of the version number of the configuration data.	R

MINR[7:0] Bits (Minor Version)

These bits indicate the minor part of the version number of the configuration data.

MAJR[7:0] Bits (Major Version)

These bits indicate the major part of the version number of the configuration data.

23.2.2.4 Reception Register (RX)

The RX register holds the received 16 bits of data and indicators of the state of MDIO transfer.

Address(es): MDIOM1.RX B0B1 CD04h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	END	—	—	—	—	—	TA2ERR	TXOVFERR	—	TXOP[1:0]		TXPHY[4:0]				
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	DAT[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to 0	DAT[15:0]	16-Bit Data	These bits hold data which were received in a read or post-read-increment-address operation.	R
b20 to b16	TXPHY[4:0]	Transmitted PHYADR Data	These bits hold the transmitted PHYADR value.	R
b22, b21	TXOP[1:0]	Transmitted OP Data	These bits hold the transmitted OP value.	R
b23	—	Reserved	This bit is read as 0.	R
b24	TXOVFERR	Transmission Overflow Error	This bit indicates whether a transmission overflow error, i.e. writing to the TX register more than once during the same MDIO transfer, has occurred. 0: No transmission overflow error has occurred. 1: A transmission overflow error has occurred.	R
b25	TA2ERR	TA2 Reception Error	This bit indicates whether the value received as the second of the TA bits in a read or post-read-increment-address operation is correct. 0: The correct value 0 was received as the second of the TA bits in the case of reception or an address or write operation which does not involve reception has proceeded. 1: The incorrect value 1 was received as the second of the TA bits.	R
b30 to b26	—	Reserved	These bits are read as 0.	R
b31	END	MDIO Transfer Complete	This bit indicates whether the MDIO transfer has been completed. 0: The MDIO transfer has not been completed. 1: The MDIO transfer has been completed.	R

DAT[15:0] Bits (16-Bit Data)

These bits hold data which were received in a read or post-read-increment-address operation.

These bits are set to 0 in the case of an address or write operation.

The value of these bits is always updated when the RX.END bit is set to 1 following completion of the MDIO transfer.

TXPHY[4:0] Bits (Transmitted PHYADR Data)

These bits hold the transmitted PHYADR value.

The value of these bits is always updated when the RX.END bit is set to 1 following completion of the MDIO transfer.

TXOP[1:0] Bits (Transmitted OP Data)

These bits hold the transmitted OP value.

The value of these bits is always updated when the RX.END bit is set to 1 following completion of the MDIO transfer.

TXOVFERR Bit (Transmission Overflow Error)

This bit indicates whether a transmission overflow error, i.e. writing to the TX register more than once during the same MDIO transfer, has occurred. The setting 0 indicates that no transmission overflow error has occurred and the setting 1 indicates a transmission overflow error.

The value of this bit is always updated when the RX.END bit is set to 1 following completion of the MDIO transfer.

Figure 23.5 shows an example of operation of this bit in the case of making consecutive settings.

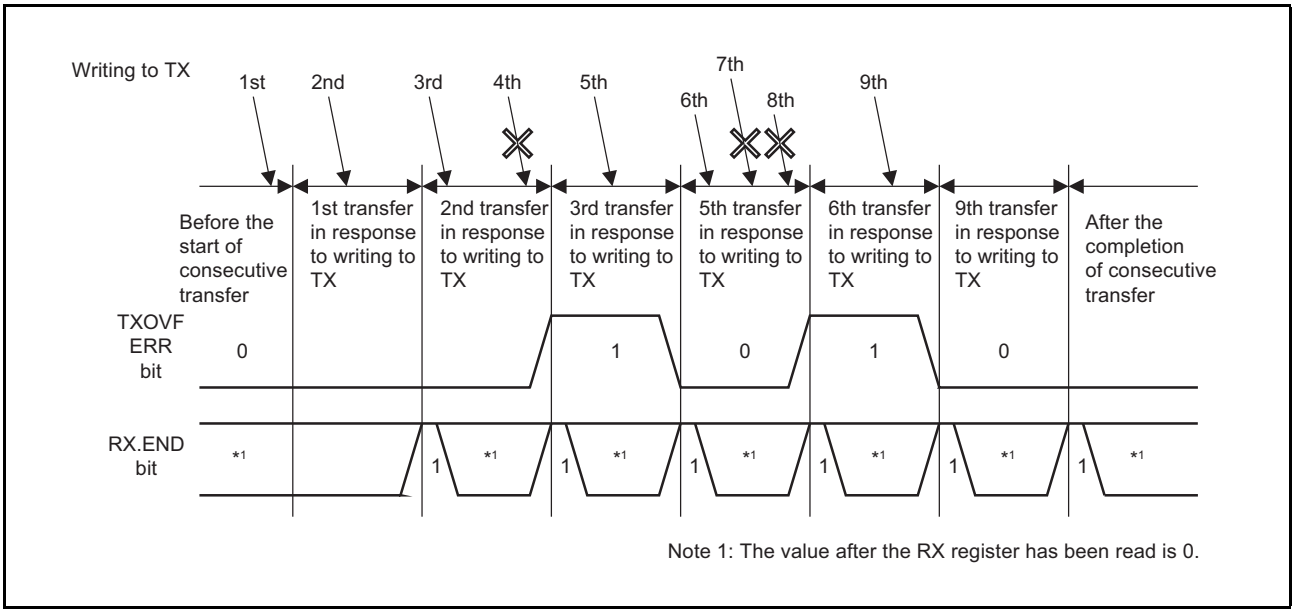


Figure 23.5 Example Operation of the TXOVFERR Bit in the Case of Making Consecutive Settings

TA2ERR Bit (TA2 Reception Error)

This bit indicates whether the value received as the second of the TA bits in a read or post-read-increment-address operation is correct. The setting 0 indicates that the correct value 0 was received and the setting 1 indicates that the incorrect value 1 was received.

This bit is set to 0 in the case of an address or write operation.

The value of this bit is always updated when the RX.END bit is set to 1 following completion of the MDIO transfer.

The setting 1 for this bit indicates that there may be a problem with the communications environment, such as the slave device not being ready to operate or the distance to the slave device being too great.

END Bit (MDIO Transfer Complete)

This bit indicates whether the MDIO transfer has been completed. The setting 1 indicates that the MDIO transfer has been completed.

This bit is always set to 1 in response to the MDIO transfer being completed and an INT_END interrupt is generated.

Reading the RX register leads to clearing of this bit to 0, which deasserts the INT_END interrupt. Since the settings of the RX register are updated every time an MDIO transfer is completed, read the RX register before completion of the next MDIO transfer after this bit has been set to 1 following the completion of the current MDIO transfer.

23.2.3 Operation

The MDIO master transmits the data required by the type of operation and receives 16 bits of data.

23.2.3.1 Types of Operation

MDIO transfer according to the IEEE802.3 clause 45 specification consists of the following four types of operation. Any of these operations requires 64 clock cycles of processing time for each MDIO transfer.

- Address operation
- Write operation
- Post-read-increment-address operation
- Read operation

(1) Format of Address and Write Operations

The format of address and write operations is as shown in Figure 23.6.

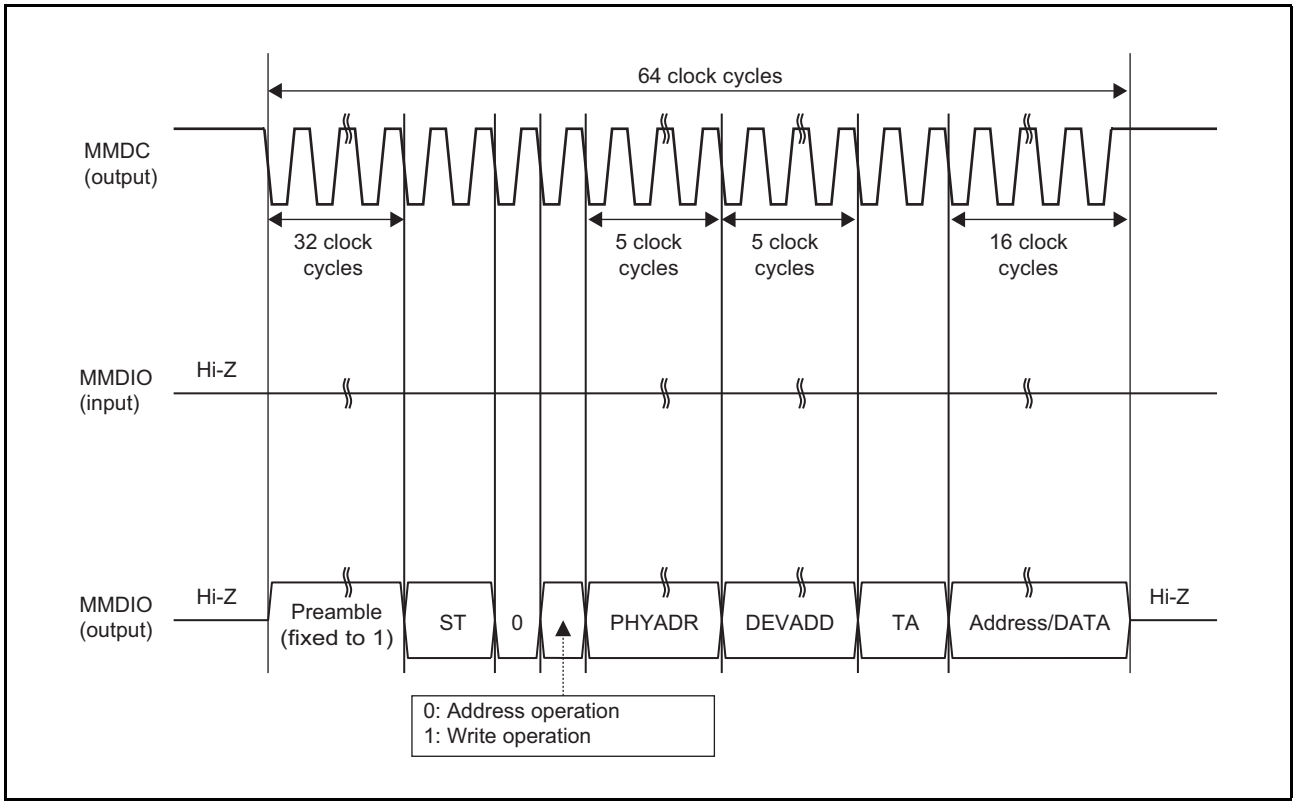


Figure 23.6 Format of Address and Write Operations

(2) Format of Post-Read-Increment-Address and Read Operations

The format of post-read-increment-address and read operations is as shown in Figure 23.7. Note that the depiction of MMDIO (input) in this figure is on the assumption that the delay in transfer is 0.

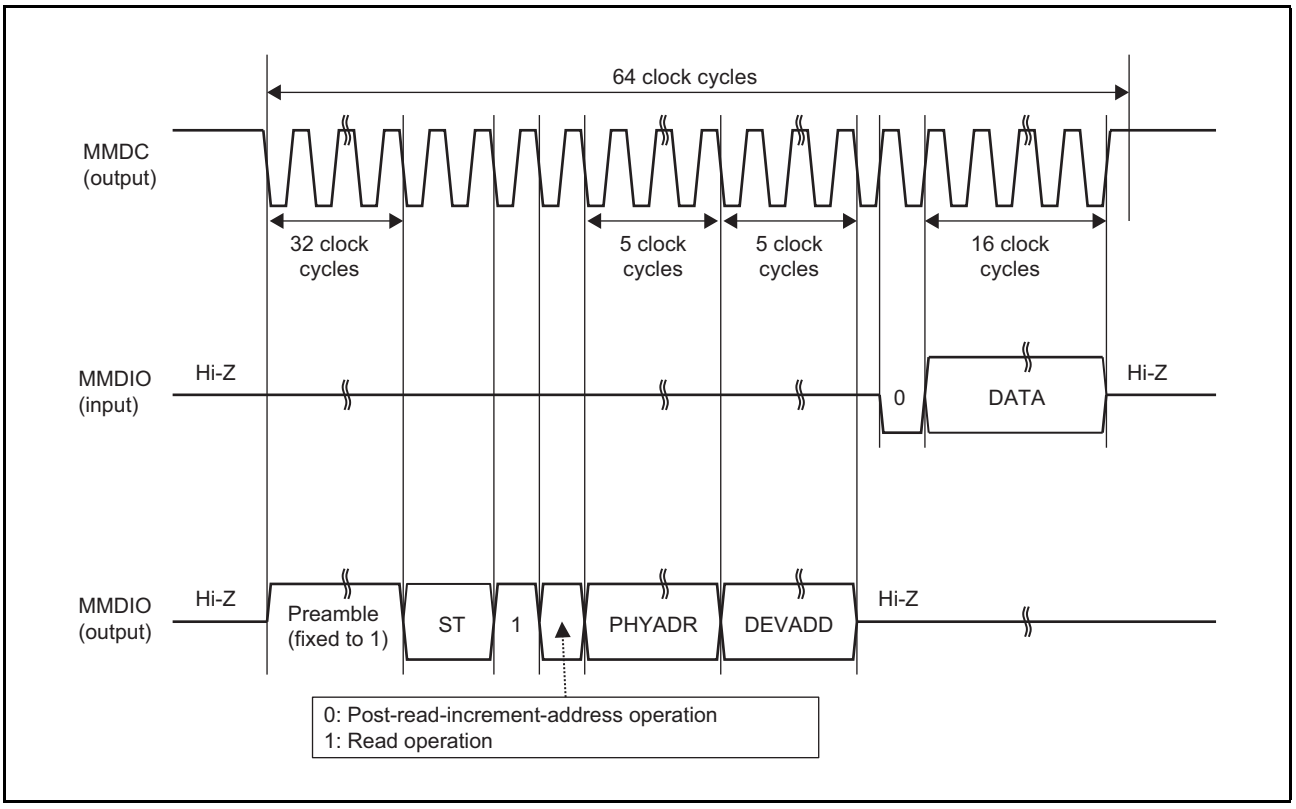


Figure 23.7 Format of Post-Read-Increment-Address and Read Operations

23.2.3.2 Idle Operation after MDIO Transfer

After the MDIO transfer has been completed, the interface becomes idle, with the output on the MMDC pin fixed to the high level and the MMDIO output becoming hi-Z. Even in consecutive MDIO transfers, operation always becomes idle for one clock cycle after the completion of each MDIO transfer. Figure 23.8 shows an example of consecutive operations in MDIO transfer.

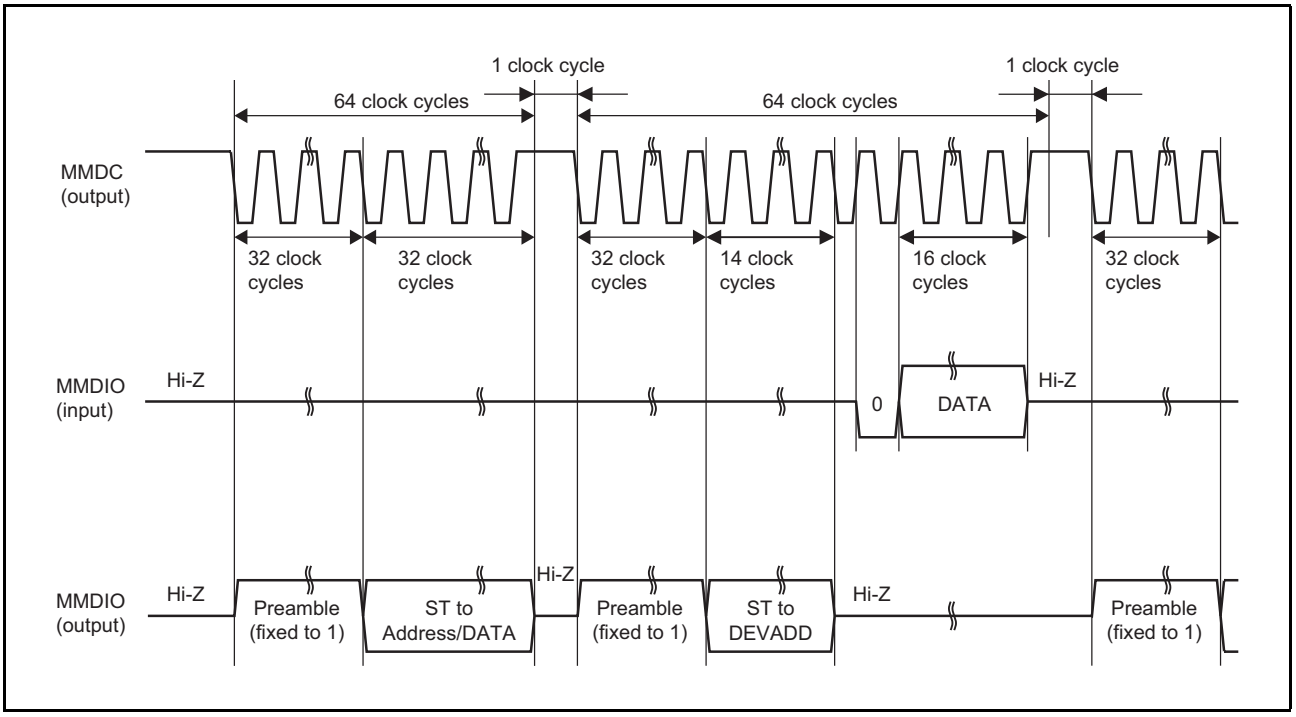


Figure 23.8 Example of Consecutive Operations in MDIO Transfer

23.2.3.3 Interrupt

An interrupt request (INT_END) is generated when the RX.END bit is set to 1 in response to the MDIO transfer being completed. To deassert the INT_END interrupt, read the RX register. This leads to deassertion of the INT_END interrupt.

23.2.4 Setting Procedures

23.2.4.1 Initialization

Figure 23.9 shows the flow for initializing the MDIO master.

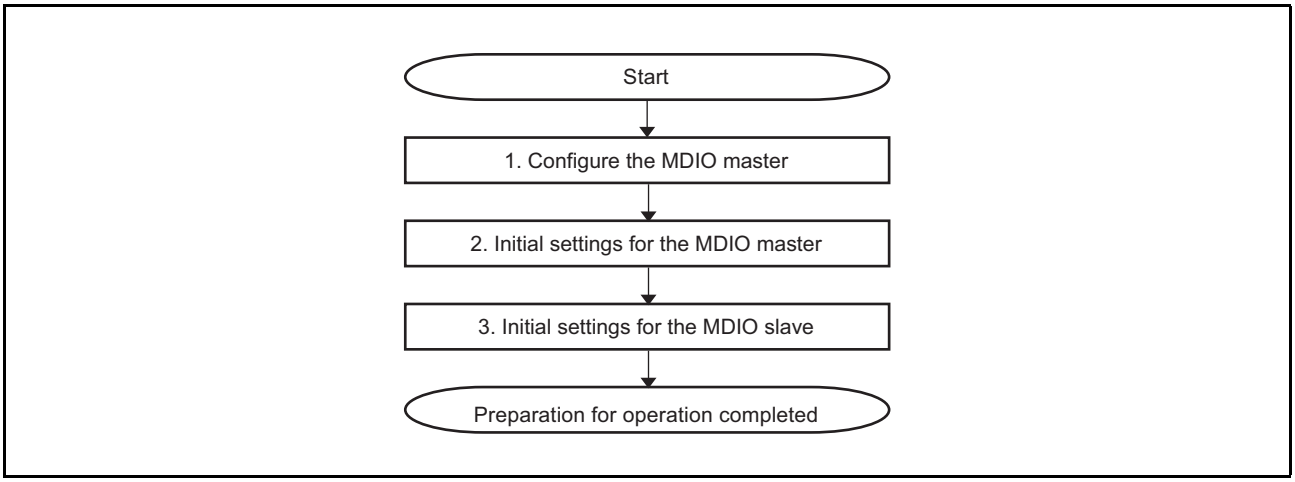


Figure 23.9 MDIO Master Initialization Flow

1. Configuration of the MDIO Master
   When configuring the MDIO master, follow the procedure below.

   1) Set the port direction register (PDR) for the pins for use by the MDIO master (listed in section 23.2.1.2, Correspondence between I/O Pins and I/O Ports) to 00b and the port mode register (PMR) to 0 to select a general I/O port.
   2) Configure the MDIO master by using the MDIO configuration library.
   3) Make Pmn pin function control registers (PmnPFS) (m = 0 to 9, A to H, J to N, P, R to U; n = 0 to 7) writable by setting the PWPR.PFSWE bit to 1 after setting the PWPR.B0WI bit to 0.
   4) Set the PmnPFS.PSEL[5:0] bits for the pins set in step 1) to 2Bh.
   5) Set the PWPR.PFSWE bit to 0 to disable writing to the PmnPFS register.
   6) Set the corresponding bits of the PMR register for the pins set in step 1) to 1, and switch the pin input/output functions of the MDIO master.
   7) The port state can be read by setting the PDR register to 10b as required.
   8) Start the MDIO master by using the MDIO configuration library.

2. Initial Settings for the MDIO Master
   Make the initial setting of the BR register.

   For example, when the clock frequency for MDIO transfer is to be 10 MHz,
   BR.CKDIV[7:0] = 19 (operation is at 200/(19+1) MHz.)

3. Initial Settings for the MDIO Slave
   Switch the power of the MIO slave on and initialize it.
   For details of the procedure for initializing the slave, see the manual, etc. for the slave you are using.

See the *RZ/T1-M Group MDIO Configuration Library User's Manual (R01UH0577)* for details of the MDIO configuration library.

23.2.4.2 Settings for Regular Operation

Figure 23.10 shows the flow for regular MDIO transfer. Specifically, it shows the settings for regular operation to proceed with MDIO transfer.

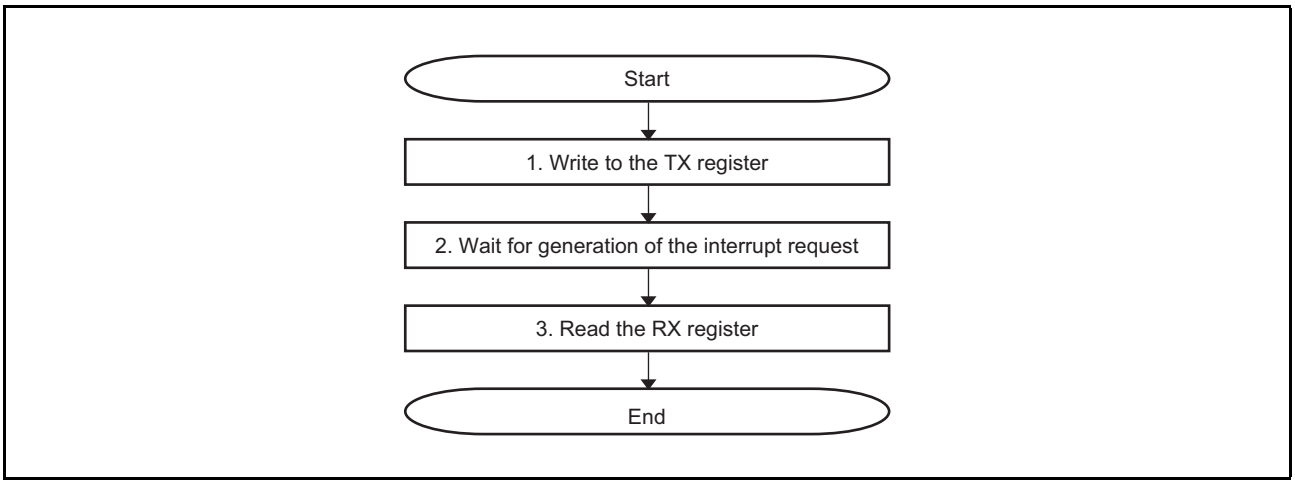


Figure 23.10 Flow for Regular MDIO Transfer

1. Writing to the TX register
   Start MDIO transfer by writing the details of the MDIO transfer to be performed to the TX register.

2. Waiting for generation of the interrupt request
   Wait for generation of the INT_END interrupt request on completion of the MDIO transfer.

3. Reading the RX register
   Read the results of the MDIO transfer from the RX register. When the RX register is read, the INT_END interrupt is also deasserted.

23.2.4.3 Settings for Consecutive Operations

Figure 23.11 shows the flow of operations for consecutive MDIO transfer. Specifically, it shows the settings for operations to proceed with n consecutive rounds of MDIO transfer.

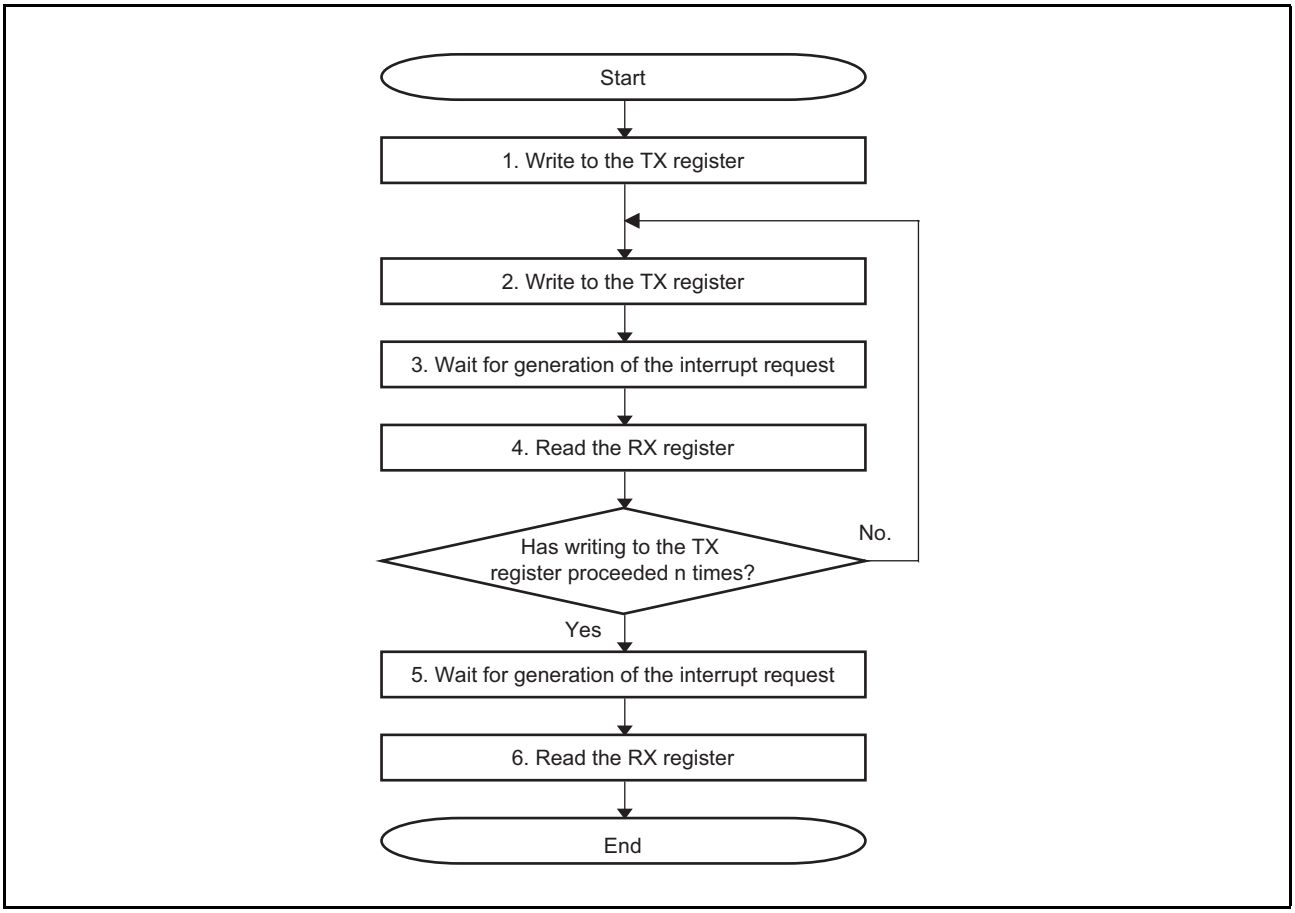


Figure 23.11 Flow for n Consecutive Rounds of MDIO Transfer

1. Writing to the TX register
   Start MDIO transfer by writing the details of the first round of MDIO transfer to the TX register.

2. Writing to the TX register
   Writing the details of the MDIO transfer to be performed the i+1th time during the ith MDIO transfer (where i is a value from 1 to n -1) leads to the i+1th MDIO transfer starting after the completion of the ith MDIO transfer.

3. Waiting for generation of the interrupt request
   Wait for generation of the INT_END interrupt request on completion of the ith MDIO transfer.

4. Reading the RX register
   Read the results of the ith MDIO transfer from the RX register. When the RX register is read, the INT_END interrupt is also deasserted. If writing to the TX register has not proceeded n times, return to step 2.

5. Waiting for generation of the interrupt request
   Wait for generation of the INT_END interrupt request on completion of the nth MDIO transfer.

6. Reading the RX register
   Read the results of the nth MDIO transfer from the RX register. When the RX register is read, the INT_END interrupt is also deasserted.

24. Serial Communications Interface with FIFO (SCIFA)

This LSI has four channels of serial communication interface (SCIFA) with FIFO that support both asynchronous and clock synchronous serial communication. The SCIFA has 16-stage FIFO buffers for transmission and reception, respectively, for each channel that enable this LSI to perform efficient high-speed continuous communication.

24.1 Overview

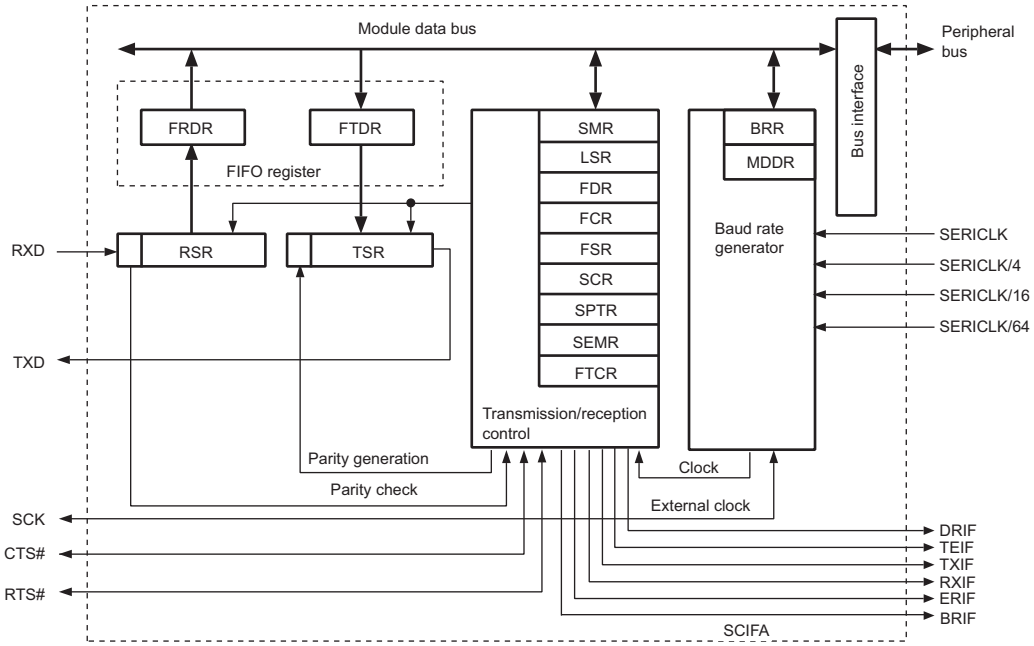
Table 24.1 lists the specifications of the SCIFA.

Table 24.1 Specifications of SCIFA

Item		Description
Channel		4 channels
Serial communication method		Asynchronous communication mode and clock synchronous communication mode*2
Transfer speed		Selectable bit rate with an on-chip baud rate generator
Full duplex communication		Transmitting section: realizes continuous data transmission using 16-stage FIFO buffer Receiving section: realizes continuous data reception using 16-stage FIFO buffer
Data transmission		Selectable either LSB-first or MSB-first transfer
Interrupt source		The following six sources: • Transmit-end (TEIF) • Transmit-FIFO-data-empty (TXIF) • Receive-FIFO-data-full (RXIF) • Receive-data-ready (DRIF)*1 • Framing error or parity error (ERIF) • Break or overrun (BRIF)
Asynchronous communication mode	Character length	7 or 8 bits
	Transmission stop bit length	1 or 2 bits
	Parity	Even, odd, or none
	Receive error detection	Detects following errors as receive error: parity error, overrun error, and framing error
	Hardware flow control	Controls data transmission and reception using the CTS# and RTS# pins.
	Break detection	Break signal detection function by hardware.
	Clock source	Selectable from internal or external clock
	Noise cancellation	Incorporates a digital noise filter in the RXD pin input path.
Clock synchronous communication mode	Character length	8 bits
	Receive error detection	Detects an overrun error as a receive error.
	Clock source	Selectable either internal or external clock
Bit rate modulation		Enables errors to be decreased by correcting the output of the on-chip baud rate generator.

Note 1. Effective only for asynchronous communication mode
Note 2. Channel 4 is used only in asynchronous mode, and does not support the clock synchronous mode.

Figure 24.1 shows a block diagram of the SCIFA.



RSR:	Receive shift register	BRR:	Bit rate register
FRDR:	Receive FIFO data register	SPTR:	Serial port register
TSR:	Transmit shift register	FCR:	FIFO control register
FTDR:	Transmit FIFO data register	FDR:	FIFO data count register
SMR:	Serial mode register	LSR:	Line status register
SCR:	Serial control register	SEMR:	Serial extended mode register
FSR:	Serial status register	FTCR:	FIFO trigger control register
MDDR:	Modulation duty register		

Figure 24.1 Block Diagram of SCIFA

Table 24.2 lists the input/output pins of the SCIFA.

Table 24.2 Pin Configuration of the SCIFA

Item	Pin Name	I/O	Function
Serial clock pin	SCK	I/O	Transmission/reception clock input/output, general output
Receive data pin	RXD	Input	Receive data input
Transmit data pin	TXD	Output	Transmit data output
Transmission/reception start control pin	CTS#	I/O	Input for hardware flow control (transmission enable signal) / general output
	RTS#	Output	Output for hardware flow control (transmission request signal) / general output

Note: Channels of each pin is omitted.

24.2 Register Descriptions

24.2.1 Receive Shift Register (RSR)

The RSR register receives serial data and temporally stores the data. The SCIFA stores the serial data input via the RXD pin into the RSR register. When one byte of data has been received, it is automatically transferred to the receive FIFO data register (FRDR).
The CPU cannot read from or write to the RSR register directly.

	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0

24.2.2 Receive FIFO Data Register (FRDR)

The FRDR register is an 8-bit, 16-stage FIFO register that stores the received serial data. When the SCIFA receives one byte of serial data, it transfers the received data from the receive shift register (RSR) to the FRDR register and completes the receive operation. Continuous reception is possible until the received 16 bytes of data are stored. If the FRDR register is read when there is no received data in the FRDR register, an undefined value is read.
When the FRDR register is full of received data, subsequently received serial data is lost.
The CPU can read the FRDR register but cannot write to it.

Address(es): SCIFA0.FRDR A006 500Ah, SCIFA1.FRDR A006 540Ah, SCIFA2.FRDR A006 580Ah, SCIFA4.FRDR A006 600Ah

	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0

24.2.3 Transmit Shift Register (TSR)

The SCIFA transfers the transmit data from the transmit FIFO data register (FTDR) to the TSR register, and then transmits the data serially to the TXD pin. After transmitting one byte of data, the SCIFA automatically transfers the next transmit data from the FTDR register into the TSR register and starts transmission.
The CPU cannot read from or write to the TSR register directly.

	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1

24.2.4 Transmit FIFO Data Register (FTDR)

The FTDR register is an 8-bit, 16-stage FIFO register that stores serial transmission data. When the SCIFA detects that the transmit shift register (TSR) is empty, it transmits data written in the FTDR register to the TSR register and starts serial transmission. Continuous serial transmission is executed until there is no transmit data left in the FTDR register. Writing the transmit data to the FTDR register should be done when a transmit data empty interrupt (TXIF) request is generated.

When the FTDR register becomes full of transmit data (16 bytes), no more data can be written. Even if new data is written, the data is ignored.

CPU can read from the FTDR register but cannot write to it.

Address(es): SCIFA0.FTDR A006 5006h, SCIFA1.FTDR A006 5406h, SCIFA2.FTDR A006 5806h, SCIFA4.FTDR A006 6006h

	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1

24.2.5 Serial Mode Register (SMR)

The SMR register specifies the SCIFA serial communication format and selects the clock source for the baud rate generator.

The CPU can always read and write to the SMR register.

Address(es): SCIFA0.SMR A006 5000h, SCIFA1.SMR A006 5400h, SCIFA2.SMR A006 5800h, SCIFA4.SMR A006 6000h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	CM	CHR	PE	PM	STOP	—	CKS[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	CKS[1:0]	Clock Select	b1b0 0 0: 1 × SERICLK*1 0 1: 1/4 × SERICLK*1 1 0: 1/16 × SERICLK*1 1 1: 1/64 × SERICLK*1	R/W
b2	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b3	STOP	Stop Bit Length	0: One stop bit 1: Two stop bits	R/W
b4	PM	Parity Mode	0: Even parity 1: Odd parity	R/W
b5	PE	Parity Enable	0: Parity bit addition or check is disabled. 1: Parity bit addition or check is enabled.	R/W
b6	CHR	Character Length	0: 8-bit data 1: 7-bit data*2	R/W
b7	CM	Communication Mode	0: Asynchronous mode 1: Clock synchronous mode	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. SERICLK: Peripheral clock

Note 2. When 7-bit data is selected, the MSB (bit 7) of the transmit FIFO data register is not transmitted.

CKS[1:0] Bits (Clock Select)

Select an internal clock source for the on-chip baud rate generator. For further information on the clock source, bit rate register settings, and baud rates, see section 24.2.8, Bit Rate Register (BRR).

STOP Bit (Stop Bit Length)

Selects one bit or two bits as the stop bit length in asynchronous mode. This setting is used only in asynchronous mode. It is ignored in clock synchronous mode because no stop bits are added. When receiving, 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, but if the second stop bit is 0, it is treated as the start bit of the next incoming character.

Note: When transmitting with one stop bit, a single 1 bit (stop bit) is added at the end of each transmission character.

Note: When transmitting with two stop bits, two 1 bits (stop bits) are added at the end of each transmission character.

PM Bit (Parity Mode)

Selects either the even or odd parity check. The setting of this bit is effective only when the parity enable (PE) bit of this register is set to 1 in asynchronous mode. The setting of this bit is ignored in clock synchronous mode, or when parity addition/check is disabled in asynchronous mode.

Note: If even parity is selected, the parity bit is added to data to be transmitted to make the total number of 1s even in the transmission character and parity bit combined. When receiving, the SCIFA verifies that the total number of 1s in the received character and parity bit combined is even.

Note: If odd parity is selected, the parity bit is added to data to be transmitted to make the total number of 1s odd in the transmission character and parity bit combined. When receiving, the SCIFA verifies that the total number of 1s in the received character and parity bit combined is odd.

PE Bit (Parity Enable)

Selects whether to add a parity bit on data transmission and whether to enable/disable the parity check on data reception in asynchronous mode. In clock synchronous mode, a parity bit is neither added nor checked, regardless of the setting of this bit.

Note: When this bit is set to 1, an even or odd parity bit specified in the PM bit is added to data to be transmitted. The SCIFA verifies whether the parity bit of the received data is even or odd as specified in the PM bit when receiving.

CHR Bit (Character Length)

Selects 7- or 8-bit data length in asynchronous mode. In clock synchronous mode, the data length is always 8 bits, regardless of the CHR setting.

CM Bit (Communication Mode)

Selects whether the SCIFA operates in asynchronous or clock synchronous mode.

24.2.6 Serial Control Register (SCR)

The SCR register enables or disables the SCIFA transmission/reception and interrupt requests, and selects the transmit/receive clock source. The CPU can always read from and write to the SCR register.

Address(es): SCIFA0.SCR A006 5004h, SCIFA1.SCR A006 5404h, SCIFA2.SCR A006 5804h, SCIFA4.SCR A006 6004h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	TIE	RIE	TE	RE	REIE	TEIE	CKE[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	CKE[1:0]	Clock Enable	In asynchronous mode: b1 b0 0 0: Internal clock or SCK pin is used for input pin (input signal is ignored). The SCK pin state depends on the SCKIO and SCKDT bits in SPTR. 0 1: Internal clock or SCK pin is used for clock output (The output clock frequency is 16 or 8 times of the bit rate). 1 0: External clock or SCK pin is used for clock input (The input clock frequency is 16 or 8 times of the bit rate). 1 1: Setting prohibited In clock synchronous mode*2: b1 b0 0 0: Internal clock or SCK pin is used for synchronous clock output. 0 1: Internal clock or SCK pin is used for synchronous clock output. 1 0: External clock or SCK pin is used for synchronous clock input. 1 1: Setting prohibited	R/W
b2	TEIE*1	Transmit End Interrupt Enable	0: Transmit end interrupt (TEIF) request is disabled. 1: Transmit end interrupt (TEIF) request is enabled.	R/W
b3	REIE	Receive Error Interrupt Enable	0: Receive-error interrupt (ERIF) and break interrupt (BRIF) requests are disabled. 1: Receive-error interrupt (ERIF) and break interrupt (BRIF) requests are enabled.	R/W
b4	RE	Receive Enable	0: Data reception is disabled. 1: Data reception is enabled.	R/W
b5	TE	Transmit Enable	0: Data transmission is disabled. 1: Data transmission is enabled.	R/W
b6	RIE	Receive Interrupt Enable	0: Receive-FIFO-data-full interrupt (RXIF), receive-data ready interrupt (DRIF), receive-error interrupt (ERIF), and break interrupt (BRIF) requests are disabled. 1: Receive-FIFO-data-full interrupt (RXIF), receive-data ready interrupt (DRIF), receive-error interrupt (ERIF), and break interrupt (BRIF) requests are enabled.	R/W
b7	TIE	Transmit Interrupt Enable	0: Transmit-FIFO-data-empty interrupt request (TXIF) is disabled. 1: Transmit-FIFO-data-empty interrupt request (TXIF) is enabled.	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. TEIF interrupt requests can be cleared by reading 1 from the TEND flag, and then clearing the setting to 0, or by setting the TEIE bit to 0.

Note 2. Channel 4 does not support the clock synchronous mode.

CKE[1:0] Bits (Clock Enable)

Select the SCIFA clock source and enable or disable clock output from the SCK pin. Depending on the settings of these bits, the SCK pin can be used for serial clock output or serial clock input. If the SCK pin is set for the synchronous clock output in the clock synchronous mode, set the CM bit in the SMR register to 1, and then set the CKE[1:0] bits. The settings of the CKE[1:0] bits are listed in Table 24.15.

REIE Bit (Receive Error Interrupt Enable)

Specifies whether to enable or disable a receive-error interrupt (ERIF) request and a break interrupt (BRIF) request. The setting of this bit is only valid when the RIE bit is set to 0.

Note: ERIF interrupt requests can be cleared by reading 1 from the ER bit in the FSR register, and then clearing the setting to 0, or by clearing both the RIE and REIE bits in this register to 0. BRIF interrupt requests can be cleared by reading 1 from the BRK bit in the FSR register, or from the ORER flag in the LSR register, and then clearing the setting to 0, or by clearing both the RIE and REIE bits in this register to 0.

RE Bit (Receive Enable)

Specifies whether to enable or disable the serial data reception.

Note: Setting this bit to 0 does not affect the receive flags (DR, ER, BRK, RDF, FER, and PER in the FSR register, and ORER in the LSR register). These flags retain their previous values.

Note: Serial reception starts when a start bit is detected in asynchronous mode, or a synchronous clock input is detected in clock synchronous mode. Before setting this bit to 1, be sure to set the serial mode register (SMR) and the FIFO control register (FCR) to select the receive format and reset the receive FIFO.

TE Bit (Transmit Enable)

Specifies whether to enable or disable the serial data transmission.

Note: Serial transmission starts after writing of data to be transmitted into the FTDR register under this condition. Before setting this bit to 1, be sure to set the serial mode register (SMR) and the FIFO control register (FCR) to select the transmit format and reset the transmit FIFO.

RIE Bit (Receive Interrupt Enable)

Specifies whether to enable or disable a receive-FIFO-data-full (RXIF) interrupt request when the RDF flag in the serial status register (FSR) is set to 1, a receive-data ready (DRIF) interrupt request when the DR flag in the FSR register is set to 1, a receive-error (ERIF) interrupt request when the ER flag in the FSR register is set to 1, and a break (BRIF) interrupt request when the BRK flag in the FSR register or the ORER flag in the line status register (LSR) is set to 1.

Note: RXIF interrupt requests can be cleared by reading 1 from the DR or RDF flag in the FSR register, then clearing the flag to 0, or by clearing the RIE bit to 0. DRIF interrupt requests can be cleared by reading 1 from the DR flag in the FSR register, and then clearing the setting to 0, or by clearing the RIE bit in this register to 0. Receive error interrupt (ERIF) requests and break interrupt (BRIF) requests can be cleared by clearing both the RIE and REIE bits in this register to 0.

TIE Bit (Transmit Interrupt Enable)

Specifies whether to enable or disable a transmit-FIFO-data-empty interrupt (TXIF) request when the serial transmit data is transferred from the transmit FIFO data register (FTDR) into the transmit shift register (TSR), the quantity of data in the transmit FIFO data register falls below the specified trigger number for transmission, and the TDFE flag in the serial status register (FSR) is set to 1.

Note: TXIF interrupt requests can be cleared either by writing a greater quantity of transmit data than the specified transmission trigger number into the FTDR register, reading 1 from the TDFE flag, and then clearing the TDFE flag to 0, or by clearing this bit to 0.

24.2.7 Serial Status Register (FSR)

The FSR register is a 16-bit register. The 8 lower-order bits indicate the status flag representing the SCIFA operating state.

The CPU can always read and write to the FSR register, but cannot write 1 to the status flags (ER, TEND, TDFE, BRK, RDF, and DR bits) in this register. These flags can be only cleared to 0 when they have first been read (after being set to 1). b3 (FER) and b2 (PER) are read-only bits that cannot be written.

Address(es): SCIFA0.FSR A006 5008h, SCIFA1.FSR A006 5408h, SCIFA2.FSR A006 5808h, SCIFA4.FSR A006 6008h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	ER	TEND	TDFE	BRK	FER	PER	RDF	DR
Value after reset:	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	DR	Receive Data Ready Flag	0: Reception is in progress, or no received data has remained in the FRDR register after normally completed receiving. 1: Next receive data has not been received.	R/(W) *1
b1	RDF	Receive FIFO Data Full Flag	0: The quantity of receive data in the FRDR register falls below the specified reception trigger number. 1: The quantity of receive data written in the FRDR register is equal to or greater than the specified reception trigger number.	R/(W) *1
b2	PER	Parity Error Flag	0: No receive parity error occurred in the next receive data read from the FRDR register. 1: A receive parity error occurred in the next receive data read from the FRDR register.	R
b3	FER	Framing Error Flag	0: No receive framing error occurred in the next data read from the FRDR register. 1: A receive framing error occurred in the next data read from the FRDR register.	R
b4	BRK	Break Detect Flag	0: No break signal is received. 1: A break signal is received.*2	R/(W) *1
b5	TDFE	Transmit FIFO Data Empty Flag	0: The quantity of transmit data written in the FTDR register exceeds the specified transmission trigger number. 1: The quantity of transmit data written in the FTDR register is equal to or less than the specified transmission trigger number.*3	R/(W) *1
b6	TEND	Transmit End Flag	0: Transmission is in the waiting state or in progress. 1: Transmission is completed.	R/(W) *1
b7	ER	Receive Error Flag	0: Reception is in progress or has normally completed. 1: A framing error or parity error has occurred during reception.	R/(W) *1
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. 0 can be only written to clear the flag after 1 is read.

Note 2. When a break signal is detected, transfer of the receive data (00h) to the FRDR register stops after the detection. When the break ends and the receive signal becomes mark state (high level), the transfer of receive data resumes.

Note 3. Since the FTDR register is a 16-byte FIFO register, the maximum quantity of data that can be written when TDFE is 1 is “16 minus the number of non-transmitted data units”. If additional data is written, the data is ignored. The quantity of data in the FTDR register is indicated by the 8 higher-order bits of the FDR register.

DR Bit (Receive Data Ready Flag)

Indicates that the quantity of data stored in the receive FIFO data register (FRDR) falls below the specified reception trigger number, and that no next data has been received yet after the elapse of 15 ETUs*1 from the last stop bit in asynchronous mode. In clock synchronous mode, this bit is not set to 1.

[Setting condition]

- DR is set to 1 when the FRDR register contains less data than the specified reception trigger number*2, and no next data has been received yet after the elapse of 15 ETUs*1 from the last stop bit.

[Clearing conditions]

When either of the following is satisfied:

- DR is cleared to 0 when DR = 1 is read and then 0 is written to the DR flag.
- DR is cleared to 0 when all received data in the FRDR register are read.

Note 1. This is equivalent to one and a half (1.5) frames in the 8-bit format with one stop bit (ETU: elementary time unit).
Note 2. The clearing condition takes precedence when all received data in the FRDR register are read.
Note: When the RE bit in SCR is cleared to 0, the DR bit is not affected and retains its previous value.

RDF Bit (Receive FIFO Data Full Flag)

Indicates that receive data has been transferred to the receive FIFO data register (FRDR), and the quantity of data in FRDR becomes equal to or greater than the specified reception trigger number.

[Setting condition]

- RDF is set to 1 when the quantity of receive data which is equal to or greater than the specified reception trigger number are stored in the FRDR register*1.

[Clearing condition]

- RDF is cleared to 0 when the FRDR register is read until the quantity of receive data in the FRDR register falls below the specified reception trigger number after 1 is read from RDF and then 0 is written to this bit.

Note 1. Since the FRDR register is a 16-byte FIFO register, the maximum quantity of data that can be read when this bit is 1 is equivalent to the specified reception trigger number. If an attempt is made to read after all the data in the FRDR register has been read, the read data is undefined. The quantity of receive data in the FRDR register is indicated by the 8 lower-order bits of the FDR register.

PER Bit (Parity Error Flag)

Indicates whether there is a parity error in the data read from the receive FIFO data register (FRDR) in asynchronous mode.

[Setting condition]

- PER is set to 1 when a parity error is present in the next data read from the FRDR register.

[Clearing condition]

- PER is cleared to 0 when no parity error is present in the next data read from the FRDR register.

FER Bit (Framing Error Flag)

Indicates whether there is a framing error in the data read from the receive FIFO data register (FRDR) in asynchronous mode.

[Setting condition]

- FER is set to 1 when a framing error is present in the next data read from the FRDR register.

[Clearing condition]

- FER is cleared to 0 when no framing error is present in the next data read from the FRDR register.

BRK Bit (Break Detect Flag)

Indicates that a break signal has been detected in receive data.

[Setting condition]

- BRK is set to 1 when data including a framing error is received, and the framing error is followed by at least one frame of data received at the space 0 level (low level).

[Clearing condition]

- BRK is cleared to 0 when software reads BRK after it has been set to 1 and then writes 0 to BRK.

TDFE Bit (Transmit FIFO Data Empty Flag)

Indicates that data has been transferred from the transmit FIFO data register (FTDR) into the transmit shift register (TSR), the quantity of data in the FTDR register becomes equal to or less than the specified transmission trigger number, and writing of transmit data to the FTDR register is enabled.

[Setting conditions]

When either of the following is satisfied:

- TDFE is set to 1 when the TE bit in SCR is 0.
- TDFE is set to 1 when the quantity of transmit data written in the FTDR register is equal to or less than the specified transmission trigger number.

[Clearing condition]

- TDFE is cleared to 0 when 0 is written in the TDFE bit after reading TDFE = 1.

TEND Bit (Transmit End Flag)

Indicates that the FTDR register contains no more valid data and transmission is completed when transmitting the last bit of the transmit data.

[Setting condition]

- TEND is set to 1 when the FTDR register does not contain transmit data when the last bit of the serial transmission data is transmitted.

[Clearing conditions]

When either of the following is satisfied:

- When transmit data is written to the FTDR register
- When 0 is written to TEND after it has been read as 1

ER Bit (Receive Error Flag)

Indicates the occurrence of a framing error, or of a parity error when receiving the parity-added data*1.

[Setting conditions]

When either of the following is satisfied:

- ER is set to 1 when the stop bit is found to be 0 after checking whether the stop bit of the received data is 1 at the end of one data receive operation*2.
- ER is set to 1 when the total number of 1s in the received data and parity bit combined does not match the even or odd parity setting specified by the PM bit in the SMR register.

[Clearing condition]

- When 0 is written to ER after it has been read as 1

Note 1. Clearing the RE bit to 0 in the SCR register does not affect this bit, which retains its previous value. Even if a receive error occurs, the receive data is transferred to the FRDR register and the receive operation is continued. Whether the data read from the FRDR register includes a receive error can be detected by the FER and PER bits in the FSR register.

Note 2. In two stop bits mode, only the first stop bit is checked; the second stop bit is not checked.

24.2.8 Bit Rate Register (BRR)

The BRR register is an 8-bit register that, together with the baud rate generator clock source selected by the CKS[1:0] bits in the serial mode register (SMR), determines the serial transmit/receive bit rate.
This register is located in the same address as that of the MDDR register and selected when the MDDRS bit in SEMR is 0. The CPU can read and write to BRR. Writing to BRR should be executed when TE = RE = 0 in the SCR register.

Address(es): SCIFA0.BRR A006 5002h, SCIFA1.BRR A006 5402h, SCIFA2.BRR A006 5802h, SCIFA4.BRR A006 6002h

	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1

The BRR setting is calculated using the following formulae.
[Asynchronous mode]

- When the baud rate generator is in normal mode (SEMR.BGDM = 0):

$$N = \frac{SERICLK}{64 \times 2^{2n-1} \times B} \times 10^6 - 1$$

(When operating on the base clock with a frequency 16 times the bit rate (SEMR.ABCS0 = 0))

$$N = \frac{SERICLK}{32 \times 2^{2n-1} \times B} \times 10^6 - 1$$

(When operating on the base clock with a frequency 8 times the bit rate (SEMR.ABCS0 = 1))

- When the baud rate generator is in double-speed mode (SEMR.BGDM = 1):

$$N = \frac{SERICLK}{32 \times 2^{2n-1} \times B} \times 10^6 - 1$$

(When operating on the base clock with a frequency 16 times the bit rate (SEMR.ABCS0 = 0))

$$N = \frac{SERICLK}{16 \times 2^{2n-1} \times B} \times 10^6 - 1$$

(When operating on the base clock with a frequency 8 times the bit rate (SEMR.ABCS0 = 1))

[Clock synchronous mode]

$$N = \frac{SERICLK}{8 \times 2^{2n-1} \times B} \times 10^6 - 1$$

B: Bit rate (bit/s)
N: Setting of the BRR register (0 ≤ N ≤ 255) (The setting must satisfy the electrical characteristics).
SERICLK: Operating frequency for peripheral modules (MHz)
n: Baud rate generator clock source (n = 0, 1, 2, 3) (For the clock sources and values of n, see Table 24.3).

Note: The MDDR register is used to adjust the bit rate. For details, see section 24.2.9, Modulation Duty Register (MDDR).

Table 24.3 SMR Register Setting

		Setting of SMR.CKS [1:0] Bits	
n	**Clock Source**	**b1**	**b0**
0	SERICLK	0	0
1	SERICLK/4	0	1
2	SERICLK/16	1	0
3	SERICLK/64	1	1

The bit rate error in asynchronous mode is calculated using the following formulae.

- When the baud rate generator is in normal mode (SEMR.BGDM = 0):

$$\text{Error (\%)} = \left\{ \frac{\text{SERICLK} \times 10^6}{(N + 1) \times B \times 64 \times 2^{2n-1}} - 1 \right\} \times 100$$

(When operating on the base clock with a frequency 16 times the bit rate (SEMR.ABCS0 = 0))

$$\text{Error (\%)} = \left\{ \frac{\text{SERICLK} \times 10^6}{(N + 1) \times B \times 32 \times 2^{2n-1}} - 1 \right\} \times 100$$

(When operating on the base clock with a frequency 8 times the bit rate (SEMR.ABCS0 = 1))

- When the baud rate generator is in double-speed mode (SEMR.BGDM = 1):

$$\text{Error (\%)} = \left\{ \frac{\text{SERICLK} \times 10^6}{(N + 1) \times B \times 32 \times 2^{2n-1}} - 1 \right\} \times 100$$

(When operating on the base clock with a frequency 16 times the bit rate (SEMR.ABCS0 = 0))

$$\text{Error (\%)} = \left\{ \frac{\text{SERICLK} \times 10^6}{(N + 1) \times B \times 16 \times 2^{2n-1}} - 1 \right\} \times 100$$

(When operating on the base clock with a frequency 8 times the bit rate (SEMR.ABCS0 = 1))

Table 24.4 list the examples of the BRR register setting in asynchronous mode, and Table 24.5 list the examples of the BRR register setting in clock synchronous mode.

Table 24.4 Bit Rates and BRR Register Settings in Asynchronous Mode

Bit Rate (bps)	SERICLK (MHz)					
	120			150		
	n	N	Error (%)	n	N	Error (%)
150						
300	3	194	0.16	3	243	0.06
600	3	97	-0.35	3	121	0.06
1200	2	194	0.16	2	243	0.06
2400	2	97	-0.35	2	121	0.06
4800	1	194	0.16	1	243	0.06
9600	1	97	-0.35	1	121	0.06
14400	1	64	0.16	1	80	0.47
19200	0	194	0.16	0	243	0.06
28800	0	129	0.16	0	162	-0.15
31250	0	119	0	0	149	0
38400	0	97	-0.35	0	121	0.06
115200				0	40	-0.76
500000						

Note: These values assume bits SEMR.ABCS0 and SEMR.BGDM are both 0. When either the SEMR.ABCS0 bit or SEMR.BGDM bit is set to 1, the bit rate is doubled. When bits SEMR.ABCS0 and SEMR.BGDM are both 1, the bit rate is quadrupled.
Configure settings so the range of error is no greater than 1%.
Values for the blank cells in the table can be set using the MDDR register. For details, see section 24.2.9, Modulation Duty Register (MDDR) and the Table 24.10.

Table 24.5 Bit Rates and BRR Register Settings in Clock Synchronous Mode

Bit Rate (bps)	SERICLK (MHz)			
	120		150	
	n	N	n	N
250				
500				
1000				
2500	3	187	3	233
5000	3	93	3	116
10000	2	187	2	233
25000	2	74	2	93
50000	1	149	1	187
100000	1	74	1	93
250000	0	119	0	149
500000	0	59	0	74
1000000	0	29	0	37
2500000	0	11	0	14
5000000	0	5	0	7

Blank: Setting is prohibited.
Note: Set the BRR register so that the range of error can fall within 1% or less.

Table 24.6 lists the maximum bit rates for various frequencies in asynchronous mode when the baud rate generator is used. Table 24.7 lists the maximum bit rates for various frequencies in clock synchronous mode when the baud rate generator is used. Table 24.8 and Table 24.9 list the maximum rates for external clock inputs in asynchronous mode and clock synchronous mode, respectively.

Table 24.6 Maximum Bit Rates for Various Frequencies with Baud Rate Generator (in Asynchronous Mode)

SERICLK (MHz)	Maximum Bit Rate (bit/s)	Settings	
		n	N
120	15000000	0	0
150	18750000	0	0

Note: These values assume bits SEMR.ABCS0 and SEMR.BGDM are both 1. When either the SEMR.ABCS0 bit or SEMR.BGDM bit is set to 1, the bit rate is 1/2. When bits SEMR.ABCS0 and SEMR.BGDM are both 0, the bit rate is 1/4.

Table 24.7 Maximum Bit Rates for Various Frequencies with Baud Rate Generator (in Clock Synchronous Mode)

SERICLK (MHz)	Discontinuous Transmission/Reception			Continuous Transmission/Reception		
	Maximum Bit Rate (bit/s)	Settings		Maximum Bit Rate (bit/s)	Settings	
		n	N		n	N
120	30000000	0	0	15000000	0	1
150	37500000	0	0	18750000	0	1

Table 24.8 Maximum Bit Rates with External Clock Input (in Asynchronous Mode)

SERICLK (MHz)	External Input Clock (MHz)	Maximum Bit Rate (bit/s)
120	30	3750000
150	37.5	4687500

Note: This is an example when the SEMR.ABCS0 bit is 1. When the ABCS0 bit is set to 0, the bit rate is 1/2 .

Table 24.9 Maximum Bit Rates with External Clock Input (in Clock Synchronous Mode)

SERICLK (MHz)	External Input Clock (MHz)	Maximum Bit Rate (bit/s)
120	10	10000000
150	12.5	12500000

24.2.9 Modulation Duty Register (MDDR)

The MDDR register corrects the bit rate adjusted by the BRR register. The value after reset of this register is FFh. When the BRME bit in SEMR is set to 1, the bit rate generated by the on-chip baud rate generator is evenly corrected according to the settings of MDDR (MDDR/256). The relationship between the MDDR register setting and the bit rate (B) is given by the following formula. The MDDR register is located in the same address as that of the BRR register and is selected when the MDDRS bit in SEMR is 1. This register is only writable when TE = RE = 0 in the SCR register. b7 in this register is fixed to 1.

Address(es): SCIFA0.MDDR A006 5002h, SCIFA1.MDDR A006 5402h, SCIFA2.MDDR A006 5802h, SCIFA4.MDDR A006 6002h

	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1

The formulae below show the relationships between the MDDR setting and the bit rate (B) when the bit rate modulation function is used.

[Asynchronous mode]

- When the baud rate generator is in normal mode (SEMR.BGDM = 0):

$$B = \frac{SERICLK \times 10^6}{64 \times 2^{2n-1} \times (256/MDDR) \times (N + 1)}$$

(When operating on the base clock with a frequency 16 times the bit rate (SEMR.ABCS0 = 0))

$$B = \frac{SERICLK \times 10^6}{32 \times 2^{2n-1} \times (256/MDDR) \times (N + 1)}$$

(When operating on the base clock with a frequency 8 times the bit rate (SEMR.ABCS0 = 1))

- When the baud rate generator is in double-speed mode (SEMR.BGDM = 1):

$$B = \frac{SERICLK \times 10^6}{32 \times 2^{2n-1} \times (256/MDDR) \times (N + 1)}$$

(When operating on the base clock with a frequency 16 times the bit rate (SEMR.ABCS0 = 0))

$$B = \frac{SERICLK \times 10^6}{16 \times 2^{2n-1} \times (256/MDDR) \times (N + 1)}$$

(When operating on the base clock with a frequency 8 times the bit rate (SEMR.ABCS0 = 1))

[Clock synchronous mode]

$$B = \frac{SERICLK \times 10^6}{8 \times 2^{2n-1} \times (256/MDDR) \times (N + 1)}$$

When the bit rate modulation is used, the bit rate average error is given by the following formulae.

- When the baud rate generator is in normal mode (SEMR.BGDM = 0):

$$\text{Error (\%)} = \left\{ \frac{\text{SERICLK} \times 10^6}{B \times 64 \times 2^{2n-1} \times (256/\text{MDDR}) \times (N+1)} - 1 \right\} 100$$

(When operating on the base clock with a frequency 16 times the bit rate (SEMR.ABCS0 = 0))

$$\text{Error (\%)} = \left\{ \frac{\text{SERICLK} \times 10^6}{B \times 32 \times 2^{2n-1} \times (256/\text{MDDR}) \times (N+1)} - 1 \right\} 100$$

(When operating on the base clock with a frequency 8 times the bit rate (SEMR.ABCS0 = 1))

- When the baud rate generator is in double-speed mode (SEMR.BGDM = 1):

$$\text{Error (\%)} = \left\{ \frac{\text{SERICLK} \times 10^6}{B \times 32 \times 2^{2n-1} \times (256/\text{MDDR}) \times (N+1)} - 1 \right\} 100$$

(When operating on the base clock with a frequency 16 times the bit rate (SEMR.ABCS0 = 0))

$$\text{Error (\%)} = \left\{ \frac{\text{SERICLK} \times 10^6}{B \times 16 \times 2^{2n-1} \times (256/\text{MDDR}) \times (N+1)} - 1 \right\} 100$$

(When operating on the base clock with a frequency 8 times the bit rate (SEMR.ABCS0 = 1))

B:	Bit rate (bits/s)
N:	BRR register setting (0 ≤ N ≤ 255) (The setting must satisfy the electrical characteristics).
SERICLK:	Operating frequency for peripheral modules (MHz)
MDDR:	MDDR setting (128 ≤ MDDR ≤ 255)
n:	Baud rate generator clock source (n = 0, 1, 2, 3) (For the clock sources and values of n, see Table 24.3).

Table 24.10 Bit Rates and Settings of BRR and MDDR Registers in Asynchronous Mode

Bit Rate (bps)	SERICLK (MHz)							
	120				150			
	n	N	MDDR	Error (%)	n	N	MDDR	Error (%)
150	3	205	135	-0.003	3	247	130	-0.018
300	3	176	232	0.001	3	205	216	-0.003
600	2	205	135	-0.003	3	102	216	-0.003
1200	2	176	232	0.001	2	205	216	-0.003
2400	1	205	135	-0.003	2	102	216	-0.003
4800	1	176	232	0.001	1	205	216	-0.003
9600	0	205	135	-0.003	1	102	216	-0.003
14400	0	176	174	0.001	0	205	162	-0.003
19200	0	176	232	0.001	0	205	216	-0.003
28800	0	117	232	0.001	0	102	162	-0.003
31250	0	59	128	0.000	0	74	128	0.000
38400	0	73	194	0.007	0	102	216	-0.003
115200	0	21	173	-0.009	0	23	151	0.003
500000	0	6	239	0.028	0	6	151	-0.077

Note: These values assume bits SEMR.ABCS0 and SEMR.BGDM are both 0. When either the SEMR.ABCS0 bit or SEMR.BGDM bit is set to 1, the bit rate is doubled. When bits SEMR.ABCS0 and SEMR.BGDM are both 1, the bit rate is quadrupled. Configure settings so the range of error is no greater than 1%.

24.2.10 FIFO Control Register (FCR)

The FCR register resets the quantity of data in the transmit FIFO data register (FTDR) and the receive FIFO data register (FRDR) and specifies the number of triggers. This register also specifies whether to enable the loop-back test.
The CPU can always read and write to the FCR register.

Address(es): SCIFA0.FCR A006 500Ch, SCIFA1.FCR A006 540Ch, SCIFA2.FCR A006 580Ch, SCIFA4.FCR A006 600Ch

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	RSTRG[2:0]			RTRG[1:0]		TTRG[1:0]		MCE	TFRST	RFRST	LOOP
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	LOOP	Loop-Back Test	0: Loop back test is disabled. 1: Loop back test is enabled.	R/W
b1	RFRST	Receive FIFO Data Register Reset	0: Normal operation 1: Resets the FRDR register.	R/W
b2	TFRST	Transmit FIFO Data Register Reset	0: Normal operation 1: Resets the FTDR register.	R/W
b3	MCE	Modem Control Enable	0: Modem signal is disabled.*1 1: Modem signal is enabled.	R/W
b5, b4	TTRG[1:0]	Transmit FIFO Data Trigger Number Select	b5 b4 0 0: 8 (8)*2 0 1: 4 (12)*2 1 0: 2 (14)*2 1 1: 0 (16)*2	R/W
b7, b6	RTRG[1:0]	Receive FIFO Data Trigger Number Select	In asynchronous mode: b7 b6 0 0: 1 0 1: 4 1 0: 8 1 1: 14 In clock synchronous mode: b7 b6 0 0: 1 0 1: 2 1 0: 8 1 1: 14	R/W
b10 to b8	RSTRG[2:0]	RTS# Output Active Trigger Number Select	b10 b8 0 0 0: 15 0 0 1: 1 0 1 0: 4 0 1 1: 6 1 0 0: 8 1 0 1: 10 1 1 0: 12 1 1 1: 14	R/W
b15 to b11	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. The CTS# input level does not affect the transmit operation. Similarly, the RTS# input level does not affect the receive operation.
Note 2. Values in parentheses mean the number of empty bytes in the FTDR register when the TDFE flag is set to 1 and a transmit FIFO data empty interrupt (TXIF) request is generated.

LOOP Bit (Loop-Back Test)

Internally connects between the transmit output pin (TXD) and the receive input pin (RXD) and between the RTS# pin and the CTS# pin, to perform loop-back testing.

RFRST Bit (Receive FIFO Data Register Reset)

Disables the receive data in the receive FIFO data register (FRDR) and makes the data to the empty state. If you set this bit to 1, be sure to clear it to 0 afterward.

TFRST Bit (Transmit FIFO Data Register Reset)

Disables the transmit data in the transmit FIFO data register (FTDR) and makes the data to the empty state. If you set this bit to 1, be sure to clear it to 0 afterward.

MCE Bit (Modem Control Enable)

Specifies whether to enable or disable the modem control signals, CTS# and RTS#.
In clock synchronous mode, this bit should always be set to 0.

TTRG[1:0] Bits (Transmit FIFO Data Trigger Number Select)

Specify the reference quantity of data for transmission (i.e., the threshold number of entries to trigger the writing of further data for transmission) for setting of the TDFE flag in the serial status register (FSR). When the number of entries for transmission in the transmission FIFO, i.e. the number of entries written to the transmit FIFO data register (FTDR) that are yet to be transmitted, falls to or below the number specified by the TTRG[1:0] bits, the TDFE flag is set to 1 and a transmit FIFO data empty interrupt (TXIF) request is generated.
The setting in these bits is valid when the TTRGS bit in the FTCR register is 0. When the TTRGS bit in the FTCR register is 1, the setting of the TFTC[4:0] bits in the FTCR register is valid.

RTRG[1:0] Bits (Receive FIFO Data Trigger Number Select)

Specify the reference quantity of receive data (i.e., the threshold number of entries to trigger the reading of received data) for setting of the RDF flag in the serial status register (FSR). When the number of entries in the reception FIFO, i.e. the number of entries yet to be read from the receive FIFO data register (FRDR), rises to or above the number specified by the RTRG[1:0] bits, the RDF flag is set to 1 and a receive FIFO data full interrupt (RXIF) request is generated.
The setting in these bits is valid when the RTRGS bit in the FTCR register is 0. When the RTRGS bit in the FTCR register is 1, the setting of the RFTC[4:0] bits in the FTCR register is valid.

RSTRG[2:0] Bits (RTS# Output Active Trigger Number Select)

When the number of entries in the reception FIFO, i.e. the number of entries yet to be read from the receive FIFO data register (FRDR), rises to or above the trigger number specified by the RSTRG[2:0] bits, the RTS# signal is in the high state.
The setting in these bits is only valid when a modem signal is enabled by the MCE bit in this register in asynchronous mode.

24.2.11 FIFO Data Count Register (FDR)

The FDR register indicates the quantity of data stored in the transmit FIFO data register (FTDR) and the receive FIFO data register (FRDR).

This register indicates the quantity of transmit data in the FTDR register with the 8 higher-order bits, and the quantity of receive data in the FRDR register with the 8 lower-order bits. The CPU can always read the FDR register but cannot write to it.

Address(es): SCIFA0.FDR A006 500Eh, SCIFA1.FDR A006 540Eh, SCIFA2.FDR A006 580Eh, SCIFA4.FDR A006 600Eh

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	T[4:0]					—	—	—	R[4:0]				
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b4 to b0	R[4:0]	Receive Data Quantity in FRDR	Indicate the quantity of receive data stored in the FRDR register.	R
b7 to b5	—	Reserved	These bits are read as 0.	R
b12 to b8	T[4:0]	Non-Transmitted Data Quantity in FTDR	Indicate the quantity of non-transmitted data stored in the FTDR register.	R
b15 to b13	—	Reserved	These bits are read as 0.	R

R[4:0] Bits

Indicate the quantity of receive data stored in the FRDR register.

00h means no received data, and 10h means that all of the received data is stored in the FRDR register.

T[4:0] Bits

Indicate the quantity of non-transmitted data stored in the FTDR register.

00h means no transmit data, and 10h means that all of the data for transmission is stored in the FTDR register.

24.2.12 Serial Port Register (SPTR)

The SPTR register controls input/output and data of the pins multiplexed to SCIFA function.
The CPU can always read and write to the SPTR register.

Note: b6, b4, b2, and b0 of this register respectively indicate the input status of their corresponding pins. See the descriptions for each bit for details. Writings to these bits in 1-bit unit are handled as read-modify-write, which may lead to undesired values to be written. To avoid this, when modifying the SPB2DT or SPB2IO bit, for example, write the other bit (the bit used in combination) at the same time.

Address(es): SCIFA0.SPTR A006 5010h, SCIFA1.SPTR A006 5410h, SCIFA2.SPTR A006 5810h, SCIFA4.SPTR A006 6010h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	RTS2IO	RTS2DT	CTS2IO	CTS2DT	SCKIO	SCKDT	SPB2IO	SPB2DT
Value after reset:	0	0	0	0	0	0	0	0	0	x	0	x	0	x	0	x

x: Undefined

Bit	Symbol	Bit Name	Description	R/W
b0	SPB2DT	Serial Port Break Data Select	Controls the TXD pin in combination with the TE bit in the SCR register and the SPB2IO bit. See Table 24.13.	R/W
b1	SPB2IO	Serial Port Break Input/Output	Controls the TXD pin in combination with the TE bit in the SCR register and the SPB2DT bit.	R/W
b2	SCKDT	SCK Port Data Select	Controls the SCK pin in combination with the CM bit in the SMR register, the SCKIO bit, and the CKE1 and CKE0 bits in the SCR register. See Table 24.15.	R/W
b3	SCKIO	SCK Port Input/Output	Controls the SCK pin in combination with the CM bit in the SMR register, the SCKDT bit, and the CKE1 and CKE0 bits in the SCR register. See Table 24.15.	R/W
b4	CTS2DT	CTS# Port Data Select	Controls the CTS# pin in combination with MCE bit in FCR and CTS2IO bit. See Table 24.12.	R/W
b5	CTS2IO	CTS# Port Output Specify		R/W
b6	RTS2DT	RTS# Port Data Select	Controls the RTS# pin in combination with MCE bit in FCR and RTS2IO bit. See Table 24.11.	R/W
b7	RTS2IO	RTS# Port Output Specify		R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

SPB2DT Bit (Serial Port Break Data Select)

This bit specifies the output level of the TXD pin when the setting of the SCR.TE bit is 0. The RXD pin input status can be read from this bit regardless of the SPB2IO bit setting. However, the RXD pin function must have been selected with the multi-function pin controller (MPC).

SPB2IO Bit (Serial Port Break Input/Output)

Controls the TXD pin in combination with the TE bit in the SCR register and the SPB2DT bit.

SCKDT Bit (SCK Port Data Select)

The SCK pin status can be read from this bit regardless of the SCKIO bit setting. (When the SCK pin is used for input, the input signal is invalid (has no means) but the pin status can be read.) However, the SCK pin function must have been selected with the multi-function pin controller (MPC).

SCKIO Bit (SCK Port Input/Output)

Specifies input or output status of the SCK pin. This bit controls the SCK pin in combination with the SCKDT bit, the CM bit in the SMR register, and the CKE1 and CKE0 bits in the SCR register.

CTS2DT Bit (CTS# Port Data Select)

The status of the CTS# pin can be read from this bit regardless of the CTS2IO bit setting. However, the CTS# pin function must have been selected with the multi-function pin controller (MPC).

RTS2DT Bit (RTS# Port Data Select)

The status of the RTS# pin can be read from this bit regardless of the RTS2IO bit setting. However, the RTS# pin function must have been selected with the multi-function pin controller (MPC).

Table 24.11 RTS# Pin Status

FCR.MCE Bit Setting	RTS2IO Bit Setting	RTS2DT Bit Setting	RTS# Pin Status
0	0	×	Setting prohibited*1
0	1	0	Low output
0	1	1	High output
1	×	×	Modem control output

×: Don't care

Note 1. There is no problem with the initial setting if the RTS# pin is not used.

Table 24.12 CTS# Pin Status

FCR.MCE Bit Setting	CTS2IO Bit Setting	CTS2DT Bit Setting	CTS# Pin Status
0	0	×	Setting prohibited*1
0	1	0	Low output
0	1	1	High output
1	×	×	Modem control input

×: Don't care

Note 1. There is no problem with the initial setting if the CTS# pin is not used.

Table 24.13 TXD Pin Status

SCR.TE Bit Setting	SPB2IO Bit Setting	SPB2DT Bit Setting	TXD Pin Status
0	0	×	Setting prohibited
0	1	0	Low output
0	1	1	High output
1	×	×	Transmit data output

×: Don't care

24.2.13 Line Status Register (LSR)

The LSR register is a 16-bit register. The PER and FER bits indicate the number of receive errors in the receive FIFO data register. 1 cannot be written to the ORER status flag. The flag should be read as 1 prior to clearing it to 0.

Address(es): SCIFA0.LSR A006 5012h, SCIFA1.LSR A006 5412h, SCIFA2.LSR A006 5812h, SCIFA4.LSR A006 6012h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	PER[3:0]				—	—	FER[3:0]				—	ORER
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ORER	Overrun Error Flag	0: Reception is in progress or has normally completed. 1: An overrun error has occurred during reception.	R/(W) *1
b1	—	Reserved	This bit is read as 0.	R
b5 to b2	FER[3:0]	Framing Error Count	Indicates the quantity of data with a framing error among the receive data stored in the receive FIFO data register (FRDR).	R
b7, b6	—	Reserved	These bits are read as 0.	R
b11 to b8	PER[3:0]	Parity Error Count	Indicates the quantity of data with a parity error among the receive data stored in the receive FIFO data register (FRDR).	R
b15 to b12	—	Reserved	These bits are read as 0.	R

Note 1. To clear the flag, 0 can be only written after 1 is read.

ORER Bit (Overrun Error Flag)

Indicates that receive operation abnormally stops due to occurrence of an overrun error. This flag is not affected and retains its previous state if the RE bit in the serial control register (SCR) is cleared to 0. The receive FIFO data register (FRDR) retains the data before an overrun error occurred, and newly received data is lost. When the ORER bit is set to 1, the SCIFA cannot continue subsequent serial reception.

[Setting condition]

- When the next serial reception is completed with the receive FIFO in full state (16-byte data is received)

[Clearing condition]

- When 0 is written to ORER after being read as 1.

Note: When the internal clock is selected while the SCIFA is in clock synchronous mode, the amount of receive data can be controlled, so no overrun occurs.

FER[3:0] Bits (Framing Error Count)

The values of bits 5 to 2 indicate the quantity of data with a framing error after the ER bit in the FSR register is set. Reading 0000 from the FER[3:0] bits means all 16-byte receive data in the FRDR register have a framing error.

PER[3:0] Bits (Parity Error Count)

The values of bits 11 to 8 indicate the quantity of data with a parity error after the ER bit in the FSR register is set. Reading 0000 from the PER[3:0] bits means all 16-byte receive data in the FRDR register have a parity error.

24.2.14 Serial Extended Mode Register (SEMR)

The SEMR register specifies either LSB or MSB first, enables the noise cancellation, operation in normal or double-speed mode of the baud rate generator, and bit rate modulation, and selects the modulation register and the sampling count (either 8 or 16 times).

Address(es): SCIFA0.SEMR A006 5014h, SCIFA1.SEMR A006 5414h, SCIFA2.SEMR A006 5814h, SCIFA4.SEMR A006 6014h

	b7	b6	b5	b4	b3	b2	b1	b0
	BGDM	—	BRME	MDDRS	DIR	NFEN	—	ABCS0
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ABCS0	Asynchronous Base Clock Select	0: Operates on a frequency 16 times the transfer rate as the base clock. 1: Operates on a frequency 8 times the transfer rate as the base clock.	R/W
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	NFEN	Noise Cancellation Enable	0: Noise cancellation for the RxD pin is disabled. 1: Noise cancellation for the RxD pin is enabled.	R/W
b3	DIR	Data Transfer Direction Select	0: Transmits the data in the FTDR register by the LSB-first method. The received data is stored in the FRDR register by the LSB-first method. 1: Transmits the data in the FTDR register by the MSB-first method. The received data is stored in the FRDR register by the MSB-first method.	R/W
b4	MDDRS	Modulation Duty Register Select	0: BRR register is accessible. 1: MDDR register is accessible.	R/W
b5	BRME	Bit Rate Modulation Enable	0: Bit rate modulation is disabled. 1: Bit rate modulation is enabled.	R/W
b6	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b7	BGDM	Baud Rate Generator Double-Speed Mode Select	0: Baud rate generator normal mode: Baud rate generator operates on the clock signal produced by dividing the clock source by two. 1: Baud rate generator double-speed mode: Baud rate generator operates on the clock signal produced by the clock source (no frequency division).	R/W

ABCS0 Bit (Asynchronous Base Clock Select)

Selects the base clock for 1-bit period in asynchronous mode.

This bit setting is valid only in asynchronous mode (i.e., when the CM bit in the SMR register is 0).

NFEN Bit (Noise Cancellation Enable)

Reduces noise of the input to the RxD pin. This function is only valid in asynchronous mode. For details, see section 24.7, Noise Cancellation.

In clock synchronous mode, this bit should always be set to 0.

DIR Bit (Data Transfer Direction Select)

Selects the serial communication format. This bit is valid only when the transmit/receive data is in 8-bit formats.*1

Note 1. Asynchronous mode or clock synchronous mode with the 8-bit data length

MDDRS Bit (Modulation Duty Register Select)

Selects the register to be enabled access to it.

BRME Bit (Bit Rate Modulation Enable)

Specifies whether to enable or disable the bit rate modulation.

BGDM Bit (Baud Rate Generator Double-Speed Mode Select)

Selects operating mode of the baud rate generator. When setting 1 in this bit, the baud rate generator included in the SCIFA operates in double-speed mode. The setting of this bit is only effective in asynchronous mode (SMR.CM bit = 0) when the internal clock is selected as the clock source (SCR.CKE[1:0] = 00b). Use normal mode under any other settings.

24.2.15 FIFO Trigger Control Register (FTCR)

The FTCR register is a 16-bit register that specifies FIFO trigger conditions. The CPU can always read from and write to the FTCR register.

Address(es): SCIFA0.FTCR A006 5016h, SCIFA1.FTCR A006 5416h, SCIFA2.FTCR A006 5816h, SCIFA4.FTCR A006 6016h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RTRGS	—	—	RFTC[4:0]					TTRGS	—	—	TFTC[4:0]				
Value after reset:	0	0	0	1	1	1	1	1	0	0	0	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b4 to b0	TFTC[4:0]	Transmit FIFO Data Trigger Number	00h: Transmit data trigger number is 0. 0Fh: Transmit data trigger number is 15. Do not set 10h to 1Fh in these bits.	R/W
b6, b5	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b7	TTRGS	Transmit Trigger Select	0: TTRG[1:0] bits in FCR are valid. 1: TFTC[4:0] bits in FTCR are valid.	R/W
b12 to b8	RFTC[4:0]	Receive FIFO Data Trigger Number	01h: Receive data trigger number is 1. 10h: Receive data trigger number is 16. Do not set 00h and 11h to 1Fh in these bits.	R/W
b14, b13	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15	RTRGS	Receive Trigger Select	0: RTRG[1:0] bits in FCR are valid. 1: RFTC[4:0] bits in FTCR are valid.	R/W

TFTC[4:0] Bits (Transmit FIFO Data Trigger Number)

Specify the reference quantity of data for transmission (i.e., the threshold number of entries to trigger the writing of further data for transmission) for setting of the TDFE flag in the serial status register (FSR).

When the number of entries for transmission in the transmission FIFO, i.e. the number of entries written to the transmit FIFO data register (FTDR) that are yet to be transmitted, falls to or below the specified trigger number for transmission, the TDFE flag is set to 1 and a transmit FIFO data empty interrupt (TXIF) request is generated.

RFTC[4:0] Bits (Receive FIFO Data Trigger Number)

Specify the reference quantity of receive data (i.e., the threshold number of entries to trigger the reading of received data) for setting of the RDF flag in the serial status register (FSR).

When the number of entries in the reception FIFO, i.e. the number of entries yet to be read from the receive FIFO data register (FRDR), rises to or above the specified trigger number for reception, the RDF flag is set to 1 and a receive FIFO data full interrupt (RXIF) request is generated.

24.3 Operation

24.3.1 Overview

For serial communication, the SCIFA can select either asynchronous mode in which characters are synchronized individually or a clock synchronous mode in which communication is synchronized with clock pulses.
The SCIFA has a 16-stage FIFO buffer for both transmission and receptions, reducing the overhead of the CPU and enabling continuous high-speed communication. The RTS# and CTS# signals are provided as modem control signals.
Selection of a transmission/reception format is enabled with the serial mode register (SMR). Table 24.14 shows the transmission format which can be selected in the serial mode register (SMR). As shown in Table 24.15, the SCIFA clock source can be set by using the CKE[1:0] bits of the serial control register (SCR).

(1) Asynchronous Mode

- Data length is selectable either 7 or 8 bits
- Parity addition and 1- or 2-bit stop bit addition are selectable.
  (The combination of the preceding selections determines the transmission/reception format and character length).
- In reception, it is possible to detect framing errors, parity errors, receive FIFO data full, overrun errors, receive data ready, and breaks.
- The stored data quantities are indicated in the FIFO data count register (FDR), respectively for transmit and receive FIFO data.
- An internal or external clock can be selected as the SCIFA clock source.
  When an internal clock is selected, the SCIFA operates using the clock of on-chip baud rate generator and can output the clock with a frequency 16 (or 8) times the bit rate.
  When an external clock is selected, the external clock input must have a frequency 16 (or 8) times the bit rate. (The on-chip baud rate generator is not used.)

(2) Clock Synchronous Mode

- The transmission/reception format is fixed to the 8-bit data length.
- In reception, it is possible to detect overrun errors (ORER).
- An internal or external clock can be selected as the SCIFA clock source.
  When an internal clock is selected, the SCIFA operates using the clock of the on-chip baud rate generator, and outputs this clock to external devices as the synchronous clock.
  When an external clock is selected, the SCIFA operates on the input synchronous clock not using the on-chip baud rate generator.

Table 24.14 SMR Register Settings and SCIFA Communication Formats

SMR Register					SCIFA Transmission/Reception Format		
b7	b6	b5	b3				
CM	CHR	PE	STOP	Mode	Data Length	Parity Bit	Stop Bit Length
0	0	0	0	Asynchronous mode	8 bits	Not set	1 bit
			1				2 bits
		1	0			Set	1 bit
			1				2 bits
	1	0	0		7 bits	Not set	1 bit
			1				2 bits
		1	0			Set	1 bit
			1				2 bits
1	×	×	×	Clock synchronous mode	8 bits	Not set	None

×: Don't care

Table 24.15 SMR, SCR, and SPTR Register Settings and SCIFA Clock Source Selection

SMR Register	SCR Register		SPTR Register				
b7	b1	b0	b3	b2			
CM	CKE[1:0]		SCKIO	SCKDT	Mode	Clock Source	SCK Pin Function
0	0	0	0	×	Asynchronous mode	Internal	Input pin (input signal invalid) (Initial state)
			1	0			SCK pin state: Low
			1	1			SCK pin state: High
		1	×	×			Outputs a clock with frequency 16/8 times the bit rate*1
	1	0	×	×		External	Inputs a clock with frequency 16/8 times the bit rate*2
		1	×	×		Setting prohibited	
1	0	×	×	×	Clock synchronous mode	Internal	Outputs the synchronous clock
	1	0	×	×		External	Inputs the synchronous clock
		1	×	×		Setting prohibited	

×: Don't care

Note 1. SEMR.ABCS0 = 0: Output a clock that has a frequency 16 times the bit rate.
SEMR.ABCS0 = 1: Output a clock that has a frequency 8 times the bit rate.

Note 2. SEMR.ABCS0 = 0: Input a clock that has a frequency 16 times the bit rate.
SEMR.ABCS0 = 1: Input a clock that has a frequency 8 times the bit rate.

24.3.2 Operation in Asynchronous Mode

In asynchronous mode, each transmitted or received character begins with a start bit and ends with a stop bit. Serial communication is synchronized one character at a time.

The transmitting and receiving sections of the SCIFA are independent, so full duplex communication is possible. The transmitter and receiver are 16-stage FIFO buffered, so data can be written and read while transmitting and receiving are in progress, enabling continuous transmission and reception.

Figure 24.2 shows the general format of asynchronous serial communication.

In asynchronous serial communication, the communication line is normally held in the mark (high) state. The SCIFA monitors the line and starts serial communication when the line goes to the space (low) state, considered as a start bit. One serial character consists of a start bit (low), data (LSB first when LSB-first transfer is selected), parity bit (high or low), and stop bit (high), in that order.

When receiving in asynchronous mode, the SCIFA synchronizes at the falling edge of the start bit. The SCIFA samples each data bit on the eighth pulse of a clock with a frequency 16 or 8 times the bit rate*1. Receive data is latched at the center of each bit.

Note 1. When the SEMR.ABCS0 bit = 0, data is sampled on the eighth pulse of a clock with a frequency 16 times the bit rate.
When the SEMR.ABCS0 bit = 1, data is sampled on the fourth pulse of a clock with a frequency 8 times the bit rate.

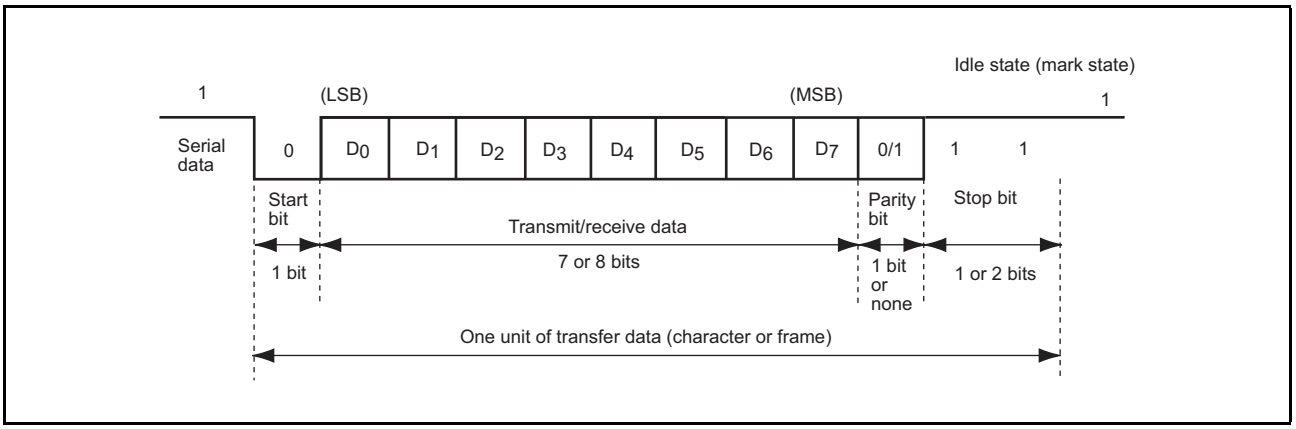


Figure 24.2 Data Format in Asynchronous Communication (8-Bit Data with Parity and Two Stop Bits when LSB-First Transfer is Selected)

(1) Transmit/Receive Formats

Table 24.16 lists the eight communications formats that can be selected in asynchronous mode. The format is selected by setting in the serial mode register (SMR).

Table 24.16 Serial Communications Formats (in Asynchronous Mode)

SMR Setting			Serial Transmit/Receive Format and Frame Length											
CHR	PE	STOP	1	2	3	4	5	6	7	8	9	10	11	12
0	0	0	START	8-bit data								STOP		
		1	START	8-bit data								STOP	STOP	
	1	0	START	8-bit data								P	STOP	
		1	START	8-bit data								P	STOP	STOP
1	0	0	START	7-bit data							STOP			
		1	START	7-bit data							STOP	STOP		
	1	0	START	7-bit data							P	STOP		
		1	START	7-bit data							P	STOP	STOP	

START: Start bit
STOP: Stop bit
P: Parity bit

(2) Clock

An SCIFA transmit/receive clock can be selected from the internal clock generated by the on-chip baud rate generator or the external clock input from the SCK pin. The clock source is selected by the settings of the CM bit in the serial mode register (SMR) or the CKE[1:0] bits in the serial control register (SCR). For clock source selection, refer to Table 24.15.
When an external clock is input at the SCK pin, it must have a frequency equal to 16/8 times the desired bit rate.
When the SCIFA operates on an internal clock, it can output a clock signal on the SCK pin. The frequency of this output clock is 16/8 times the desired bit rate.

(3) Transmitting and Receiving Data

- SCIFA Initialization (in Asynchronous Mode)

Before transmitting or receiving data, clear the TE and RE bits to 0 in the serial control register (SCR), and then initialize the SCIFA as follows.

When changing operating mode or communication format, always clear the TE and RE bits in the SCR register to 0 before following the procedure given below. Clearing TE to 0 initializes the transmit shift register (TSR). Clearing TE and RE to 0, however, does not initialize the serial status register (FSR), transmit FIFO data register (FTDR), or receive FIFO data register (FRDR), which retain their previous contents. Clear TE to 0 after all transmit data has been transmitted and the TEND flag in the FSR register is set. The TE bit can be cleared to 0 during transmission, but the transmit data (the TXD pin output level) after the TE bit is cleared to 0 depends on the settings of the SPB2IO and SPB2DT bits in the SPTR register. Set the TFRST bit in the FCR register to 1 and reset the FTDR register before TE is set to 1 again to start transmission.

When an external clock is used, the clock should not be stopped during initialization or subsequent operation. SCIFA operation becomes unreliable if the clock is stopped. Figure 24.3 shows a sample flowchart for initializing the SCIFA in asynchronous mode.

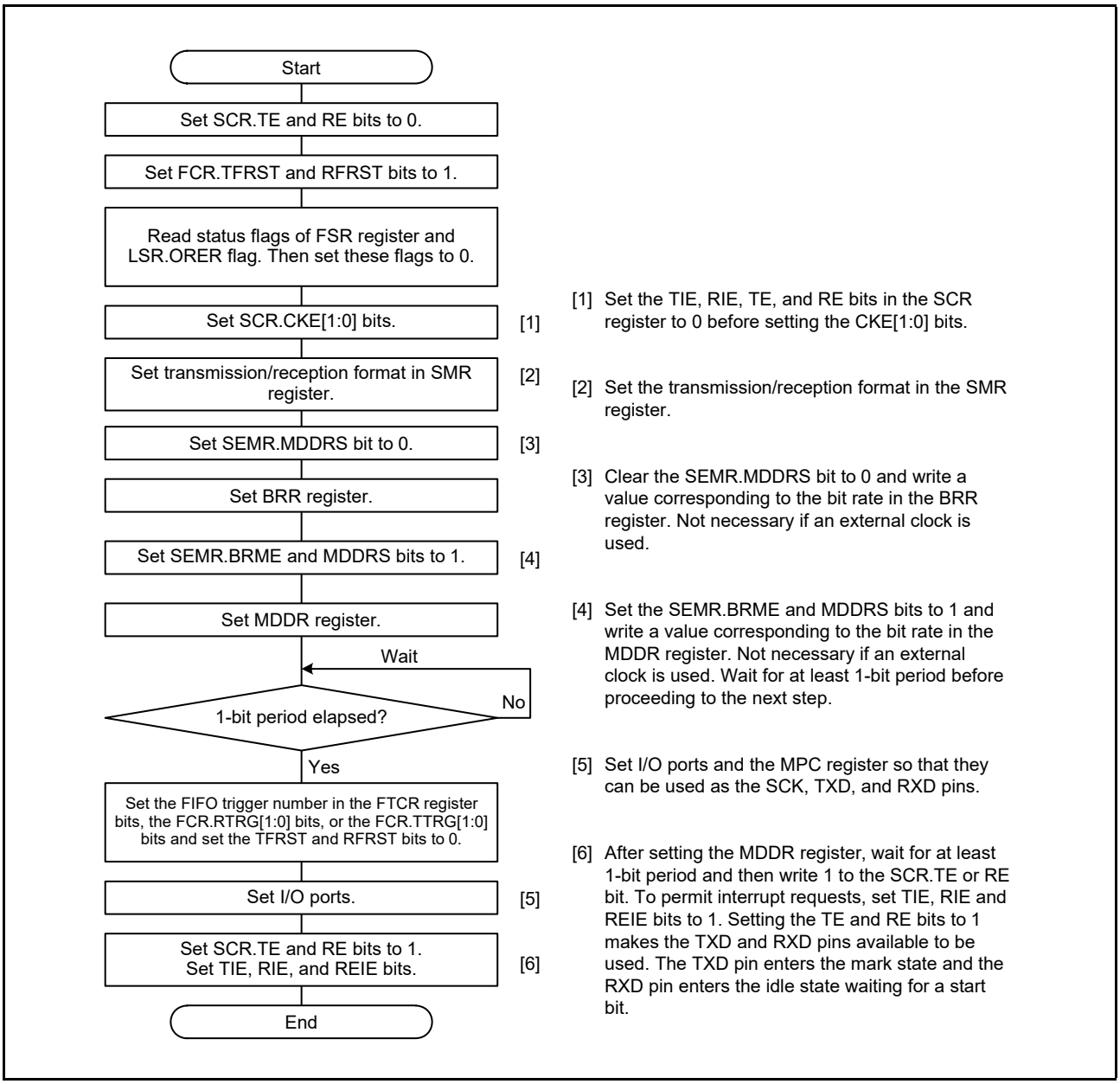
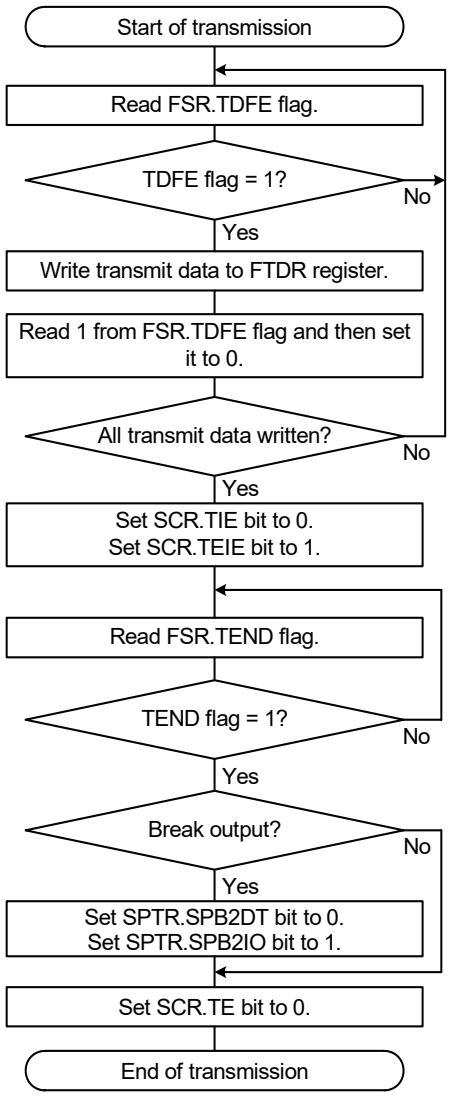


Figure 24.3 Sample Flowchart for SCIFA Initialization in Asynchronous Mode

- Transmitting Serial Data (in Asynchronous Mode)

Figure 24.4 shows a sample flowchart for serial transmission in asynchronous mode.
Follow the procedure given below for serial data transmission after enabling the SCIFA for transmission.



[1] Check the SCIFA status and write the transmit data:
Read the FSR register, confirm that the TDFE flag is set to 1, and then write the transmit data to the FTDR register. The transmit data quantity that can be written is "16 - FDR.T[4:0]".
A change of the TDFE flag from 0 to 1 can be known by a TXIF interrupt request.

[2] Serial transmission continuation procedure (when the FTDR register is written by the CPU):
To continue serial transmission, confirm that the TDFE flag in the FSR register is set to 1, and then write 0 to the flag.

[3] Serial transmission end procedure (when TXIF or TEIF interrupt is used):
After the last transmission data is written to the FTDR register, set the TIE bit to 0 and the TEIE bit to 1 in the SCR register. If the DMAC is activated to write the transmit data to the FTDR register, enable the transmit end interrupt. This will generate a CPU interrupt request when the last transmit data is written in the FTDR register. The TIE and TEIE bit settings will be changed by the CPU interrupt process.
This procedure is unnecessary if the TXIF or TEIF interrupt is not used.

[4] Serial transmission end confirmation:
Read the FSR register and confirm that the TEND flag has changed to 1.
A change of the TEND flag from 0 to 1 can be known by a TEIF interrupt request.

[5] Break output during serial transmission:
To cause a break to be output during serial transmission, set the SPB2DT bit to 0 and the SPB2IO bit to 1 in the SPTR register. Then set the SCR.TE bit to 0.*1

In Step [1], the number of data bytes that can be written can be checked by the FDR.T[4:0] flag (non-transmitted data quantity in the FTDR register).

Note 1. The SPB2DT bit of the SPTR register indicates the input status of the RXD pin. Writings to the SPB2DT and SPB2IO bits of this register in 1-bit unit are handled as read-modify-write, which may lead to undesired values to be written to the SPB2DT bit. To avoid this, when modifying the SPB2DT or SPB2IO bit, for example, write the other bit (the bit used in combination) at the same time.

Figure 24.4 Sample Flowchart for Transmitting Serial Data in Asynchronous Mode

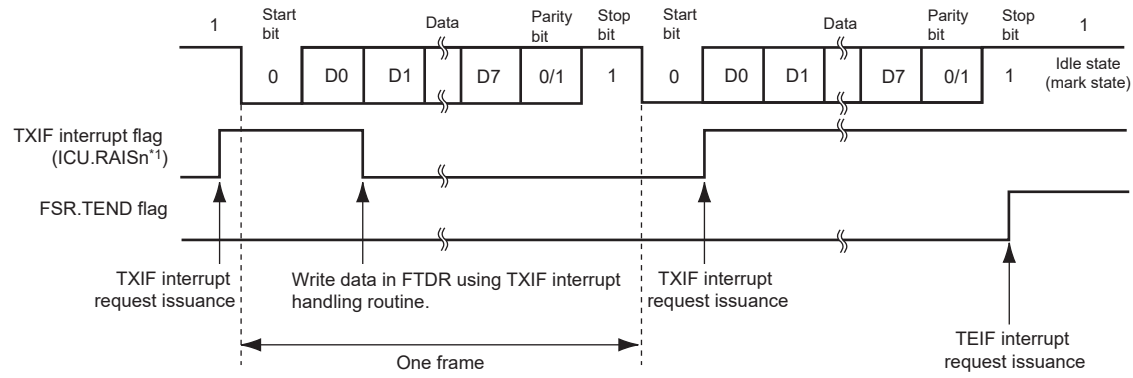
In asynchronous mode, the SCIFA performs serial transmission as described below.

1. When data is written into the transmit FIFO data register (FTDR) by the TXIF interrupt processing routine, the SCIFA transfers the data from the FTDR register to the transmit shift register (TSR) and starts transmission. Confirm that the TDFE flag in the serial status register (FSR) is set to 1 before writing transmit data to the FTDR register. The number of data bytes that can be written is “16 minus the number of non-transmitted data units”.
2. When data is transferred from the FTDR register to the TSR register and transmission is started, consecutive transmit operations are performed until there is no transmit data left in the FTDR register. When the number of transmit data bytes in the FTDR register becomes equal to or less than the transmission trigger number specified in the FIFO control register (FCR) or FIFO trigger control register (FTCR), the TDFE flag is set. If the TIE bit in the serial control register (SCR) is set to 1 at this time, a transmit-FIFO-data-empty interrupt (TXIF) request is generated.

   The serial transmit data is output from the TXD pin in the following order.

   (a) Start bit: One-bit 0 is output.
   (b) Transmit data: 8- or 7-bit data is output in LSB-first order (when LSB-first transfer is selected).
   (c) Parity bit: One parity bit (even or odd parity) is output. (A format in which a parity bit is not output can also be selected.)
   (d) Stop bit(s): One or two 1 bits (stop bits) are output.
   (e) Mark state: 1 is output continuously until the start bit that starts the next transmission is output.
3. The SCIFA checks the transmit data of the FTDR register at the timing for sending the stop bit. If data is present, the data is transferred from the FTDR register to the TSR register, the stop bit is output, and then serial transmission of the next frame is started. If there is no data to be transmitted, the TEND flag in the FSR register is set to 1, the stop bit is output, and then the SCIFA enters the mark state (high level) in which 1 is output continuously.

Figure 24.5 shows an example of the operation for transmission in asynchronous mode.



Note 1. For the corresponding interrupt number, see section 12, Interrupt Controller (ICUA).

Figure 24.5 Example of Transmit Operation in Asynchronous Mode (8-Bit Data with Parity and One Stop Bit when LSB-First Transfer is Selected)

4. When modem control is enabled, transmission can be stopped/resumed by the input level to the CTS# pin. When a high level is input to the CTS# pin during transmission, the SCIFA enters the mark state (high level) after completion of one-frame data transmission. When a low level is input to the CTS# pin, output of the next data to be transmitted begins with a start bit. Figure 24.6 shows an example of the operation for transmission when using the modem control function.

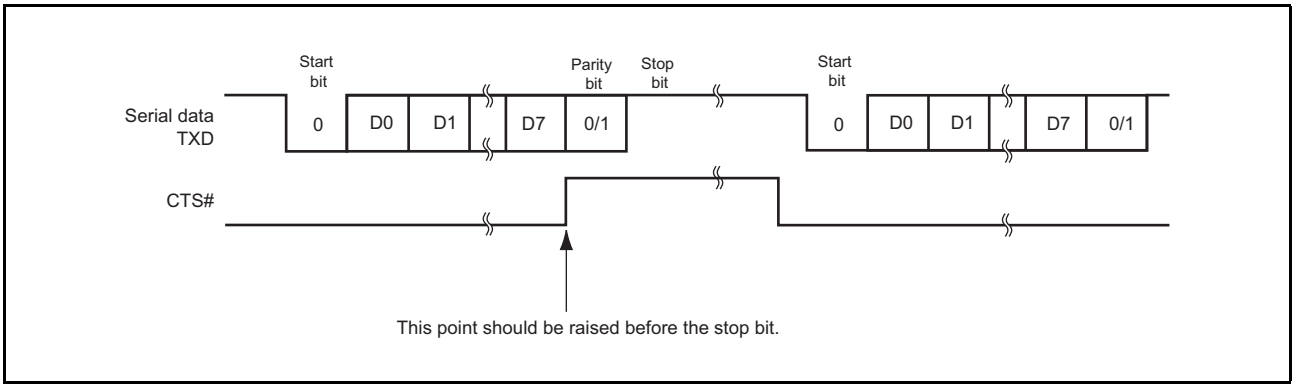
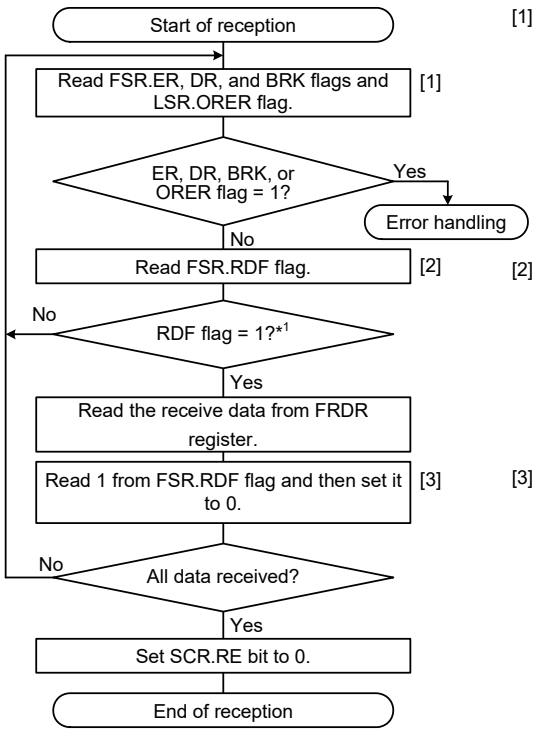


Figure 24.6 Example of Transmit Operation in Asynchronous Mode Using Modem Control Function (CTS#)

- Receiving Serial Data (in Asynchronous Mode)

Figure 24.7 and Figure 24.8 show sample flowcharts for serial reception in asynchronous mode.
Follow the procedure given below for serial data reception after enabling the SCIFA for reception.



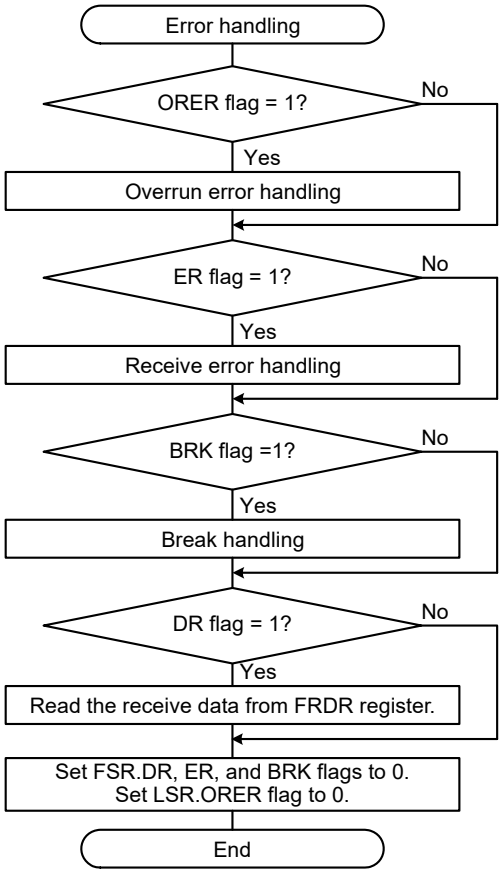
[1] Receive error handling and break detection:
When a receive error occurs, read the ER, DR, and BRK flags in the FSR register and the LSR.ORER flag to identify the error. After handling the error as specified, clear the ER, DR, BRK, and ORER flags to 0.
In the case where a framing error occurs, a break can be detected by reading the value of the RXD pin.
Error occurrence can be known by a BRIF, ERIF, or DRIF interrupt request.

[2] Check the SCIFA status and read the receive data:
Read the FSR register, confirm that the RDF flag is set to 1, and then read the receive data at least as much as the amount specified as the receive data trigger number from those stored in the FRDR register. The receive data quantity that can be read can be checked by the FDR.R[4:0] flag.
A change of the RDF flag from 0 to 1 can be known by an RXIF interrupt request.

[3] Serial reception continuation procedure (when the FRDR register is read by the CPU):
To continue serial reception, confirm that the RDF flag in the FSR register is set to 1, and then write 0 to the flag.

Note 1. The number of all the received data units should be an integral multiple of the receive data trigger number. If not, process returns to Step [1] and enters an infinite loop. In this case, it is necessary to set the timeout processing using a timer.

Figure 24.7 Sample Flowchart for Receiving Serial Data in Asynchronous Mode (1)



- Whether a framing error or parity error has occurred in the received data that is read from the FRDR register can be checked with the FER and PER flags in the FSR register.
- When a break signal is received, the received data is not transferred to the FRDR register while the BRK flag is set to 1. However, note that the last data in the FRDR register is 00h and the break data in which a framing error occurred is stored.

Figure 24.8 Sample Flowchart for Receiving Serial Data in Asynchronous Mode (2)

In asynchronous mode, the SCIFA performs serial reception as described below.

1. The SCIFA monitors the communication line, and if a 0 start bit is detected, it performs internal synchronization to start reception.
2. The received data is stored into the RSR register in LSB-to-MSB order (when LSB-first transfer is selected).
3. The parity bit and stop bit are received.
   After receiving these bits, the SCIFA carries out the following checks.
   (a) Stop bit check: The SCIFA checks whether the stop bit is 1. If there are two stop bits, only the first is checked.
   (b) The SCIFA checks whether receive data can be transferred from the receive shift register (RSR) to the receive FIFO data register (FRDR).
   (c) Parity bit check: The SCIFA checks whether the parity bit is an expected value.
   (d) Overrun error check: The SCIFA checks whether the ORER flag is 0, indicating that the overrun error has not occurred.
   (e) Break check: The SCIFA checks whether the BRK flag is 0, indicating that the break state is not set.

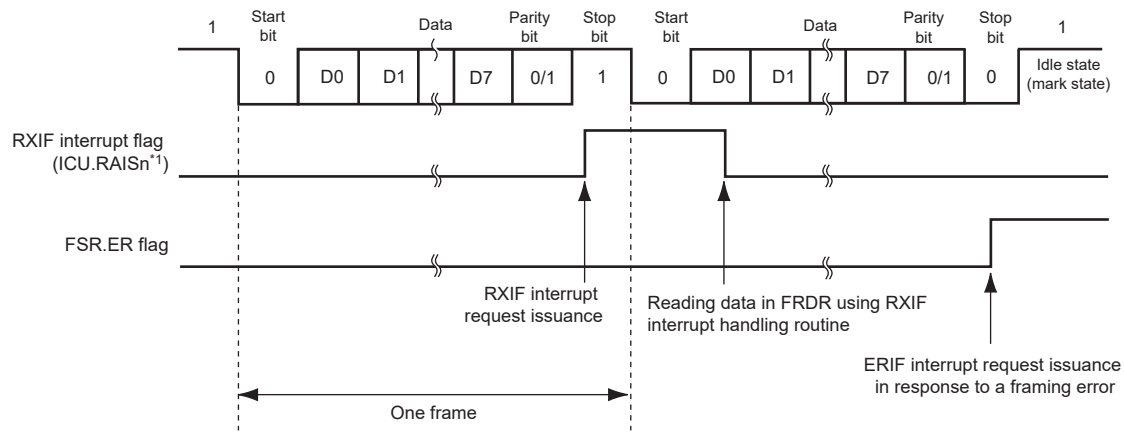
If all the above checks are passed, the receive data is stored in the FRDR register.

Note: When a parity error or a framing error occurs, reception is not suspended.

4. When receive data units equaling or exceeding the specified reception trigger number are stored in the receive FIFO data register (FRDR) and the RDF flag is changed to 1, a receive FIFO data full interrupt (RXIF) request is generated while the RIE bit in the SCR register is set to 1. When the quantity of data in the FRDR register falls below the specified reception trigger number and the RIE bit in the SCR register is set to 1, a receive data ready interrupt (DRIF) request is generated if no next data is received after the elapse of 15 ETUs*1 from the last stop bit (the DR flag in the FSR register is 1). When the ER flag in the FSR register is changed to 1, a receive error interrupt (ERIF) request is generated while the RIE or REIE bit in the SCR register is set to 1. When the BRK or ORER flag is changed to 1 in the FSR register, a break reception interrupt (BRIF) request is generated while the RIE or REIE bit in the SCR register is set to 1.

Note 1. It is equivalent to 1 and half frames of 8-bit format with one stop bit (ETU: Element Time Unit).

Figure 24.9 shows an example of the operation for reception in asynchronous mode.



Note 1. For the corresponding interrupt number, see section 12, Interrupt Controller (ICUA).

Figure 24.9 Example of SCIFA Receive Operation in Asynchronous Mode (8-Bit Data with Parity and One Stop Bit when LSB-First Transfer is Selected)

5. When modem control is enabled, the RTS# signal that indicates the FRDR register has space is output. When the RTS# pin is at low level, reception is possible. The RTS# pin being at the high level indicates that the number of entries in the FRDR register is equal to or greater than the threshold for output of the active level of the RTS# signal and that the transmission of further data needs to be suspended until the FRDR register has enough space. Figure 24.10 shows an example of the operation for reception in asynchronous mode when using the modem control function.

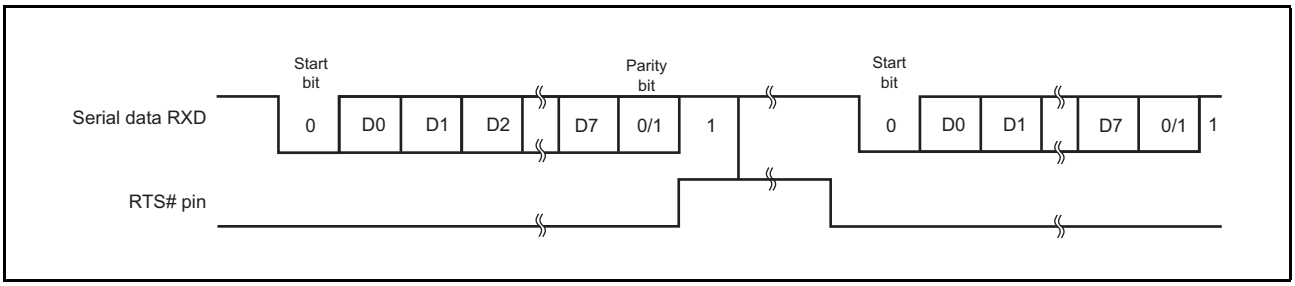
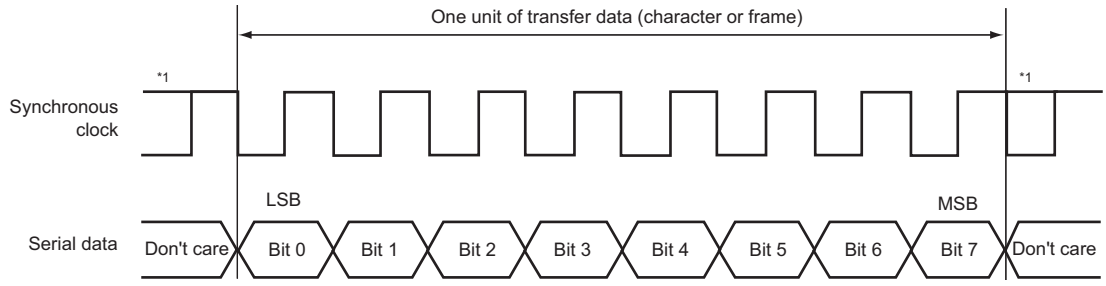


Figure 24.10 Example of SCIFA Receive Operation in Asynchronous Mode Using Modem Control Function (RTS#)

24.3.3 Operation in Clock Synchronous Mode

In clock synchronous mode, the SCIFA transmits and receives data in synchronization with clock pulses. This mode is suitable for high-speed serial communication.
Full-duplex communication is possible because the SCIFA transmitter and receiver are independent and share the same clock. Since the transmitter and the receiver have 16-stage FIFO buffers, respectively, continuous transmission or reception is possible by reading or writing data while transmission or reception is in progress.
Figure 24.11 shows the general format in clock synchronous serial communication.



Note 1. High level except in continuous transmission/reception

Figure 24.11 Data Format in Clock Synchronous Communication (when LSB-First Transfer is Selected)

In clock synchronous serial communication, each data bit is output on the communication line from one falling edge of the synchronous clock to the next. Data is guaranteed valid at the rising edge of the synchronous clock.
In each character, the serial data bits are transmitted in order from the LSB (first) to the MSB (last). After output of the MSB, the communication line remains in the state of the MSB (when LSB-first transfer is selected).
In clock synchronous mode, the SCIFA receives data by synchronizing with the rising edge of the synchronous clock.

(1) Transmit/Receive Formats

The data length is fixed at eight bits.
No parity bit can be added.

(2) Clock

An internal clock generated by the on-chip baud rate generator or an external clock input from the SCK pin can be selected as the SCIFA transmit/receive clock according to the settings of the CM bit in the serial mode register (SMR) or the CKE[1:0] bits in the serial control register (SCR).
When the SCIFA operates on an internal clock, it outputs the synchronous clock signal at the SCK pin. Eight synchronous clock pulses are output per transmitted or received character. Unless the SCIFA is transmitting or receiving, the synchronous clock signal remains in the high state. When the SCIFA only receives data on an internal clock, the internal clock signal outputs while the RE bit in the SCR register is 1 until the number of data units in the receive FIFO reaches the specified reception trigger number.

(3) Transmitting and Receiving Data

- SCIFA Initialization (in Clock Synchronous Mode)

Before transmitting or receiving, clear the TE and RE bits to 0 in the serial control register (SCR), and then initialize the SCIFA by performing the following procedure.
Similarly, before changing the mode or communication format, clear the TE and RE bits to 0, and then change it by performing the following procedure. Clearing TE to 0 initializes the transmit shift register (TSR). Clearing RE to 0, however, does not initialize the RDF, PER, FER, and ORER flags and the receive FIFO data register (FRDR), which retain their previous contents.

Figure 24.12 shows a sample flowchart for initializing the SCIFA in clock synchronous mode.

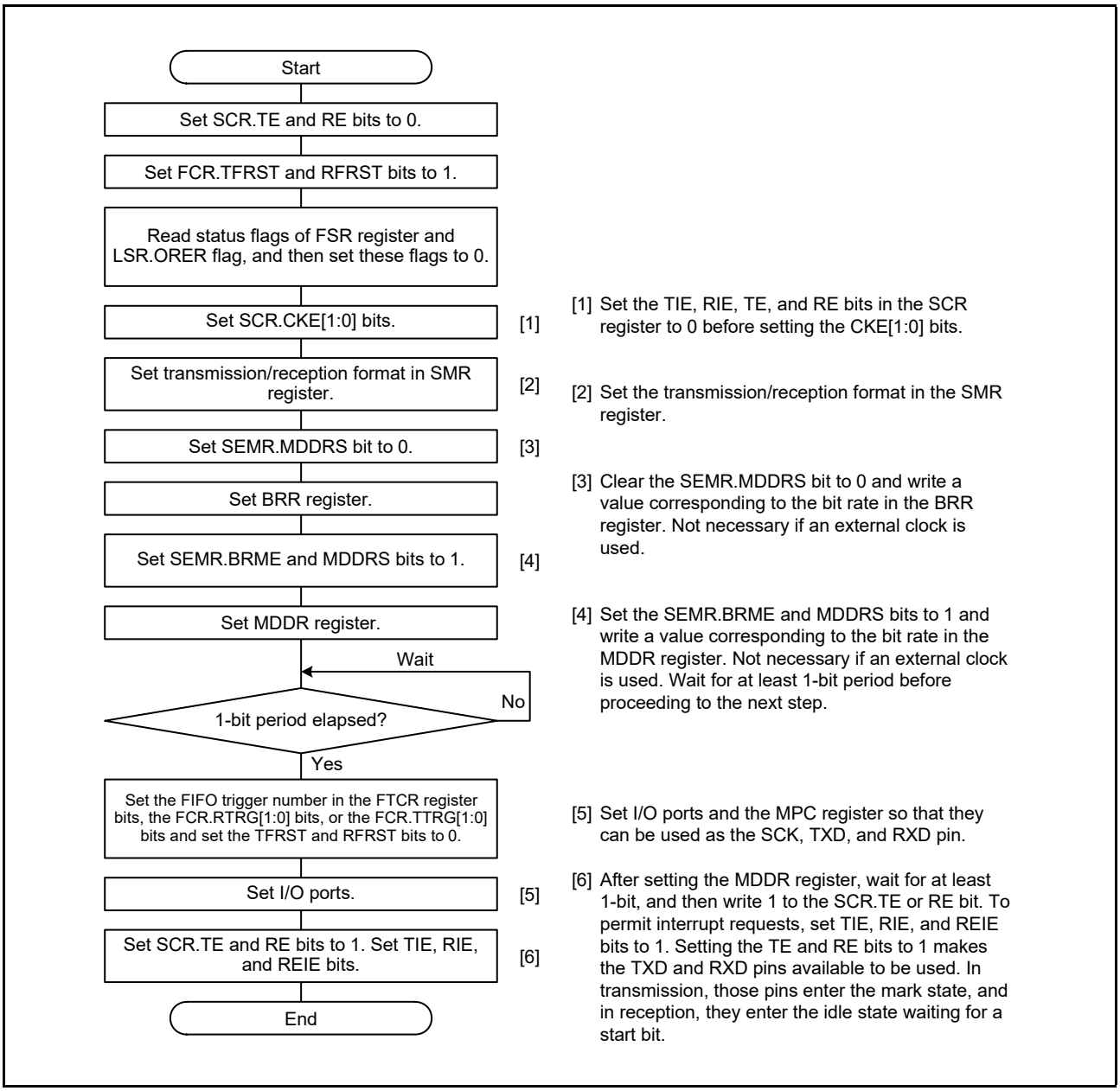


Figure 24.12 Sample Flowchart for SCIFA Initialization in Clock Synchronous Mode

- Transmitting Serial Data (in Clock Synchronous Mode)

Figure 24.13 shows a sample flowchart for transmitting serial data in clock synchronous mode.
Follow the procedure given below for serial data transmission after enabling the SCIFA for transmission.

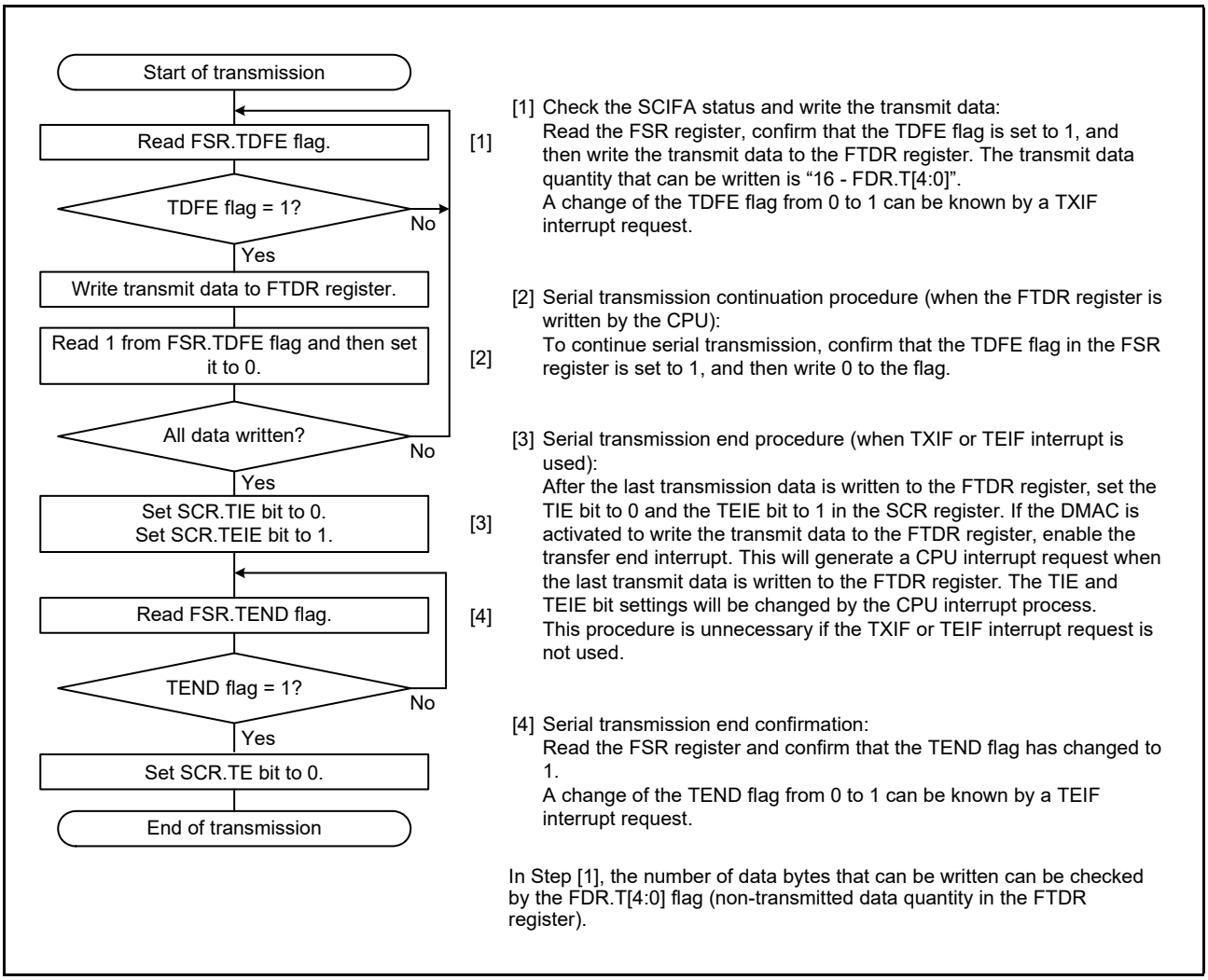
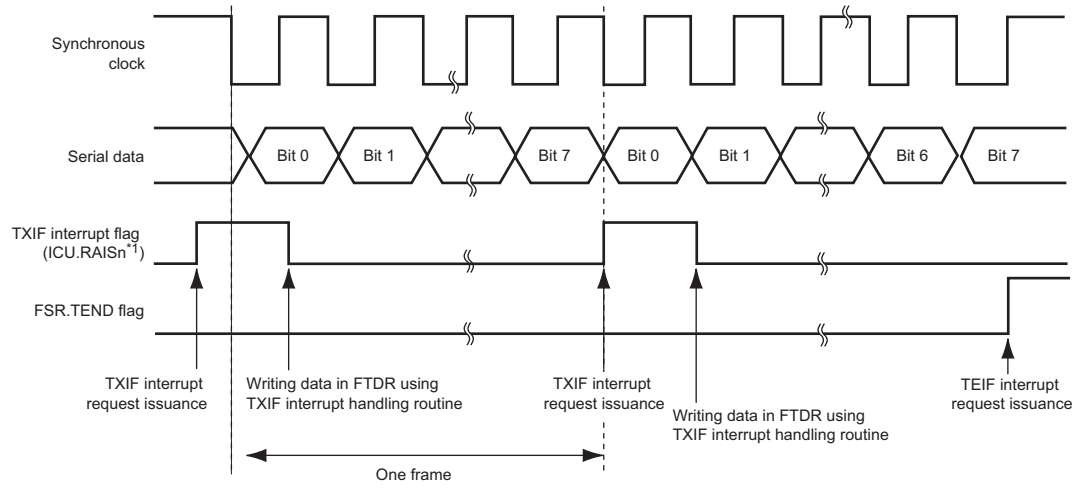


Figure 24.13 Sample Flowchart for Transmitting Serial Data in Clock Synchronous Mode

In clock synchronous mode, the SCIFA performs serial transmission as described below.

1. When data is written into the transmit FIFO data register (FTDR) by the TXIF interrupt processing routine, the SCIFA transfers the data from the FTDR register to the transmit shift register (TSR) and starts transmission. Confirm that the TDFE flag in the serial status register (FSR) is set to 1 before writing transmit data to the FTDR register. The number of data bytes that can be written is “16 minus the specified number of non-transmitted data units”.
2. When data is transferred from the FTDR register to the TSR register and transmission is started, consecutive transmit operations are performed until there is no transmit data left in the FTDR register. When the number of transmit data bytes in the FTDR register becomes equal to or less than the transmission trigger number set in the FIFO control register (FCR) or FIFO trigger control register (FTCR), the TDFE flag in the FSR register is set. If the TIE bit in the serial control register (SCR) is set to 1 at this time, a transmit-FIFO-data-empty interrupt (TXIF) request is generated.
   If clock output mode is selected, the SCIFA outputs eight synchronous clock pulses. If an external clock source is selected, the SCIFA outputs data in synchronization with the input clock. Data is output from the TXD pin in order from the LSB (b0) to the MSB (b7) (when LSB-first transfer is selected).
3. The SCIFA checks the transmit data of the FTDR register at the timing for sending the MSB (bit 7). If data is present, the data is transferred from the FTDR register to the TSR register, and then serial transmission of the next frame is started. If there is no data, the TXD pin holds the output level of the last data after the TEND flag in the FSR register is set to 1 and the MSB (bit 7) is output.
4. After the end of serial transmission, the SCK pin is held in the high state.

Figure 24.14 shows an example of SCIFA transmit operation in clock synchronous mode.

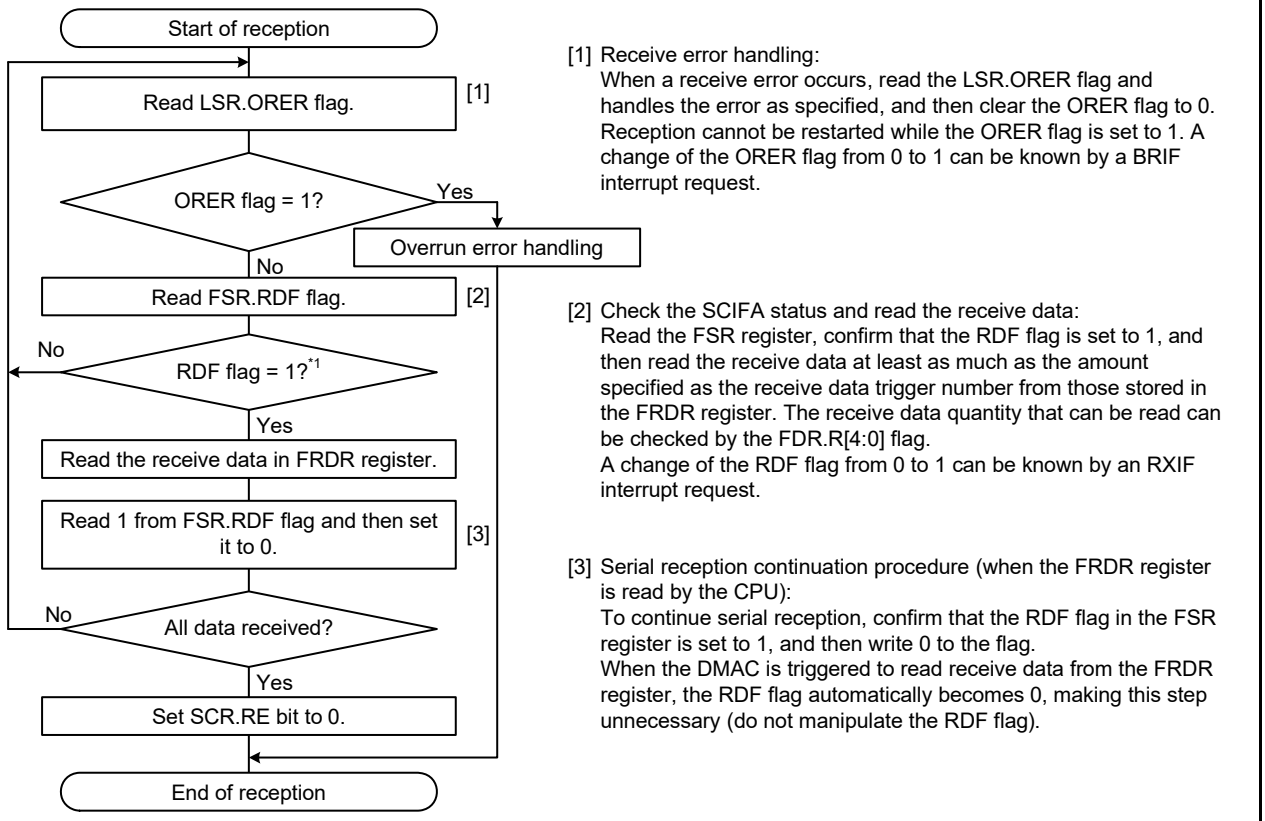


Note 1. For the corresponding interrupt number, see section 12, Interrupt Controller (ICUA).

Figure 24.14 Example of SCIFA Transmit Operation in Clock Synchronous Mode (when LSB-First Transfer is Selected)

- Receiving Serial Data (in Clock Synchronous Mode)

Figure 24.15 shows sample flowcharts for receiving serial data in clock synchronous mode.
Follow the procedure given below for serial data reception after enabling the SCIFA for reception. When switching from asynchronous mode to clock synchronous mode without SCIFA initialization, make sure that the ORER, PER, and FER flags in the line status register (LSR) are cleared to 0.



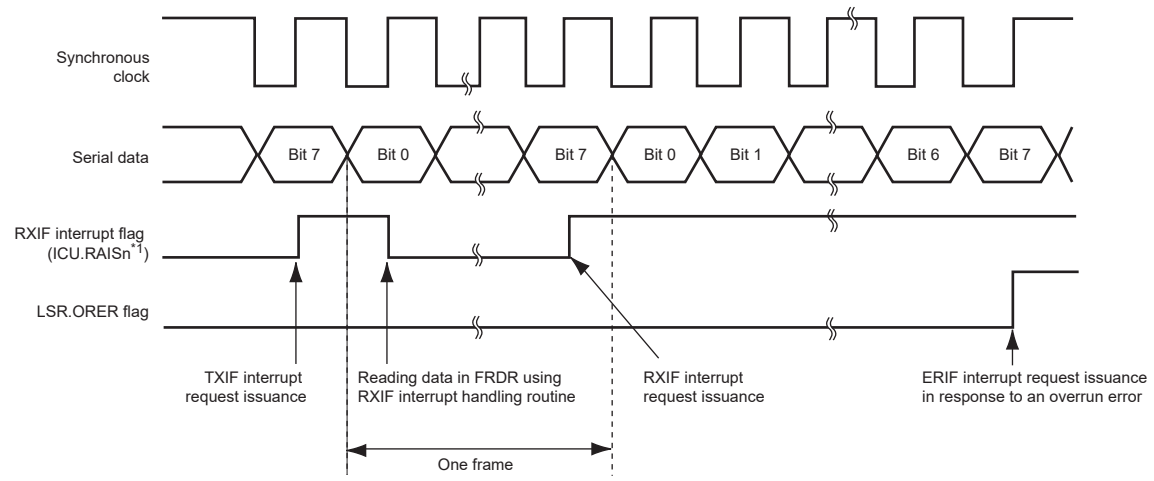
Note 1. The number of all the received data units should be an integral multiple of the receive data trigger number. If not, process returns to Step [1] and enters an infinite loop. In this case, it is necessary to set the timeout processing using a timer.

Figure 24.15 Sample Flowchart for Receiving Serial Data in Clock Synchronous Mode

In clock synchronous mode, the SCIFA performs serial reception as described below.

1. The SCIFA synchronizes with the synchronous clock input or output and starts reception.
2. Receive data is stored into the receive shift register (RSR) in order from the LSB to the MSB (when LSB-first transfer is selected). After receiving the data, the SCIFA checks whether the receive data can be transferred from the RSR register to the FRDR register. If data can be transferred, the SCIFA stores the received data in the FRDR register. If an overrun error is detected during the error check, further reception is not performed.
3. After the received data units equaling or exceeding the specified reception trigger number are stored in the FRDR register and the RDF flag is set to 1, a receive-data-full interrupt (RXIF) request is generated when the RIE bit in the serial control register (SCR) is set to 1. When the ORER flag in the line status register (LSR) is set to 1 and the RIE or REIE bit in the SCR register is also set to 1, a break interrupt (BRIF) request is generated.

Figure 24.16 shows an example of SCIFA receive operation in clock synchronous mode.



Note 1. For the corresponding interrupt number, see section 12, Interrupt Controller (ICUA).

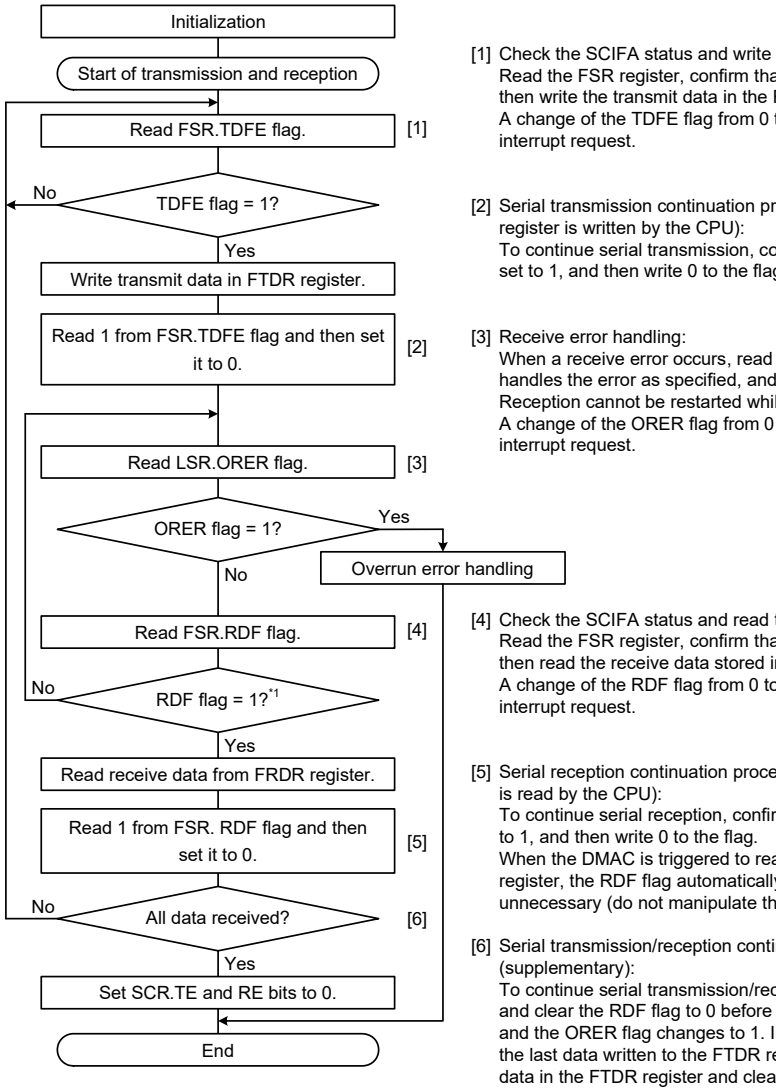
Figure 24.16 Example of SCIFA Receive Operation (when LSB-First Transfer is Selected)

- Transmitting and Receiving Serial Data Simultaneously (in Clock Synchronous Mode)

Figure 24.17 shows a sample flowchart for transmitting and receiving serial data simultaneously in clock synchronous mode.

In simultaneous transmission/reception of serial data, number of receive data = number of transmit data = number of transmit data to be written to the FTDR register.

Follow the procedure given below for the simultaneous transmission/reception of serial data, after enabling the SCIFA for transmission/reception.



[1] Check the SCIFA status and write the transmit data:
Read the FSR register, confirm that the TDFE flag is set to 1, and then write the transmit data in the FTDR register.
A change of the TDFE flag from 0 to 1 can be known by a TXIF interrupt request.

[2] Serial transmission continuation procedure (when the FTDR register is written by the CPU):
To continue serial transmission, confirm that the FSR.TDFE flag is set to 1, and then write 0 to the flag.

[3] Receive error handling:
When a receive error occurs, read the LSR.ORER flag and handles the error as specified, and then clear the ORER flag to 0.
Reception cannot be restarted while the ORER flag is set to 1.
A change of the ORER flag from 0 to 1 can be known by a BRIF interrupt request.

[4] Check the SCIFA status and read the receive data:
Read the FSR register, confirm that the RDF flag is set to 1, and then read the receive data stored in the FRDR register.
A change of the RDF flag from 0 to 1 can be known by an RXIF interrupt request.

[5] Serial reception continuation procedure (when the FRDR register is read by the CPU):
To continue serial reception, confirm that the FSR.RDF flag is set to 1, and then write 0 to the flag.
When the DMAC is triggered to read receive data from the FRDR register, the RDF flag automatically becomes 0, making this step unnecessary (do not manipulate the RDF flag).

[6] Serial transmission/reception continuation procedure (supplementary):
To continue serial transmission/reception, read the FRDR register and clear the RDF flag to 0 before the FRDR register overflows and the ORER flag changes to 1. In addition, before the last bit of the last data written to the FTDR register is transmitted, write the data in the FTDR register and clear the TDFE flag to 0.
When the DMAC is triggered to write data to the FTDR register or read data from the FRDR register, the TDFE and RDF flags automatically become 0, making this step unnecessary (do not manipulate the TDFE or RDF flag).

Note: When switching the operation from transmission or reception to the simultaneous transmission and reception, clear the TE and RE bits to 0 and then simultaneously set these bits to 1.

Note 1. The number of all the received data units should be an integral multiple of the receive data trigger number. If not, process returns to Step [1] and enters an infinite loop. In this case, it is necessary to set the timeout processing using a timer.

Figure 24.17 Sample Flowchart for Simultaneous Transmitting/Receiving Serial Data in Clock Synchronous Mode

24.4 Bit Modulation

Using the bit rate modulation, the bit rate can be corrected by skipping the specified number of clock pulses input to the baud rate generator. To correct the bit rate, only the number of clock pulses specified in the MDDR register are enabled among 256 internal clock pulses specified by the CKS[1:0] bits in the SMR register in a way that forms average intervals.

Figure 24.18 shows an example where SERICLK is selected by the CKS[1:0] bits in SMR and BRR and MDDR are set to 0 and 160, respectively, in asynchronous mode. In this example, the cycle of the base clock is evenly corrected (256/160) and the bit rate is also corrected (160/256). Note that skipping an internal clock causes bias and expansion or contraction is generated in the pulse width of the base clock.

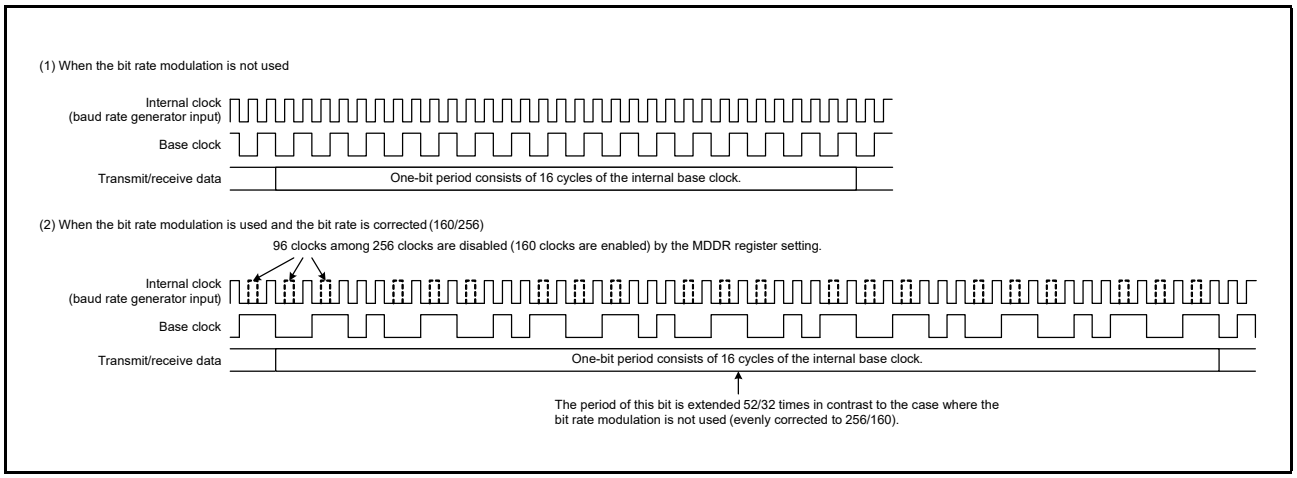


Figure 24.18 Example of Internal Base Clock when Bit Modulation is Used

24.5 Interrupt Sources

The SCIFA has six interrupt sources: transmit-FIFO-data-empty (TXIF), receive-error (ERIF), receive-FIFO-data-full (RXIF), break (BRIF), transmit-end (TEIF), receive-data-ready (DRIF). The TEIF and DRIF interrupts, the ERIF and BRIF interrupts share the same vector numbers, respectively.

Table 24.17 shows the interrupt sources and priority. The interrupt sources can be enabled or disabled using the TIE, RIE, REIE, TEIE bits in the SCR register and are separately input to the interrupt controller.

When the quantity of transmit data written in the FTDR register as a result of transmission is equal to or less than the specified transmission trigger number, the TDFE flag in the serial status register (FSR) is set to 1 and a TXIF interrupt request is generated.

When the data units equaling or exceeding the specified transmission trigger number are stored in the receive FIFO register (FRDR) and the RDF flag in the FSR register is set to 1, a receive data full interrupt (RXIF) request is generated.

When the quantity of received data in the FRDR register is below the specified reception trigger number and no next data has been received yet even after the period of 15 ETUs elapsed*1 from the last stop bit, the DR flag in the FSR register is set to 1 and a receive data ready interrupt (DRIF) request is generated. In clock synchronous mode, a DRIF interrupt request is not generated.

When the BRK flag in the FSR register or the ORER flag in the LSR register is set to 1, a BRIF interrupt request is issued.

When the ER flag in the FSR register is set to 1, an ERIF interrupt request is issued.

When the TEND flag in the FSR register is set to 1, a TEIF interrupt request is issued.

When the RIE bit is cleared to 0 and the REIE bit in the SCR register is set to 1, an ERIF and BRIF interrupt requests are issued but an RXIF and a DRIF interrupt requests are not.

An TXIF interrupt indicates that transmit data can be written and an RXIF interrupt indicates that receive data is stored in the FRDR register.

Note 1. It is equivalent to 1 and half frames of 8-bit format with one stop bit (ETU: Element Time Unit).

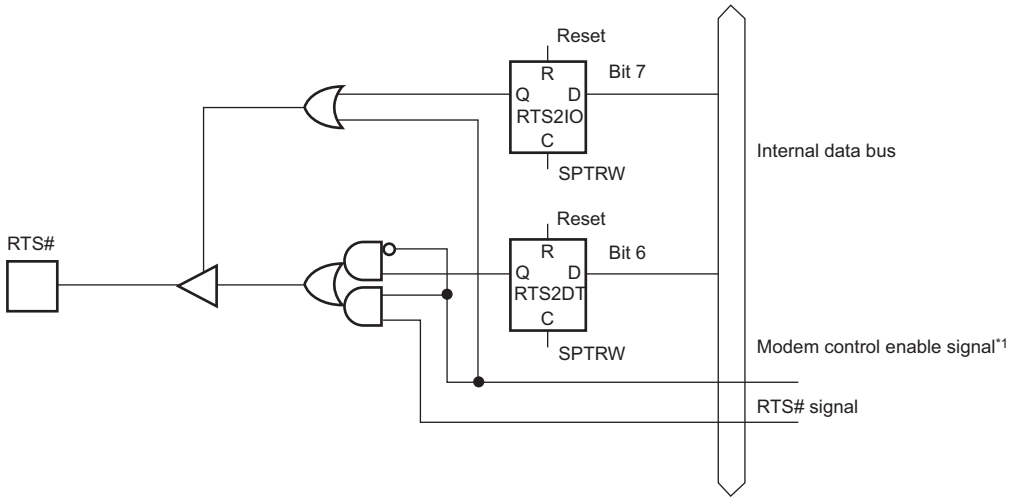
Table 24.17 SCIFA Interrupt Sources

Name	Level/ Edge	Interrupt Source	Interrupt Enable Bit	DMAC Activation	Priority
BRIF	Level	Interrupt caused by break (BRK) or overrun (ORER).	RIE or REIE	Impossible	High
ERIF	Level	Interrupt caused by framing or parity (ER).	RIE or REIE	Impossible	
RXIF	Level	Interrupt caused by receive FIFO data full (RDF).	RIE	Possible	
TXIF	Level	Interrupt caused by transmit FIFO data empty (TDFE).	TIE	Possible	
TEIF	Level	Interrupt caused by transmit end (TEND).	TEIE	Impossible	
DRIF	Level	Interrupt caused by receive data ready (DR).	RIE	Impossible	Low

Note: The TEIF and DRIF interrupts share the same vector number.

24.6 Serial Port Register (SPTR) and SCIFA-Related Pins

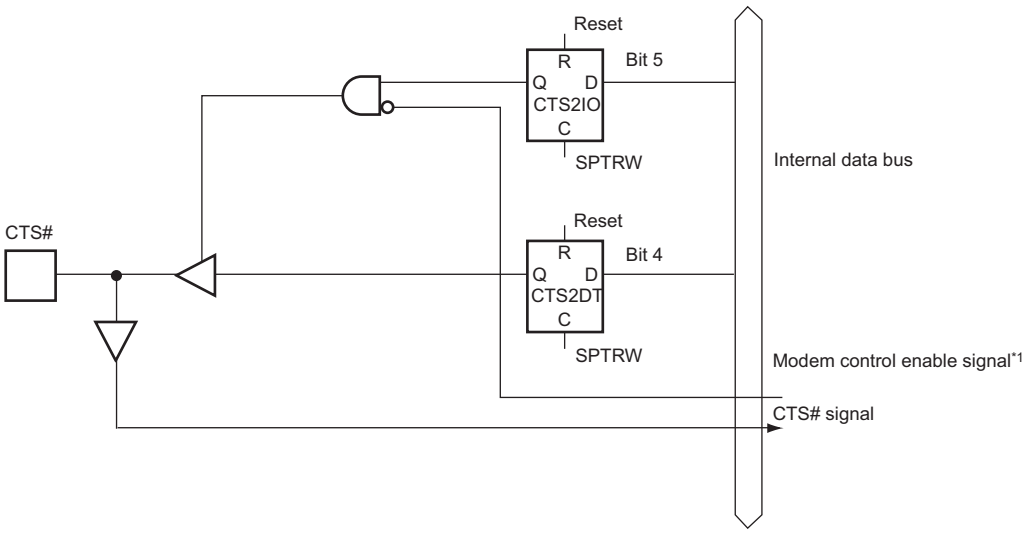
Figure 24.19 to Figure 24.22 show the relationships between the SPTR register and the SCIFA-related pins.



SPTRW: SPTR write

Note 1. These signals allow the RTS# pin to enter the modem control state with the MCE bits in FCR.

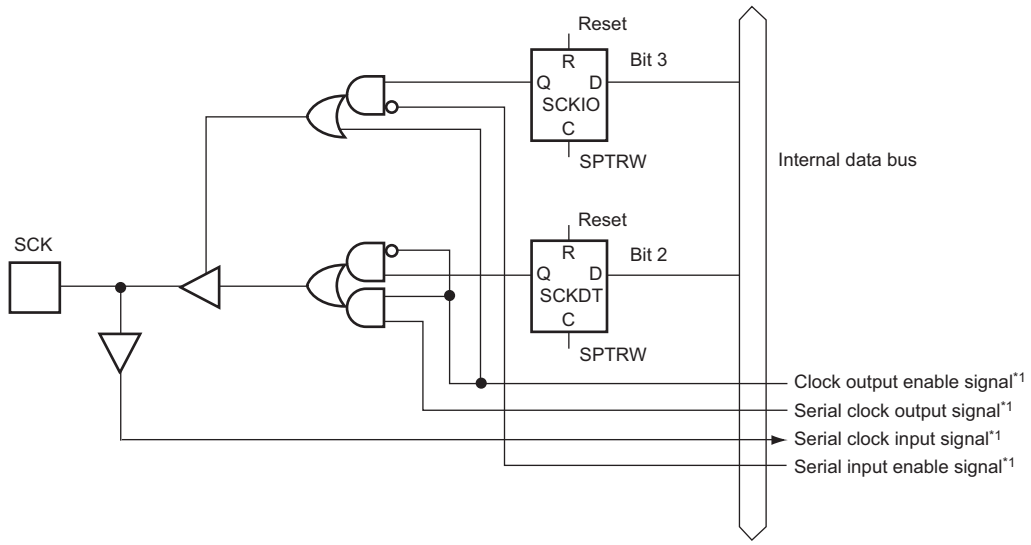
Figure 24.19 RTS2IO Bit and RTS2DT Bit in the SPTR Register, and RTS# Pin



SPTRW: SPTR write

Note 1. These signals allow the CTS# pin to enter the modem control state with the MCE bits in FCR.

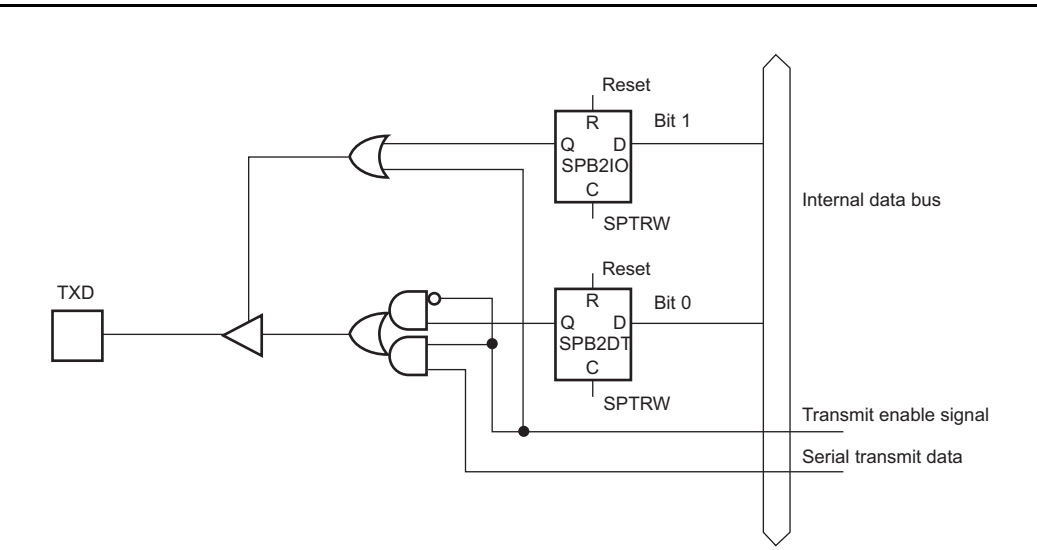
Figure 24.20 CTS2IO Bit and CTS2DT Bit in the SPTR Register, and CTS# Pin



SPTRW: SPTR write

Note 1. These signals control the SCK pin according to the settings of the CM bit in SMR and CKE[1:0] bits in SCR.

Figure 24.21 SCKIO Bit and SCKDT Bit in the SPTR Register, and SCK Pin



SPTRW: SPTR write

Figure 24.22 SPB2IO Bit and SPB2DT Bit in the SPTR Register, and TXD Pin

24.7 Noise Cancellation

Figure 24.23 shows the configuration of the noise filter used for noise cancellation. The noise filter consists of a two-stage flip-flop circuits and a match detection circuit. When the input signals of the noise filter and the output signals of the two-stage flip-flop circuits completely match, the matched level is conveyed as an internal signal. Unless otherwise matched, the previous value is retained. (When the levels sampled on three consecutive cycles of the sampling clock of the noise filter match, the signal is considered valid. If three consecutive sampled values do not match, the signal is considered to be noise rather than a received signal).

In asynchronous mode, the noise cancellation can be applied to the receive signal input to the RXDn pin. The receive level of the RXDn pin is taken in the flip-flop circuit of the noise filter on the base clock (the clock with a frequency 16, 8, or 4 times the transfer rate*1).

If the base clock is stopped once with the noise filter enabled and then the base clock input is restarted again, the noise filter operation resumes from the state where the clock was stopped. When SCR.RE is set to 0 during input of the base clock, the noise filter outputs 0 as the internal RxDn signal. The internal match detector continues operating even while operations for reception are stopped, and the result from the last time previous consecutive samples matched is output at the same time as operations for reception are resumed.

Note 1. A frequency 16 times bit rate when the SEMR.ABCS0 bit and the SEMR.BGDM bit are both 0, a frequency 8 times bit rate when either the SEMR.ABCS0 bit or the SEMR.BGDM bit is 1, and a frequency 4 times bit rate when the SEMR.ABCS0 bit and the SEMR.BGDM bit are both 1.

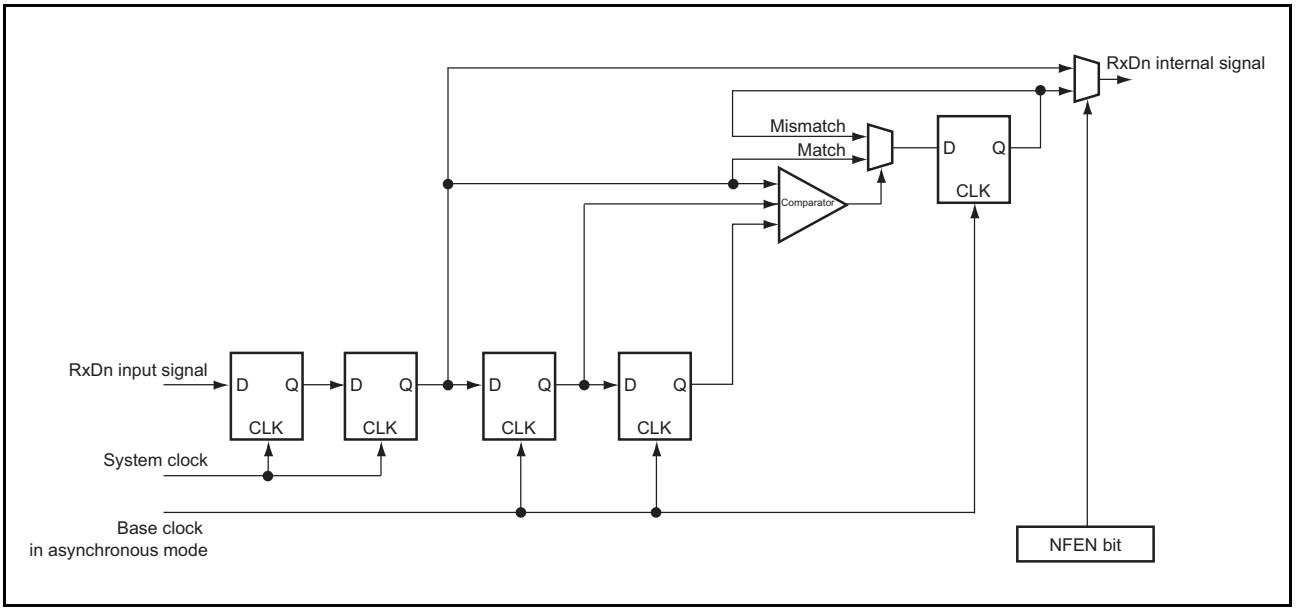


Figure 24.23 Block Diagram of Digital Noise Filter Circuit

24.8 Usage Notes

The following is the notes on using the SCIFA.

24.8.1 FTDR Register Writing and TDFE Flag

The TDFE flag in the serial status register (FSR) is set when the number of transmit data bytes written in the transmit FIFO data register (FTDR) has fallen below the transmission trigger number set by bits TTRG[1:0] in the FIFO control register (FCR) or bits TFTC[4:0] in the FIFO trigger control register (FTCR). After the TDFE flag is set, transmit data up to the number of empty bytes in the FTDR register can be written, allowing efficient continuous transmission.
However, if the number of data bytes written in the FTDR register is equal to or less than the specified transmission trigger number, the TDFE flag will be set to 1 again even after being read as 1 and cleared to 0.
In case of data writing to the FTDR register by the DMAC, the FSR.TDFE flag remains 1 after the DMAC transfer completes. However, DMAC transfer is possible regardless of the FSR.TDFE flag setting.
The number of transmit data bytes in the FTDR register can be checked by the 8 higher-order bits of the FIFO data count register (FDR).

24.8.2 FRDR Register Reading and RDF Flag

The RDF flag in the serial status register (FSR) is set when the number of receive data bytes in the receive FIFO data register (FRDR) has become equal to or greater than the reception trigger number set by bits RTRG[1:0] in the FIFO control register (FCR) or bits RFTC[4:0] in the FIFO trigger control register (FTCR). After the RDF flag is set, receive data equivalent to the trigger number can be read from the FRDR register, allowing efficient continuous reception.
However, if the number of data bytes in the FRDR register exceeds the reception trigger number, the RDF flag will be set to 1 again even after being read as 1 and cleared to 0.
In case of data reading from the FRDR register by the DMAC, the FSR.RDF flag remains 1 after the DMAC transfer completes. However, DMAC transfer is possible regardless of the FSR.RDF flag setting.
The number of receive data bytes in the FRDR register can be checked by the 8 lower-order bits of the FIFO data count register (FDR).

24.8.3 Break Detection and Processing

When a framing error (FER) is detected, a break signal can be detected by reading the RXD pin value directly. In a break, the input from the RXD pin becomes all low. Therefore, the FER flag in the serial status register (FSR) is set to 1 and the parity error flag (PER) may also be set to 1.
Upon detection of a break signal, the SCIFA stops the received data transfer to the FRDR register but continues the receive operation.

24.8.4 Writing to the SPTR Register

b6, b4, b2, and b0 of the SPTR register respectively indicate the input status of their corresponding pins. (See the description of each bit of section 24.2.12, Serial Port Register (SPTR) for details.)
Writings to these bits in 1-bit unit are handled as read-modify-write, which may lead to undesired values to be written. To avoid this, when modifying the SPB2DT or SPB2IO bit, for example, write the other bit (the bit used in combination) at the same time.

24.8.5 Break Signal Transmission

The output signal from the TXD pin is determined by the SPB2IO bit and the SPB2DT bit in the serial port register (SPTR). The break signal can be sent by using these bits.
The TXD pin does not function as a transmit data output pin during the period from when the SCIFA is initialized to when the TE bit in the SCR register is set to 1 (transmission possible). The TXD pin status during this period is replaced by the SPB2DT bit value. Therefore, the SPB2IO and SPB2DT bits in the SPTR register must have been set to 1 (high output) at first (mark (high) status).

To transmit the break signal during serial transmission, set the SPB2IO bit in the SPTR register to 1, clear the SPB2DT bit to 0 (specify a low level), and then clear the TE bit in the SCR register to 0 (transmission stop). Clearing the TE bit to 0 initializes the transmitter regardless of the current transmission status, and outputs a low level from the TXD pin.

24.8.6 Receive Data Sampling Timing and Receive Margin in Asynchronous Mode

The SCIFA operates on a base clock with a frequency 16 times the transfer rate*1. In reception, the SCIFA internally latches the received data at the rising edge of the eighth base clock pulse*1. The timing is shown in Figure 24.24.

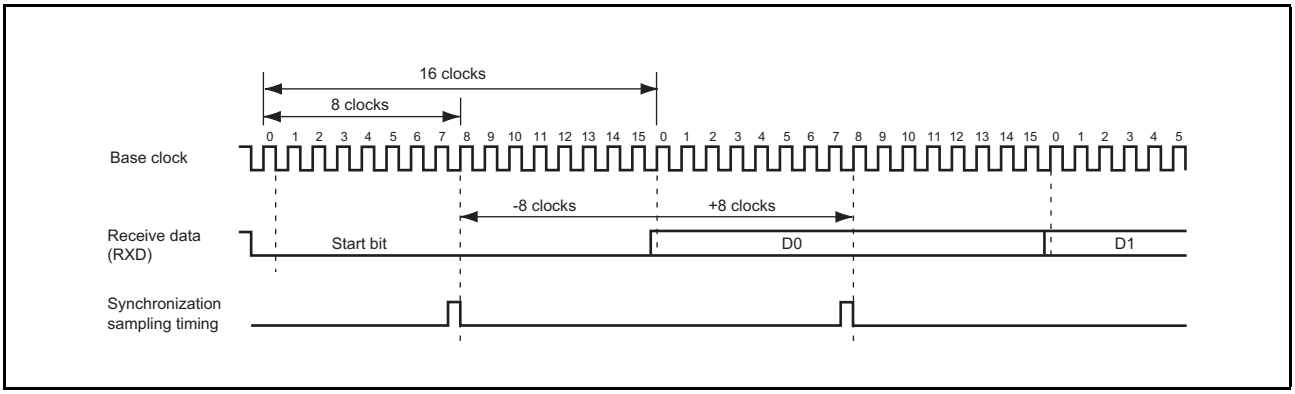


Figure 24.24 Receive Data Sampling Timing in Asynchronous Mode

Note 1. This is an example when the SEMR.ABCS0 bit is 0. When the ABCS0 bit is 1, a frequency of 8 times the bit rate is used as a base clock and receive data is sampled at the rising edge of the 4th pulse of the base clock. The receive margin in asynchronous mode can therefore be expressed as shown in equation 1.

Equation 1:

$$M = \{(0.5 - \frac{1}{2N}) - (L - 0.5)F - \frac{|D - 0.5|}{N}(1 + F)\} \times 100[\%]$$

Where: M: Receive margin (%)
N: Ratio of clock frequency to bit rate (N = 16)
D: Clock duty (D = 0 to 1.0)
L: Frame length (L = 9 to 12)
F: Absolute deviation of clock frequency

From equation 1, if F = 0 and D = 0.5, the receive margin is 46.875%, as given by equation 2.

Equation 2:

When D = 0.5 and F = 0:
M = (0.5 - 1/(2 ×16)) ×100% = 46.875%

This is a theoretical value. A reasonable margin to allow in system designs is 20% to 30%.

24.8.7 Note on FER Flag and PER Flag in Serial Status Register (FSR)

The FER flag and PER flag in the serial status register (FSR) are status flags that apply to next entry to be read from the receive FIFO data register (FRDR). After the CPU reads the receive FIFO data register, the flags of framing errors and parity errors in the receive data will be cleared. To check the received data for the states of framing errors and parity errors, only read the receive FIFO data register after reading the serial status register.

24.8.8 Notes on External Clock Input in Clock Synchronous Mode

Before setting the TE and RE bits in the serial control register (SCR) to 1, wait for four or more cycles of the peripheral operating clock after the external clock (SCK) is changed from 0 (low) to 1 (high). To input the external clock (SCK) (to start communication), wait for one or more cycles of the external clock after the TE and RE bits in the SCR register are set to 1.

24.8.9 Module Standby Mode Setting

SCIFA operation can be disabled or enabled using the standby control register. As the initial setting, the SCIFA operation is halted. Register access is enabled by clearing module standby mode. For details, refer to section 9, Low-Power Consumption Function.

24.8.10 Notes on Operation for Reception when an Internal Clock is Selected in Clock Synchronous Mode

When an internal clock is selected as the clock for reception in clock-synchronous mode, if the number of data stored through the receive FIFO data register (FRDR) becomes equal to or greater than the specified reception trigger number, the RDF flag is set, the RXIF interrupt request is generated and, at the same time, output of the synchronizing clock and reception of serial data are stopped. Once the number of data are again less than the specified reception trigger number, output of the synchronizing clock and the reception of serial data are restarted. In addition, if an internal clock is selected for reception in clock synchronous mode, the ORER flag is not set to 1 since no overrun occurs. Accordingly, overruns (indicated by the ORER flag) cannot be used as a BRIF interrupt source.

24.8.11 Notes on Initialization of the SCIFA

In the SCIFA initialization process, clearing of the TE and RE bits of the serial control register (SCR) should be taken place at the same time or the clearing of the RE bit should precede that of the TE bit. This is because clearing the TE bit to 0 while the RE bit remains 1 enables data reception and may lead to start of unintended data reception.

25. I²C Bus Interface (RIICa)

This LSI has two I2C bus interfaces (RIIC modules).
The RIIC module conforms with and provides a subset of the NXP I2C bus (Inter-IC-Bus) interface functions.

25.1 Overview

Table 25.1 lists the specifications of the RIIC, Figure 25.1 shows a block diagram of the RIIC, and Figure 25.2 shows an example of I/O pin connections to external circuits (I2C bus configuration example). Table 25.2 lists the I/O pins of the RIIC.

Table 25.1 RIIC Specifications (1 / 2)

Item	Description
Communications format	• I2C bus format • Master mode or slave mode selectable • Automatic securing of the various setup times, hold times, and bus-free times for the transfer rate
Transfer rate	Up to 400 kbps: fast mode
SCL clock	For master operation, the duty cycle of the SCL clock is selectable in the range from 4% to 96%.
Issuing and detecting conditions	Start, restart, and stop conditions are automatically generated. Start conditions (including restart conditions) and stop conditions are detectable.
Slave address	• Up to three different slave addresses can be set. • 7-bit and 10-bit address formats are supported (along with the use of both at once). • General call addresses and device ID addresses are detectable.
Acknowledgement	• For transmission, the acknowledge bit is automatically loaded. Transfer of the next data for transmission can be automatically suspended on detection of a not-acknowledge bit. • For reception, the acknowledge bit is automatically transmitted. If a wait between the eighth and ninth clock cycles has been selected, software control of the value in the acknowledge field in response to the received value is possible.
Wait function	• In reception, the following periods of waiting can be obtained by holding the SCL clock at the low level: Waiting between the eighth and ninth clock cycles Waiting between the ninth clock cycle and the first clock cycle of the next transfer
SDA output delay function	Timing of the output of transmitted data, including the acknowledge bit, can be delayed.
Arbitration	• Synchronized operation of multiple SCL clocks to avoid contention with other masters (providing support for multiple masters). • Loss in arbitration as a master Detection of mismatches of state between the SDA signal and another signal on the SDA line when a start condition is issued. Detection of a start condition being issued while in the bus busy state. Detection of mismatches of state between the data being transmitted and the signal on the SDA line in transmission as a master. • Loss in arbitration on NACK transmission Detection of mismatches of state between the data being transmitted and the signal on the SDA line in transmission of a not-acknowledge signal. • Loss in arbitration as a slave Detection of mismatches of state between the data being transmitted and the signal on the SDA line in transmission as a slave.
Timeout function	The internal timeout function is capable of detecting long-interval stop of the SCL clock.
Noise cancellation	The interface incorporates digital noise filters for both the SCL and SDA signals, and the width for noise cancellation by the filters is adjustable by software.
Interrupt sources	Four sources: • Error in transfer or occurrence of events Detection of arbitration, NACK, timeout, a start condition including a restart condition, or a stop condition • Receive-data-full (including matching with a slave address) • Transmit-data-empty (including matching with a slave address) • Transmission complete
Low-power consumption function	Module-stop state can be set.

Table 25.1 RIIC Specifications (2 / 2)

Item	Description
RIIC operating modes	• Four modes: Master transmission mode, master reception mode, slave transmission mode, and slave reception mode
Event link function (output)	Four sources: • Error in transfer or occurrence of events Detection of arbitration, NACK, timeout, a start condition including a restart condition, or a stop condition • Receive-data-full (including matching with a slave address) • Transmit-data-empty (including matching with a slave address) • Transmission complete

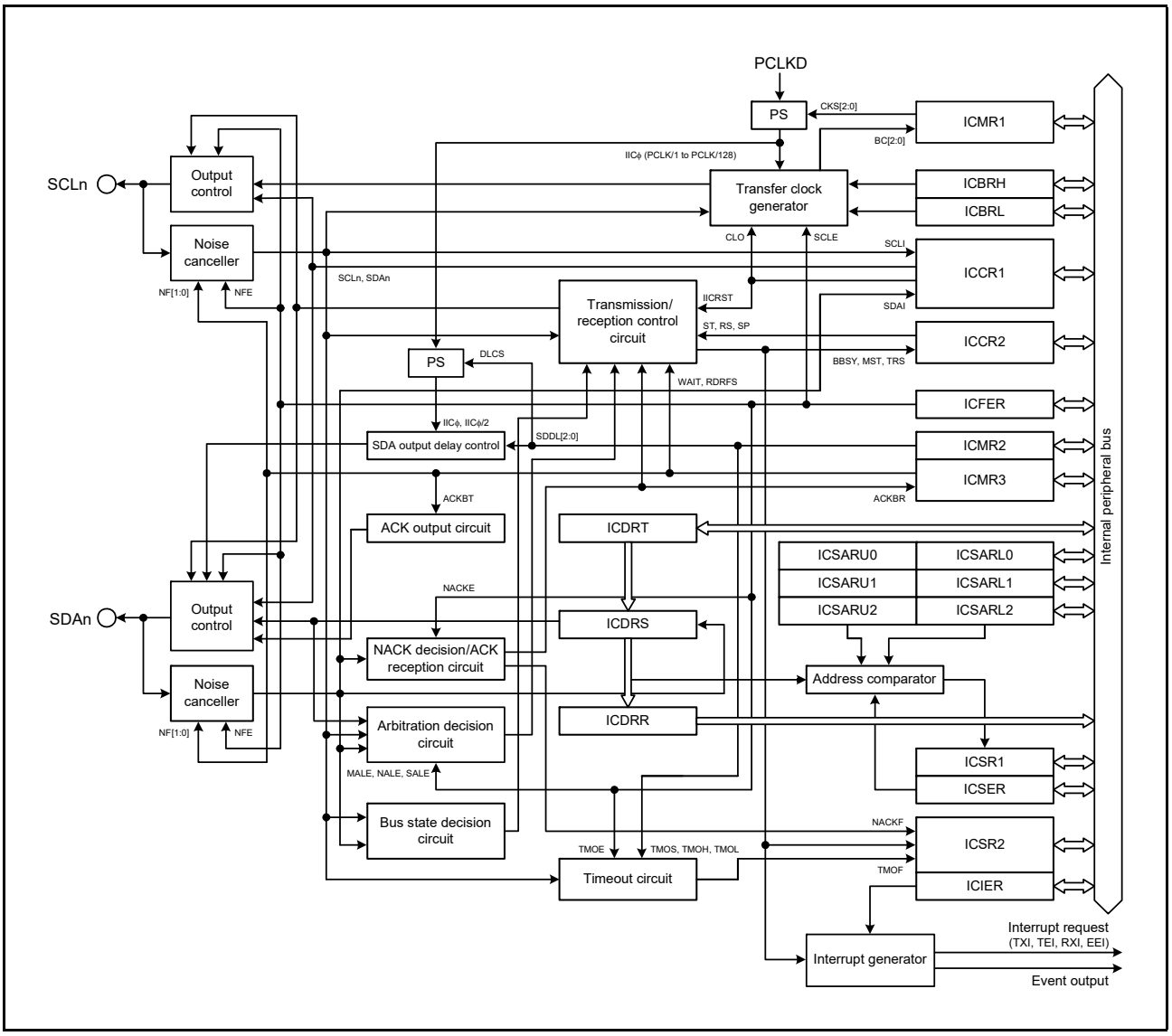


Figure 25.1 RIIC Block Diagram

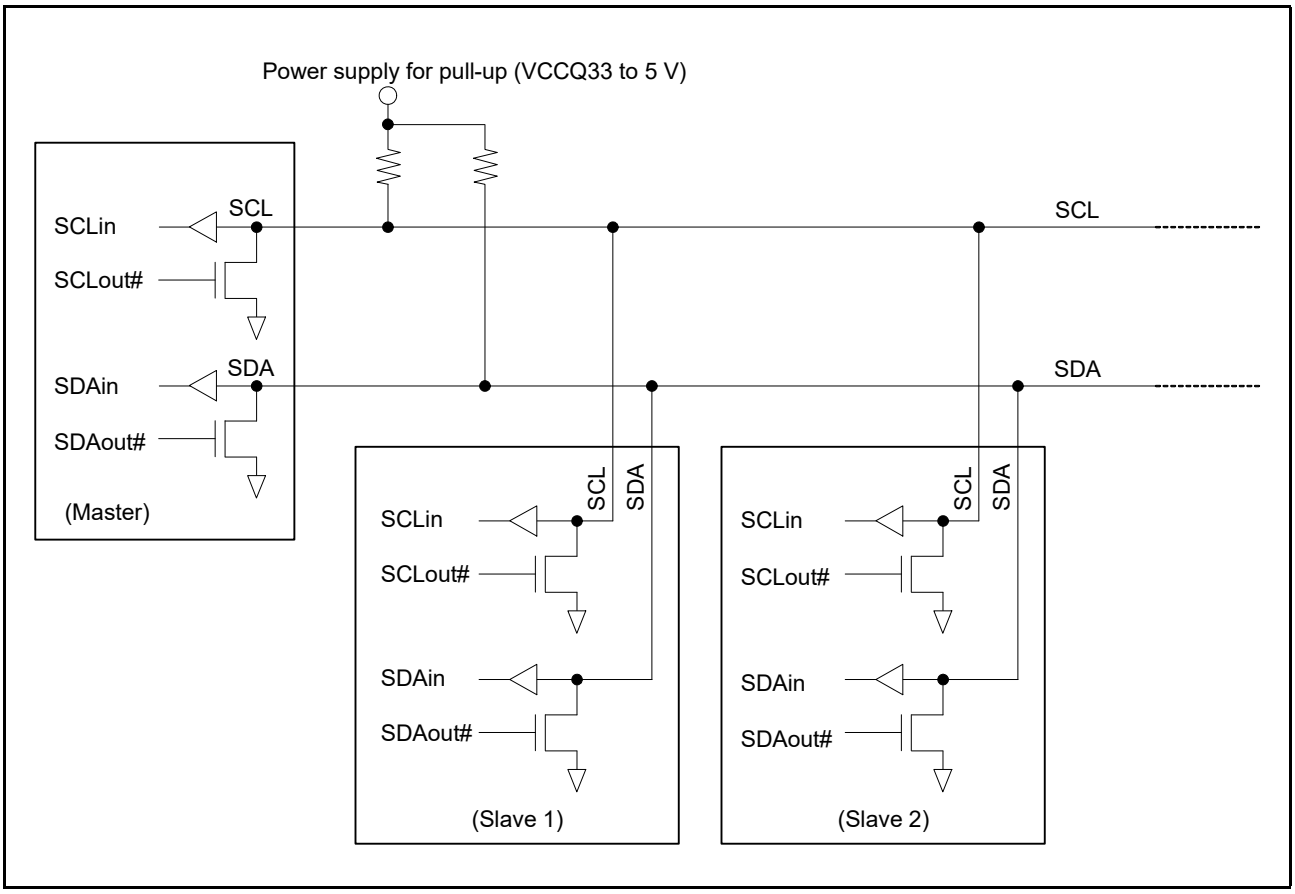


Figure 25.2 Connections to the External Circuit by the I/O Pins (I²C Bus Configuration Example)

The input level of the signals for RIIC is CMOS.

Table 25.2 Pin Configuration

Channel	Pin Name	I/O	Function
RIIC0	SCL0	I/O	RIIC0 serial clock I/O pin
	SDA0	I/O	RIIC0 serial data I/O pin
RIIC1	SCL1	I/O	RIIC1 serial clock I/O pin
	SDA1	I/O	RIIC1 serial data I/O pin

25.2 Register Descriptions

25.2.1 I²C Bus Control Register 1 (ICCR1)

The ICCR1 register controls the SDAn and SCLn signals output by the RIIC (n = 0, 1).

Address(es): RIIC0.ICCR1 A008 0900h, RIIC1.ICCR1 A008 0940h

	b7	b6	b5	b4	b3	b2	b1	b0
	ICE	IICRST	CLO	SOWP	SCLO	SDAO	SCLI	SDAI
Value after reset:	0	0	0	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b0	SDAI	SDA Line Monitor	0: SDAn line is low. 1: SDAn line is high.	R
b1	SCLI	SCL Line Monitor	0: SCLn line is low. 1: SCLn line is high.	R
b2	SDAO	SDA Output Control/Monitor	• Read: 0: SDAn pin output is low. 1: SDAn pin output is high. • Write: 0: SDAn pin output is set to low. 1: SDAn pin output is changed to high impedance. (High level output is achieved through an external pull-up resistor.)	R/W
b3	SCLO	SCL Output Control/Monitor	• Read: 0: SCLn pin output is low. 1: SCLn pin output is high. • Write: 0: SCLn pin output is set to low. 1: SCLn pin output is changed to high impedance. (High level output is achieved through an external pull-up resistor.)	R/W
b4	SOWP	SCLO/SDAO Write Protect	0: Bits SCLO and SDAO can be written. 1: Bits SCLO and SDAO are protected. (This bit is always read as 1.)	R/W
b5	CLO	Extra SCL Clock Cycle Output	0: Does not output an extra SCL clock cycle (default). 1: Outputs an extra SCL clock cycle. (The CLO bit is cleared automatically after one clock cycle is output.)	R/W
b6	IICRST	I²C Bus Interface Internal Reset	0: Releases the RIIC reset or internal reset. 1: Initiates the RIIC reset or internal reset. (Clears the bit counter and the SCLn/SDAn output latch)	R/W
b7	ICE	I²C Bus Interface Enable	0: Disable (SCLn and SDAn pins in inactive state) 1: Enable (SCLn and SDAn pins in active state) (Combined with the IICRST bit to select either RIIC or internal reset.)	R/W

SDAO Bit (SDA Output Control/Monitor) and SCLO Bit (SCL Output Control/Monitor)

These bits are used to directly control the SDAn and SCLn signals output from the RIIC.

When writing to these bits, also write 0 to the SOWP bit.

The result of setting these bits is input to the RIIC via the input buffer. When slave mode is selected, a start condition may be detected and the bus may be released depending on the bit settings.

Do not rewrite these bits during a start condition, stop condition, restart condition, or during transmission or reception. Operation after rewriting under the above conditions is not guaranteed.

When reading these bits, the state of signals output from the RIIC can be read.

CLO Bit (Extra SCL Clock Cycle Output)

This bit is used to output an extra SCL clock cycle for debugging or error processing.
Normally, set the bit to 0. Setting the bit to 1 in a normal communication state causes a communication error.
For details on this function, refer to section 25.11.2, Extra SCL Clock Cycle Output Function.

IICRST Bit (I²C Bus Interface Internal Reset)

This bit is used to reset the internal states of the RIIC.
Setting this bit to 1 initiates an RIIC reset or internal reset.
Whether an RIIC reset or internal reset is initiated is determined according to the combination with the ICE bit. Table 25.3 lists the resets of the RIIC.
The RIIC reset resets all registers and internal states of the RIIC, and the internal reset resets the bit counter (ICMR1.BC[2:0] bits), the I²C bus shift register (ICDRS), and the I²C bus status registers (ICSR1 and ICSR2) as well as the internal states of the RIIC. For the reset conditions for each register, refer to section 25.13, Resets and Register and Function States When Issuing Each Condition.
An internal reset initiated with the IICRST bit set to 1 during operation (with the ICE bit set to 1) resets the internal states of the RIIC without initializing the port settings and the control and setting registers of the RIIC when the bus or RIIC hangs up due to a communication error.
If the RIIC hangs up in a low level output state, resetting the internal states cancels the low level output state and releases the bus with the SCLn pin and SDAn pin at a high impedance.

Note: If an internal reset is initiated using the IICRST bit for a bus hang-up occurred during communication with the master device in slave mode, the states may become different between the slave device and the master device (due to the difference in the bit counter information). For this reason, do not initiate an internal reset in slave mode, but initiate restoration processing from the master device. If an internal reset is necessary because the RIIC hangs up with the SCLn line in a low level output state in slave mode, initiate an internal reset and then issue a restart condition from the master device or resume communication from the start condition issuance after issuing a stop condition. If communication is restarted by initiating a reset solely in the slave device without issuing a start condition or restart condition from the master device, synchronization will be lost because the master and slave devices operate asynchronously.

Table 25.3 RIIC Resets

IICRST	ICE	State	Specifications
1	0	RIIC reset	Resets all registers and internal states of the RIIC.
	1	Internal reset	Resets the ICMR1.BC[2:0] bits, the ICSR1, ICSR2, and ICDRS registers, and the internal states of the RIIC.

ICE Bit (I²C Bus Interface Enable)

This bit selects the active or inactive state of the SCLn and SDAn pins. It can also be combined with the IICRST bit to initiate two types of resets. See Table 25.3, RIIC Resets, for the types of resets.
Set the ICE bit to 1 when using the RIIC. The SCLn and SDAn pins are placed in the active state when the ICE bit is set to 1.
Set the ICE bit to 0 when the RIIC is not to be used. The SCLn and SDAn pins are placed in the inactive state when the ICE bit is set to 0.

25.2.2 I²C Bus Control Register 2 (ICCR2)

The ICCR2 register controls start condition issuance, restart condition issuance, and stop condition issuance.

Address(es): RIIC0.ICCR2 A008 0901h, RIIC1.ICCR2 A008 0941h

	b7	b6	b5	b4	b3	b2	b1	b0
	BBSY	MST	TRS	—	SP	RS	ST	—
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b1	ST	Start Condition Issuance Request	0: Does not request to issue a start condition. 1: Requests to issue a start condition.	R/W
b2	RS	Restart Condition Issuance Request	0: Does not request to issue a restart condition. 1: Requests to issue a restart condition.	R/W
b3	SP	Stop Condition Issuance Request	0: Does not request to issue a stop condition. 1: Requests to issue a stop condition.	R/W
b4	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b5	TRS	Transmission/Reception Mode	0: Reception mode 1: Transmission mode	R/W *1
b6	MST	Master/Slave Mode	0: Slave mode 1: Master mode	R/W *1
b7	BBSY	Bus Busy Detection Flag	0: The I²C bus is released (bus free state). 1: The I²C bus is occupied (bus busy state).	R

Note 1. When the ICMR1.MTWP bit is set to 1, the MST and TRS bits can be written to.

ST Bit (Start Condition Issuance Request)

This bit is used to request transition to master mode and issuance of a start condition.

When this bit is set to 1 to request to issue a start condition, a start condition is issued when the BBSY flag is set to 0 (bus free state).

For details on the start condition issuance, refer to section 25.10, Start Condition/Restart Condition/Stop Condition Issuing Function.

[Setting condition]

- When 1 is written to the ST bit

[Clearing conditions] One of the following conditions is satisfied:

- When 0 is written to the ST bit
- When a start condition has been issued (A start condition is detected)
- When the AL (arbitration-lost) flag in ICSR2 is set to 1
- When 1 is written to the IICRST bit in ICCR1 to apply an RIIC reset or an internal reset

Note: Set the ST bit to 1 (start condition issuance request) when the BBSY flag is set to 0 (bus free state). Note that setting the ST bit to 1 (start condition issuance request) when the BBSY flag is set to 1 (bus busy state) is handled as a start condition issuance error and arbitration may be lost.

RS Bit (Restart Condition Issuance Request)

This bit is used to request that a restart condition be issued in master mode.

When this bit is set to 1 to request to issue a restart condition, a restart condition is issued when the BBSY flag is set to 1 (bus busy state) and the MST bit is set to 1 (master mode).

For details on the restart condition issuance, refer to section 25.10, Start Condition/Restart Condition/Stop Condition Issuing Function.

[Setting condition]

- When 1 is written to the RS bit with the BBSY flag in ICCR2 set to 1 (1 cannot be written with the BBSY flag in ICCR2 set to 0)

[Clearing conditions] One of the following conditions is satisfied:

- When 0 is written to the RS bit
- When a restart condition has been issued (A start condition is detected)
- When the AL (arbitration-lost) flag in ICSR2 is set to 1
- When 1 is written to the IICRST bit in ICCR1 to apply an RIIC reset or an internal reset

Note: Do not set the RS bit to 1 while issuing a stop condition.

Note: We recommend that you issue the restart condition in master transmission mode. If 1 (requests to issue a restart condition) is written to the RS bit in slave mode, the restart condition is not issued but the RS bit remains set to 1. If the operating mode changes to master mode with the bit not being cleared, note that the restart condition may be issued.

SP Bit (Stop Condition Issuance Request)

This bit is used to request that a stop condition be issued in master mode.

When this bit is set to 1 to request to issue a stop condition, a stop condition is issued when the BBSY flag is set to 1 (bus busy state) and the MST bit is set to 1 (master mode).

For details on the stop condition issuance, refer to section 25.10, Start Condition/Restart Condition/Stop Condition Issuing Function.

[Setting condition]

- When 1 is written to the SP bit with both the BBSY flag and the MST bit in ICCR2 set to 1

[Clearing conditions] One of the following conditions is satisfied:

- When 0 is written to the SP bit
- When a stop condition has been issued (A stop condition is detected)
- When the AL (arbitration-lost) flag in ICSR2 is set to 1
- When a start condition and a restart condition are detected
- When 1 is written to the IICRST bit in ICCR1 to apply an RIIC reset or an internal reset

Note: Writing to the SP bit is not possible while the setting of the BBSY flag is 0 (bus free state).

Note: Do not set the SP bit to 1 while a restart condition is being issued.

TRS Bit (Transmission/Reception Mode)

This bit indicates transmit or reception mode.
The RIIC is in reception mode when the TRS bit is set to 0 and is in transmission mode when the bit is set to 1.
Combination of this bit and the MST bit indicates the operating mode of the RIIC.
The value of TRS bit is automatically changed to 1 for transmission mode or 0 for reception mode by issuing or detection of a start condition and setting of the R/W# bit. Although writing to the TRS bit is possible when the MTWP bit in ICMR1 is set to 1, writing to this bit is not necessary during normal usage.
[Setting conditions] One of the following conditions is satisfied:

- When a start condition is issued normally according to the start condition issuance request (when a start condition is detected with the BBSY flag set to 0 (bus free state) and the ST bit set to 1)
- When a restart condition is issued normally according to the restart condition issuance request (when a restart condition is detected with the RS bit set to 1)
- When the R/W# bit added to the slave address is set to 0 in master mode
- When the address received in slave mode matches the address enabled in ICSER, with the R/W# bit set to 1
- When 1 is written to the TRS bit with the MTWP bit in ICMR1 set to 1

[Clearing conditions] One of the following conditions is satisfied:

- When a stop condition is detected
- The AL (arbitration-lost) flag in ICSR2 being set to 1
- In master mode, reception of a slave address to which an R/W# bit with the value 1 is appended
- In slave mode, a match between the received address and the address enabled in ICSER when the value of the received R/W# bit is 0 (including cases where the received address is the general call address)
- In slave mode, a restart condition is detected (a start condition is detected with ICCR2.BBSY = 1 and ICCR2.MST = 0)
- When 0 is written to the TRS bit with the MTWP bit in ICMR1 set to 1
- When 1 is written to the IICRST bit in ICCR1 to apply an RIIC reset or an internal reset

The R/W# bit, which is contained in transmit data, indicates the transmit and receive direction. Data is transferred from the slave device to the master device when the R/W# bit is 1, or from the master device to the slave device when the R/W# bit is 0.

MST Bit (Master/Slave Mode)

This bit indicates master or slave mode.
The RIIC is in slave mode when the MST bit is set to 0 and is in master mode when the bit is set to 1. Combination of this bit and the TRS bit indicates the operating mode of the RIIC.
The value of the MST bit is automatically changed to 1 for master mode or 0 for slave mode by issuing of a start condition and issuing or detection of a stop condition, etc. Although writing to the MST bit is possible when the MTWP bit in ICMR1 is set to 1, writing to this bit is not necessary during normal usage.
[Setting conditions] One of the following conditions is satisfied:

- When a start condition is issued normally according to the start condition issuance request (when a start condition is detected with the BBSY flag set to 0 (bus free state) and the ST bit set to 1)
- When 1 is written to the MST bit with the MTWP bit in ICMR1 set to 1

[Clearing conditions] One of the following conditions is satisfied:

- When a stop condition is detected
- When the AL (arbitration-lost) flag in ICSR2 is set to 1
- When 0 is written to the MST bit with the MTWP bit in ICMR1 set to 1
- When 1 is written to the IICRST bit in ICCR1 to apply an RIIC reset or an internal reset

BBSY Flag (Bus Busy Detection Flag)

The BBSY flag indicates whether the I2C bus is occupied (bus busy state) or released (bus free state).
This bit is set to 1 when the SDAn line changes from high to low under the condition of SCLn line = high, assuming that a start condition has been issued.
When the SDAn line changes from low to high under the condition of SCLn line = high, this bit is set to 0 after the bus free time (specified in ICBRL) start condition is not detected, assuming that a stop condition has been issued.
[Setting condition]

- When a start condition is detected

[Clearing conditions] One of the following conditions is satisfied:

- When the bus free time (specified in ICBRL) start condition is not detected after detecting a stop condition
- When 1 is written to the IICRST bit in ICCR1 with the ICE bit in ICCR1 set to 0 (RIIC reset)

25.2.3 I²CBus Mode Register 1 (ICMR1)

The ICMR1 register specifies the number of bits of the down counter and the reference clock source.

Address(es): RIIC0.ICMR1 A008 0902h, RIIC1.ICMR1 A008 0942h

	b7	b6	b5	b4	b3	b2	b1	b0
	MTWP	CKS[2:0]			BCWP	BC[2:0]		
Value after reset:	0	0	0	0	1	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	BC[2:0]	Bit Counter	b2 b0 0 0 0: 9 bits 0 0 1: 2 bits 0 1 0: 3 bits 0 1 1: 4 bits 1 0 0: 5 bits 1 0 1: 6 bits 1 1 0: 7 bits 1 1 1: 8 bits	R/W *1
b3	BCWP	BC Write Protect	0: Enables a value to be written in the BC[2:0] bits. (This bit is read as 1.)	R/W *1
b6 to b4	CKS[2:0]	Internal Reference Clock Selection	Selects the internal reference clock source (IICϕ) for the RIIC. b6 b4 0 0 0: PCLKD/1 clock 0 0 1: PCLKD/2 clock 0 1 0: PCLKD/4 clock 0 1 1: PCLKD/8 clock 1 0 0: PCLKD/16 clock 1 0 1: PCLKD/32 clock 1 1 0: PCLKD/64 clock 1 1 1: PCLKD/128 clock	R/W
b7	MTWP	MST/TRS Write Protect	0: Disables writing to the MST and TRS bits in ICCR2. 1: Enables writing to the MST and TRS bits in ICCR2.	R/W

Note 1. Rewrite the BC[2:0] bits and set the BCWP bit to 0 at the same time.

BC[2:0] Bits (Bit Counter)

These bits function as a counter that indicates the number of bits remaining to be transferred at the detection of a rising edge on the SCLn line. Although these bits are writable and readable, it is not necessary to access these bits under normal conditions.

To write to these bits, specify the number of bits to be transferred plus one (data is transferred with an additional acknowledge bit) between transferred frames when the SCLn line is at a low level.

The values of the BC[2:0] bits return to 000b at the end of a data transfer including the acknowledge bit or when a start condition including a restart condition is detected.

25.2.4 I²C Bus Mode Register 2 (ICMR2)

The ICMR2 register specifies various settings regarding the timeout detection function and SDA output delay function.

Address(es): RIIC0.ICMR2 A008 0903h, RIIC1.ICMR2 A008 0943h

	b7	b6	b5	b4	b3	b2	b1	b0
	DLCS	SDDL[2:0]			—	TMOH	TMOL	TMOS
Value after reset:	0	0	0	0	0	1	1	0

Bit	Symbol	Bit Name	Description	R/W
b0	TMOS	Timeout Detection Time Selection	0: Long mode is selected. 1: Short mode is selected.	R/W
b1	TMOL	Timeout L Count Control	0: Count is disabled while the SCLn line is at a low level. 1: Count is enabled while the SCLn line is at a low level.	R/W
b2	TMOH	Timeout H Count Control	0: Count is disabled while the SCLn line is at a high level. 1: Count is enabled while the SCLn line is at a high level.	R/W
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b6 to b4	SDDL[2:0]	SDA Output Delay Counter	• When ICMR2.DLCS = 0 (IICϕ) b6 b4 0 0 0: No output delay 0 0 1: 1 IICϕ cycle 0 1 0: 2 IICϕ cycles 0 1 1: 3 IICϕ cycles 1 0 0: 4 IICϕ cycles 1 0 1: 5 IICϕ cycles 1 1 0: 6 IICϕ cycles 1 1 1: 7 IICϕ cycles • When ICMR2.DLCS = 1 (IICϕ/2) b6 b4 0 0 0: No output delay 0 0 1: 1 or 2 IICϕ cycles 0 1 0: 3 or 4 IICϕ cycles 0 1 1: 5 or 6 IICϕ cycles 1 0 0: 7 or 8 IICϕ cycles 1 0 1: 9 or 10 IICϕ cycles 1 1 0: 11 or 12 IICϕ cycles 1 1 1: 13 or 14 IICϕ cycles	R/W
b7	DLCS	SDA Output Delay Clock Source Selection	0: The internal reference clock (IICϕ) is selected as the clock source of the SDA output delay counter. 1: The internal reference clock divided by 2 (IICϕ/2) is selected as the clock source of the SDA output delay counter.*1	R/W

Note 1. The setting DLCS = 1 (IICϕ/2) only becomes valid when SCL is at the low level. When SCL is at the high level, the setting DLCS = 1 becomes invalid and the clock source becomes the internal reference clock (IICϕ).

TMOS Bit (Timeout Detection Time Selection)

This bit is used to select long mode or short mode for the timeout detection time when the timeout function is enabled (TMOE bit = 1 in ICFER). When this bit is set to 0, long mode is selected. When this bit is set to 1, short mode is selected. In long mode, the timeout detection internal counter functions as a 16 bit-counter. In short mode, the counter functions as a 14 bit-counter. While the SCLn line is in the state that enables this counter as specified by bits TMOH and TMOL, the counter counts up in synchronization with the internal reference clock (IICϕ) as a count source.
For details on the timeout function, refer to section 25.11.1, Timeout Function.

TMOL Bit (Timeout L Count Control)

This bit is used to enable or disable the internal counter of the timeout function to count up while the SCLn line is held low when the timeout function is enabled (TMOE bit = 1 in ICFER).

TMOH Bit (Timeout H Count Control)

This bit is used to enable or disable the internal counter of the timeout function to count up while the SCLn line is held high when the timeout function is enabled (TMOE bit = 1 in ICFER).

SDDL[2:0] Bits (SDA Output Delay Counter)

The SDA output can be delayed by the SDDL[2:0] setting. This counter works with the clock source selected by the DLCS bit. The setting of this function can be used for all types of SDA output, including the transmission of the acknowledge bit.

Set the SDA output delay time to meet the I^{2}C bus standard (within the data enable time/acknowledge enable time*1) and to be within 250 ns (SCL-clock low-level period - the data setup time). Note that, if a value outside the standard is set, communication with communication devices may malfunction or it may seemingly become a start condition or stop condition depending on the bus state.

For details on this function, refer to section 25.5, Facility for Delaying SDA Output.

Note 1. Data enable time/acknowledge enable time
900 ns (up to 400 kbps: fast mode [Fm])

25.2.5 I²CBus Mode Register 3 (ICMR3)

The ICMR3 register specifies the settings for acknowledgement and wait and digital noise filter.

Address(es): RIIC0.ICMR3 A008 0904h, RIIC1.ICMR3 A008 0944h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	WAIT	RDRFS	ACKWP	ACKBT	ACKBR	NF[1:0]	
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	NF[1:0]	Noise Filter Stage Selection	b1 b0 0 0: Noise of up to one IICϕ cycle is filtered out (single-stage filter). 0 1: Noise of up to two IICϕ cycles is filtered out (2-stage filter). 1 0: Noise of up to three IICϕ cycles is filtered out (3-stage filter). 1 1: Noise of up to four IICϕ cycles is filtered out (4-stage filter).	R/W
b2	ACKBR	Receive Acknowledge	0: A 0 is received as the acknowledge bit (ACK reception). 1: A 1 is received as the acknowledge bit (NACK reception).	R
b3	ACKBT	Transmit Acknowledge	0: A 0 is sent as the acknowledge bit (ACK transmission). 1: A 1 is sent as the acknowledge bit (NACK transmission).	R/W *1
b4	ACKWP	ACKBT Write Protect	0: Modification of the ACKBT bit is disabled. 1: Modification of the ACKBT bit is enabled.	W*1
b5	RDRFS	RDRF Flag Set Timing Selection	0: The RDRF flag is set at the rising edge of the ninth SCL clock cycle. (The SCLn line is not held low at the falling edge of the eighth clock cycle.) 1: The RDRF flag is set at the rising edge of the eighth SCL clock cycle. (The SCLn line is held low at the falling edge of the eighth clock cycle.) Low-hold is released at the ninth cycle since a value is written to the ACKBT bit.	R/W *2
b6	WAIT	WAIT	0: No WAIT (The period between ninth clock cycle and first clock cycle is not held low.) 1: WAIT (The period between ninth clock cycle and first clock cycle is held low.) Low-hold is released by reading ICDRR.	R/W *2
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

Note 1. Write to the ACKBT bit only while the ACKWP bit is already 1. If it is attempted to write 1 to both the ACKWP and ACKBT bits at the same time, the ACKBT bit will not be set to 1.

Note 2. The WAIT and RDRFS bits are valid only in reception mode. These bits are invalid in transmission mode.

NF[1:0] Bits (Number of Noise Filter Stages Select)

These bits are used to select the number of stages in the digital noise filter.

For details on the digital noise filter function, refer to section 25.6, Digital Noise-Filter Circuits.

ACKBR Bit (Receive Acknowledge)

This bit is used to store the acknowledge bit information received from the receive device in transmission mode.
[Setting condition]

- When 1 (Not Acknowledge) is received as the acknowledge bit with the TRS bit in ICCR2 set to 1

[Clearing conditions] One of the following conditions is satisfied:

- When 0 (Acknowledge) is received as the acknowledge bit with the TRS bit in ICCR2 set to 1
- When 1 is written to the IICRST bit in ICCR1 while the ICE bit in ICCR1 is 0 (RIIC reset)

ACKBT Bit (Transmit Acknowledge)

This bit is used to set the bit to be sent at the acknowledge timing in reception mode.
[Setting condition]

- When 1 (Not Acknowledge) is written to this bit with the ACKWP bit set to 1

[Clearing conditions] One of the following conditions is satisfied:

- When 0 (Acknowledge) is written to this bit with the ACKWP bit set to 1
- When stop condition issuance is detected (when a stop condition is detected with the SP bit in ICCR2 set to 1)
- When 1 is written to the IICRST bit in ICCR1 while the ICE bit in ICCR1 is 0 (RIIC reset)

ACKWP Bit (ACKBT Write Protect)

This bit is used to control the modification of the ACKBT bit.

RDRFS Bit (RDRF Flag Set Timing Selection)

This bit is used to select the RDRF flag set timing in reception mode and also to select whether to hold the SCLn line low at the falling edge of the eighth SCL clock cycle.
When the RDRFS bit is 0, the SCLn line is not held low at the falling edge of the eighth SCL clock cycle, and the RDRF flag is set to 1 at the rising edge of the ninth SCL clock cycle.
When the RDRFS bit is 1, the RDRF flag is set to 1 at the rising edge of the eighth SCL clock cycle and the SCLn line is held low at the falling edge of the eighth SCL clock cycle. Low-hold state is released at the ninth clock cycle or later, by writing the ACKBT bit.
After data is received with this setting, the SCLn line is automatically held low before the acknowledge bit is sent. This enables processing to send ACK (ACKBT = 0) or NACK (ACKBT = 1) according to receive data.

WAIT Bit (WAIT)

This bit is used to control whether to hold the period between the ninth SCL clock cycle and the first SCL clock cycle low until the receive data buffer (ICDRR) is completely read each time single-byte data is received in reception mode.
When the WAIT bit is 0, the receive operation is continued without holding the period between the ninth and the first SCL clock cycle low. When both the RDRFS and WAIT bits are 0, continuous receive operation is enabled with the double buffer.
When the WAIT bit is 1, the SCLn line is held low from the falling edge of the ninth clock cycle until the ICDRR value is read each time single-byte data is received. This enables receive operation in byte units.

Note: When writing 0 to the WAIT bit, be sure to read the ICDRR beforehand.

25.2.6 I²CBus Function Enable Register (ICFER)

The ICFER register specifies the settings for various arbitration functions.

Address(es): RIIC0.ICFER A008 0905h, RIIC1.ICFER A008 0945h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	SCLE	NFE	NACKE	SALE	NALE	MALE	TMOE
Value after reset:	0	1	1	1	0	0	1	0

Bit	Symbol	Bit Name	Description	R/W
b0	TMOE	Timeout Function Enable	0: The timeout function is disabled. 1: The timeout function is enabled.	R/W
b1	MALE	Master Arbitration-Lost Detection Enable	0: Master arbitration-lost detection is disabled. (Disables the arbitration-lost detection function and does not clear the MST and TRS bits in ICCR2 automatically when arbitration is lost.) 1: Master arbitration-lost detection is enabled. (Enables the arbitration-lost detection function and clears the MST and TRS bits in ICCR2 automatically when arbitration is lost.)	R/W
b2	NALE	NACK Transmission Arbitration-Lost Detection Enable	0: NACK transmission arbitration-lost detection is disabled. 1: NACK transmission arbitration-lost detection is enabled.	R/W
b3	SALE	Slave Arbitration-Lost Detection Enable	0: Slave arbitration-lost detection is disabled. 1: Slave arbitration-lost detection is enabled.	R/W
b4	NACKE	NACK Reception Transfer Suspension Enable	0: Transfer operation is not suspended during NACK reception (transfer suspension disabled). 1: Transfer operation is suspended during NACK reception (transfer suspension enabled).	R/W
b5	NFE	Digital Noise Filter Circuit Enable	0: No digital noise filter circuit is used. 1: A digital noise filter circuit is used.	R/W
b6	SCLE	SCL Synchronous Circuit Enable	0: No SCL synchronous circuit is used. 1: An SCL synchronous circuit is used.	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

TMOE Bit (Timeout Function Enable)

This bit is used to enable or disable the timeout function.
For details on the timeout function, refer to section 25.11.1, Timeout Function.

MALE Bit (Master Arbitration-Lost Detection Enable)

This bit is used to specify whether to use the arbitration-lost detection function in master mode. Normally, set this bit to 1.
For details on master arbitration lost detection function, see section 25.9.1, Master Arbitration-Lost Detection (MALE Bit).

NALE Bit (NACK Transmission Arbitration-Lost Detection Enable)

This bit is used to specify whether to cause arbitration to be lost when ACK is detected during transmission of NACK in reception mode (such as when slaves with the same address exist on the bus or when two or more masters select the same slave device simultaneously with different number of receive bytes).
For details on NACK transmission arbitration-lost detection function, section 25.9.2, Function to Detect Loss of Arbitration during NACK Transmission (NALE Bit).

SALE Bit (Slave Arbitration-Lost Detection Enable)

This bit is used to specify whether to cause arbitration to be lost when a value different from the value being transmitted is detected on the bus in slave transmission mode (such as when slaves with the same address exist on the bus or when a mismatch with the transmit data occurs due to noise).
For details on slave arbitration-lost detection function, see section 25.9.3, Slave Arbitration-Lost Detection (SALE Bit).

NACKE Bit (NACK Reception Transfer Suspension Enable)

This bit is used to specify whether to continue or discontinue the transfer operation when NACK is received from the slave device in transmission mode. Normally, set this bit to 1.
When NACK is received with the NACKE bit set to 1, the next transfer operation is suspended.
When the NACKE bit is 0, the next transfer operation is continued regardless of the received acknowledge content.
For details on the NACK reception transfer suspension function, refer to section 25.8.2, NACK Reception Transfer Suspension Function.

SCLE Bit (SCL Synchronous Circuit Enable)

This bit is used to specify whether to synchronize the SCL clock with the SCL input clock. Normally, set this bit to 1.
When the SCLE bit is set to 0 (no SCL synchronous circuit used), the RIIC does not synchronize the SCL clock with the SCL input clock. In this setting, the RIIC outputs the SCL clock with the transfer rate set in ICBRH and ICBRL regardless of the SCLn line state. For this reason, if the bus load of the I^2C bus line is much larger than the specification value or if the SCL clock output overlaps in multiple masters, the short-cycle SCL clock that does not meet the specification may be output. When no SCL synchronous circuit is used, it also affects the issuance of a start condition, restart condition, and stop condition, and the continuous output of extra SCL clock cycles.
This bit must not be set to 0 except for checking the output of the set transfer rate.
For details on SCL synchronous circuit function, see section 25.4, SCL Synchronization Circuit.

25.2.7 I²C Bus Status Enable Register (ICSER)

The ICSER register specifies the settings for enabling slave addresses and ID address detection.

Address(es): RIIC0.ICSER A008 0906h, RIIC1.ICSER A008 0946h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	DIDE	—	GCAE	SAR2E	SAR1E	SAR0E
Value after reset:	0	0	0	0	1	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b0	SAR0E	Slave Address Register 0 Enable	0: Slave address in ICSARL0 and ICSARU0 is disabled. 1: Slave address in ICSARL0 and ICSARU0 is enabled.	R/W
b1	SAR1E	Slave Address Register 1 Enable	0: Slave address in ICSARL1 and ICSARU1 is disabled. 1: Slave address in ICSARL1 and ICSARU1 is enabled.	R/W
b2	SAR2E	Slave Address Register 2 Enable	0: Slave address in ICSARL2 and ICSARU2 is disabled. 1: Slave address in ICSARL2 and ICSARU2 is enabled.	R/W
b3	GCAE	General Call Address Enable	0: General call address detection is disabled. 1: General call address detection is enabled.	R/W
b4	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b5	DIDE	Device-ID Address Detection Enable	0: Device-ID address detection is disabled. 1: Device-ID address detection is enabled.	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

SARyE Bit (Slave Address Register y Enable) (y = 0 to 2)

This bit is used to enable or disable the slave address set in ICSARLy and ICSARUy.

When this bit is set to 1, the slave address set in ICSARLy and ICSARUy is enabled and is compared with the received slave address.

When this bit is set to 0, the slave address set in ICSARLy and ICSARUy is disabled and is ignored even if it matches the received slave address.

GCAE Bit (General Call Address Enable)

This bit is used to specify whether to ignore the general call address (0000 000b + 0 [W]: All 0) when it is received.

When this bit is set to 1, if the received slave address matches the general call address, the RIIC recognizes the received slave address as the general call address independently of the slave addresses set in ICSARLy and ICSARUy (y = 0 to 2) and performs data receive operation.

When this bit is set to 0, the received slave address is ignored even if it matches the general call address.

DIDE Bit (Device-ID Address Detection Enable)

This bit is used to specify whether to recognize and execute the Device-ID address when a device ID (1111 100b) is received in the first frame after a start condition or restart condition is detected.

When this bit is set to 1, if the received first frame matches the device ID, the RIIC recognizes that the Device-ID address has been received. When the following R/W# bit is 0 [W], the RIIC recognizes the second and the following frames as slave addresses and continues the receive operation.

When this bit is set to 0, the RIIC ignores the received first frame even if it matches the device ID address and recognizes the first frame as a normal slave address.

For details on the device-ID address detection, refer to section 25.7.3, Device-ID Address Detection.

25.2.8 I²C Bus Interrupt Enable Register (ICIER)

The ICIER register enables or disables interrupt requests regarding RIIC.

Address(es): RIIC0.ICIER A008 0907h, RIIC1.ICIER A008 0947h

	b7	b6	b5	b4	b3	b2	b1	b0
	TIE	TEIE	RIE	NAKIE	SPIE	STIE	ALIE	TMOIE
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	TMOIE	Timeout Interrupt Request Enable	0: Timeout interrupt request (TMOI) is disabled. 1: Timeout interrupt request (TMOI) is enabled.	R/W
b1	ALIE	Arbitration-Lost Interrupt Request Enable	0: Arbitration-lost interrupt request (ALI) is disabled. 1: Arbitration-lost interrupt request (ALI) is enabled.	R/W
b2	STIE	Start Condition Detection Interrupt Request Enable	0: Start condition detection interrupt request (STI) is disabled. 1: Start condition detection interrupt request (STI) is enabled.	R/W
b3	SPIE	Stop Condition Detection Interrupt Request Enable	0: Stop condition detection interrupt request (SPI) is disabled. 1: Stop condition detection interrupt request (SPI) is enabled.	R/W
b4	NAKIE	NACK Reception Interrupt Request Enable	0: NACK reception interrupt request (NAKI) is disabled. 1: NACK reception interrupt request (NAKI) is enabled.	R/W
b5	RIE	Receive Data Full Interrupt Request Enable	0: Receive data full interrupt request (RXI) is disabled. 1: Receive data full interrupt request (RXI) is enabled.	R/W
b6	TEIE	Transmit End Interrupt Request Enable	0: Transmit end interrupt request (TEI) is disabled. 1: Transmit end interrupt request (TEI) is enabled.	R/W
b7	TIE	Transmit Data Empty Interrupt Request Enable	0: Transmit data empty interrupt request (TXI) is disabled. 1: Transmit data empty interrupt request (TXI) is enabled.	R/W

TMOIE Bit (Timeout Interrupt Request Enable)

This bit is used to enable or disable timeout interrupt requests (TMOI) when the TMOF flag in ICSR2 is set to 1. A TMOI interrupt request is canceled by setting the TMOF flag or the TMOIE bit to 0.

ALIE Bit (Arbitration-Lost Interrupt Request Enable)

This bit is used to enable or disable arbitration-lost interrupt requests (ALI) when the AL flag in ICSR2 is set to 1. An ALI interrupt request is canceled by setting the AL flag or the ALIE bit to 0.

STIE Bit (Start Condition Detection Interrupt Request Enable)

This bit is used to enable or disable start condition detection interrupt requests (STI) when the START flag in ICSR2 is set to 1. An STI interrupt request is canceled by setting the START flag or the STIE bit to 0.

SPIE Bit (Stop Condition Detection Interrupt Request Enable)

This bit is used to enable or disable stop condition detection interrupt requests (SPI) when the STOP flag in ICSR2 is set to 1. An SPI interrupt request is canceled by setting the STOP flag or the SPIE bit to 0.

NAKIE Bit (NACK Reception Interrupt Request Enable)

This bit is used to enable or disable NACK reception interrupt requests (NAKI) when the NACKF flag in ICSR2 is set to 1. An NAKI interrupt request is canceled by setting the NACKF flag or the NAKIE bit to 0.

RIE Bit (Receive Data Full Interrupt Request Enable)

This bit is used to enable or disable receive data full interrupt requests (RXI) when the RDRF flag in ICSR2 is set to 1.

TEIE Bit (Transmit End Interrupt Request Enable)

This bit is used to enable or disable transmit end interrupt requests (TEI) when the TEND flag in ICSR2 is set to 1. An TEI interrupt request is canceled by setting the TEND flag or the TEIE bit to 0.

TIE Bit (Transmit Data Empty Interrupt Request Enable)

This bit is used to enable or disable transmit data empty interrupt requests (TXI) when the TDRE flag in ICSR2 is set to 1.

25.2.9 I²C Bus Status Register 1 (ICSR1)

The ICSR1 register is a status register that indicates the status of detection of various addresses.

Address(es): RIIC0.ICSR1 A008 0908h, RIIC1.ICSR1 A008 0948h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	DID	—	GCA	AAS2	AAS1	AAS0
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	AAS0	Slave Address 0 Detection Flag	0: Slave address 0 is not detected. 1: Slave address 0 is detected.	R/(W) *1
b1	AAS1	Slave Address 1 Detection Flag	0: Slave address 1 is not detected. 1: Slave address 1 is detected.	R/(W) *1
b2	AAS2	Slave Address 2 Detection Flag	0: Slave address 2 is not detected. 1: Slave address 2 is detected.	R/(W) *1
b3	GCA	General Call Address Detection Flag	0: General call address is not detected. 1: General call address is detected.	R/(W) *1
b4	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b5	DID	Device-ID Address Detection Flag	0: Device-ID address is not detected. 1: Device-ID address is detected. • This bit is set to 1 when the first frame received immediately after a start condition is detected matches a value of (device ID address (1111 100b) + 0[W]).	R/(W) *1
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Only 0 can be written to clear the flag.

AASy Flag (Slave Address y Detection Flag) (y = 0 to 2)

[Setting conditions]

For 7-bit address format: ICSARUy.FS = 0

- When the received slave address matches the SVA[6:0] value in ICSARLy with the SARyE bit in ICSER set to 1 (slave address y detection enabled)
  This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the frame.

For 10-bit address format: ICSARUy.FS = 1

- When the received slave address matches a value of (11110b + SVA[1:0] in ICSARUy) and the following address matches the ICSARLy value with the SARyE bit in ICSER set to 1 (slave address y detection enabled)
  This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the frame.

[Clearing conditions] One of the following conditions is satisfied:

- When 0 is written to the AASy bit after reading AASy = 1
- When a stop condition is detected
- When 1 is written to the IICRST bit in ICCR1 to apply an RIIC reset or an internal reset

For 7-bit address format: ICSARUy.FS = 0

- When the received slave address does not match the SVA[6:0] value in ICSARLy with the SARyE bit in ICSER set to 1 (slave address y detection enabled)
  This flag is set to 0 at the rising edge of the ninth SCL clock cycle in the frame.

For 10-bit address format: ICSARUy.FS = 1

- When the received slave address does not match a value of (11110b + SVA[1:0] in ICSARUy) with the SARyE bit in ICSER set to 1 (slave address y detection enabled)
  This flag is set to 0 at the rising edge of the ninth SCL clock cycle in the frame.
- When the received slave address matches a value of (11110b + SVA[1:0] in ICSARUy) and the following address does not match the ICSARLy value with the SARyE bit in ICSER set to 1 (slave address y detection enabled)
  This flag is set to 0 at the rising edge of the ninth SCL clock cycle in the frame.

GCA Flag (General Call Address Detection Flag)

[Setting condition]

- When the received slave address matches the general call address (0000 000b + 0 [W]) with the GCAE bit in ICSER set to 1 (general call address detection is enabled)
  This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the frame.

[Clearing conditions] One of the following conditions is satisfied:

- When 0 is written to the GCA bit after reading GCA = 1
- When a stop condition is detected
- When the received slave address does not match the general call address (0000 000b + 0 [W]) with the GCAE bit in ICSER set to 1 (general call address detection is enabled)
  This flag is set to 0 at the rising edge of the ninth SCL clock cycle in the frame.
- When 1 is written to the IICRST bit in ICCR1 to apply an RIIC reset or an internal reset

DID Flag (Device-ID Address Detection Flag)

[Setting condition]

- When the first frame received immediately after a start condition or restart condition is detected matches a value of (device ID address (1111 100b) + 0 [W]) with the DIDE bit in ICSER set to 1 (Device-ID address detection is enabled)
  This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the frame.

[Clearing conditions] One of the following conditions is satisfied:

- When 0 is written to the DID bit after reading DID = 1
- When a stop condition is detected
- When the first frame received immediately after a start condition or restart condition is detected does not match a value of (device ID address (1111 100b)) with the DIDE bit in ICSER set to 1 (Device-ID address detection is enabled)
  This flag is set to 0 at the rising edge of the ninth SCL clock cycle in the frame.
- When the first frame received immediately after a start condition or restart condition is detected matches a value of (device ID address (1111 100b) + 0 [W]) and the second frame does not match any of slave addresses 0 to 2 with the DIDE bit in ICSER set to 1 (Device-ID address detection is enabled)
  This flag is set to 0 at the rising edge of the ninth SCL clock cycle in the frame.
- When 1 is written to the IICRST bit in ICCR1 to apply an RIIC reset or an internal reset

25.2.10 I²C Bus Status Register 2 (ICSR2)

The ICSR2 register is a status register that indicates the status of detection of various conditions.

Address(es): RIIC0.ICSR2 A008 0909h, RIIC1.ICSR2 A008 0949h

	b7	b6	b5	b4	b3	b2	b1	b0
	TDRE	TEND	RDRF	NACKF	STOP	START	AL	TMOF
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	TMOF	Timeout Detection Flag	0: Timeout is not detected. 1: Timeout is detected.	R/(W) *1
b1	AL	Arbitration-Lost Flag	0: Arbitration is not lost. 1: Arbitration is lost.	R/(W) *1
b2	START	Start Condition Detection Flag	0: Start condition is not detected. 1: Start condition is detected.	R/(W) *1
b3	STOP	Stop Condition Detection Flag	0: Stop condition is not detected. 1: Stop condition is detected.	R/(W) *1
b4	NACKF	NACK Detection Flag	0: NACK is not detected. 1: NACK is detected.	R/(W) *1
b5	RDRF	Receive Data Full Flag	0: ICDRR contains no receive data. 1: ICDRR contains receive data.	R/(W) *1
b6	TEND	Transmit End Flag	0: Data is being transmitted. 1: Data has been transmitted.	R/(W) *1
b7	TDRE	Transmit Data Empty Flag	0: ICDRT contains transmit data. 1: ICDRT contains no transmit data.	R

Note 1. Only 0 can be written to clear the flag.

TMOF Flag (Timeout Detection Flag)

This flag is set to 1 when the RIIC recognizes timeout after the SCLn line state remains unchanged for a certain period.

[Setting condition]

- When the SCLn line state remains unchanged for the period specified by the ICMR2.TMOH, TMOL, and TMOS bits while the ICFER.TMOE bit is 1 (the timeout function is enabled) in master mode or in slave mode and the received slave address matches.

[Clearing conditions] One of the following conditions is satisfied:

- When 0 is written to the TMOF bit after reading TMOF = 1
- When 1 is written to the IICRST bit in ICCR1 to apply an RIIC reset or an internal reset

AL Flag (Arbitration-Lost Flag)

This flag shows that bus mastership has been lost (loss in arbitration) due to a bus conflict or some other reason when a start condition is issued or an address and data are transmitted. The RIIC monitors the level on the SDAn line during transmission and, if the level on the line does not match the value of the bit being output, sets the value of the AL flag to 1 to indicate that the bus is occupied by another device.

The RIIC can also set the flag to indicate the detection of loss of arbitration during NACK transmission in master mode or during data transmission in slave mode.

[Setting conditions] One of the following conditions is satisfied:

When master arbitration-lost detection is enabled: ICFER.MALE = 1

- When the internal SDA output state does not match the SDAn line level at the rising edge of SCL clock except for the ACK period during data (including slave address) transmission in master transmission mode (when the SDAn line is driven low while the internal SDA output is at a high level (the SDAn pin is in the high-impedance state))
- When a start condition is detected while the ST bit in ICCR2 is 1 (start condition issuance request) or the internal SDA output state does not match the SDAn line level
- When the ST bit in ICCR2 is set to 1 (start condition issuance request) with the BBSY flag in ICCR2 set to 1.

When NACK transmission arbitration-lost detection is enabled: ICFER.NALE = 1

- When the internal SDA output state does not match the SDAn line level at the rising edge of SCL clock in the ACK period during NACK transmission in reception mode

When slave arbitration-lost detection is enabled: ICFER.SALE = 1

- When the internal SDA output state does not match the SDAn line level at the rising edge of SCL clock except for the ACK period during data transmission in slave transmission mode

[Clearing conditions] One of the following conditions is satisfied:

- When 0 is written to the AL bit after reading AL = 1
- When 1 is written to the IICRST bit in ICCR1 to apply an RIIC reset or an internal reset

Table 25.4 Relationship between Arbitration-Lost Generation Sources and Arbitration-Lost Enable Functions

ICFER			ICSR2		
MALE	NALE	SALE	AL	Error	Arbitration-Lost Generation Source
1	×	×	1	Start condition issuance error	When internal SDA output state does not match SDAn line level when a start condition is detected while the ST bit in ICCR2 is 1
					When ST in ICCR2 is set to 1 with BBSY in ICCR2 set to 1
			1	Transmit data mismatch	When transmit data (including slave address) does not match the bus state in master transmission mode
×	1	×	1	NACK transmission mismatch	When ACK is detected during transmission of NACK in master reception mode or slave reception mode
×	×	1	1	Transmit data mismatch	When transmit data does not match the bus state in slave transmission mode

×: Don't care

START Flag (Start Condition Detection Flag)

[Setting condition]

- When a start condition (or a restart condition) is detected

[Clearing conditions] One of the following conditions is satisfied:

- When 0 is written to the START bit after reading START = 1
- When a stop condition is detected
- When 1 is written to the IICRST bit in ICCR1 to apply an RIIC reset or an internal reset

STOP Flag (Stop Condition Detection Flag)

[Setting condition]

- When a stop condition is detected

[Clearing conditions] One of the following conditions is satisfied:

- When 0 is written to the STOP bit after reading STOP = 1
- When 1 is written to the IICRST bit in ICCR1 to apply an RIIC reset or an internal reset

NACKF Flag (NACK Detection Flag)

[Setting condition]

- When acknowledge is not received (NACK is received) from the receive device in transmission mode with the NACKE bit in ICFER set to 1 (transfer suspension enabled)

[Clearing conditions] One of the following conditions is satisfied:

- When 0 is written to the NACKF bit after reading NACKF = 1
- When 1 is written to the IICRST bit in ICCR1 to apply an RIIC reset or an internal reset

Note: When the NACKF flag is set to 1, the RIIC suspends data transmission/reception. Writing to ICDRT in transmission mode or reading from ICDRR in reception mode with the NACKF flag set to 1 does not enable data transmit/receive operation. To restart data transmission/reception, set the NACKF flag to 0.

RDRF Flag (Receive Data Full Flag)

[Setting conditions] One of the following conditions is satisfied:

- When receive data has been transferred from ICDRS to ICDRR
  This flag is set to 1 at the rising edge of the eighth or ninth SCL clock cycle (selected by the RDRFS bit in ICMR3)
- When the received slave address matches after a start condition (or a restart condition) is detected with the TRS bit in ICCR2 set to 0

[Clearing conditions] One of the following conditions is satisfied:

- When 0 is written to the RDRF bit after reading RDRF = 1
- When data is read from ICDRR
- When 1 is written to the IICRST bit in ICCR1 to apply an RIIC reset or an internal reset

TEND Flag (Transmit End Flag)

[Setting condition]

- At the rising edge of the ninth SCL clock cycle while the TDRE flag is 1

[Clearing conditions] One of the following conditions is satisfied:

- When 0 is written to the TEND bit after reading TEND = 1
- When data is written to ICDRT
- When a stop condition is detected
- When 1 is written to the IICRST bit in ICCR1 to apply an RIIC reset or an internal reset

TDRE Flag (Transmit Data Empty Flag)

[Setting conditions] One of the following conditions is satisfied:

- When data has been transferred from ICDRT to ICDRS and ICDRT becomes empty
- When the TRS bit in ICCR2 is set to 1 (including when 1 is written to the bit)
- When the received slave address matches while the TRS bit is 1

[Clearing conditions] One of the following conditions is satisfied:

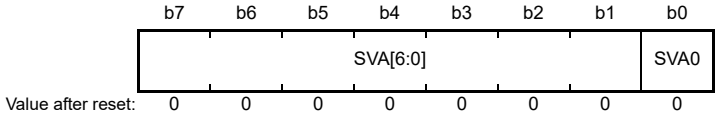
- When data is written to ICDRT
- When the TRS bit in ICCR2 is set to 0 (including when 0 is written to the bit)
- When 1 is written to the IICRST bit in ICCR1 to apply an RIIC reset or an internal reset

Note: When the NACKF flag is set to 1 while the NACKE bit in ICFER is 1, the RIIC suspends data transmission/reception. Here, if the TDRE flag is 0 (next transmit data has been written), data is transferred to the ICDRS register and the ICDRT register becomes empty at the rising edge of the ninth clock cycle, but the TDRE flag is not set to 1.

25.2.11 I²C Slave Address Register Ly (ICSARLy) (y = 0 to 2)

The ICSARLy register specifies the settings for the slave address.

Address(es): RIIC0.ICSARL0 A008 090Ah, RIIC1.ICSARL0 A008 094Ah, RIIC0.ICSARL1 A008 090Ch, RIIC1.ICSARL1 A008 094Ch, RIIC0.ICSARL2 A008 090Eh, RIIC1.ICSARL2 A008 094Eh



Bit	Symbol	Bit Name	Description	R/W
b0	SVA0	10-Bit Address LSB	A slave address is set.	R/W
b7 to b1	SVA[6:0]	7-Bit Address/10-Bit Address Lower Bits	A slave address is set.	R/W

SVA0 Bit (10-Bit Address LSB)

When the 10-bit address format is selected (ICSARUy.FS = 1), this bit functions as the LSB of a 10-bit address and forms the lower 8 bits of a 10-bit address in combination with the SVA[6:0] bits.
When the SARyE bit in ICSER is set to 1 (ICSARLy and ICSARUy enabled) and the ICSARUy.FS bit is 1, this bit is valid. While the ICSARUy.FS bit or SARyE bit is 0, the setting of this bit is ignored.

SVA[6:0] Bits (7-Bit Address/10-Bit Address Lower Bits)

When the 7-bit address format is selected (ICSARUy.FS = 0), these bits function as a 7-bit address. When the 10-bit address format is selected (ICSARUy.FS = 1), these bits function as the lower 8 bits of a 10-bit address in combination with the SVA0 bit.
While the SARyE bit in ICSER is 0, the setting of these bits is ignored.

25.2.12 I²C Slave Address Register Uy (ICSARUy) (y = 0 to 2)

The ICSARUy register specifies the format of the slave address.

Address(es): RIIC0.ICSARU0 A008 090Bh, RIIC1.ICSARU0 A008 094Bh, RIIC0.ICSARU1 A008 090Dh, RIIC1.ICSARU1 A008 094Dh, RIIC0.ICSARU2 A008 090Fh, RIIC1.ICSARU2 A008 094Fh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	SVA[1:0]		FS
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	FS	7-Bit/10-Bit Address Format Selection	0: The 7-bit address format is selected. 1: The 10-bit address format is selected.	R/W
b2, b1	SVA[1:0]	10-Bit Address Upper Bits	A slave address is set.	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

FS Bit (7-Bit/10-Bit Address Format Selection)

This bit is used to select 7-bit address or 10-bit address for slave address y (in ICSARLy and ICSARUy).

When the SARyE bit in ICSER is set to 1 (ICSARLy and ICSARUy enabled) and the ICSARUy.FS bit is 0, the 7-bit address format is selected for slave address y, the SVA[6:0] setting in ICSARLy is valid, and the settings of the SVA[1:0] bits and the SVA0 bit in ICSARLy are ignored.

When the SARyE bit in ICSER is set to 1 (ICSARLy and ICSARUy enabled) and the ICSARUy.FS bit is 1, the 10-bit address format is selected for slave address y and the settings of the SVA[1:0] bits and ICSARLy are valid.

While the SARyE bit in ICSER is 0 (ICSARLy and ICSARUy disabled), the setting of the ICSARUy.FS bit is invalid.

SVA[1:0] Bits (10-Bit Address Upper Bits)

When the 10-bit address format is selected (FS = 1), these bits function as the upper 2 bits of a 10-bit address.

When the SARyE bit in ICSER is set to 1 (ICSARLy and ICSARUy enabled) and the ICSARUy.FS bit is 1, these bits are valid. While the ICSARUy.FS bit or SARyE bit is 0, the setting of these bits is ignored.

25.2.13 I²C Bus Bit Rate Low-Level Register (ICBRL)

The ICBRL register specifies the low-level period of SCL clock and the delay cycle to which the SDA signal is added.

Address(es): RIIC0.ICBRL A008 0910h, RIIC1.ICBRL A008 0950h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	BRL[4:0]				
Value after reset:	1	1	1	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b4 to b0	BRL[4:0]	Bit Rate Low-Level Period	Low-level period of SCL clock	R/W
b7 to b5	—	Reserved	These bits are read as 1. The write value should be 1.	R/W

ICBRL works to generate the data setup time for automatic SCL low-hold operation (refer to section 25.8, Automatic Low-Hold Function for SCL); when the RIIC is used only in slave mode, this register needs to be set to a value longer than the data setup time*1.
ICBRL counts the low-level period with the internal reference clock source (IICϕ) specified by the CKS[2:0] bits in ICMR1.
If the digital noise filter is enabled (the NFE bit in ICFER is 1), set the ICBRL register to a value at least one greater than the number of stages in the noise filter. Regarding the number of stages in the noise filter, see the description of the ICMR3.NF[1:0] bits.

Note 1. Data setup time (tSU: DAT)
100 ns (up to 400 Kbps: fast mode [Fm])

25.2.14 I²C Bus Bit Rate High-Level Register (ICBRH)

The ICBRH register is a 5-bit register to set the high-level period of SCL clock, and ICBRH is valid in master mode. If the RIIC is used only in slave mode, this register need not set the high-level period.
ICBRH counts the high-level period with the internal reference clock source (IICϕ) specified by the CKS[2:0] bits in ICMR1.

Address(es): RIIC0.ICBRH A008 0911h, RIIC1.ICBRH A008 0951h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	BRH[4:0]				
Value after reset:	1	1	1	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b4 to b0	BRH[4:0]	Bit Rate High-Level Period	High-level period of SCL clock	R/W
b7 to b5	—	Reserved	These bits are read as 1. The write value should be 1.	R/W

If the digital noise filter is enabled (the NFE bit in ICFER is 1), set the ICBRH register to a value at least one greater than the number of stages in the noise filter. Regarding the number of stages in the noise filter, see the description of the ICMR3.NF[1:0] bits.

The I²C transfer rate and the SCL clock duty are calculated using the following expression.
Transfer rate = 1 / {[(ICBRH + 1) + (ICBRL + 1)] / IICϕ*1 + SCLn line rising time [tr] + SCLn line falling time [tf]}
Duty cycle = {SCLn line rising time [tr]*2 + (ICBRH + 1) / IICϕ} / {SCLn line falling time [tf]*2 + (ICBRL + 1) / IICϕ}

Note 1. IICϕ = Set value in ICMR1.CKS[2:0]
Note 2. The SCLn line rising time [tr] and SCLn line falling time [tf] depend on the total bus line capacitance [Cb] and the pull-up resistor [Rp]. For details, see the I²C bus standard from NXP Semiconductors.

Table 25.5 lists examples of ICBRH/ICBRL settings.

Table 25.5 Examples of ICBRH/ICBRL Settings for Transfer Rate

	Operating Frequency PCLKD (MHz)		
	75		
Transfer Rate (Kbps)	**CKS[2:0]**	**ICBRH**	**ICBRL**
10	111b	26 (FAh)	30(FEh)
50	101b	20 (F4h)	23(F7h)
100	100b	19 (F3h)	23(F7h)
400	010b	11 (EBh)	24(F8h)

Note: This is an example of the setting when the rising time (tr) of the SCLn line is 300 ns and the falling time (tf) of the SCLn line is 300 ns. For the specified values of the rising time (tr) and the falling time (tf) of the SCLn line, see the I²C bus specifications from NXP Semiconductors.

25.2.15 I²C Bus Transmit Data Register (ICDRT)

The ICDRT register stores transmit data.

Address(es): RIIC0.ICDRT A008 0912h, RIIC1.ICDRT A008 0952h

	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1

When ICDRT detects a space in the I2C bus shift register (ICDRS), it transfers the transmit data that has been written to ICDRT to ICDRS and starts transmitting data in transmission mode.
The double-buffer structure of ICDRT and ICDRS allows continuous transmit operation if the next transmit data has been written to ICDRT while the ICDRS data is being transmitted.
ICDRT can always be read and written. Write transmit data to ICDRT once when a transmit data empty interrupt (TXI) request is generated.

25.2.16 I²C Bus Receive Data Register (ICDRR)

The ICDRR register stores receive data.

Address(es): RIIC0.ICDRR A008 0913h, RIIC1.ICDRR A008 0953h

	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0

When 1 byte of data has been received, the received data is transferred from the I2C bus shift register (ICDRS) to ICDRR to enable the next data to be received.
The double-buffer structure of ICDRS and ICDRR allows continuous receive operation if the received data has been read from ICDRR while ICDRS is receiving data.
ICDRR cannot be written. Read data from ICDRR once when a receive data full interrupt (RXI) request is generated.
If ICDRR receives the next receive data before the current data is read from ICDRR (while the RDRF flag in ICSR2 is 1), the RIIC automatically holds the SCLn clock low one cycle before the RDRF flag is set to 1 next.

25.2.17 I²CBus Shift Register (ICDRS)

The ICDRS register is a shift register to transmit and receive data.

	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	—	—	—	—	—	—	—	—

During transmission, transmit data is transferred from ICDRT to ICDRS and is sent from the SDAn pin. During reception, data is transferred from ICDRS to ICDRR after 1 byte of data has been received.
ICDRS cannot be accessed directly.

25.3 Operation

25.3.1 Communication Data Format

The I²C bus format consists of 8-bit data and 1-bit acknowledge. The frame following a start condition or restart condition is an address frame used to specify a slave device with which the master device communicates. The specified slave is valid until a new slave is specified or a stop condition is issued.
Figure 25.3 shows the I²C bus format, and Figure 25.4 shows the I²C bus timing.

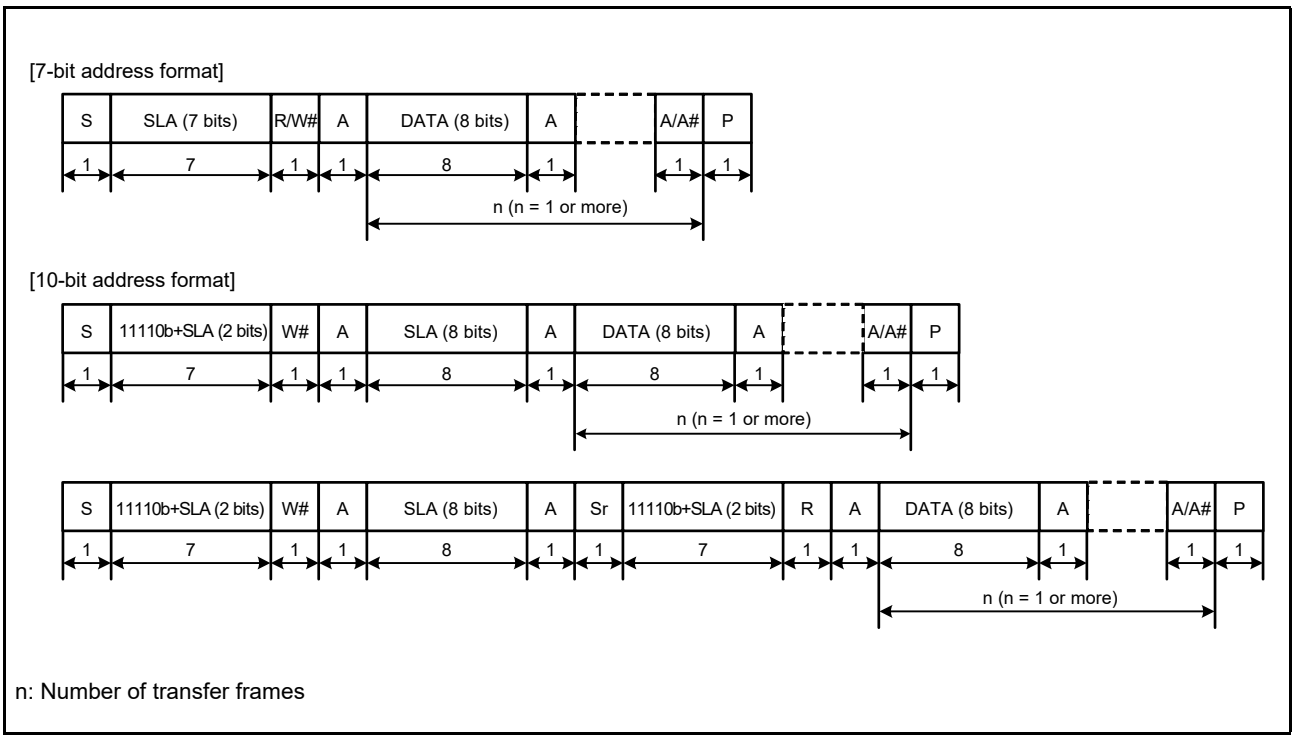


Figure 25.3 I²C Bus Format

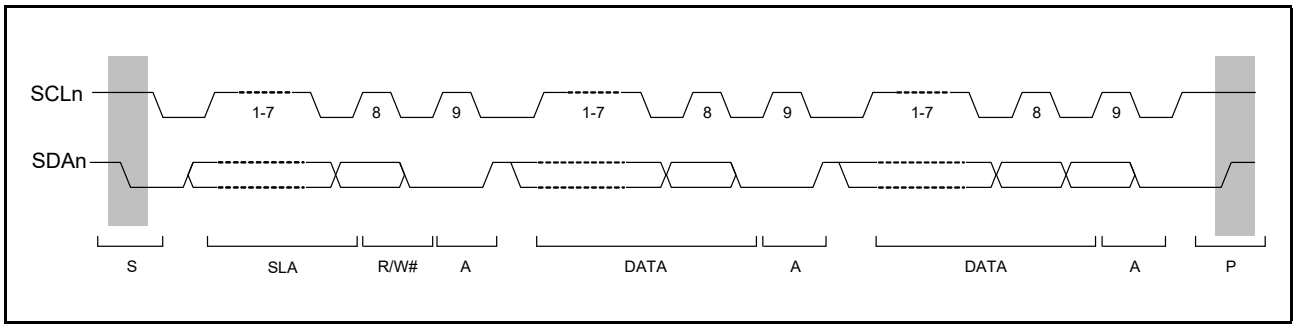


Figure 25.4 I²C Bus Timing (SLA = 7 Bits)

S: Start condition. The SDAn line is changed from the high level to the low level while the SCLn line of the master device is at a high level.

SLA: Slave address, by which the master device selects a slave device.

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 the SDAn line low. (In master transmission mode, the slave device returns acknowledge. In master reception mode, the master device returns acknowledge.)

A#: Not Acknowledge. The receive device drives the SDAn line high.

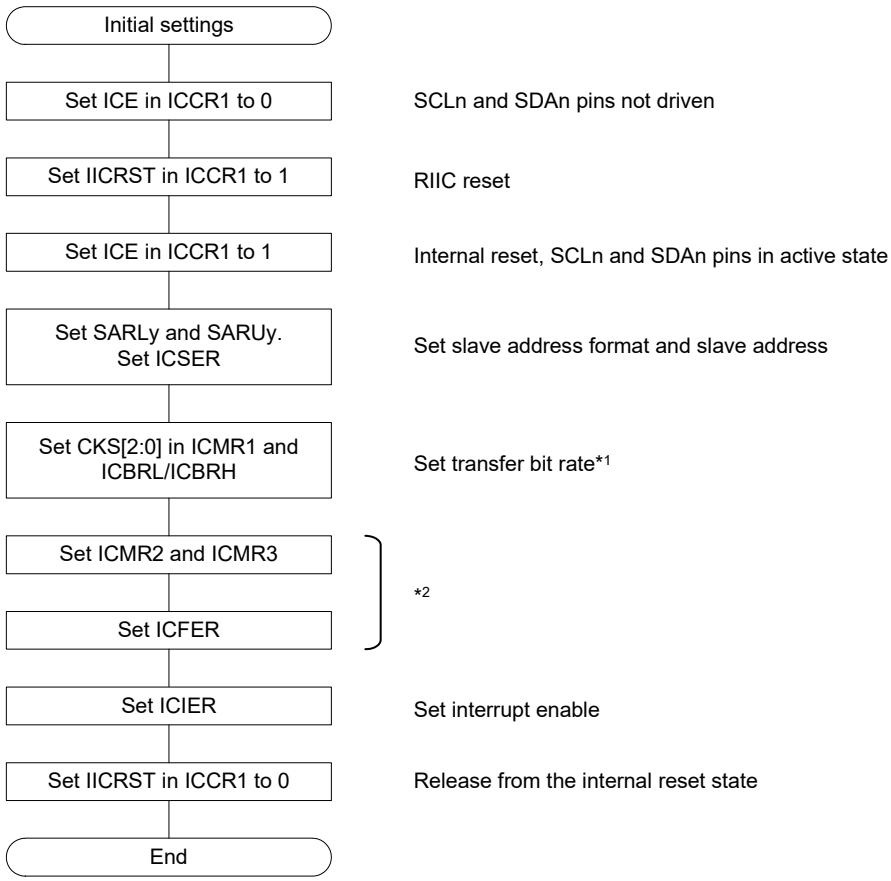
Sr: Restart condition. The SDAn line is changed from the high level to the low level after the setup time has elapsed with the SCLn line at the high level.

DATA: Transmitted or received data. The bit length of the transmitted or received data is set in ICMR1.BC[2:0].

P: Stop condition. The SDAn line is changed from the low level to the high level while the SCLn line is at a high level.

25.3.2 Initial Settings

Before starting data transmission and reception, initialize the RIIC according to the procedure in Figure 25.5.
Set the ICCR1.ICE bit to 1 (internal reset) after setting the ICCR1.IICRST bit to 1 (RIIC reset) with the ICCR1.ICE bit set to 0 (SCLn and SDAn pins in inactive state). This initializes the various flags and internal state of the ICSR1 register. After that, set registers ICSARLy, ICSARUy, ICSER, ICMR1, ICBRH, and ICBRL (y = 0 to 2), and set the other registers as necessary (for initial settings of the RIIC, see Figure 25.5). When the necessary register settings have been completed, set the ICCR1.IICRST bit to 0 (releases the RIIC reset). This step is not necessary if initialization of the RIIC has already been completed.



y = 0 to 2
Note 1. When the RIIC is used only in slave mode, set the ICBRL register to a value longer than the data setup time.
Note 2. Set these registers as necessary.

Figure 25.5 Example of RIIC Initialization Flowchart

25.3.3 Master Transmit Operation

In master transmit operation, the RIIC outputs the SCL clock and transmitted data signals as the master device, and the slave device returns acknowledgements. Figure 25.6 shows an example of usage of master transmission and Figure 25.7 to Figure 25.9 show the timing of operations in master transmission.
The following describes the procedure and operations for master transmission.

(1) Initial settings. For details, refer to section 25.3.2, Initial Settings.
(2) Read the BBSY flag in ICCR2 to check that the bus is open, and then set the ST bit in ICCR2 to 1 (start condition issuance request). Upon receiving the request, the RIIC issues a start condition. At the same time, the BBSY flag and the START flag in ICSR2 are automatically set to 1 and the ST bit is automatically set to 0. At this time, if the start condition is detected and the internal levels for the SDA output state and the levels on the SDAn line have matched while the ST bit is 1, the RIIC recognizes that issuing of the start condition as requested by the ST bit has been successfully completed, and the MST and TRS bits in ICCR2 are automatically set to 1, placing the RIIC in master transmission mode. The TDRE flag in ICSR2 is also automatically set to 1 in response to setting of the TRS bit to 1.
(3) Check that the TDRE flag in ICSR2 is 1, and then write the value for transmission (the slave address and the R/W# bit) to ICDRT. Once the data for transmission are written to ICDRT, the TDRE flag is automatically set to 0, the data are transferred from ICDRT to ICDRS, and the TDRE flag is again set to 1. After the byte containing the slave address and R/W# bit has been transmitted, the value of the TRS bit is automatically updated to select master transmit or master reception mode in accord with the value of the transmitted R/W# bit. If the value of the R/W# bit was 0, the RIIC continues in master transmission mode.
Since the ICSR2.NACKF flag being 1 at this time indicates that no slave device recognized the address or there was an error in communications, write 1 to the ICCR2.SP bit to issue a stop condition.
For data transmission with an address in the 10-bit format, start by writing 1111 0b, the 2 higher-order bits of the slave address, and W to ICDRT as the first address transmission. Then, as the second address transmission, write the 8 lower-order bits of the slave address to ICDRT.
(4) After confirming that the TDRE flag in ICSR2 is 1, write the data for transmission to the ICDRT register. The RIIC automatically holds the SCLn line low until the data for transmission are ready or a stop condition is issued.
(5) After all bytes of data for transmission have been written to the ICDRT register, wait until the value of the TEND flag in ICSR2 returns to 1, and then set the SP bit in ICCR2 to 1 (stop condition issuance request). Upon receiving a stop condition issuance request, the RIIC issues the stop condition.
(6) Upon detecting the stop condition, the RIIC automatically sets the MST and TRS bits in ICCR2 to 00b and enters slave reception mode. Furthermore, it automatically sets the TDRE and TEND flags in ICSR2 to 0, and sets the STOP flag in ICSR2 to 1.
(7) After checking that the ICSR2.STOP flag is 1, set the ICSR2.NACKF and STOP flags to 0 for the next transfer operation.

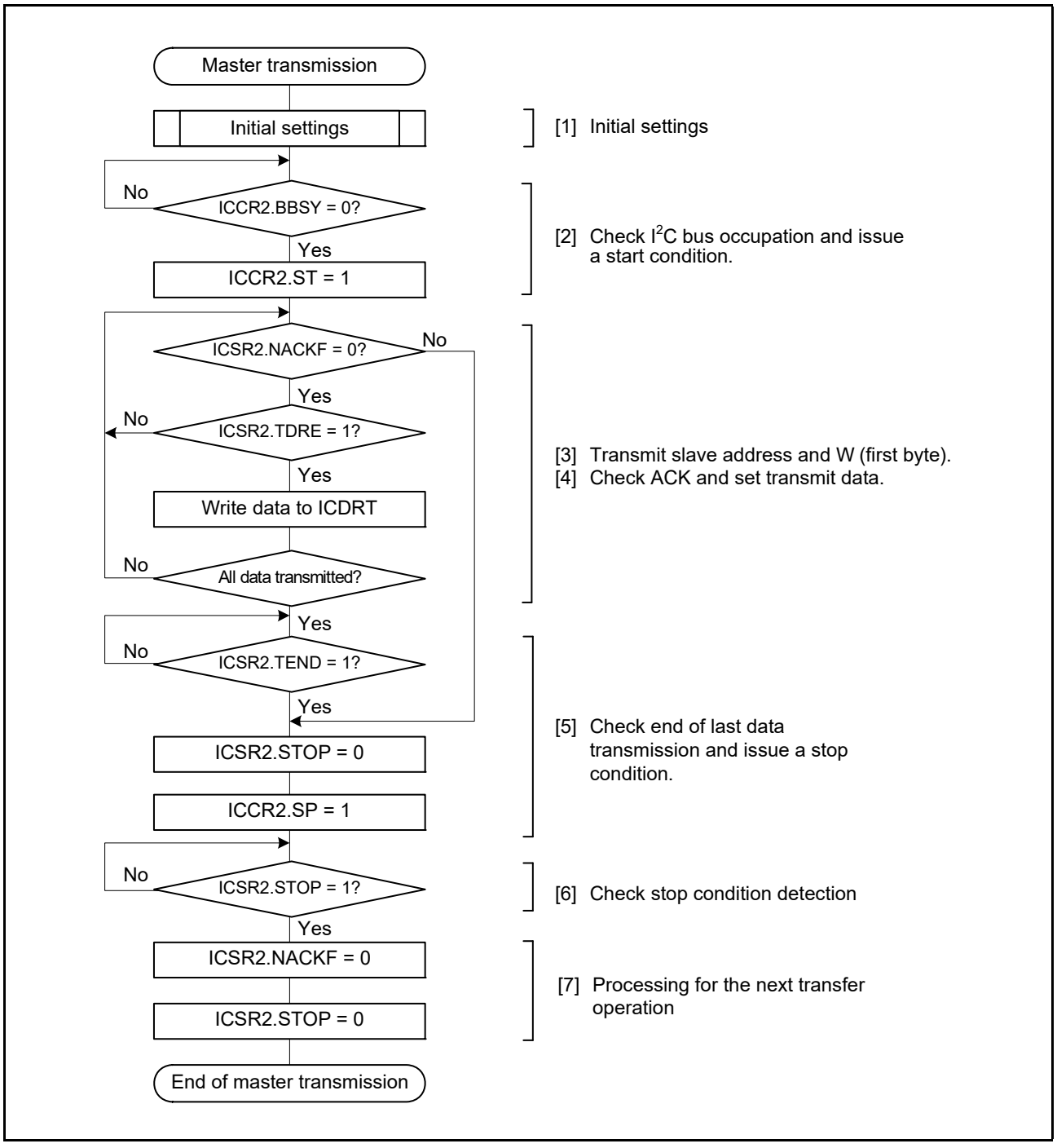


Figure 25.6 Example of Master Transmission Flowchart

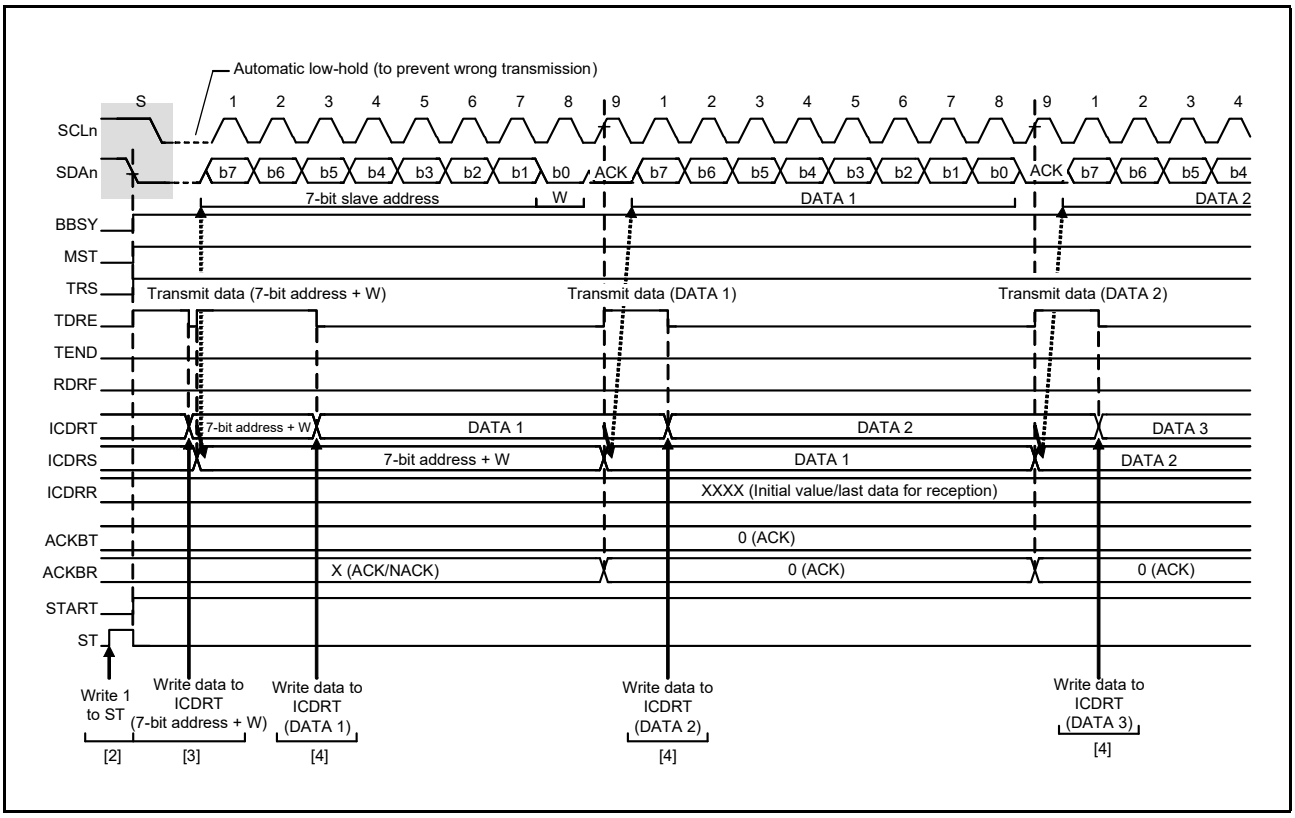


Figure 25.7 Master Transmit Operation Timing (1) (7-Bit Address Format)

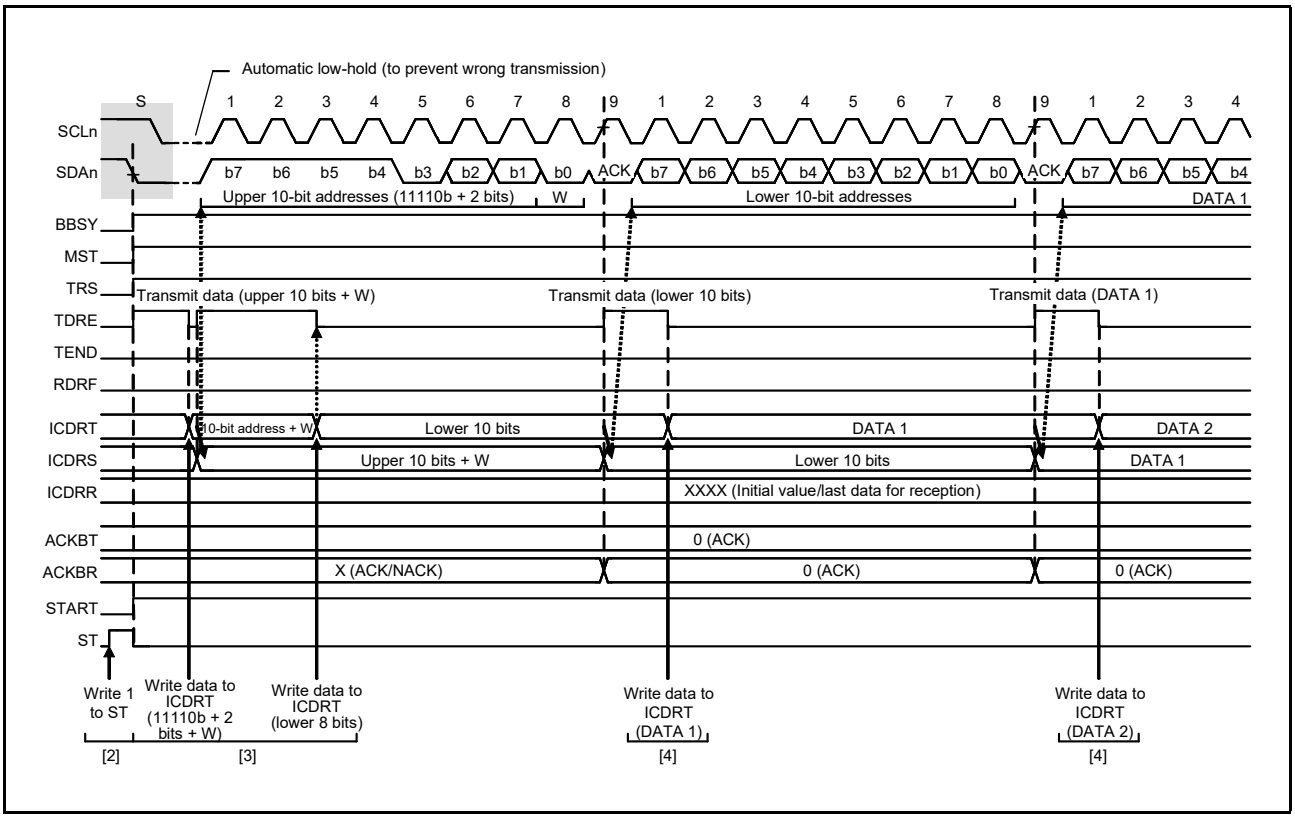


Figure 25.8 Master Transmit Operation Timing (2) (10-Bit Address Format)

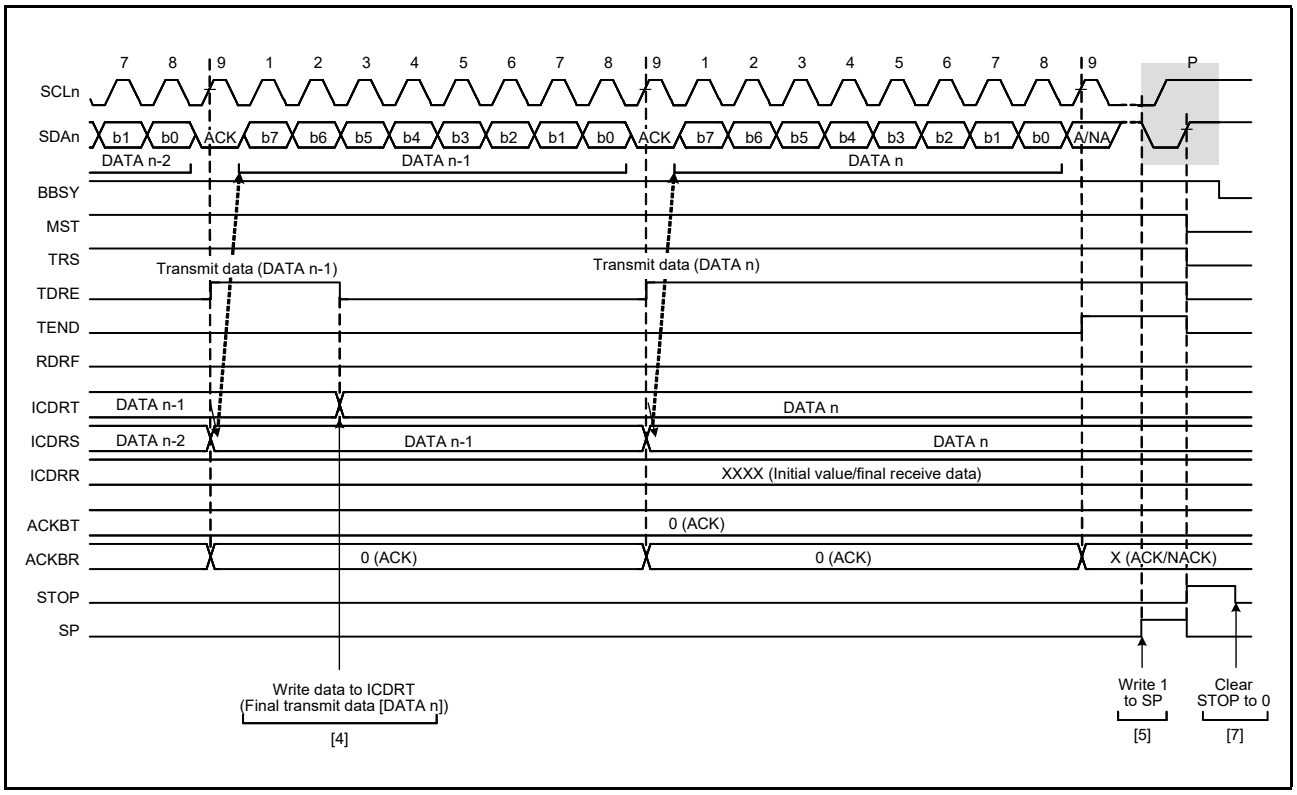


Figure 25.9 Master Transmit Operation Timing (3)

25.3.4 Master Receive Operation

In master receive operation, the RIIC as a master device outputs the SCL clock, receives data from the slave device, and returns acknowledgements. Since the RIIC must start by sending a slave address to the corresponding slave device, this part of the procedure is performed in master transmission mode, but the subsequent steps are in master reception mode. Figure 25.10 and Figure 25.11 show examples of usage of master reception in the 7-bit address format and Figure 25.12 to Figure 25.14 show the timing of operations in master reception.
The following describes the procedure and operations for master reception.

(1) Initial settings. For details, refer to section 25.3.2, Initial Settings.
(2) Read the BBSY flag in ICCR2 to check that the bus is open, and then set the ST bit in ICCR2 to 1 (start condition issuance request). Upon receiving the request, the RIIC issues a start condition. When the RIIC detects the start condition, the BBSY flag and the START flag in ICSR2 are automatically set to 1 and the ST bit is automatically set to 0. At this time, if the start condition is detected and the levels for the SDA output and the levels on the SDAn line have matched while the ST bit is 1, the RIIC recognizes that issuing of the start condition as requested by the ST bit has been successfully completed, and the MST and TRS bits in ICCR2 are automatically set to 1, placing the RIIC in master transmission mode. The TDRE flag in ICSR2 is also automatically set to 1 in response to setting of the TRS bit to 1.
(3) Check that the TDRE flag in ICSR2 is 1, and then write the value for transmission (the first byte indicates the slave address and value of the R/W# bit) to ICDRT. Once the data for transmission are written to ICDRT, the TDRE flag is automatically set to 0, the data are transferred from ICDRT to ICDRS, and the TDRE flag is again set to 1. Once the byte containing the slave address and R/W# bit has been transmitted, the value of the ICCR2.TRS bit is automatically updated to select transmit or reception mode in accord with the value of the transmitted R/W# bit. If the value of the R/W# bit was 1, the TRS bit is set to 0 on the rising edge of the ninth cycle of SCL clock, placing the RIIC in master reception mode. At this time, the TDRE flag is set to 0 and the ICSR2.RDRF flag is automatically set to 1.
Since the ICSR2.NACKF flag being 1 at this time indicates that no slave device recognized the address or there was an error in communications, write 1 to the ICCR2.SP bit to issue a stop condition.
For master reception from a device with a 10-bit address, start by using master transmission to issue the 10-bit address, and then issue a restart condition. After that, transmitting 1111 0b, the two higher-order bits of the slave address, and the R bit places the RIIC in master reception mode.
(4) Dummy read ICDRR after confirming that the RDRF flag in ICSR2 is 1; this makes the RIIC start output of the SCL clock and start data reception.
(5) After 1 byte of data has been received, the RDRF flag in ICSR2 is set to 1 on the rising edge of the eighth or ninth cycle of SCL clock (the clock signal) as selected by the RDRFS bit in ICMR3. Reading out ICDRR at this time will produce the received data, and the RDRF flag is automatically set to 0 at the same time. Furthermore, the value of the acknowledgement field received during the ninth cycle of SCL clock is returned as the value set in the ICMR3.ACKBT bit. If the next byte to be received is the next to last byte, set the ICMR3.WAIT bit to 1 (for wait insertion) before reading the ICDRR (containing the second byte from last). As well as enabling NACK output even in the case of delays in processing to set the ICMR3.ACKBT bit to 1 (NACK) in step (6), due to other interrupts, etc., this fixes the SCLn line to the low level on the falling edge of the ninth clock cycle in reception of the last byte, so the state is such that issuing a stop condition is possible.
(6) When the ICMR3.RDRFS bit is 0 and the slave device must be notified that it is to end transfer for data reception after transfer of the next (final) byte, set the ICMR3.ACKBT bit to 1 (NACK).

(7) After reading out the byte before last from the ICDRR register, if the value of the ICSR2.RDRF flag is confirmed to be 1, write 1 to the SP bit in ICCR2 (stop condition issuance request) and then read the last byte from ICDRR. When ICDRR is read, the RIIC is released from the wait state and issues the stop condition after low-level output in the ninth clock cycle is completed or the SCLn line is released from the low-hold state.

(8) Upon detecting the stop condition, the RIIC automatically sets the MST and TRS bits in ICCR2 to 00b and enters slave reception mode. Furthermore, detection of the stop condition leads to setting of the ICSR2.STOP flag to 1.

(9) After checking that the ICSR2.STOP flag is 1, set the ICSR2.NACKF and STOP flags to 0 for the next transfer operation.

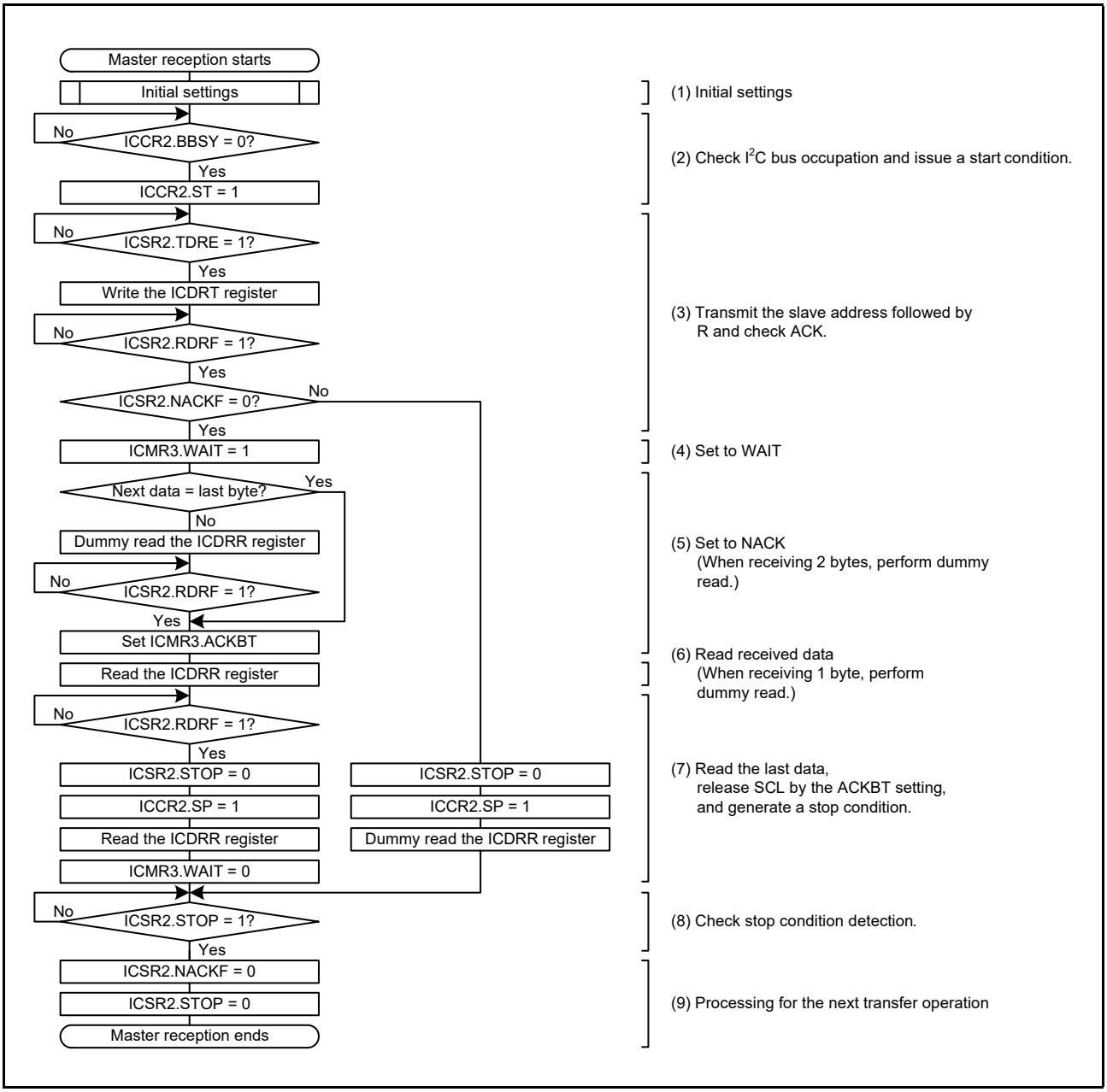


Figure 25.10 Example of Master Reception (7-Bit Address Format, 1 or 2 bytes)

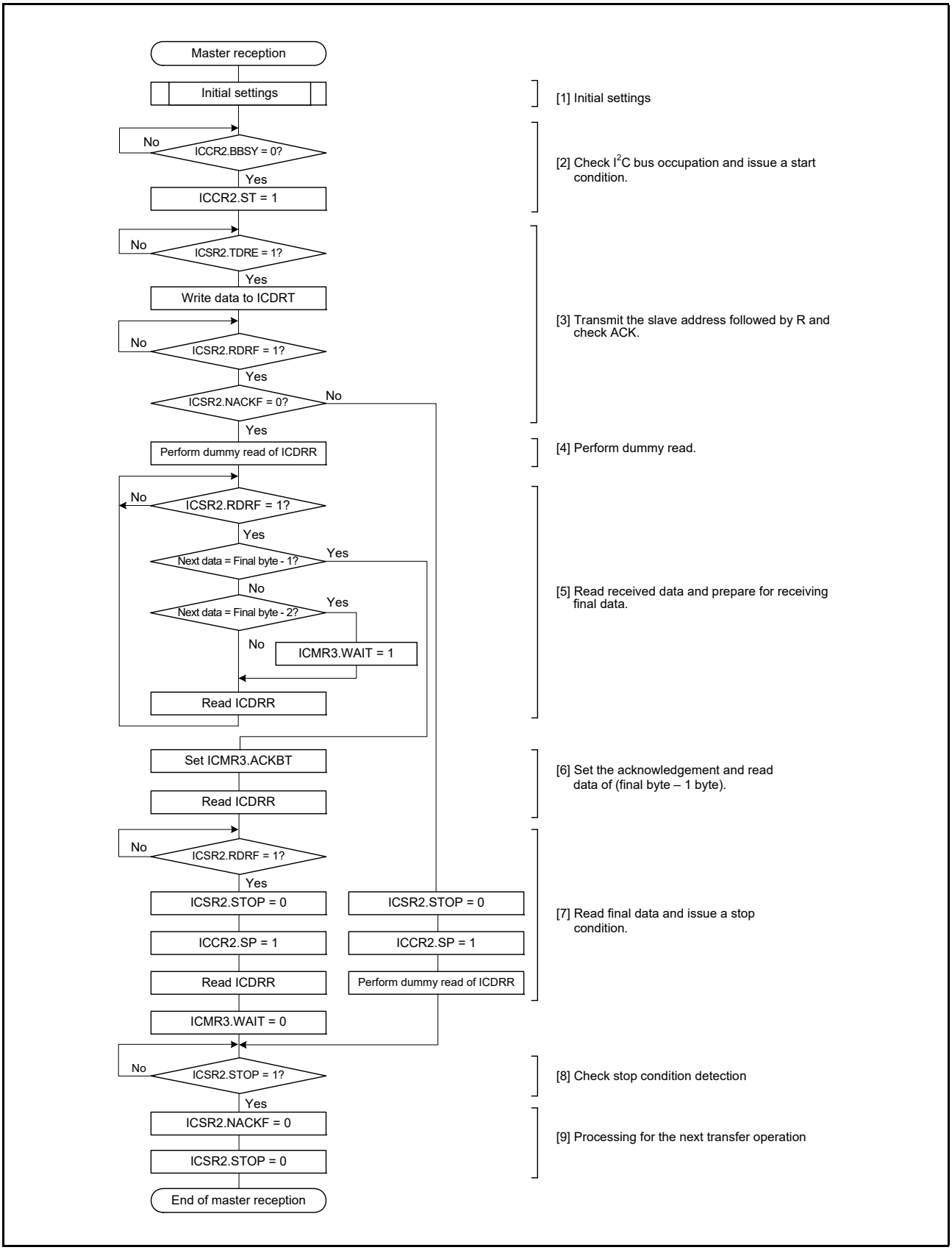


Figure 25.11 Example of Master Reception (7-Bit Address Format, 3 Bytes or More)

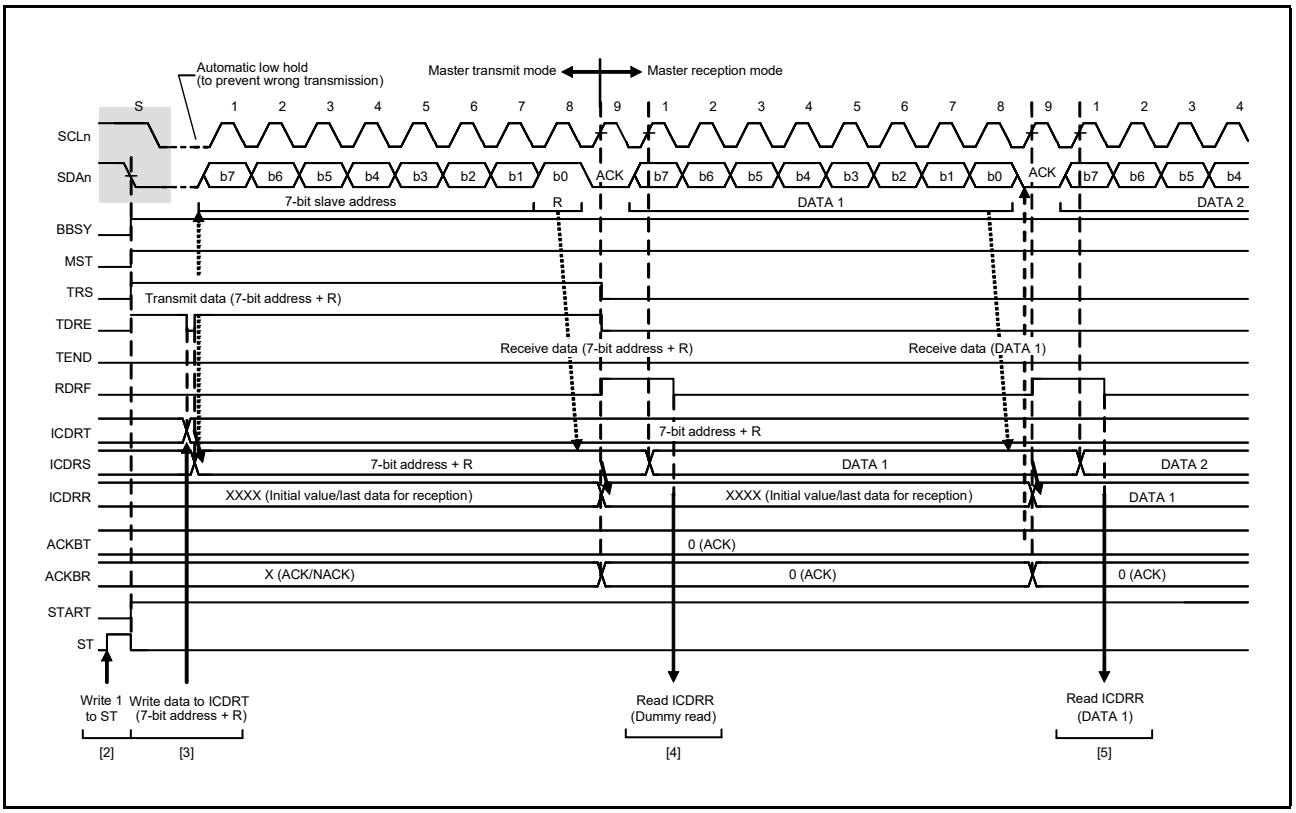


Figure 25.12 Master Receive Operation Timing (1) (7-Bit Address Format, when RDRFS = 0)

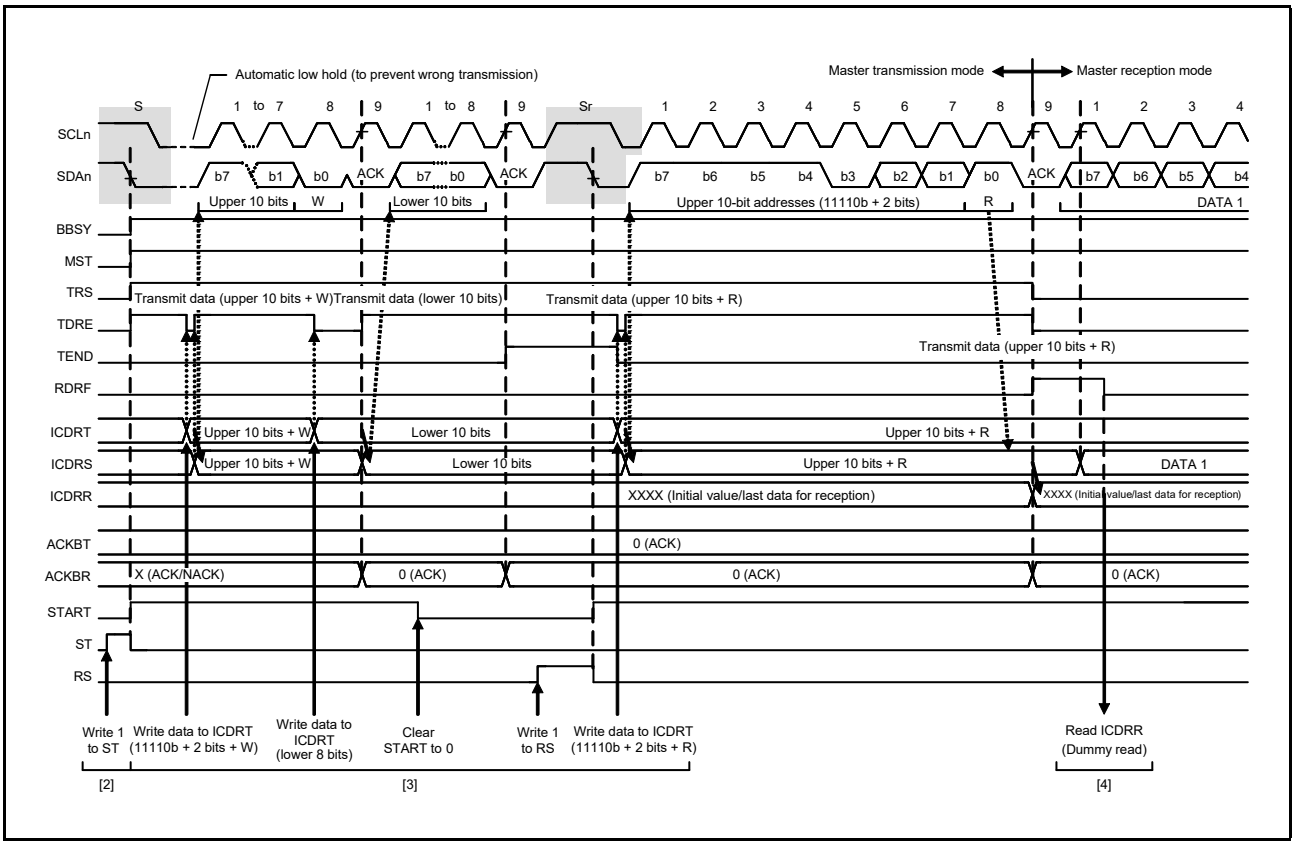


Figure 25.13 Master Receive Operation Timing (2) (10-Bit Address Format, when RDRFS = 0)

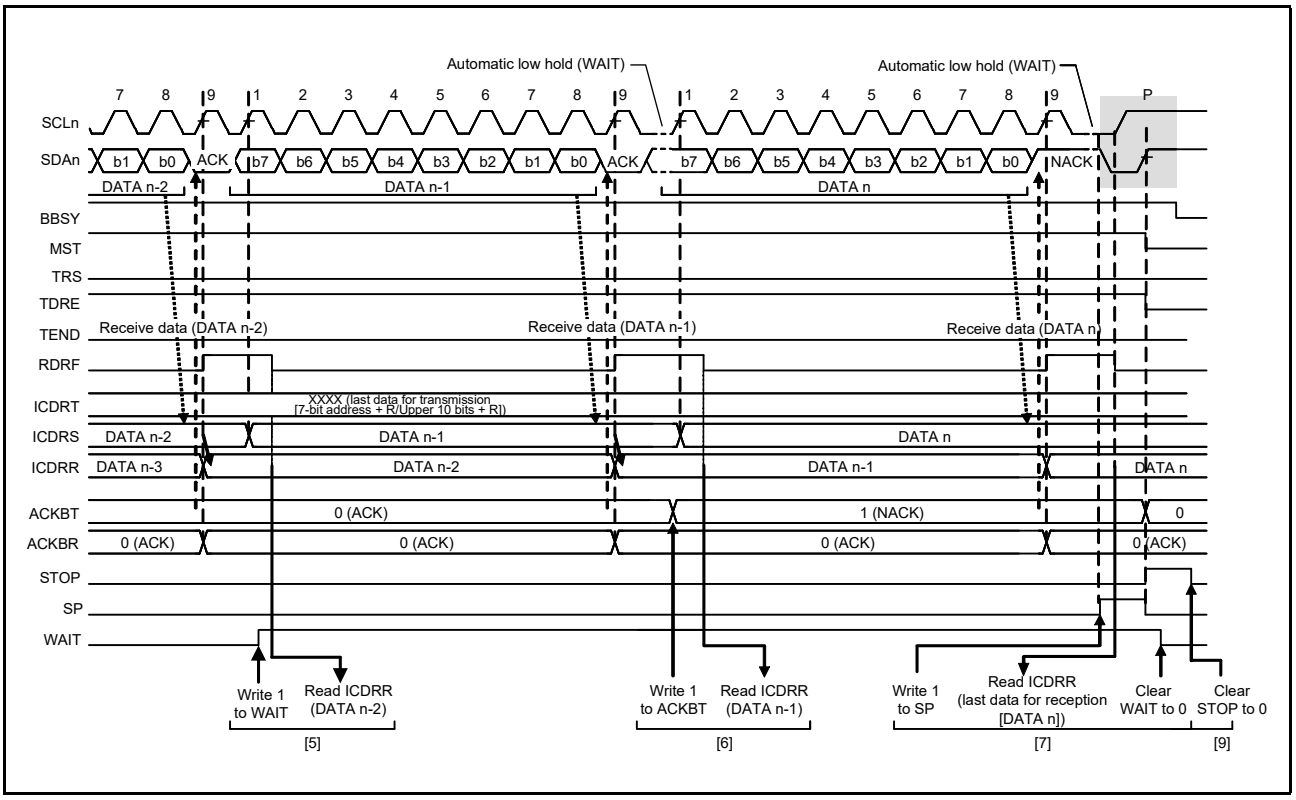


Figure 25.14 Master Receive Operation Timing (3) (when RDRFS = 0)

25.3.5 Slave Transmit Operation

In slave transmit operation, the master device outputs the SCL clock, the RIIC transmits data as a slave device, and the master device returns acknowledgements.
Figure 25.15 shows an example of usage of slave transmission and Figure 25.16 and Figure 25.17 show the timing of operations in slave transmission.
The following describes the procedure and operations for slave transmission.

(1) Initial settings. For details, refer to section 25.3.2, Initial Settings.
After initial settings, the RIIC will stay in the standby state in slave reception mode until it receives a slave address that it matches.

(2) After receiving a matching slave address, the RIIC sets the corresponding bit among ICSR1.GCA and AASy (y = 0 to 2) to 1 on the rising edge of the ninth cycle of the SCL clock and outputs the value set in the ICMR3.ACKBT bit as the acknowledge bit on the ninth cycle of the SCL clock.
With the 7-bit address format, if the value of the R/W# bit that was also received at this time is 1, the RIIC automatically places itself in slave transmission mode by setting both the TRS bit and the TDRE flag in ICSR2 to 1.
With the 10-bit address format, after receiving a matching slave address, check that the ICSR2.STOP flag and the ICSR2.RDRF flag are 0 and 1, respectively, and dummy-read the ICDRR register. Here, the received value that is dummy read is the lower-order 8 bits of the address. Following the dummy read, data reception is restarted on detection of a restart condition. If the value of the R/W# bit that was also received at this time is 1, the RIIC automatically places itself in slave transmission mode by setting both the ICCR2.TRS bit and the ICSR2.TDRE flag to 1.

(3) After the ICSR2.TDRE flag is confirmed to be 1, write the data for transmission to the ICDRT register. At this time, if the RIIC receives no acknowledge from the master device (receives an NACK signal) while the ICFER.NACKE bit is 1, the RIIC suspends transfer of the next data.

(4) Wait until the ICSR2.TEND flag is set to 1 while the ICSR2.TDRE flag is 1, after the ICSR2.NACKF flag is set to 1 or the last byte for transmission is written to the ICDRT register. When the ICSR2.NACKF flag or the TEND flag is 1, the RIIC drives the SCLn line low on the ninth falling edge of SCL clock.

(5) When the ICSR2.NACKF flag or the ICSR2.TEND flag is 1, dummy read ICDRR to complete the processing. This releases the SCLn line.

(6) Upon detecting the stop condition, the RIIC automatically sets bits ICSR1.GCA and AASy (y = 0 to 2), flags ICSR2.TDRE and TEND, and the ICCR2.TRS bit to 0, and enters slave reception mode.

(7) After checking that the ICSR2.STOP flag is 1, set the ICSR2.NACKF and STOP flags to 0 for the next transfer operation.

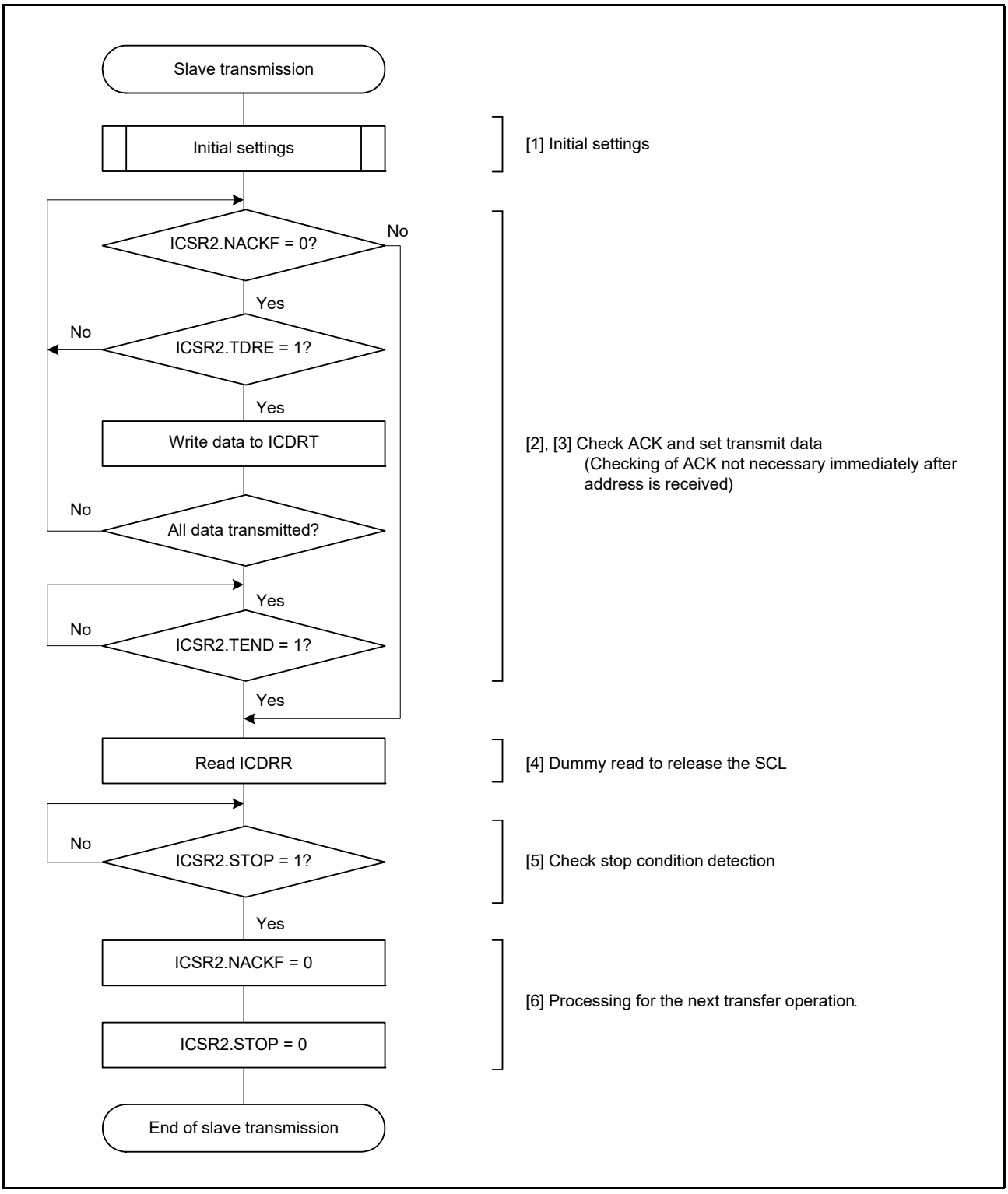


Figure 25.15 Example of Slave Transmission Flowchart

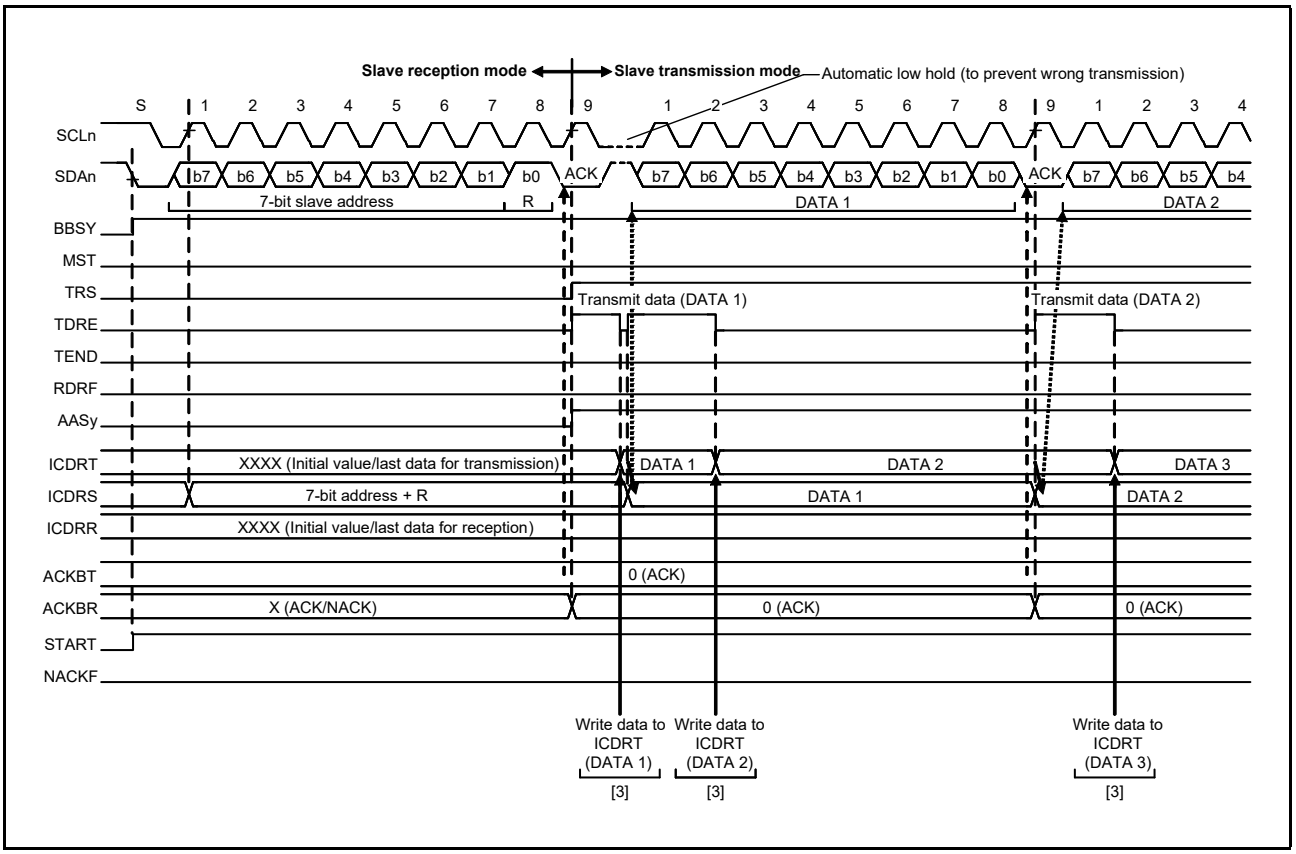


Figure 25.16 Slave Transmit Operation Timing (1) (7-Bit Address Format)

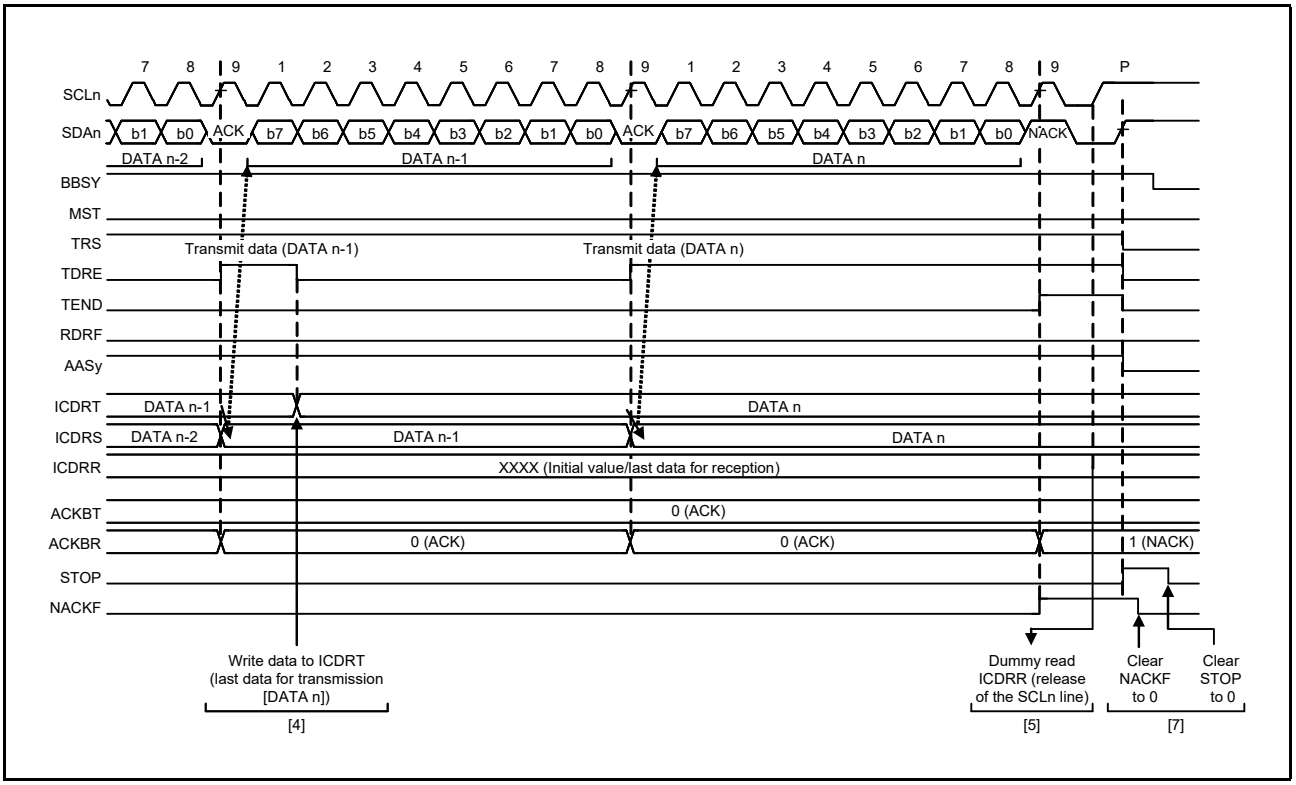


Figure 25.17 Slave Transmit Operation Timing (2)

25.3.6 Slave Receive Operation

In slave receive operation, the master device outputs the SCL clock and transmit data, and the RIIC returns acknowledgements as a slave device.

Figure 25.18 shows an example of usage of slave reception and Figure 25.19 and Figure 25.20 show the timing of operations in slave reception.

The following describes the procedure and operations for slave reception.

(1) Initial settings. For details, refer to section 25.3.2, Initial Settings.
After initial settings, the RIIC will stay in the standby state in slave reception mode until it receives a slave address that it matches.

(2) After receiving a matching slave address, the RIIC sets one of the corresponding bits ICSR1.GCA and AASy (y = 0 to 2) to 1 on the rising edge of the ninth cycle of SCL clock (the clock signal) and outputs the value set in the ICMR3.ACKBT bit to the acknowledge bit on the ninth cycle of SCL clock. If the value of the R/W# bit that was also received at this time is 0, the RIIC continues to place itself in slave reception mode and sets the RDRF flag in ICSR2 to 1.

(3) After the ICSR2.STOP flag is confirmed to be 0 and the ICSR2.RDRF flag to be 1, dummy read ICDRR (the dummy value consists of the slave address and R/W# bit when the 7-bit address format is selected, or the lower 8 bits when the 10-bit address format is selected). After the dummy-read, the RIIC clears the RDRF flag to 0 and starts reception of data.

(4) On completion of the reception of 1 byte of data, the value of the ICSR2.RDRF flag becomes 1 on the rising edge of the 8th or 9th clock cycle of the SCL clock, according to the setting made in the ICMR3.RDRFS bit. At this time, reading the ICDRR register returns the received value and the RDRF flag is automatically cleared to 0 in response. If reading of ICDRR is delayed and a next byte is received while the RDRF flag is still set to 1, the RIIC holds the SCLn line low from the falling edge of one SCL cycle before the timing with which RDRF should be set. In this case, reading ICDRR releases the SCLn line from being held at the low level.
When the ICSR2.STOP flag is 1 and the ICSR2.RDRF flag is also 1, read ICDRR until all the data is completely received.

(5) Upon detecting the stop condition, the RIIC automatically clears bits ICSR1.GCA and AASy (y = 0 to 2) to 0.

(6) After checking that the ICSR2.STOP flag is 1, set the ICSR2.STOP flag to 0 for the next transfer operation.

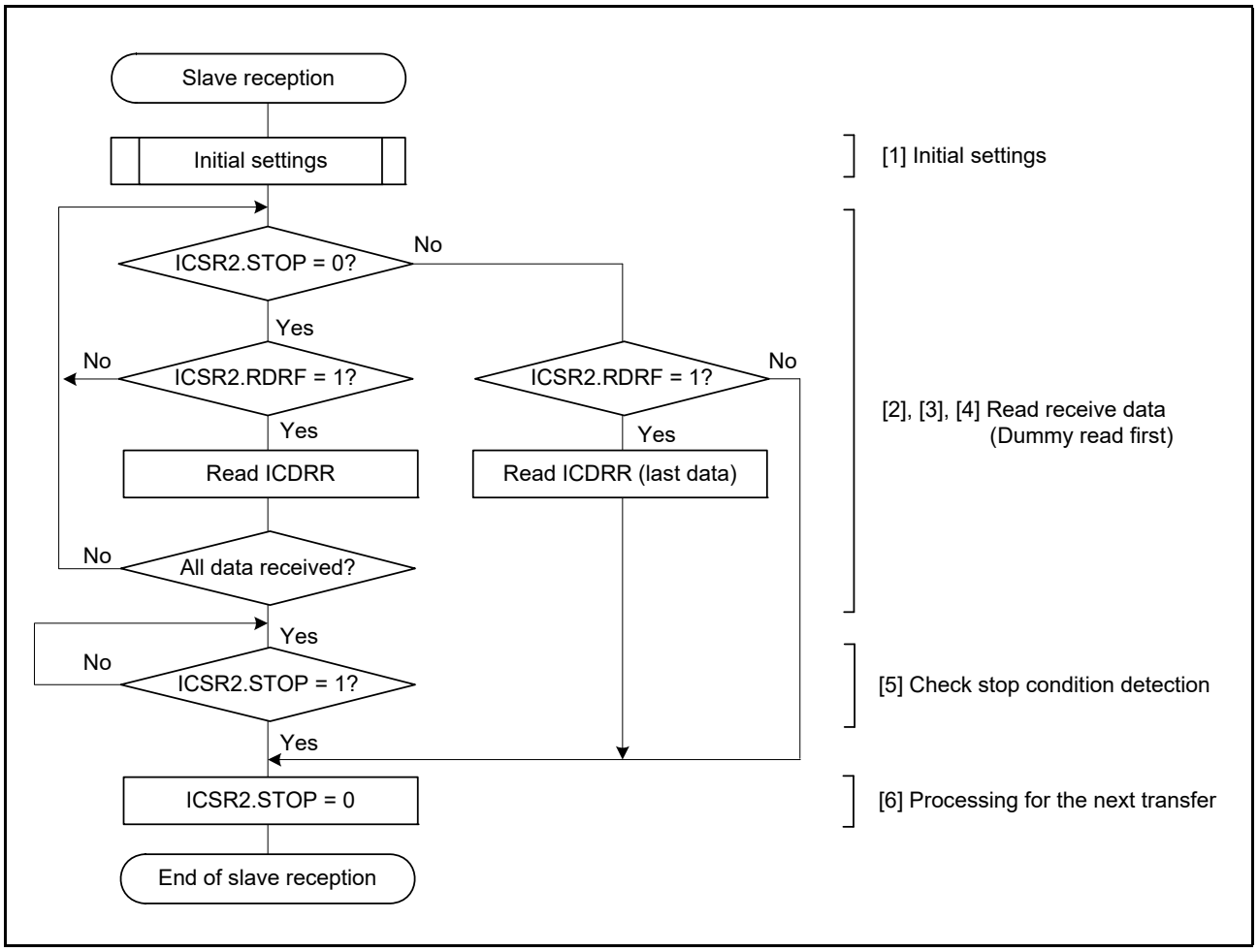


Figure 25.18 Example of Slave Reception Flowchart

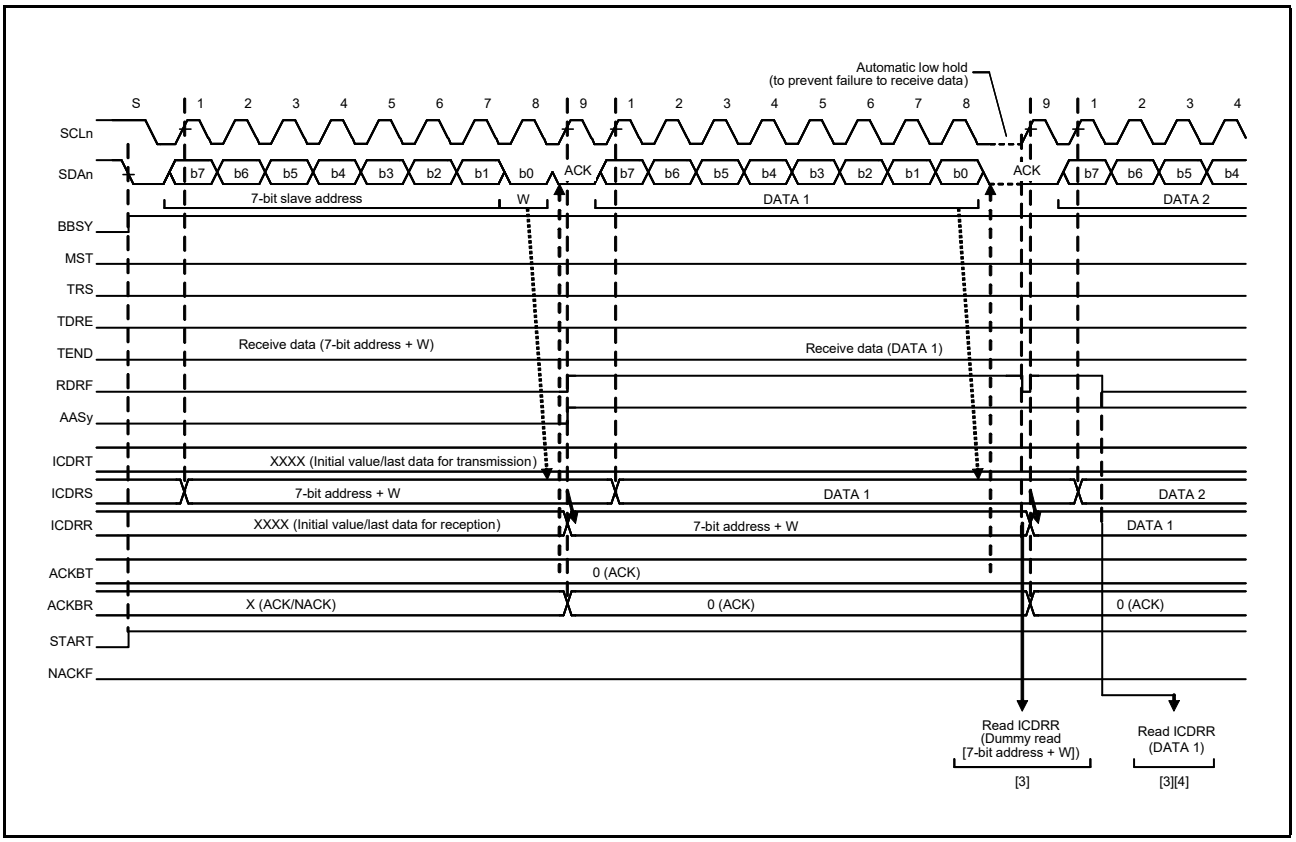


Figure 25.19 Slave Receive Operation Timing (1) (7-Bit Address Format, when RDRFS = 0)

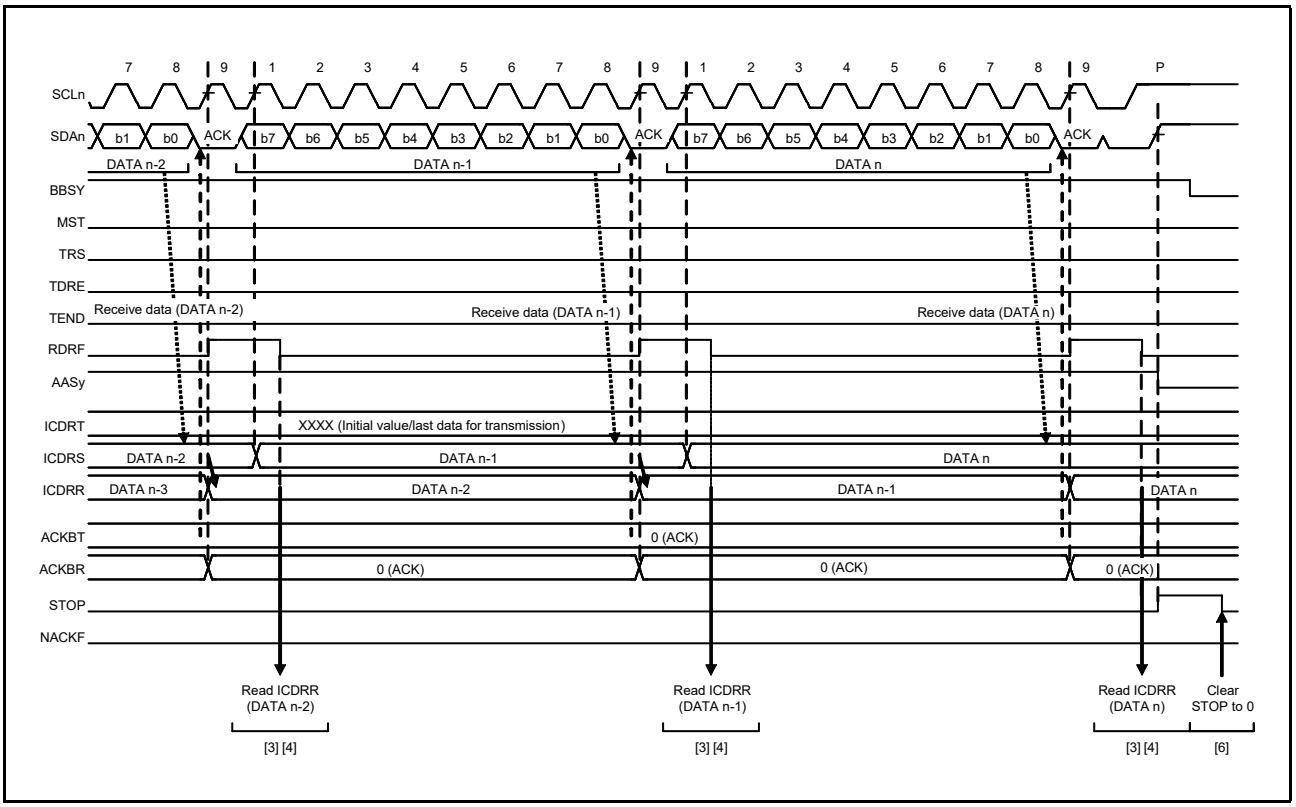


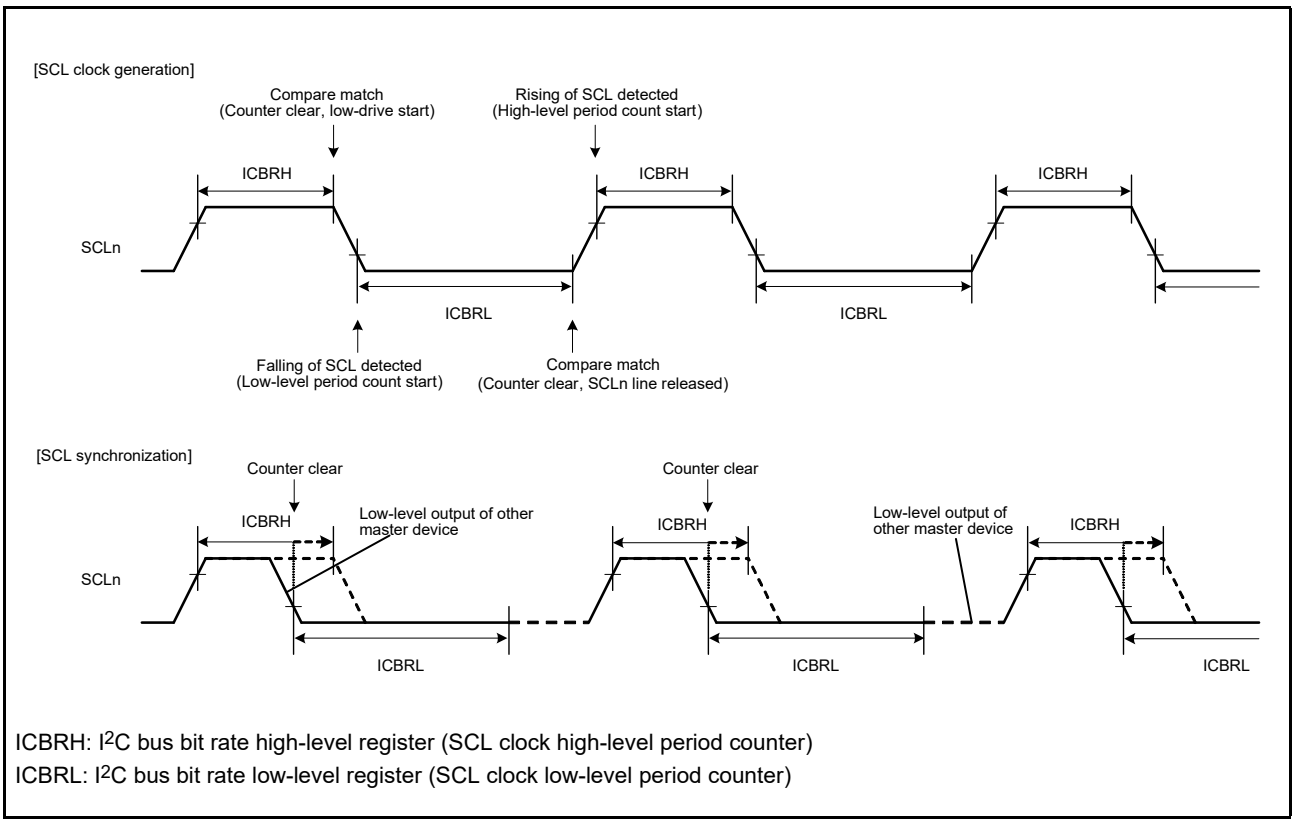
Figure 25.20 Slave Receive Operation Timing (2) (when RDRFS = 0)

25.4 SCL Synchronization Circuit

The RIIC starts counting out the value for width at high level specified in ICBRH when it detects a rising edge on the SCLn line, drives the SCLn line low once counting of the width at high level is complete, and then generates the SCL clock. When the RIIC detects the falling edge of the SCLn line, it starts counting out the width at low level period specified in ICBRL, stops driving the SCLn line (releases the line) once counting of the width at low level is complete, and then generates the SCL clock.

If multiple master devices are connected to the I²C bus, a collision of SCL signals may arise due to contention with another master device. In such cases, the master devices have to synchronize their SCL signals. Since this synchronization of SCL signals must be bit by bit, the RIIC is equipped with a facility (the SCL synchronization circuit) to obtain bit-by-bit synchronization of the SCL clock signals by monitoring the SCLn line while in master mode.

When the RIIC has detected a rising edge on the SCLn line and thus started counting out the width at high level specified in ICBRH, and the level on the SCLn line falls because an SCL signal is being generated by another master device, the RIIC stops counting when it detects the falling edge, drives the level on the SCLn line low, and starts counting out the width at low level specified in ICBRL. When the RIIC finishes counting out the width at low level, it stops driving the SCLn line to the low level (i.e. releases the line). At this time, if the width at low level of the SCL clock signal from the other master device is longer than the width at low level set in the RIIC, the width at low level of the SCL signal will be extended. Once the width at low level for the other master device has ended, the SCL signal rises because the SCLn line has been released. When the RIIC finishes outputting the low-level period of the SCL clock, the SCLn line is released and the SCL clock rises. That is, in cases of contention of SCL signals from more than one master, the width at high level of the SCL signal is synchronized with that of the clock having the narrower width, and the width at low level of the SCL signal is synchronized with that of the clock having the broader width. However, such synchronization of the SCL signal is only enabled when the SCLE bit in ICFER is set to 1.



ICBRH: I²C bus bit rate high-level register (SCL clock high-level period counter)
ICBRL: I²C bus bit rate low-level register (SCL clock low-level period counter)

Figure 25.21 Generation and Synchronization of the SCL Signal from the RIIC

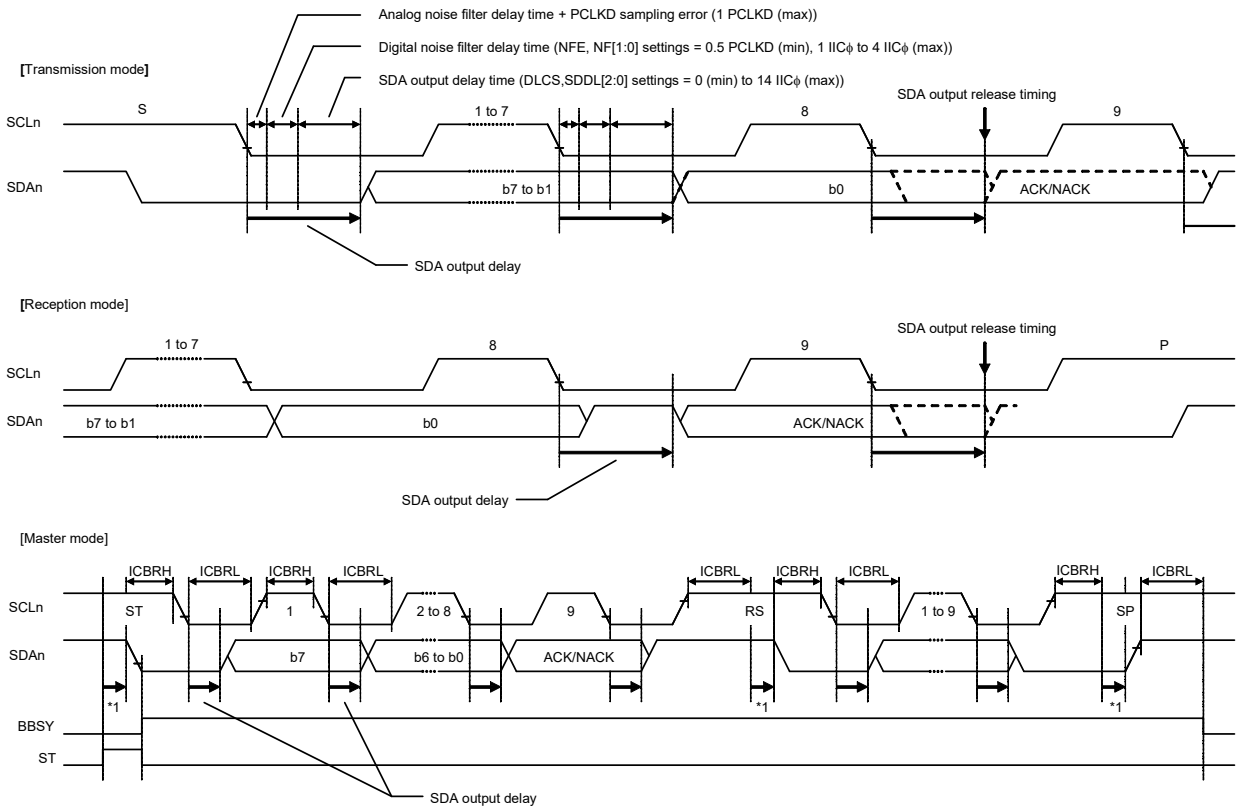
25.5 Facility for Delaying SDA Output

The RIIC module incorporates a facility for delaying output on the SDA line. The delay can be applied to all output (issuing of the start, restart, and stop conditions, data, and the ACK and NACK signals) on the SDA line.

With the SDA output delay facility, SDA output is delayed from detection of a falling edge of the SCL signal to ensure that the SDA signal is output within the interval over which the SCL clock is at the low level. Doing this leads to usage with the aim of preventing erroneous operation of communications devices.

The output delay facility is enabled by setting the SDDL[2:0] bits in ICMR2 to any value other than 000b, and disabled by setting the same bits to 000b.

While the SDA output delay facility is enabled (i.e. while the SDDL[2:0] bits in ICMR2 are set to any value other than 000b), the DLCS bit in ICMR2 selects the clock source for counting by the SDA output delay counter as the internal base clock (IICϕ) for the RIIC module or as a clock signal derived by dividing the frequency of the internal base clock by 2 (IICϕ/2). The counter counts the number of cycles set in the SDDL[2:0] bits in ICMR2. After counting of the set number of cycles of delay is completed, the RIIC module places the required output (start, restart, or stop condition, data, or an ACK or NACK signal) on the SDA line.



Note 1. The output delay facility is set by the DLCS, and SDDL[2:0] bits when a start (ST), restart (RS), or stop (SP) condition is issued.

Figure 25.22 SDA Output Delay Facility

25.6 Digital Noise-Filter Circuits

The states of the SCLn and SDAn pins are conveyed to the internal circuitry through analog noise-filter and digital noise-filter circuits. Figure 25.23 is a block diagram of the digital noise-filter circuit.

The on-chip digital noise-filter circuit of the RIIC consists of four flip-flop circuit stages connected in series and a match-detection circuit.

The number of effective stages in the digital noise filter is selected by the NF[1:0] bits in ICMR3. The selected number of effective stages determines the noise-filtering capability as a period from one to four IICϕ cycles.

The input signal to the SCLn pin (or SDAn pin) is sampled on falling edges of the IICϕ signal. When the input signal level matches the output level of the number of effective flip-flop circuit stages as selected by the NF[1:0] bits in ICMR3, the signal level is conveyed to the subsequent stage. If the signal levels do not match, the previous value is retained.

If the ratio between the frequency of the internal operating clock (PCLKD) and the transfer rate is small (e.g. data transfer at 400 Kbps with PCLKD = 4 MHz), the characteristics of the digital noise filter may lead to the elimination of needed signals as noise. In such cases, it is possible to disable the digital noise-filter circuit (by setting the ICFER.NFE bit to 0) and use only the analog noise-filter circuit.

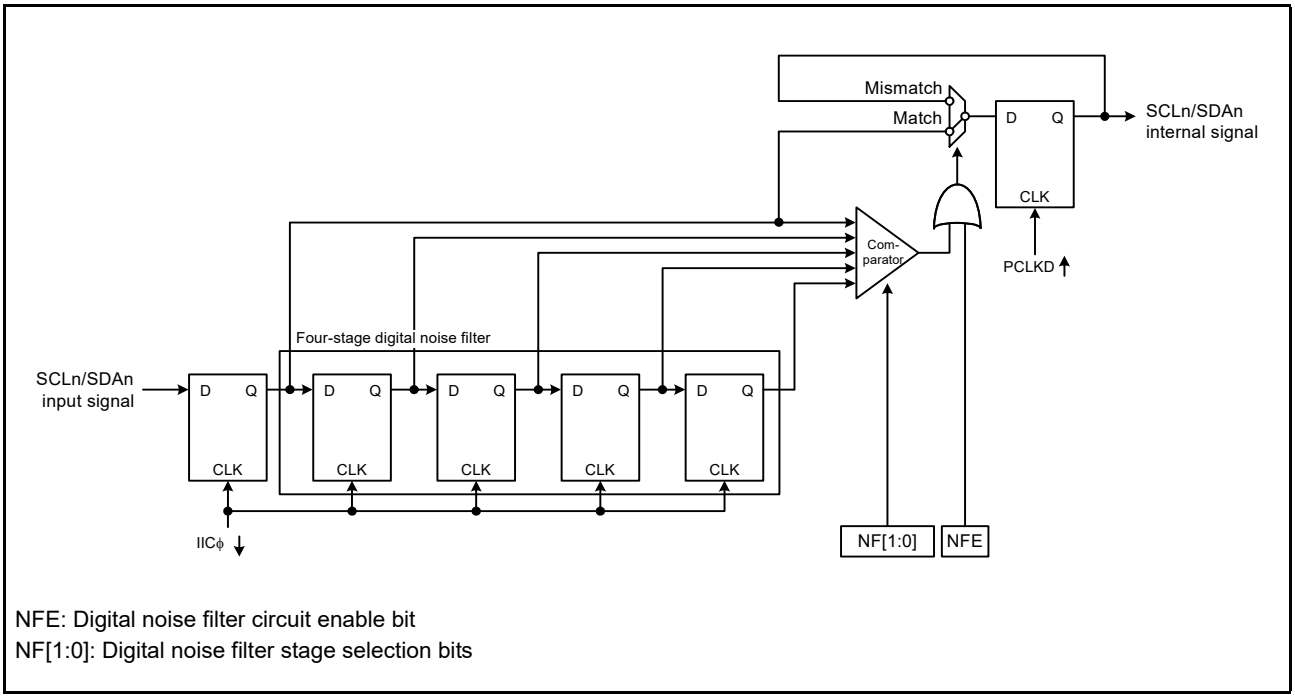


Figure 25.23 Block Diagram of Digital Noise Filter Circuit

25.7 Address Match Detection

The RIIC can set three unique slave addresses in addition to the general call address and host address, and also can set 7-bit or 10-bit slave addresses.

25.7.1 Slave-Address Match Detection

The RIIC can set three unique slave addresses, and has a slave address detection function for each unique slave address. When the SARyE bit (y = 0 to 2) in ICSER is set to 1, the slave addresses set in ICSARUy and ICSARLy (y = 0 to 2) can be detected.

When the RIIC detects a match of the set slave address, the corresponding AASy flag (y = 0 to 2) in ICSR1 is set to 1 at the rising edge of the ninth SCL clock cycle, and the RDRF flag in ICSR2 or the TDRE flag in ICSR2 is set to 1 by the following R/W# bit. This causes a receive data full interrupt (RXI) or transmit data empty interrupt (TXI) to be generated. The AASy flag is used to identify which slave address has been specified.

Figure 25.24 to Figure 25.26 show the AASy flag set timing in three cases.

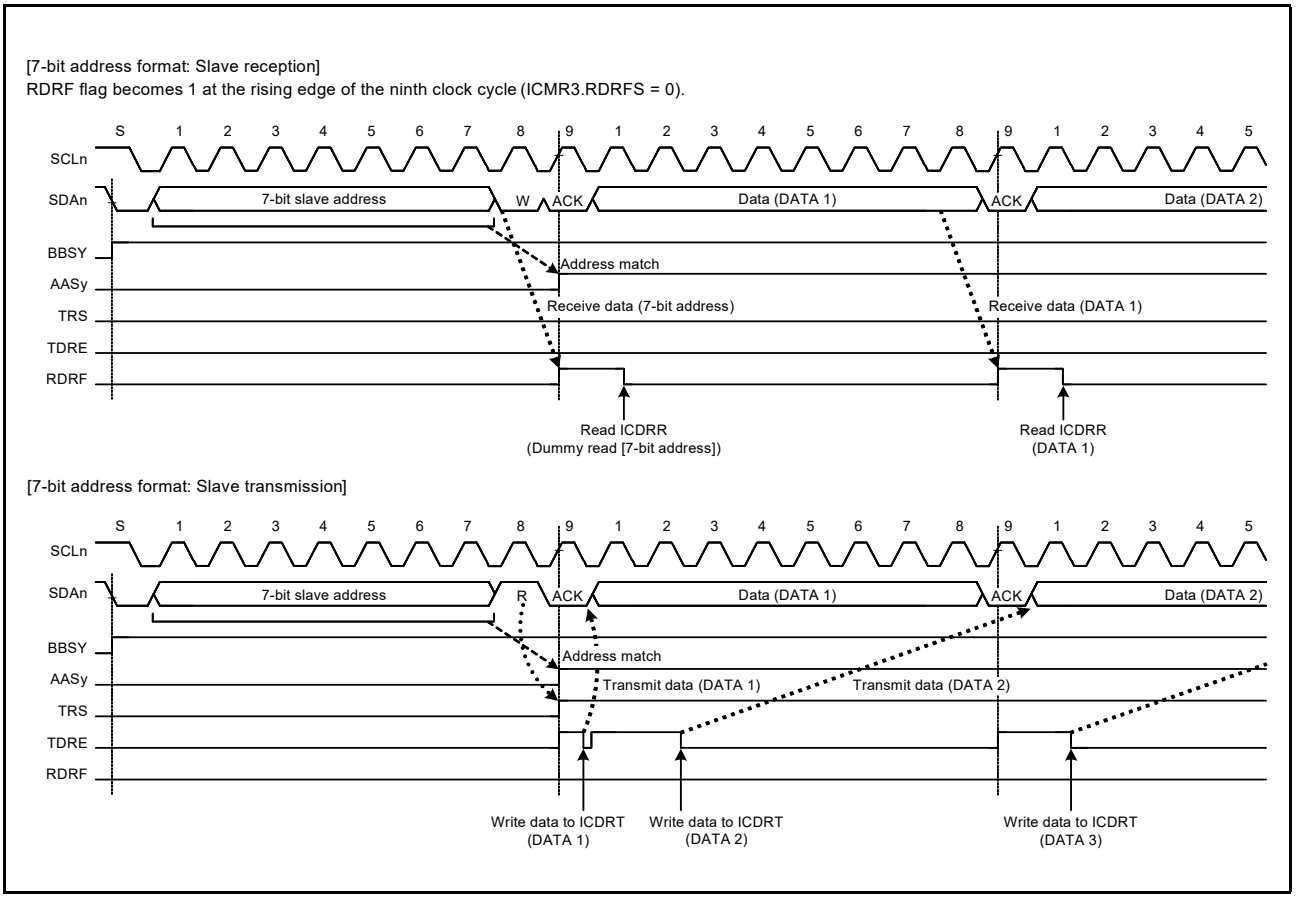


Figure 25.24 AASy Flag Set Timing with 7-Bit Address Format Selected

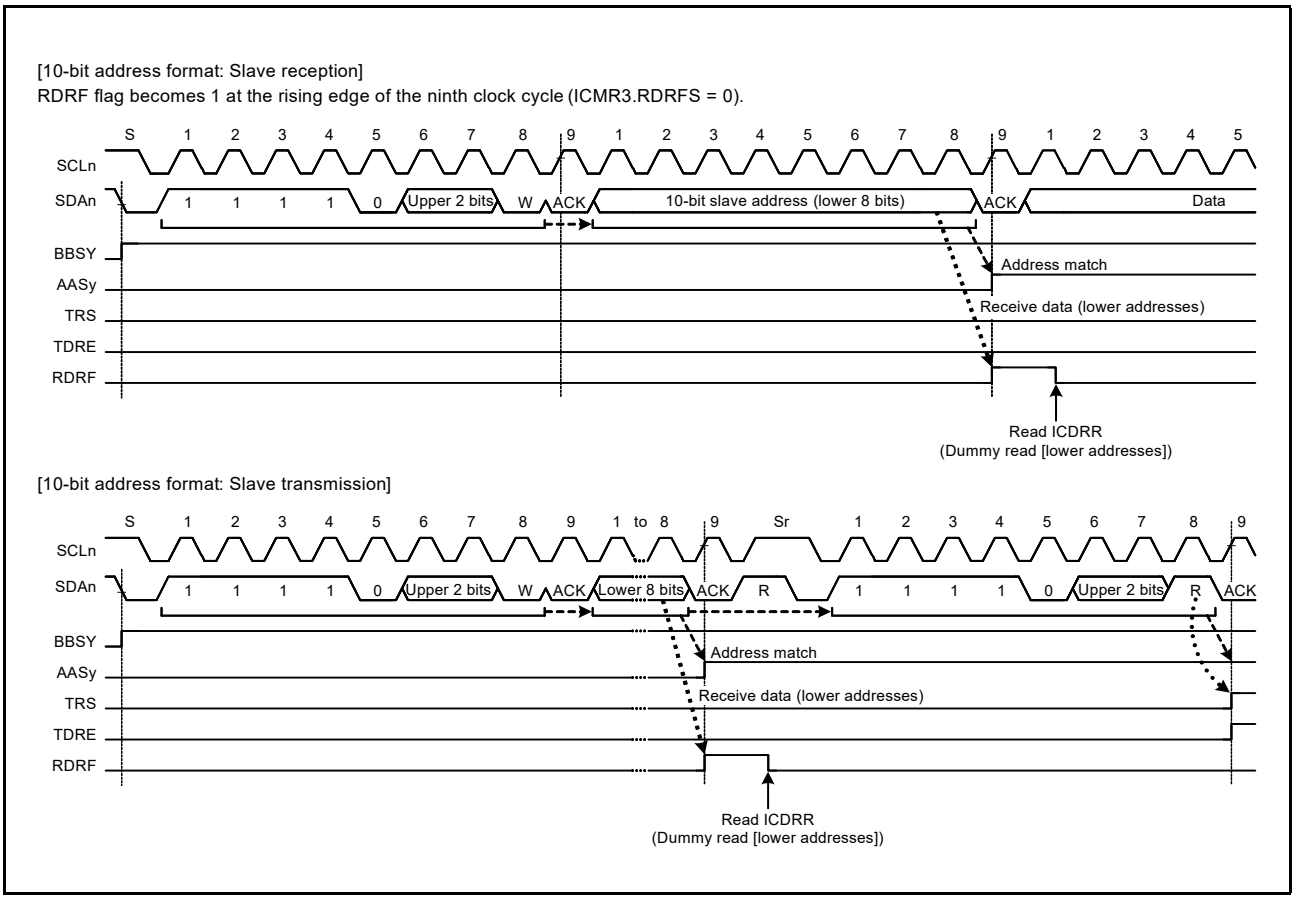


Figure 25.25 AASy Flag Set Timing with 10-Bit Address Format Selected

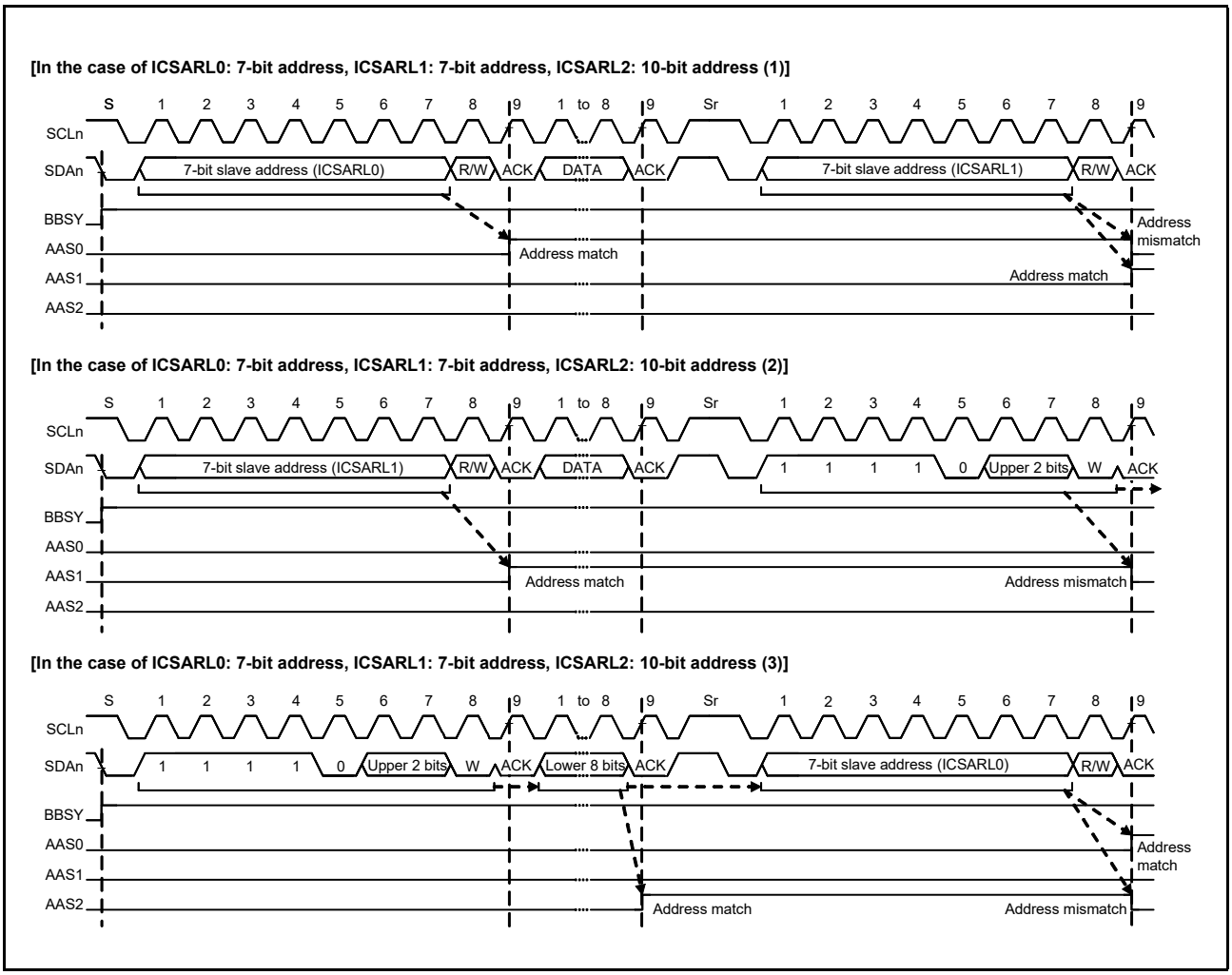


Figure 25.26 AASy Flag Set/Clear Timing with 7-Bit/10-Bit Address Formats Mixed

25.7.2 Detection of the General Call Address

The RIIC has a facility for detecting the general call address (0000 000b + 0 [W]). This is enabled by setting the GCAE bit in ICSER to 1.

If the address received after a start or restart condition is issued is 0000 000b + 1[R] (start byte), the RIIC recognizes this as the address of a slave device with an “all-zero” address but not as the general call address.

When the RIIC detects the general call address, both the GCA flag in ICSR1 and the RDRF flag in ICSR2 are set to 1 on the rising edge of the ninth cycle of SCL clock. This leads to the generation of a receive data full interrupt (RXI). The value of the GCA flag can be confirmed to recognize that the general call address has been transmitted.

Operation after detection of the general call address is the same as normal slave receive operation.

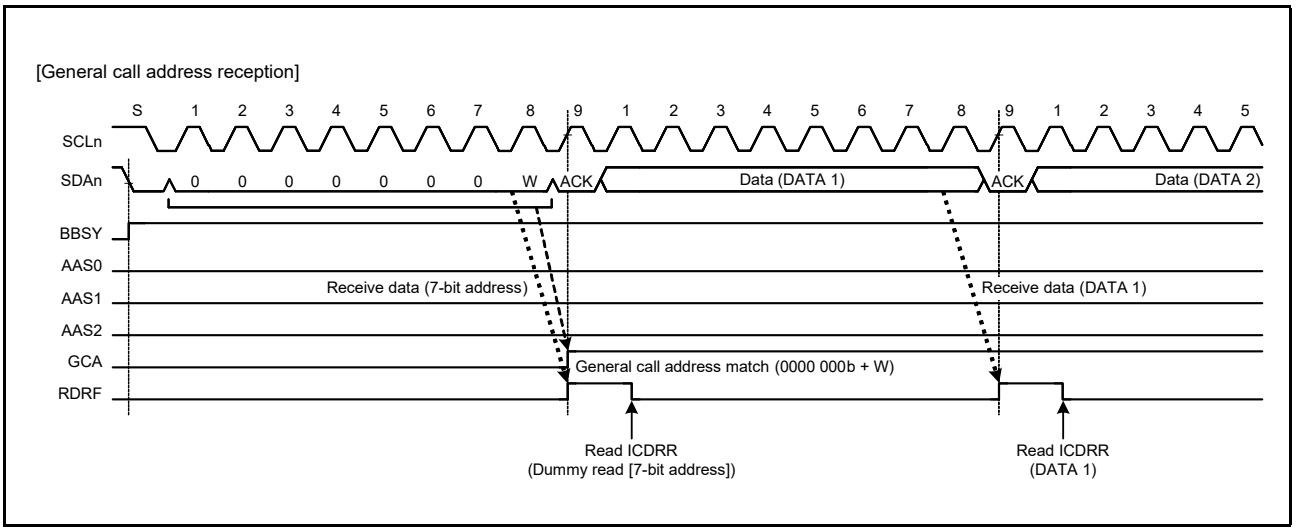


Figure 25.27 Timing of GCA Flag Setting during Reception of General Call Address

25.7.3 Device-ID Address Detection

The RIIC module has a facility for detecting device-ID addresses conforming with the I2C bus specification (Rev. 03). When the RIIC receives 1111 100b as the first byte after a start condition or restart condition was issued with the DIDE bit in ICSER set to 1, the RIIC recognizes the address as a device ID, sets the DID flag in ICSR1 to 1 on the rising edge of the eighth SCL clock cycle when the following R/W# bit is 0, and then compares the second and subsequent bytes with its own slave address. If the address matches the value in the slave address register, the RIIC sets the corresponding AASy flag (y = 0 to 2) in ICSR1 to 1.

After that, when the first byte received after a start or restart condition is issued matches the device ID address (1111 100b) again and the following R/W# bit is 1, the RIIC does not compare the second and subsequent bytes and sets the ICSR2.TDRE flag to 1.

In the device-ID address detection function, the RIIC sets the DID flag to 0 if a match with the RIIC's own slave address is not obtained or a match with the device ID address is not obtained after a match with the RIIC's own slave address and the detection of a restart condition. In this case, if the first byte after detection of a start or restart condition matches the device ID address (1111 100b) and the R/W# bit is 0, the RIIC sets the DID flag to 1 and compares the second and subsequent bytes with the RIIC's slave address. If the R/W# bit is 1, the DID flag holds the previous value and the RIIC does not compare the second and subsequent bytes. Therefore, the reception of a device-ID address can be checked by reading the DID flag after confirming that TDRE = 1.

Furthermore, prepare the device-ID fields (three bytes: 12 bits indicating the manufacturer + 9 bits identifying the part + 3 bits indicating the revision) that must be sent to the host after reception of a continuous device-ID field as normal data for transmission. For details of the information that must be included in device-ID fields, contact NXP Semiconductors.

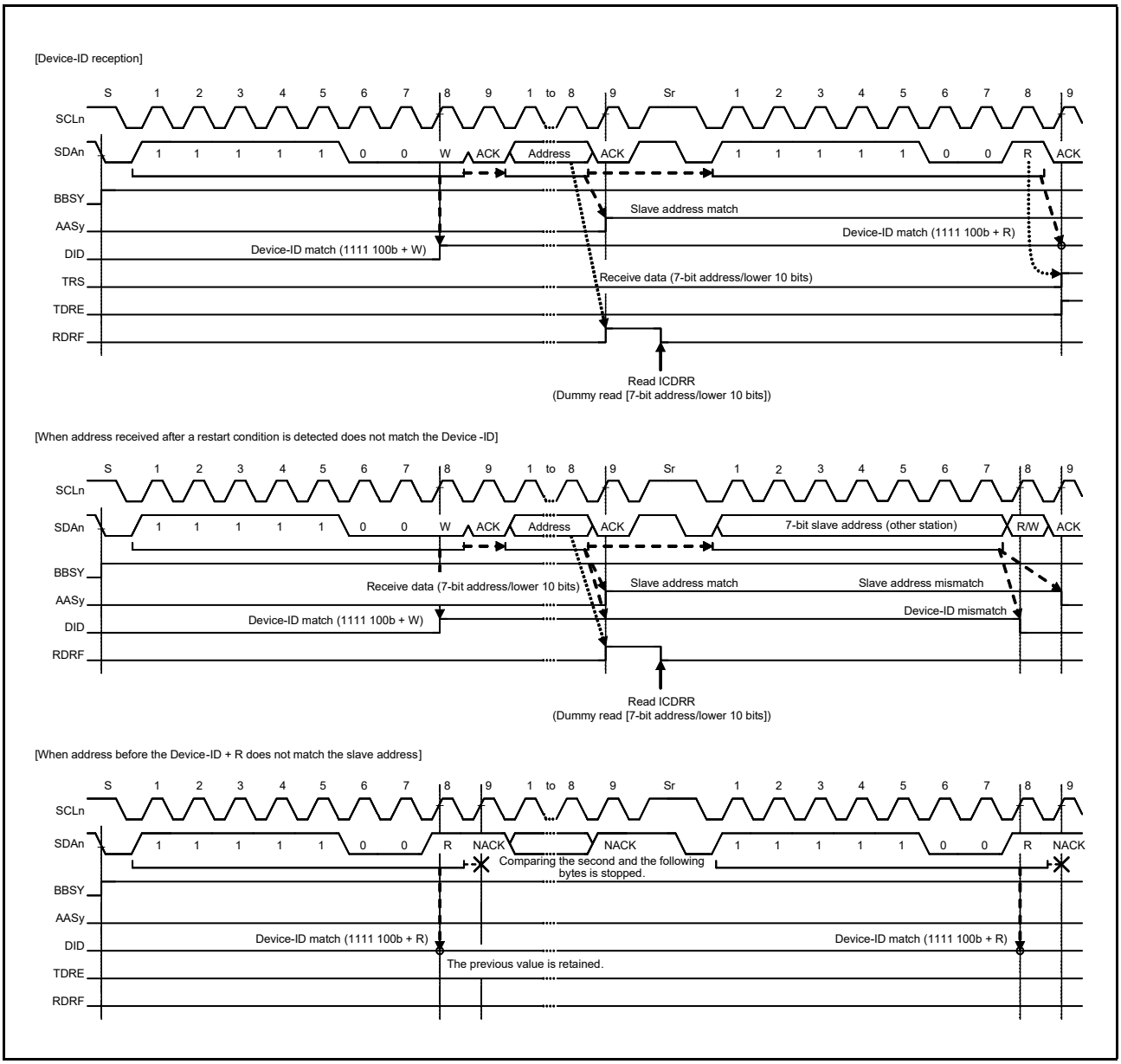


Figure 25.28 AASy/DID Flag Set/Clear Timing during Reception of Device-ID

25.8 Automatic Low-Hold Function for SCL

25.8.1 Function to Prevent Wrong Transmission of Transmit Data

If the shift register (ICDRS) is empty when data have not been written to the I2C bus transmit data register (ICDRT) with the RIIC in transmission mode (TRS bit = 1 in ICCR2), the SCLn line is automatically held at the low level over the intervals shown below. This low-hold period is extended until data for transmission have been written, which prevents the unintended transmission of erroneous data.

Master transmission mode

- Low-level interval after a start condition or restart condition is issued
- Low-level interval between the ninth clock cycle of one transfer and the first clock cycle of the next

Slave transmission mode

- Low-level interval between the ninth clock cycle of one transfer and the first clock cycle of the next

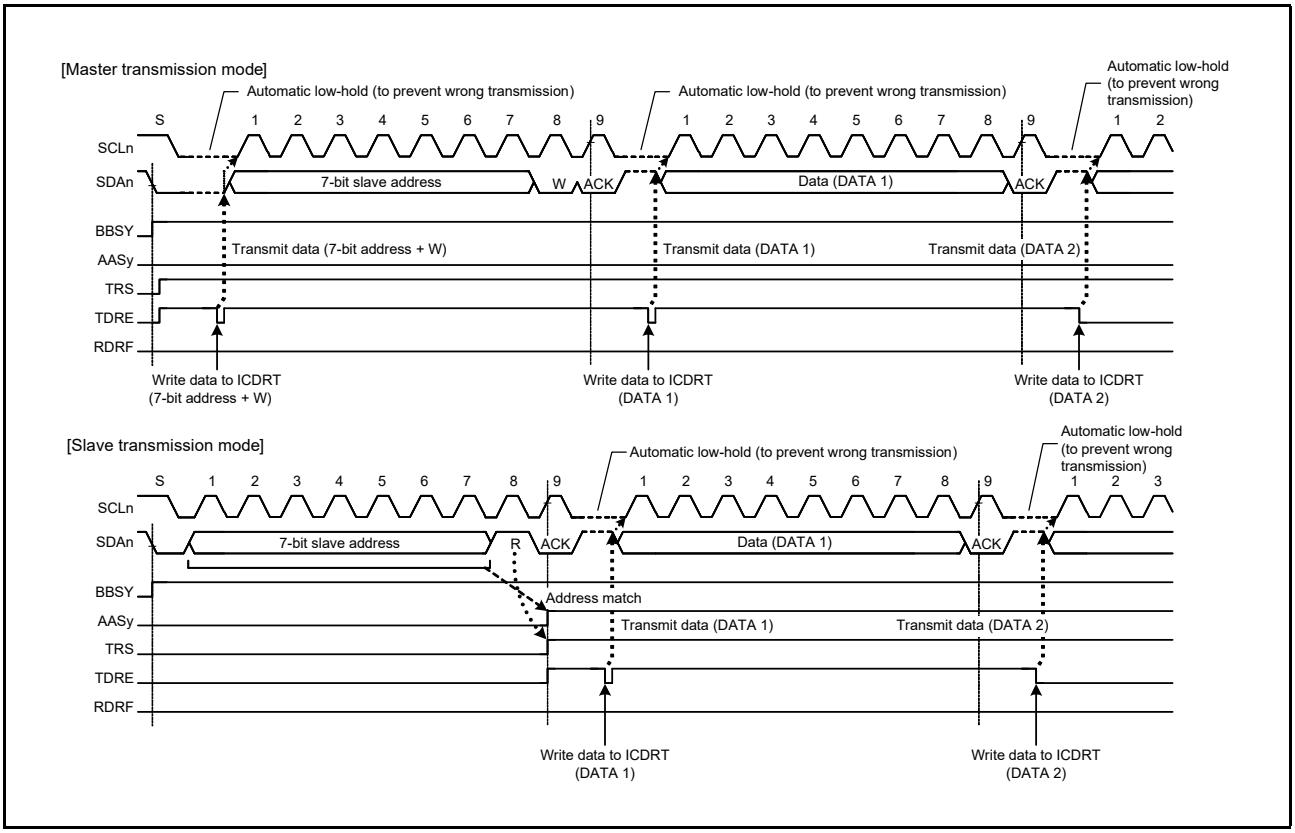


Figure 25.29 Automatic Low-Hold Operation in Transmission Mode

25.8.2 NACK Reception Transfer Suspension Function

The RIIC has a function to suspend transfer operation when NACK is received in transmission mode (TRS bit = 1 in ICCR2). This function is enabled when the NACKE bit in ICFER is set to 1 (transfer suspension enabled). If the next transmit data has already been written (TDRE flag = 0 in ICSR2) when NACK is received, next data transmission at the falling edge of the ninth SCL clock cycle is automatically suspended. This prevents the SDAn line output level from being held low when the MSB of the next transmit data is 0.

If the transfer operation is suspended by this function (NACKF flag = 1 in ICSR2), transmit operation and receive operation are discontinued. To restore transmit/receive operation, be sure to set the NACKF flag to 0. In master transmission mode, set the NACKF flag to 0, issue a restart or stop condition, and then issue a start condition again.

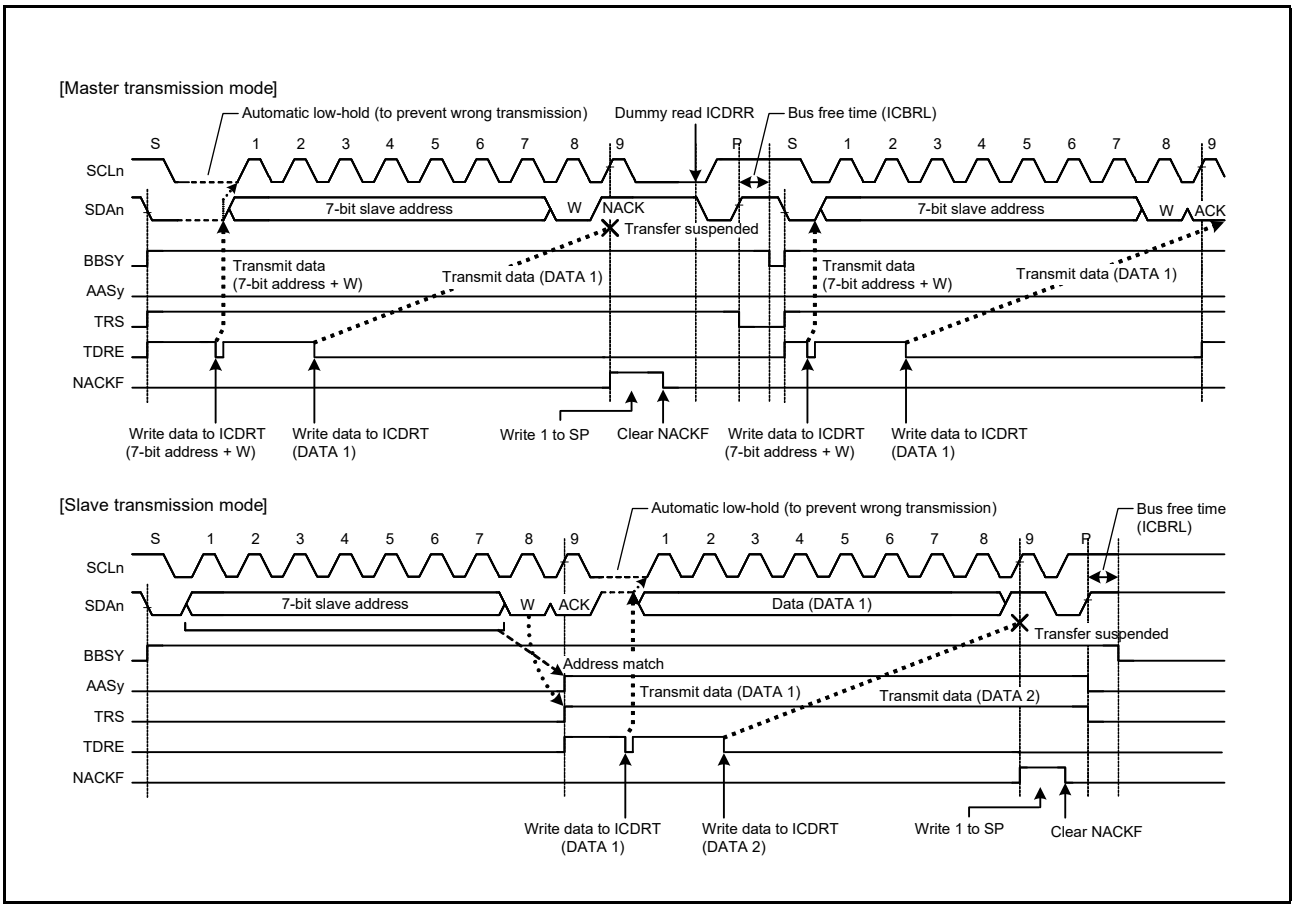


Figure 25.30 Suspension of Data Transfer When NACK is Received (NACKE = 1)

25.8.3 Function to Prevent Failure to Receive Data

If response processing is delayed when receive data (ICDRR) read is delayed for a period of one transfer frame or more with receive data full (RDRF flag = 1 in ICSR2) in reception mode (TRS = 0 in ICCR2), the RIIC holds the SCLn line low automatically immediately before the next data is received to prevent failure to receive data.
This function to prevent failure to receive data using the automatic low-hold function is also enabled even if the read processing of the final receive data is delayed and, in the meantime, the RIIC's own slave address is designated after a stop condition is issued. This function does not disturb other communication because the RIIC does not hold the SCLn line low when a mismatch with its own slave address occurs after a stop condition is issued.
Sections in which the SCLn line is held low can be selected with a combination of the WAIT and RDRFS bits in ICMR3.

(1) 1-Byte Receive Operation and Automatic Low-Hold Function Using the WAIT Bit

When the WAIT bit in ICMR3 is set to 1, the RIIC performs 1-byte receive operation using the WAIT bit function. When the ICMR3.RDRFS bit is 0, the ACKBT bit value in ICMR3 is automatically sent to the acknowledge bit in the period from the falling edge of the eighth SCL clock cycle to the falling edge of the ninth SCL clock cycle. When the falling edge of the ninth SCL clock cycle is detected, the SCLn line is automatically held low by the WAIT bit function. This low-hold is released by reading data from ICDRR, which enables bytewise receive operation.
The WAIT bit function is enabled for receive frames after a match with the RIIC's own slave address (including the general call address and host address) is obtained in master reception mode or slave reception mode.

(2) 1-Byte Receive Operation (ACK/NACK Transmission Control) and Automatic Low-Hold Function Using the RDRFS Bit

When the RDRFS bit in ICMR3 is set to 1, the RIIC performs 1-byte receive operation using the RDRFS bit function. When the RDRFS bit is set to 1, the RDRF flag (receive data full) in ICSR2 is set to 1 at the rising edge of the eighth SCL clock cycle, and the SCLn line is automatically held low at the falling edge of the eighth SCL clock cycle. This low-hold is released by writing a value to the ACKBT bit in ICMR3, but cannot be released by reading data from ICDRR, which enables receive operation by the ACK/NACK transmission control according to the data received in byte units.
The RDRFS bit function is enabled for receive frames after a match with the RIIC's own slave address (including the general call address and host address) is obtained in master reception mode or slave reception mode.

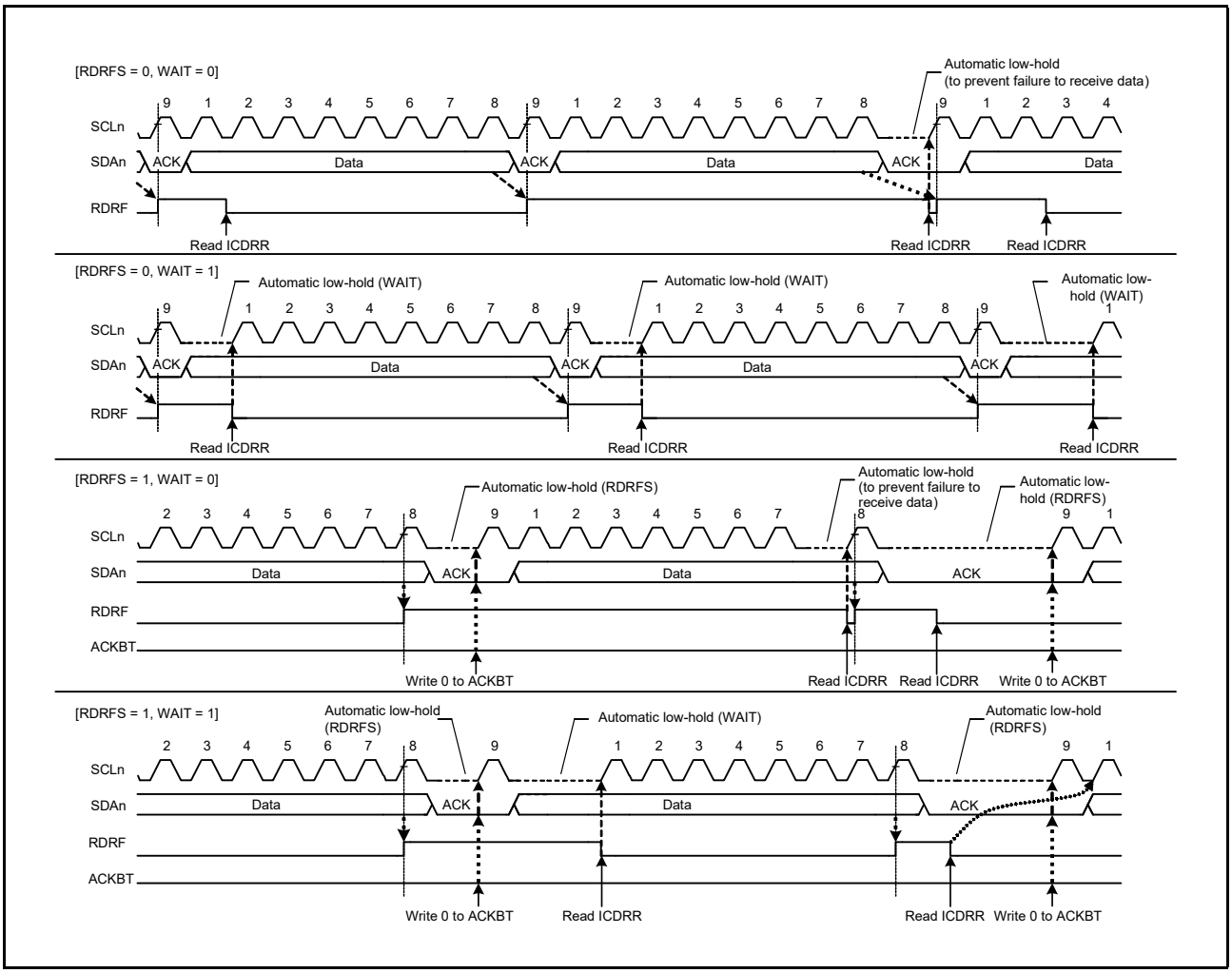


Figure 25.31 Automatic Low-Hold Operation in Reception Mode (Using RDRFS and WAIT Bits)

25.9 Arbitration-Lost Detection Functions

In addition to the normal arbitration-lost detection function defined by the I2C bus standard, the RIIC has functions to prevent double-issue of a start condition, to detect arbitration-lost during transmission of NACK, and to detect arbitration-lost in slave transmission mode.

25.9.1 Master Arbitration-Lost Detection (MALE Bit)

The RIIC drives the SDAn line low to issue a start condition. However, if the SDAn line has already been driven low by another master device issuing a start condition, the RIIC regards its own issuing of a start condition as an error and considers this a loss in arbitration, so priority is given to transfer by the other master device. Similarly, if a request to issue a start condition is made by setting the ST bit in ICCR2 to 1 while the bus is busy (BBSY flag = 1 in ICCR2), the RIIC regards this as a double-issuing-of-start-condition error and considers itself to have lost in arbitration, thus preventing a failure of transfer due to issuing of a start condition while transfer is in progress.
When a start condition is issued successfully, the data for transmission including the address bits (i.e. the internal SDA output level) and the level on the SDAn line do not match (the high output as the internal SDA output; i.e. the SDAn pin is in the high-impedance state, and the low level is detected on the SDAn line), the RIIC loses in arbitration.
After a loss in arbitration of mastership, the RIIC immediately enters slave reception mode. If a slave address (including the general call address) matches its own address at this time, the RIIC continues in slave operation.
A loss in arbitration of mastership is detected when the following conditions are met while the MALE bit in ICFER is 1 (master arbitration-lost detection enabled).
[Master arbitration-lost conditions]

- Non-matching of the internal level for output on SDA and the level on the SDAn line after a start condition was issued by setting the ST bit in ICCR2 to 1 while the BBSY flag in ICCR2 was set to 0 (erroneous issuing of a start condition)
- Setting of the ST bit in ICCR2 to 1 (start condition double-issue error) while the BBSY flag in ICCR2 is set to 1
- When the transmit data excluding acknowledge (internal SDA output level) does not match the level on the SDAn line in master transmission mode (MST and TRS bits = 11b in ICCR2)

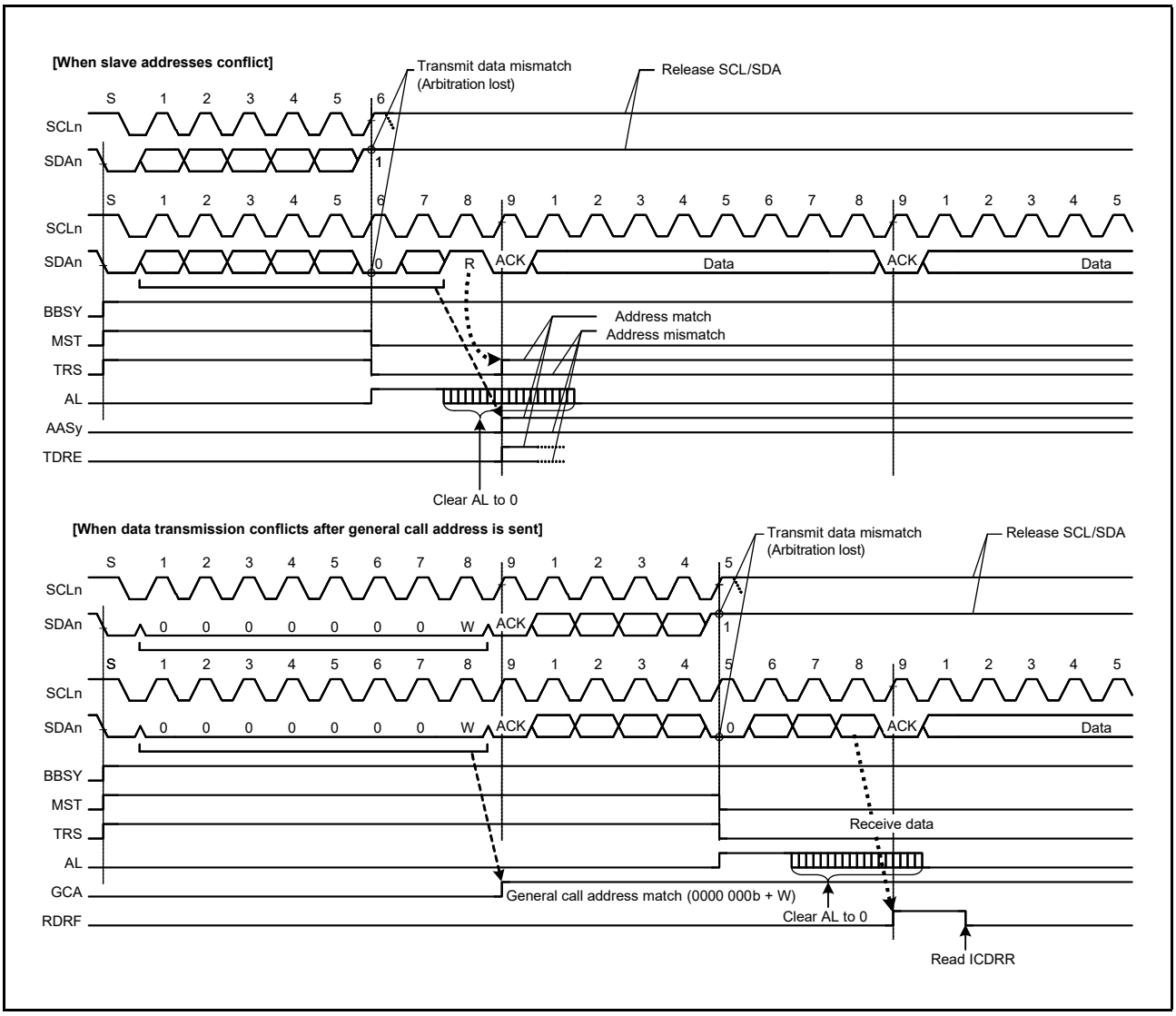


Figure 25.32 Examples of Master Arbitration-Lost Detection (MALE = 1)

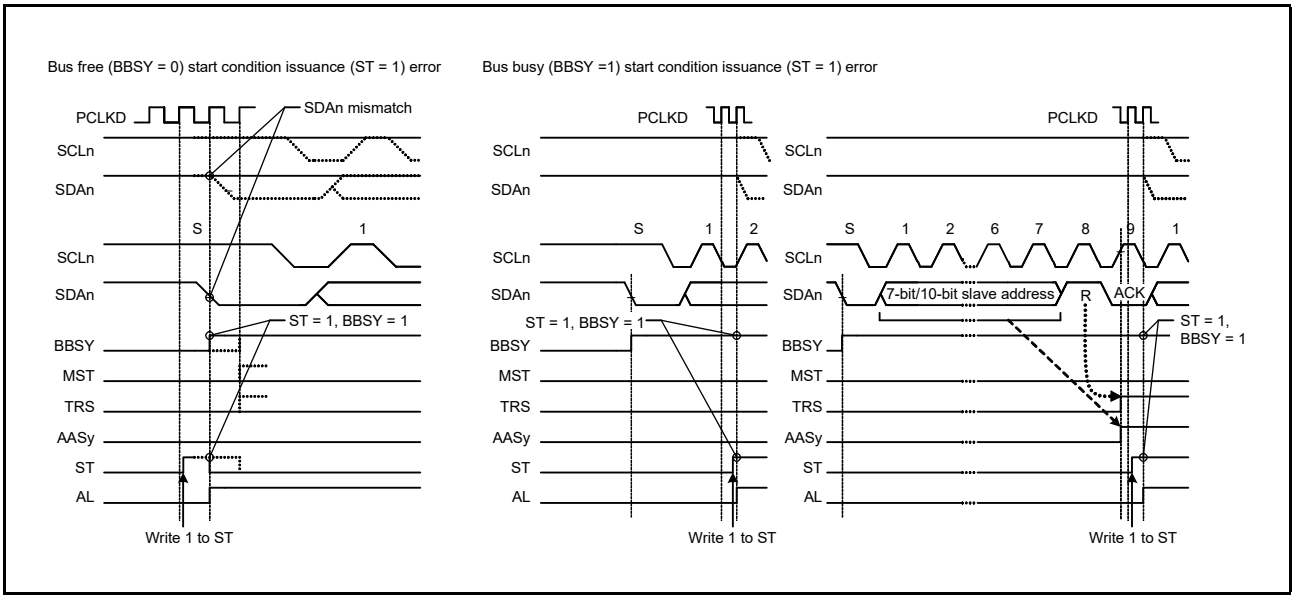


Figure 25.33 Arbitration-Lost When a Start Condition is Issued (MALE = 1)

25.9.2 Function to Detect Loss of Arbitration during NACK Transmission (NALE Bit)

The RIIC has a function to cause arbitration to be lost if the internal SDA output level does not match the level on the SDAn line (the high output as the internal SDA output; i.e. the SDAn pin is in the high-impedance state, and the low level is detected on the SDAn line) during transmission of NACK in reception mode. Arbitration is lost due to a conflict of NACK transmission and ACK transmission when two or more master devices receive data from the same slave device simultaneously in a multi-master system. Such conflict occurs when multiple master devices send/receive the same information through a single slave device. Figure 25.34 shows an example of arbitration-lost detection during transmission of NACK.

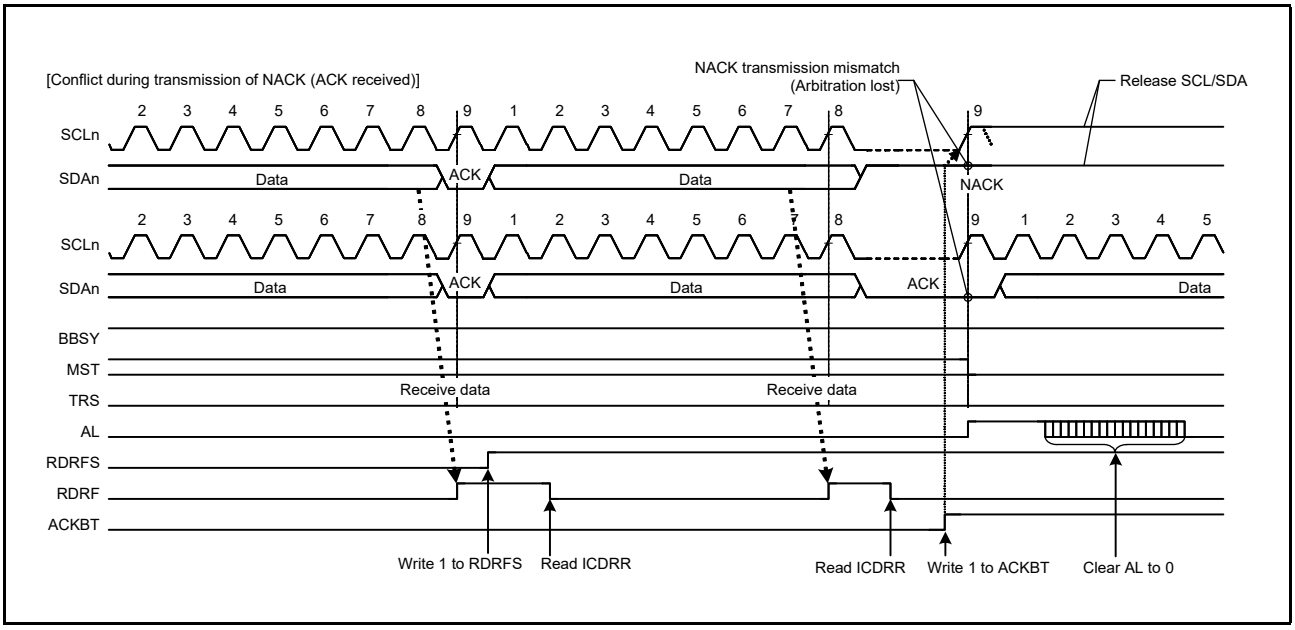


Figure 25.34 Example of Arbitration-Lost Detection during Transmission of NACK (NALE = 1)

The following explains arbitration-lost detection using an example where two master devices (master A and master B) and a single slave device are connected through the bus. In this example, master A receives two bytes of data from the slave device, and master B receives four bytes of data from the slave device.

If master A and master B access the slave device simultaneously, because the slave address is identical, arbitration is not lost in both master A and master B during access to the slave device. Therefore, both master A and master B recognize that they have obtained the bus mastership and operate as such. Here, master A sends NACK when it has received two final bytes of data from the slave device. Meanwhile, master B sends ACK because it has not received necessary four bytes of data. At this time, the NACK transmission from master A and the ACK transmission from master B conflict. In general, if a conflict like this occurs, master A cannot detect ACK transmitted by master B and issues a stop condition. Therefore, the issuance of the stop condition conflicts with the SCL clock output of master B, which disturbs communication.

When the RIIC receives ACK during transmission of NACK, it detects a defeat in conflict with other master devices and causes arbitration to be lost.

If arbitration is lost during transmission of NACK, the RIIC immediately cancels the slave match condition and enters slave reception mode. This prevents a stop condition from being issued, preventing a communication failure on the bus.

The RIIC detects arbitration-lost during transmission of NACK when the following condition is met with the NALE bit in ICFER set to 1 (arbitration-lost detection during NACK transmission enabled).

[Condition for arbitration-lost during NACK transmission]

- When the internal SDA output level does not match the SDAn line (ACK is received) during transmission of NACK (ACKBT bit = 1 in ICMR3)

25.9.3 Slave Arbitration-Lost Detection (SALE Bit)

The RIIC has a function to cause arbitration to be lost if the data for transmission (i.e. the internal SDA output level) and the level on the SDAn line do not match (the high output as the internal SDA output; i.e. the SDAn pin is in the high-impedance state, and the low level is detected on the SDAn line) in slave transmission mode.
When it loses slave arbitration, the RIIC is immediately released from the slave-matched state and enters slave reception mode.
The RIIC detects slave arbitration-lost when the following condition is met with the SALE bit in ICFER set to 1 (slave arbitration-lost detection enabled).

[Condition for slave arbitration-lost]

- When transmit data excluding acknowledge (internal SDA output level) does not match the SDAn line in slave transmission mode (MST and TRS bits = 01b in ICCR2)

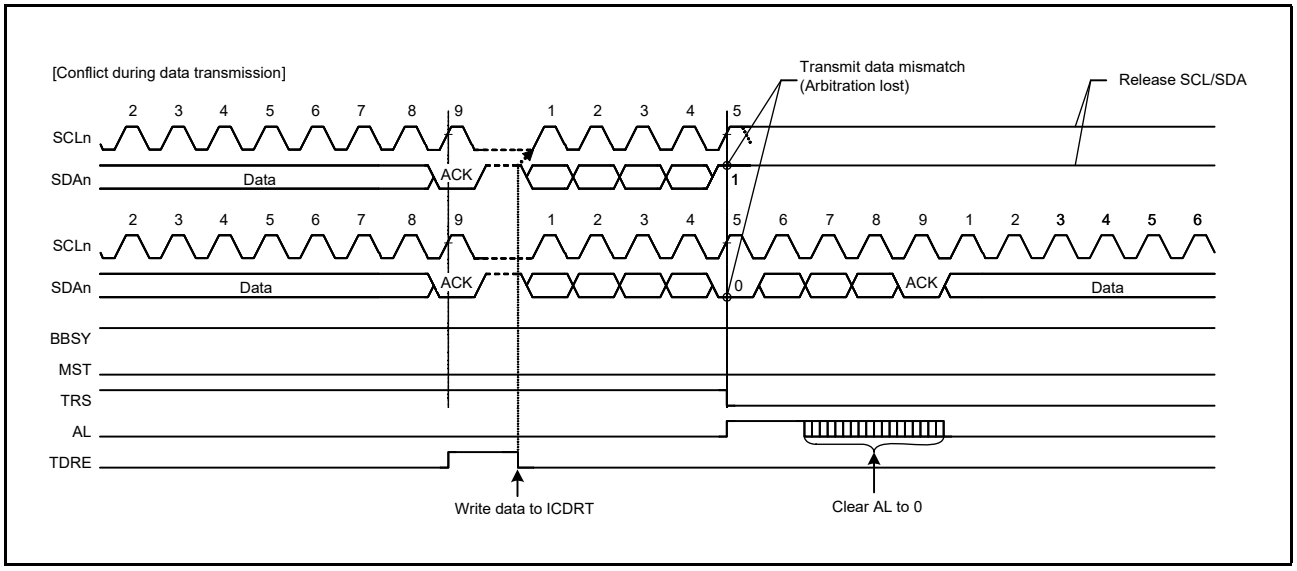


Figure 25.35 Example of Slave Arbitration-Lost Detection (SALE = 1)

25.10 Start Condition/Restart Condition/Stop Condition Issuing Function

25.10.1 Issuing a Start Condition

The RIIC issues a start condition when the ST bit in ICCR2 is set to 1.
When the ST bit is set to 1, a start condition issuance request is made and the RIIC issues a start condition when the BBSY flag in ICCR2 is 0 (bus free state). When a start condition is issued normally, the RIIC automatically shifts to the master transmission mode.
A start condition is issued in the following sequence.

[Start condition issuance]
(1) Drive the SDAn line low (high level to low level).
(2) Ensure the start condition hold time set in ICBRH.
(3) Drive the SCLn line low (high level to low level).
(4) Detect low level of the SCLn line and ensure the low-level period of SCLn line set in ICBRL.

25.10.2 Issuing a Restart Condition

The RIIC issues a restart condition when the RS bit in ICCR2 is set to 1.
When the RS bit is set to 1, a restart condition issuance request is made and the RIIC issues a restart condition when the BBSY flag in ICCR2 is 1 (bus busy state) and the MST bit in ICCR2 is 1 (master mode).
A restart condition is issued in the following sequence.

[Restart condition issuance]
(1) Release the SDAn line.
(2) Ensure the low-level period of SCLn line set in ICBRL.
(3) Release the SCLn line (low level to high level).
(4) Detect a high level of the SCLn line and ensure the restart condition setup time set in ICBRL.
(5) Drive the SDAn line low (high level to low level).
(6) Ensure the restart condition hold time set in ICBRH.
(7) Drive the SCLn line low (high level to low level).
(8) Detect a low level of the SCLn line and ensure the low-level period of SCLn line set in ICBRL.

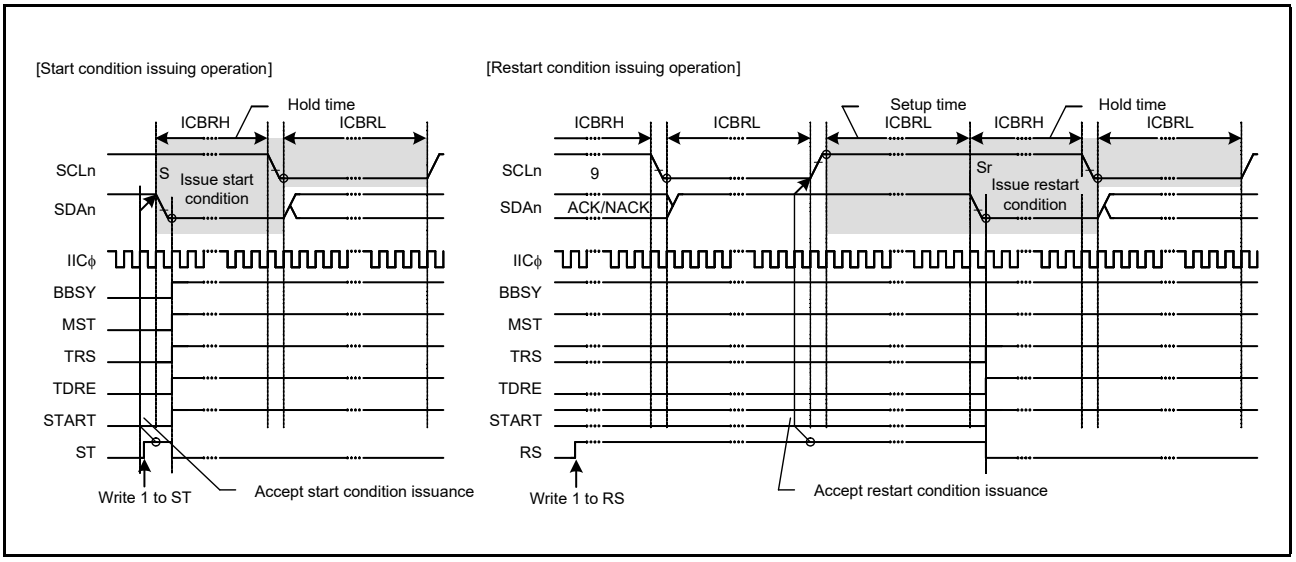


Figure 25.36 Start Condition/Restart Condition Issue Timing (ST and RS Bits)

25.10.3 Issuing a Stop Condition

The RIIC issues a stop condition when the SP bit in ICCR2 is set to 1.
When the SP bit is set to 1, a stop condition issuance request is made and the RIIC issues a stop condition when the BBSY flag in ICCR2 is 1 (bus busy state) and the MST bit in ICCR2 is 1 (master mode).
A stop condition is issued in the following sequence.

[Stop condition issuance]

- Drive the SDAn line low (high level to low level).
- Ensure the low-level period of SCLn line set in ICBRL.
- Release the SCLn line (low level to high level).
- Detect a high level of the SCLn line and ensure the stop condition setup time set in ICBRH.
- Release the SDAn line (low level to high level).
- Ensure the bus free time set in ICBRL.
- Clear the BBSY flag to 0 (to release the bus mastership).

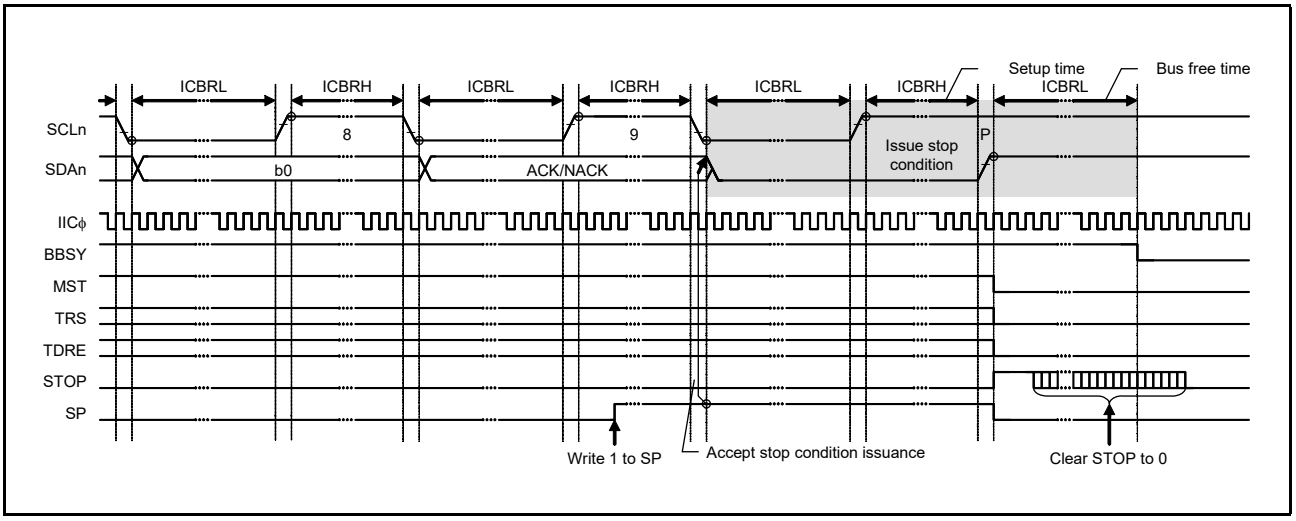


Figure 25.37 Stop Condition Issue Timing (SP Bit)

25.11 Bus Hanging

If the clock signals from the master and slave devices go out of synchronization due to noise or other factors, the I²C bus might hang with a fixed level on the SCLn line and/or SDAn line.
As measures against the bus hanging, the RIIC has a timeout function to detect hanging by monitoring the SCLn line, a function for the output of an extra SCL clock cycle to release the bus from a hung state due to clock signals being out of synchronization, the RIIC reset function, and internal reset function.
By checking the SCLO, SDAO, SCLI, and SDAI bits in ICCR1, it is possible to see whether the RIIC or its partner in communications is placing the low level on the SCLn or SDAn line.

25.11.1 Timeout Function

The RIIC includes a timeout function for detecting when the SCLn line has been stuck longer than the predetermined time. The RIIC can detect an abnormal bus state by monitoring that the SCLn line is stuck low or high for a predetermined time.
The timeout function monitors the SCLn line state and counts the low-level period or high-level period using the internal counter. The timeout function resets the internal counter each time the SCLn line changes (rising or falling), but continues to count unless the SCLn line changes. If the internal counter overflows due to no SCLn line change, the RIIC can detect the timeout and report the bus hung state.
This timeout function is enabled when the ICFER.TMOE bit is 1. It detects a hung state that the SCLn line is stuck low or high during the following conditions:

- The bus is busy (ICCR2.BBSY flag is 1) in master mode (ICCR2.MST bit is 1).
- The RIIC's own slave address is detected (ICSR1 register is not 00h) and the bus is busy (ICCR2.BBSY flag is 1) in slave mode (ICCR2.MST bit is 0).
- The bus is free (ICCR2.BBSY flag is 0) while generation of a start condition is requested (ICCR2.ST bit is 1).

The internal counter of the timeout function works using the internal reference clock (IICϕ) set by the CKS[2:0] bits in ICMR1 as a count source. It functions as a 16-bit counter when long mode is selected (TMOS bit = 0 in ICMR2) or a 14-bit counter when short mode is selected (TMOS bit = 1).
The SCLn line level (low/high or both levels) during which this counter is activated can be selected by the setting of the TMOH and TMOL bits in ICMR2. If both TMOL and TMOH bits are set to 0, the internal counter does not work.

Note: When the timeout function for detecting is to be used, see section 25.2.4, I²C Bus Mode Register 2 (ICMR2), section 25.3, Operation, and section 25.3.2, Initial Settings.

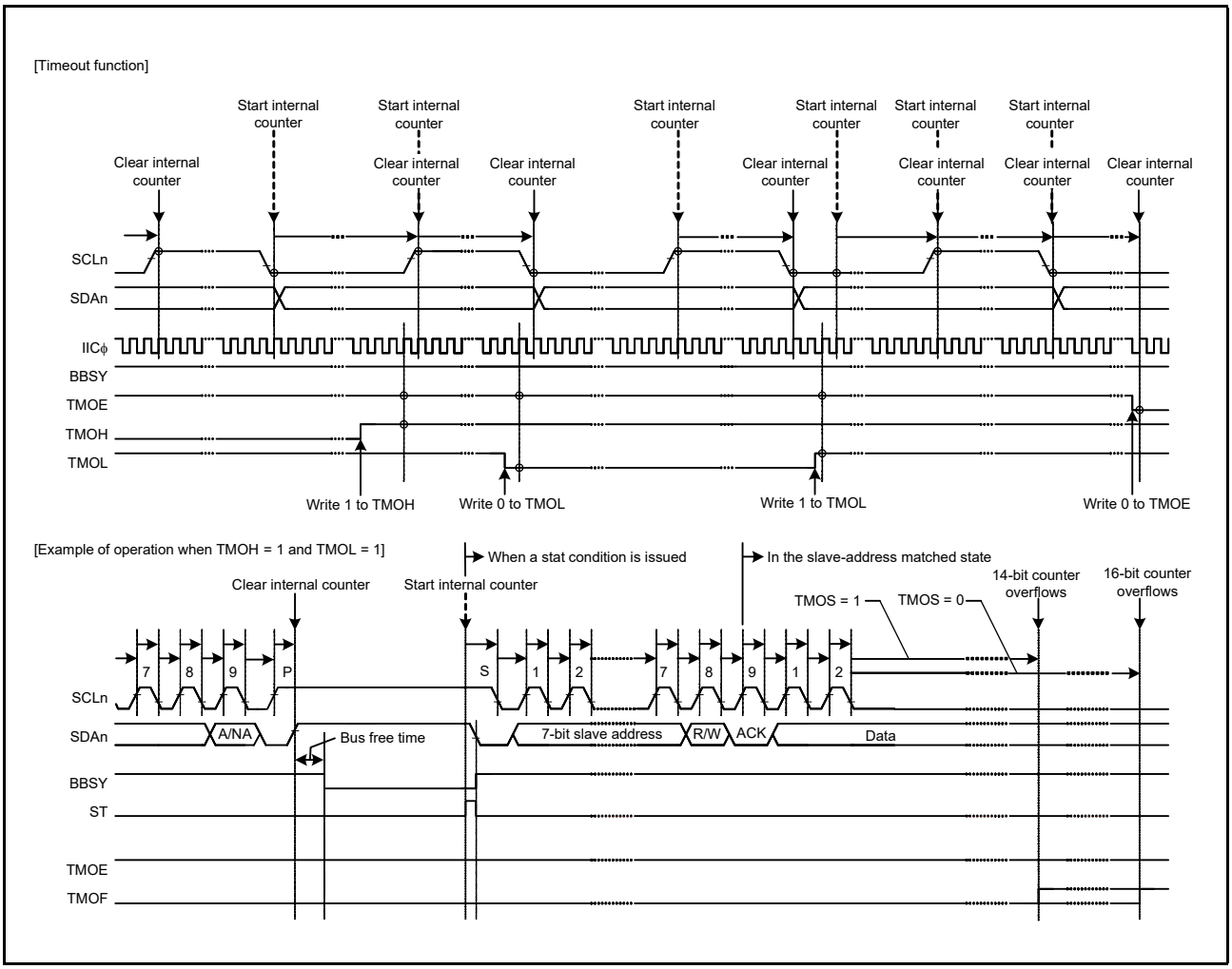


Figure 25.38 Timeout Function (TMOE, TMOS, TMOH, and TMOL Bits)

25.11.2 Extra SCL Clock Cycle Output Function

In master mode, the RIIC module has a facility for the output of extra SCL clock cycles to release the SDAn line of the slave device from being held at the low level due to the master being out of synchronization with the slave device.
This function is mainly used in master mode to release the SDAn line of the slave device from the state of being fixed to the low level by including extra cycles of SCL output from the RIIC with single cycles of the SCL clock as the unit in the case of a bus error where the RIIC cannot issue a stop condition because the slave device is holding the SDAn line at the low level. Do not use this facility in normal situations. Using it when communications are proceeding correctly will lead to malfunctions.
When the CLO bit in ICCR1 is set to 1 in master mode, a single cycle of the SCL clock at the frequency corresponding to the transfer rate settings (settings of the CKS[2:0] bits in ICMR1, and of the ICBRH and ICBRL registers) is output as an extra clock cycle. After output of this single cycle of the SCL clock, the CLO bit is automatically set to 0. Therefore, further extra clock cycles can be output consecutively by the software program writing 1 to the CLO bit after having read CLO = 0.
When the RIIC module is in master mode and the slave device is holding the SDAn line at the low level because synchronization with the slave device has been lost due to the effects of noise, etc., the output of a stop condition is not possible. The facility for output of an extra cycle of the SCL clock can be used to output extra cycles of SCL one by one to make the slave device release the SDAn line from being held at the low level, thus recovering the bus from an unusable state. Release of the SDAn line by the slave device can be monitored by reading the SDAI bit in ICCR1. After confirming release of the SDAn line by the slave device, complete communications by reissuing the stop condition.
Use this facility with the MALE bit (master arbitration-lost detection disabled) in ICFER set to 0. If the MALE bit is set to 1 (master arbitration-lost detection enabled), arbitration is lost when the value of the SDAO bit in ICCR1 does not match the state of the SDAn line, so take care on this point.

[Output conditions for using the CLO bit in ICCR1]

- When the bus is free (BBSY flag in ICCR2 = 0) or in master mode (MST bit = 1 and BBSY flag = 1 in ICCR2)
- When the communication device does not hold the SCLn line low

Figure 25.39 shows the operation timing of the extra SCL clock cycle output function (CLO bit).

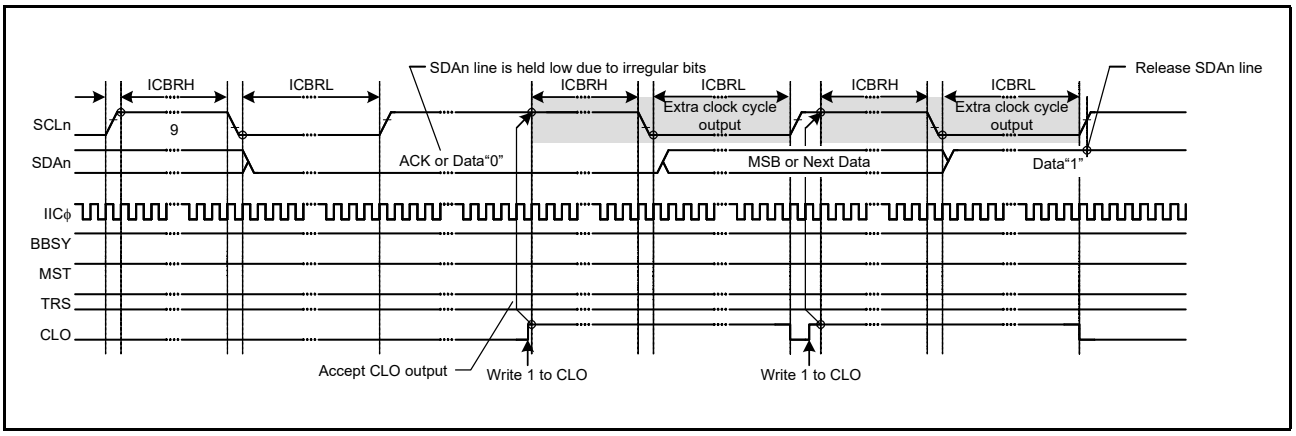


Figure 25.39 Extra SCL Clock Cycle Output Function (CLO Bit)

25.11.3 RIIC Reset and Internal Reset

The RIIC module incorporates a function for resetting itself. There are two types of reset. One is referred to as an RIIC reset; this initializes all registers including the BBSY flag in ICCR2. The other is referred to as an internal reset; this releases the RIIC from the slave-address matched state and initializes the internal counter while retaining other settings. After issuing a reset, be sure to set the IICRST bit in ICCR1 to 0.

Both types of reset are effective for release from bus-hung states since both restore the output state of the SCLn and SDAn pins to the high impedance state.

Do not issue a reset during slave operation because it may lead to a loss of synchronization between the master device clock and the slave device clock. Note that monitoring of the bus state, such as for the presence of a start condition, is not possible during an RIIC reset (ICE and IICRST bits = 01b in ICCR1).

For a detailed description of the RIIC and internal resets, refer to section 25.13, Resets and Register and Function States When Issuing Each Condition.

25.12 Interrupt Sources

The RIIC issues four types of interrupt request: transfer error or event generation (arbitration-lost, NACK detection, timeout detection, start condition detection, and stop condition detection), receive data full, transmit data empty, and transmit end.

Table 25.6 lists details of the several interrupt requests. The receive data full and transmit data empty are both capable of activating data transfer by the DMAC.

Table 25.6 Interrupt Sources

Symbol	Interrupt Source	Interrupt Flag	DMAC Activation	Interrupt Condition
EEI	Transfer error/ event generation	AL	Not possible	AL = 1 • ALIE = 1
		NACKF		NACKF = 1 • NAKIE = 1
		TMOF		TMOF = 1 • TMOIE = 1
		START		START = 1 • STIE = 1
		STOP		STOP = 1 • SPIE = 1
RXI*1	Receive data full	—	Possible	RDRF = 1 • RIE = 1
TXI*2	Transmit data empty	—	Possible	TDRE = 1 • TIE = 1
TEI*3	Transmit end	TEND	Not possible	TEND = 1 • TEIE = 1

Note: There is a delay time between the execution of a write instruction for a peripheral module by the CPU and actual writing to the module. Thus, when an interrupt flag has been cleared or masked, read the relevant flag again to check whether clearing or masking has been completed, and then return from interrupt handling. Returning from interrupt handling without checking that writing to the module has been completed creates a possibility of repeated processing of the same interrupt.

Note 1. The RDRF flag in ICSR2 (a condition for RXI) is automatically set to 0 when data are read from ICDRR.

Note 2. The TDRE flag in ICSR2 (a condition for TXI) is automatically set to 0 when data for transmission are written to ICDRT or a stop condition is detected (STOP flag = 1 in ICSR2).

Note 3. When using the TEI interrupt, clear the TEND flag in ICSR2 in the TEI interrupt handling.
Note that the TEND flag in ICSR2 is automatically set to 0 when data for transmission are written to ICDRT or a stop condition is detected (STOP flag = 1 in ICSR2).

Clear or mask the each flag during interrupt handling.

25.13 Resets and Register and Function States When Issuing Each Condition

The RIIC has reset, RIIC reset, and internal reset functions. Table 25.7 lists the resets and register and function states when issuing each condition.

Table 25.7 Resets and Register and Function States When Issuing Each Condition

		Chip Reset	RIIC Reset (ICE = 0, IICRST = 1)	Internal Reset (ICE = 1, IICRST = 1)	Start Condition/ Restart Condition Detection	Stop Condition Detection
ICCR1	ICE, IICRST	At a reset	Retained	Retained	Retained	Retained
	SCLO, SDAO		At a reset	At a reset		
	Others			Retained		
ICCR2	BBSY	At a reset	At a reset	Retained	Retained	Retained
	ST			At a reset	At a reset	Retained
	Others					At a reset
ICMR1	BC[2:0]	At a reset	At a reset	At a reset	At a reset	Retained
	Others			Retained	Retained	
ICMR2		At a reset	At a reset	Retained	Retained	Retained
ICMR3		At a reset	At a reset	Retained	Retained	Retained
ICFER		At a reset	At a reset	Retained	Retained	Retained
ICSER		At a reset	At a reset	Retained	Retained	Retained
ICIER		At a reset	At a reset	Retained	Retained	Retained
ICSR1		At a reset	At a reset	At a reset	Retained	At a reset
ICSR2	TDRE, TEND	At a reset	At a reset	At a reset	Retained	At a reset
	START				Retained	
	STOP				Retained	Retained
	Others				Retained	Retained
ICSARL0, ICSARL1, ICSARL2 ICSARU0, ICSARU1, ICSARU2		At a reset	At a reset	Retained	Retained	Retained
ICBRH, ICBRL		At a reset	At a reset	Retained	Retained	Retained
ICDRT		At a reset	At a reset	Retained	Retained	Retained
ICDRR		At a reset	At a reset	Retained	Retained	Retained
ICDRS		At a reset	At a reset	At a reset	Retained	Retained
Timeout function		At a reset	At a reset	Operation	Operation	Operation
Bus free time measurement		At a reset	At a reset	Operation	Operation	Operation

25.14 Event Link Output

The RIIC0 handles event output for the event link controller (ELC) corresponding to the following sources.

(1) Transfer error event

When a transfer error event occurs, the corresponding event signal can be output for another module via the ELC.

(2) Receive data full

When a receive data register becomes full, the corresponding event signal can be output for another module via the ELC.

(3) Transmit data empty

When a transmit data register becomes empty, the corresponding event signal can be output for another module via the ELC.

(4) Transmit end

On completion of transfer, the corresponding event signal can be output for another module via the ELC.

25.14.1 Interrupt Handling and Event Linking

The RIIC module produces four kinds of interrupt: transfer error or event generation (arbitration-lost detection, detection of NACK, detection of timeout, detection of a start condition, or detection of a stop condition), receive data full, transmit data empty, and transmit end interrupts. Each of these has an enable bit to control enabling and disabling of the interrupt signal. An interrupt request signal is output for the CPU when an interrupt source condition is satisfied while the setting of the corresponding enable bit is enabled.

The corresponding event link output signals are sent to other modules as event signals via the ELC when the interrupt source conditions are satisfied, regardless of the settings of the interrupt enable bits.

For details on interrupt sources, see Table 25.6.

25.15 Usage Notes

25.15.1 Setting Module Stop Function

Module-stop state can be entered or canceled using module stop control register B (MSTPCRB). The initial setting is for operation of the RIIC to be stopped. RIIC register access is enabled by clearing the module-stop state.
For details on module stop control registers B, refer to section 9, Low-Power Consumption Function.

26. Serial Peripheral Interface (RSPIa)

26.1 Overview

This LSI includes two channels of serial peripheral interface (RSPI).
The RSPI channels are capable of high-speed, full-duplex synchronous serial communications with multiple processors and peripheral devices.
Table 26.1 lists the specifications of the RSPI, and Figure 26.1 shows a block diagram of the RSPI.
In this section, a lower-case letter m in RSPI command register m (SPCMDm) indicates a value from 0 to 7.

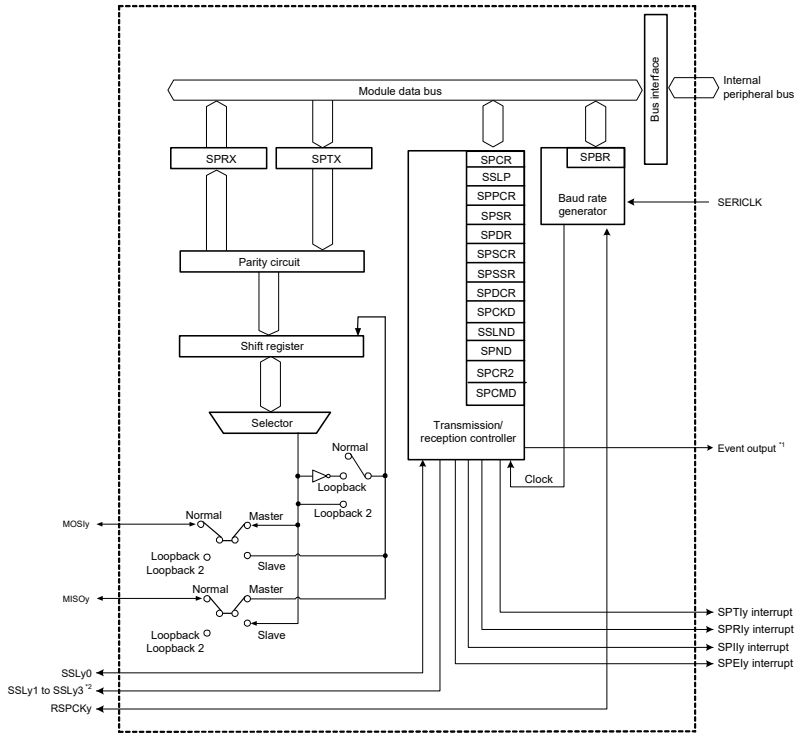
Table 26.1 RSPI Specifications (1 / 2)

Item	Description
Number of channels	Two channels
RSPI transfer functions	• Use of MOSI (master out/slave in), MISO (master in/slave out), SSL (slave select), and RSPCK (RSPI clock) signals allows serial communications through SPI operation (four-wire method) or clock synchronous operation (three-wire method). • Transmit-only operation is available. • Communication mode: Full-duplex or transmit-only can be selected. • Switching of the polarity of RSPCK • Switching of the phase of RSPCK
Data format	• MSB first/LSB first selectable • Transfer bit length is selectable as 8, 9, 10, 11, 12, 13, 14, 15, 16, 20, 24, or 32 bits. • 128-bit transmission/reception buffers • Up to four frames can be transferred in one round of transmission/reception (each frame consisting of up to 32 bits).
Bit rate	• In master mode, the on-chip baud rate generator generates RSPCK by frequency-dividing SERICLK (the division ratio ranges from divided by 4 to divided by 4096). • In slave mode, the minimum SERICLK clock divided by 8 can be input as RSPCK (the maximum frequency of RSPCK is that of SERICLK divided by 8). Width at high level: 4 cycles of SERICLK; width at low level: 4 cycles of SERICLK
Buffer configuration	• Double buffer configuration for the transmission/reception buffers • 128 bits for the transmission/reception buffers
Error detection	• Mode fault error detection • Overrun error detection*1 • Parity error detection
SSL control function	• Four SSL pins for channel 0 (SSL00 to SSL03) and two SSL pins for channel 1 (SSL10, SSL11) • In single-master mode, SSL00 to SSL03, SSL10, and SSL11 are output. • In multi-master mode: SSL00 and SSL10 pins for input, and SSL01 to SSL03 and SSL11 pins for either output or unused. • In slave mode: SSL00 and SSL10 pins for input, and SSL01 to SSL03 and SSL11 pins for unused. • 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 stop 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 eight 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, MSB/LSB first, burst, RSPCK delay, SSL negation delay, and next-access delay • A transfer can be initiated by writing to the transmission buffer. • MOSI signal value specifiable in SSL negation • RSPCK auto-stop function
Interrupt sources	• Interrupt sources Reception buffer full interrupt Transmission buffer empty interrupt RSPI error interrupt (mode fault, overrun, parity error) RSPI idle interrupt (RSPI idle)

Table 26.1 RSPI Specifications (2 / 2)

Item	Description
Event link function*2 (output)	• The following events can be output to the event link controller. Reception buffer full signal Transmission buffer empty signal Mode fault, overrun, or parity error signal RSPI idle signal Transmission-completed signal
Others	• Function for switching between CMOS output and open-drain output • Function for initializing the RSPI • Loopback mode
Low-power consumption function	Module-stop state can be set.

Note 1. In master reception and when the RSPCK auto-stop function is enabled, an overrun error does not occur because the transfer clock is stopped at the timing of overrun error detection.
Note 2. Only for channel 0 (RSPI0)



SPCR:	RSPI control register	SPDCR:	RSPI data control register
SPCR2:	RSPI control register 2	SPCKD:	RSPI clock delay register
SSLP:	RSPI slave select polarity register	SSLND:	RSPI slave select negation delay register
SPPCR:	RSPI pin control register	SPND:	RSPI next-access delay register
SPSR:	RSPI status register	SPCMD:	RSPI command registers 0 to 7 (eight registers)
SPDR:	RSPI data register		
SPSCR:	RSPI sequence control register	SPBR:	RSPI bit rate register
SPSSR:	RSPI sequence status register	SPTX:	RSPI transmission buffer
		SPRX:	RSPI reception buffer

Note 1. Only for channel 0.
Note 2. The number of pins differs for each channel (SSL01 to SSL03 for channel 0; SSL11 for channel 1).
y = 0, 1 (for all channels)

Figure 26.1 RSPI Block Diagram

Table 26.2 lists the I/O pins used in the RSPI.
The RSPI automatically switches the I/O direction of the SSLy0 pin. SSLy0 is set as an output when the RSPI is a single master and as an input when the RSPI is a multi-master or a slave. Pins RSPCKy, MOSIy, and MISOy are automatically set as inputs or outputs according to the setting of master or slave and the level input on the SSLy0 pin.
Refer to section 26.3.2, Controlling RSPI Pins.

Table 26.2 RSPI Pin Configuration

Channel	Pin Name	I/O	Function
RSPI0	RSPCK0	I/O	Clock I/O
	MOSI0	I/O	Master transmit data I/O
	MISO0	I/O	Slave transmit data I/O
	SSL00	I/O	Slave selection signal I/O
	SSL01	Output	Slave selection signal output
	SSL02	Output	Slave selection signal output
	SSL03	Output	Slave selection signal output
RSPI1	RSPCK1	I/O	Clock I/O
	MOSI1	I/O	Master transmit data I/O
	MISO1	I/O	Slave transmit data I/O
	SSL10	I/O	Slave selection signal I/O
	SSL11	Output	Slave selection signal output

Note: The number of SSL pins differs from channel to channel. The indices to identify the channels are 0 and 1.

26.2 Register Descriptions

26.2.1 RSPI Control Register (SPCR)

The SPCR register controls the settings for operation of the RSPI.
The SPCR.MSTR, SPCR.MODFEN, and SPCR.TXMD bits should be set while the setting of the SPCR.SPE bit is 0. Writing to these bits is prohibited while the setting of the SPCR.SPE bit is 1.

Address(es): RSPI0.SPCR A006 8000h, RSPI1.SPCR A006 8400h

	b7	b6	b5	b4	b3	b2	b1	b0
	SPRIE	SPE	SPTIE	SPEIE	MSTR	MODFEN	TXMD	SPMS
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SPMS	RSPI Mode Select	0: SPI operation (four-wire method) 1: Clock synchronous operation (three-wire method)	R/W
b1	TXMD	Communications Operating Mode Select	0: Full-duplex synchronous serial communications 1: Serial communications consisting of only transmit operations	R/W
b2	MODFEN	Mode Fault Error Detection Enable	0: Disables the detection of mode fault error 1: Enables the detection of mode fault error	R/W
b3	MSTR	Master/Slave Mode Select	0: Slave mode 1: Master mode	R/W
b4	SPEIE	Error Interrupt Enable	0: Disables the generation of error interrupt requests 1: Enables the generation of error interrupt requests	R/W
b5	SPTIE	Transmission Buffer Empty Interrupt Enable	0: Disables the generation of transmission buffer empty interrupt requests 1: Enables the generation of transmission buffer empty interrupt requests	R/W
b6	SPE	RSPI Function Enable	0: Disables the RSPI function 1: Enables the RSPI function	R/W
b7	SPRIE	Reception Buffer Full Interrupt Enable	0: Disables the generation of reception buffer full interrupt requests 1: Enables the generation of reception buffer full interrupt requests	R/W

SPMS Bit (RSPI Mode Select)

The SPMS bit selects SPI operation (four-wire method) or clock synchronous operation (three-wire method).
The SSLy0 to SSLy3 pins are not used in clock synchronous operation. The three pins RSPCKy, MOSIy, and MISOy handle communications. If clock synchronous operation is to proceed in master mode (SPCR.MSTR = 1), the SPCMDm.CPHA bit can be set to either 0 or 1. Set the CPHA bit to 1 if clock synchronous operation is to proceed in slave mode (SPCR.MSTR = 0). Do not set the CPHA bit to 0 when clock synchronous operation is to proceed in slave mode (SPCR.MSTR = 0). (y = 0, 1)

TXMD Bit (Communications Operating Mode Select)

The TXMD bit selects the operation for full-duplex synchronous serial communications or the operation for transmission only.
When performing communications with the TXMD bit set to 1, the RSPI performs only transmit operations and not receive operations (refer to section 26.3.6, Communications Operating Mode).
When the TXMD bit is set to 1, reception buffer full interrupt requests cannot be used.

MODFEN Bit (Mode Fault Error Detection Enable)

The MODFEN bit enables or disables the detection of mode fault error (refer to section 26.3.8, Error Detection). In addition, the RSPI determines the I/O direction of the SSLy0 to SSLy3 (y = 0, 1) pins based on combinations of the MODFEN and MSTR bits (refer to section 26.3.2, Controlling RSPI Pins).

MSTR Bit (RSPI Master/Slave Mode Select)

The MSTR bit selects master/slave mode of the RSPI. According to MSTR bit settings, the RSPI determines the direction of pins RSPCKy, MOSIy, and MISOy, and SSLy0 to SSLy3 (y = 0, 1).

SPEIE Bit (RSPI Error Interrupt Enable)

The SPEIE bit enables or disables the generation of RSPI error interrupt requests when the RSPI detects a mode fault error and sets the SPSR.MODF flag to 1, when the RSPI detects an overrun error and sets the SPSR.OVRF flag to 1, or when the RSPI detects a parity error and sets the SPSR.PERF flag to 1 (refer to section 26.3.8, Error Detection).

SPTIE Bit (Transmission Buffer Empty Interrupt Enable)

The SPTIE bit enables or disables a transmission buffer empty interrupt request generated when the RSPI detects that the transmission buffer is empty.
If the RSPI function is disabled (i.e. the SPE bit is set to 0), the transmission buffer being empty is detected. If the SPTIE bit is set to 1 at this time, a transmission buffer empty interrupt is generated.
Note that a transfer buffer empty interrupt request is also generated when the SPTIE and SPE bits are set to 1 at the same time at the start of transmission.

SPE Bit (RSPI Function Enable)

The SPE bit enables or disables the RSPI function.
When the SPSR.MODF bit is 1, the SPE bit cannot be set to 1. For details, refer to section 26.3.8, Error Detection.
Setting the SPE bit to 0 disables the RSPI function, and initializes a part of the module function. For details, refer to section 26.3.9, Initializing RSPI. Furthermore, when the setting of the SPTIE bit is 1 (enabling the generation of transmission buffer empty interrupts), a transmission buffer empty interrupt request is generated by the state of the SPE bit changing from 0 to 1 or from 1 to 0.

SPRIE Bit (RSPI Reception Buffer Full Interrupt Enable)

The SPRIE bit enables or disables RSPI reception buffer full interrupt requests when the RSPI has detected the reception buffer being full after the completion of a serial transfer.

26.2.2 RSPI Slave Select Polarity Register (SSLP)

The SSLP register controls the active sense of the RSPIy (y = 0, 1) slave select signals.
The SSLP register should be set while the setting of the SPCR.SPE bit is 0. Writing to this register is prohibited while the setting of the SPCR.SPE bit is 1.

Address(es): RSPI0.SSLP A006 8001h, RSPI1.SSLP A006 8401h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	SSL3P	SSL2P	SSL1P	SSL0P
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SSL0P	SSL0 Signal Polarity Setting	0: SSLy0 signal is active low 1: SSLy0 signal is active high	R/W
b1	SSL1P	SSL1 Signal Polarity Setting	0: SSLy1 signal is active low 1: SSLy1 signal is active high	R/W
b2	SSL2P	SSL2 Signal Polarity Setting*1	0: SSLy2 signal is active low 1: SSLy2 signal is active high	R/W
b3	SSL3P	SSL3 Signal Polarity Setting*1	0: SSLy3 signal is active low 1: SSLy3 signal is active high	R/W
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Bits for channels 1 are reserved. These bits are read as 0. The write value should be 0.

26.2.3 RSPI Pin Control Register (SPPCR)

The SPPCR register controls the output setting for RSPI output pins.

The SPPCR register should be set while the setting of the SPCR.SPE bit is 0. Writing to this register is prohibited while the setting of the SPCR.SPE bit is 1.

Address(es): RSPI0.SPPCR A006 8002h, RSPI1.SPPCR A006 8402h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	MOIFE	MOIFV	—	SPOM	SPLP2	SPLP
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SPLP	RSPI Loopback	0: Normal mode 1: Loopback mode (data is inverted for transmission)	R/W
b1	SPLP2	RSPI Loopback 2	0: Normal mode 1: Loopback mode (data is not inverted for transmission)	R/W
b2	SPOM	Output Pin Mode	0: CMOS output 1: Open-drain output	R/W
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b4	MOIFV	MOSI Idle Fixed Value	0: The level output on the MOSIy pin during MOSI idling corresponds to low. 1: The level output on the MOSIy pin during MOSI idling corresponds to high.	R/W
b5	MOIFE	MOSI Idle Value Fixing Enable	0: MOSI output value equals final data from previous transfer 1: MOSI output value equals the value set in the MOIFV bit	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

SPLP Bit (RSPI Loopback)

The SPLP bit selects the mode of the RSPI pins.

When the SPLP bit is set to 1, the RSPI shuts off the path between the MISOy pin and the shift register if the SPCR.MSTR bit is 1, and between the MOSIy pin and the shift register if the SPCR.MSTR bit is 0, and connects (inverts) the input path and output path for the shift register (loopback mode).

(y = 0, 1)

SPLP2 Bit (RSPI Loopback 2)

The SPLP2 bit selects the mode of the RSPI pins.

When the SPLP2 bit is set to 1, the RSPI shuts off the path between the MISOy pin and the shift register if the SPCR.MSTR bit is 1, and between the MOSIy pin and the shift register if the SPCR.MSTR bit is 0, and connects the input path and output path for the shift register (loopback mode).

(y = 0, 1)

SPOM Bit (Output Pin Mode)

The SPOM bit is used to set the RSPI output pins to CMOS output or open-drain output. For details, see section 26.3.2, Controlling RSPI Pins.

MOIFV Bit (MOSI Idle Fixed Value)

If the MOIFE bit is 1 in master mode, the MOIFV bit determines the MOSIy pin output value as low or high during the SSL negation period (including the SSL retention period during a burst transfer).

MOIFE Bit (MOSI Idle Value Fixing Enable)

The MOIFE bit selects whether the MOSIy output level is to be fixed over the SSL negation period (including the SSL retention period during a burst transfer) when the RSPI is in master mode. When the MOIFE bit is 0, the RSPI continues to output the value of the last bit from the previous serial transfer during the SSL negation period to the MOSIy pin. When the MOIFE bit is 1, the RSPI outputs a fixed low or high level on the MOSIy pin according to the setting of the MOIFV bit.
(y = 0, 1)

26.2.4 RSPI Status Register (SPSR)

The SPSR register indicates the state of RSPI transfer.

Address(es): RSPI0.SPSR A006 8003h, RSPI1.SPSR A006 8403h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	PERF	MODF	IDLNF	OVRF
Value after reset:	x	0	x	0	0	0	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	R/W
b0	OVRF	Overrun Error Flag	0: No overrun error occurs 1: An overrun error occurs	R/(W) *1
b1	IDLNF	RSPI Idle Flag	0: RSPI is in the idle state 1: RSPI is in the transfer state	R
b2	MODF	Mode Fault Error Flag	0: No mode fault error occurs 1: A mode fault error occurs	R/(W) *1
b3	PERF	Parity Error Flag	0: No parity error occurs 1: A parity error occurs	R/(W) *1
b4	—	Reserved	This bit is read as 0. The write value should be 0.	R/(W)
b5	—	Reserved	The read value is undefined. The write value should be 1.	R/(W)
b6	—	Reserved	This bit is read as 0. The write value should be 0.	R/(W)
b7	—	Reserved	The read value is undefined. The write value should be 1.	R/(W)

Note 1. Only 0 can be written to clear the flag after reading 1.

OVRF Flag (Overrun Error Flag)

The OVRF flag indicates the occurrence of an overrun error. In master mode (when the SPCR.MSTR bit is 1) and when the RSPCK clock auto-stop function is enabled (the SPCR1.SCKASE bit is 1), an overrun error does not occur; accordingly this flag does not become 1. For details, see section 26.3.8.1, Overrun Error.

[Setting condition]

- When the next serial transfer ends while the SPCR.TXMD bit is 0 and the reception buffer is full.

[Clearing condition]

- When SPSR is read while the OVRF flag is 1, and then writes the value 0 to the OVRF flag.

IDLNF Flag (RSPI Idle Flag)

The IDLNF flag indicates the transfer status of the RSPI.

[Setting condition]

Master mode

- Condition 1 and condition 2 are not satisfied in master mode under the [Clearing condition] below.

Slave mode

- The SPCR.SPE bit is 1 (RSPI function is enabled)

[Clearing condition]

Master mode

The following 1 is satisfied (condition 1) or all of the following 2 to 4 are satisfied (condition 2).

1. The SPCR.SPE bit is 0 (RSPI is initialized)
2. The transmission buffer (SPTX) is empty (data for the next transfer is not set)
3. The SPSSR.SPCP[2:0] bits are 000b (beginning of sequence control)
4. The RSPI internal sequencer has entered the idle state (status in which operations up to the next-access delay have finished)

Slave mode

- The SPCR.SPE bit is 0 (RSPI is initialized)

MODF Flag (Mode Fault Error Flag)

Indicates the occurrence of a mode fault error.

[Setting condition]

Multi-master mode

- When the input level of the SSLyi pin changes to the active level while the SPCR.MSTR bit is 1 (master mode) and the SPCR.MODFEN bit is 1 (mode fault error detection is enabled), the RSPI detects a mode fault error

Slave mode

- When the SSLyi pin is negated before the RSPCK cycle necessary for data transfer ends while the SPCR.MSTR bit is 0 (slave mode) and the SPCR.MODFEN bit is 1 (mode fault error detection is enabled), the RSPI detects a mode fault error

The active level of the SSLyi signal is determined by the SSLP.SSLiP bit (SSLi signal polarity setting bit).

(y = 0, 1 (for all channels); i = 0 to 3)

[Clearing condition]

- When SPSR is read while the MODF flag is 1, and then writes the value 0 to the MODF flag

PERF Flag (Parity Error Flag)

Indicates the occurrence of a parity error.

[Setting condition]

- When a serial transfer ends while the SPCR.TXMD bit is 0 and the SPCR2.SPPE bit is 1, the RSPI detects a parity error

[Clearing condition]

- When SPSR is read while the PERF flag is 1, and then writes the value 0 to the PERF flag

26.2.5 RSPI Data Register (SPDR)

SPDR is the interface with the buffers that hold data for transmission and reception by the RSPI.
When accessing in longwords (the SPLW bit is 1), access SPDR.
When accessing in words (the SPLW bit is 0), access the higher-order 16 bits (H) of SPDR.

Address(es): RSPI0.SPDR A006 8004h, RSPI1.SPDR A006 8404h

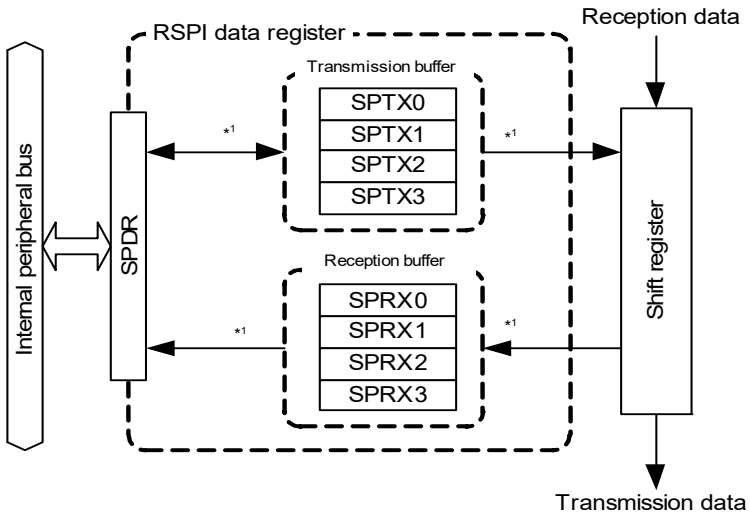
	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Address(es): RSPI0.SPDR A006 8004h, RSPI1.SPDR A006 8404h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	H															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

The transmission buffer (SPTX) and reception buffer (SPRX) are independent but are both mapped to SPDR. Figure 26.2 shows the configuration of SPDR.



Note 1. The destination buffer and stage for access is automatically switched by hardware.

Figure 26.2 Configuration of SPDR

The transmit and reception buffers each have four stages. The number of stages to be used is selectable by the number of frames specification bits in the RSPI data control register (SPDCR.SPFC[1:0]). The eight stages of the buffer are all mapped to the single address of SPDR.
Data written to SPDR are written to a transmission-buffer stage (SPTXn) (n = 0 and 1) and then transmitted from the buffer. The reception buffer holds received data on completion of reception. The reception buffer is not updated if an overrun is generated.
Furthermore, if the data length is other than 32 bits, bits not referred to in SPTXn (n = 0 and 1) are stored in the corresponding bits in SPRXn (n = 0 and 1). For example, if the data length is 9 bits, received data are stored in the SPRXn[8:0] bits and the SPTXn[31:9] bits are stored in the SPRXn[31:9] bits.

(1) Bus Interface

SPDR is the interface with 32-bit wide transmit and reception buffers, each of which has four stages, for a total of 32 bytes. In other words, the 32 bytes are mapped to the 4-byte address space for SPDR. Furthermore, the unit of access for SPDR is selected by the RSPI longword access/word access specification bit in the RSPI data control register (SPDCR.SPLW).
Data for transmission should be flush with the LSB end of the register. Received data are stored flush with the LSB end.
Operations involved in writing to and reading from SPDR are described below.

(a) Writing

Data written to SPDR are written to a transmission buffer (SPTXn). This is not influenced by the value of the SPDCR.SPRDTD bit unlike when reading from SPDR.
The transmission buffer includes a transmission buffer write pointer which is automatically updated to indicate the next stage each time data are written to SPDR.
Figure 26.3 shows the configuration of the bus interface with the transmission buffer in the case of writing to SPDR.

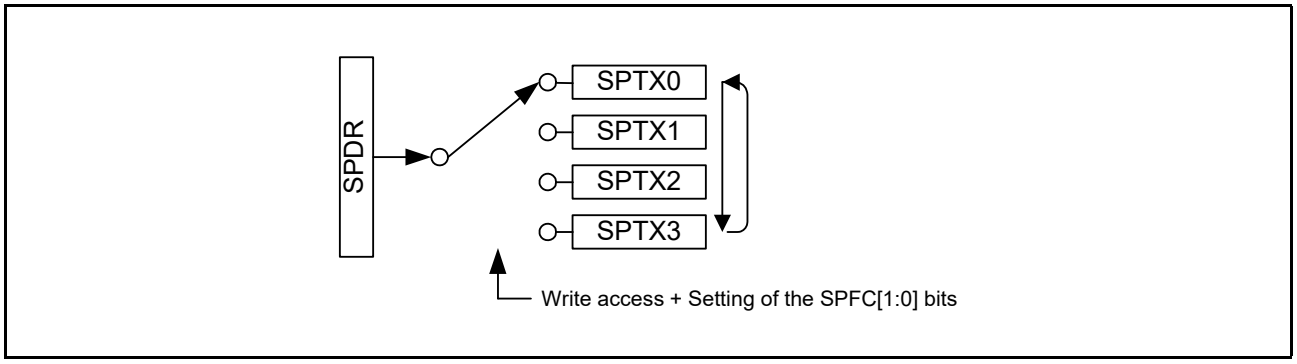


Figure 26.3 Configuration of SPDR (Writing)

The sequence for switching the transmission buffer write pointer differs with the setting of the number of frames specification bits in the RSPI data control register (SPDCR.SPFC[1:0]).

- Settings of the SPFC[1:0] bits and sequence of switching the pointer among SPTX0 to SPTX3.
  When the SPFC[1:0] bits are 00b: SPTX0 → SPTX0 → SPTX0 → …
  When the SPFC[1:0] bits are 01b: SPTX0 → SPTX1 → SPTX0 → SPTX1 → …
  When the SPFC[1:0] bits are 10b: SPTX0 → SPTX1 → SPTX2 → SPTX0 → SPTX1 → …
  When the SPFC[1:0] bits are 11b: SPTX0 → SPTX1 → SPTX2 → SPTX3 → SPTX0 → SPTX1 → …

When 1 is written to the RSPI function enable bit in the RSPI control register (SPCR.SPE) while the bit's current value is 0, SPTX0 will be the destination the next time writing proceeds.
When writing to the transmission buffer (SPTXn) after generation of the transmission buffer empty interrupt, write the number of frames set by the number of frames specification bits (SPFC[1:0]) in the RSPI data control register (SPDCR). Even if the number of frames is written to the transmission buffer (SPTXn), the value of the buffer is not updated after completion of the writing and before generation of the next transmission buffer empty interrupt.

(b) Reading

SPDR can be read to read the value of a reception buffer (SPRXn) or a transmission buffer (SPTXn). The setting of the RSPI receive/transmit data selection bit in the RSPI data control register (SPDCR.SPRDTD) selects whether reading is of the receive or transmission buffer.
The sequence of reading the SPDR register is controlled by independent pointers, reception buffer read pointer and transmission buffer read pointer.
Figure 26.4 shows the configuration of the bus interface with the receive and transmission buffers in the case of reading from SPDR.

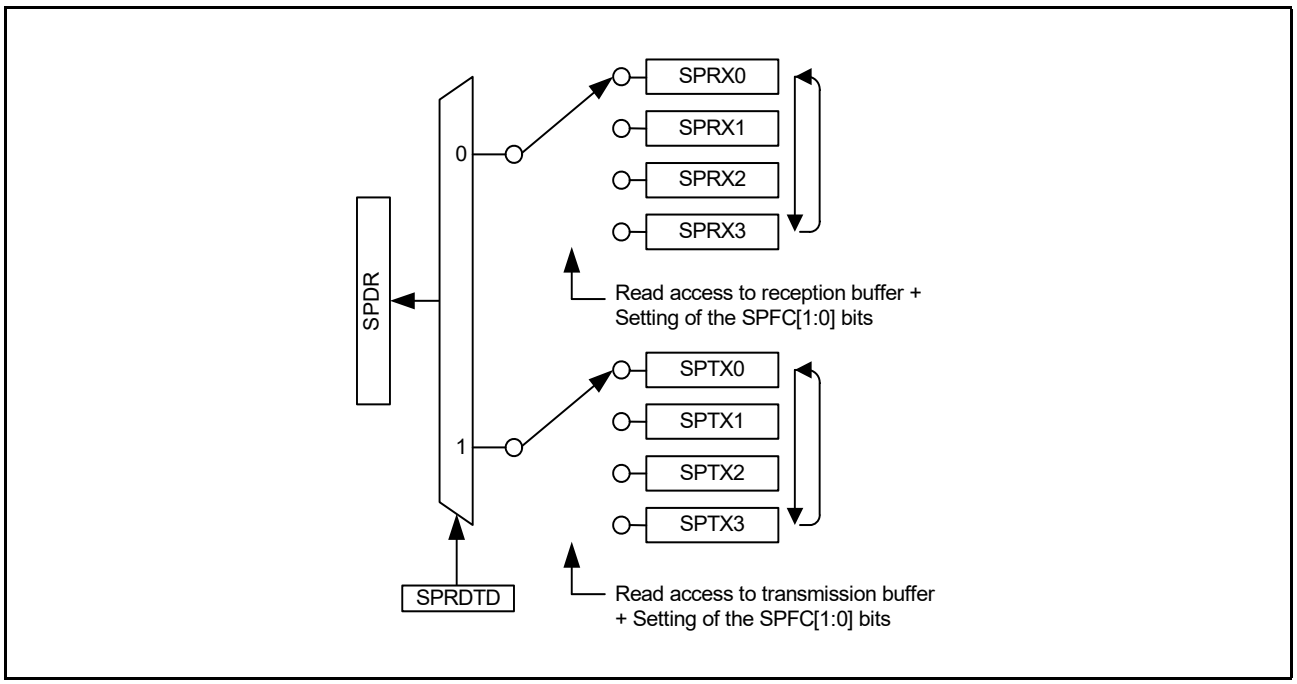


Figure 26.4 Configuration of SPDR (Reading)

Reading the reception buffer switches the reception buffer read pointer to the next buffer automatically.
The sequence of switching the reception buffer read pointer is the same as that for the transmission buffer write pointer. However, when 1 is written to the RSRI function enable bit in the RSPI control register (SPCR.SPE) while the bit's current value is 1, SPRX0 will be indicated by the buffer read pointer the next time reading proceeds.
The transmission buffer read pointer is updated when writing to SPDR, and not updated when reading from the transmission buffer. When reading from the transmission buffer, the value most recently written to SPDR is read. However, after generation of the transmission buffer empty interrupt, the values read from the transmission buffer are all 0 in the interval after completion of writing the number of frames of data specified in the number of frames specification bits (SPDCR.SPFC[1:0]) and before generation of the next buffer empty interrupt.

26.2.6 RSPI Sequence Control Register (SPSCR)

SPSCR sets the sequence length when the RSPI operates in master mode. When changing the SPSCR.SPSLN[2:0] bits while both the SPCR.MSTR and SPCR.SPE bits are 1, the bits should be changed while the SPSR.IDLNF flag is 0.

Address(es): RSPI0.SPSCR A006 8008h, RSPI1.SPSCR A006 8408h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	SPSLN[2:0]		
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	SPSLN[2:0]	RSPI Sequence Length Specification	b2 b0 Sequence Length — Referenced SPCMD0 to SPCMD7 (No.) 0 0 0: 1 — 0→0→… 0 0 1: 2 — 0→1→0→… 0 1 0: 3 — 0→1→2→0→… 0 1 1: 4 — 0→1→2→3→0→… 1 0 0: 5 — 0→1→2→3→4→0→… 1 0 1: 6 — 0→1→2→3→4→5→0→… 1 1 0: 7 — 0→1→2→3→4→5→6→0→… 1 1 1: 8 — 0→1→2→3→4→5→6→7→0→… The order in which the SPCMD0 to SPCMD07 registers are to be referenced is changed in accordance with the sequence length that is set in these bits. The relationship among the setting of these bits, sequence length, and SPCMD0 to SPCMD7 registers referenced by the RSPI is shown above. However, the RSPI in slave mode always references SPCMD0.	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

SPSLN[2:0] Bits (RSPI Sequence Length Specification)

The SPSLN[2:0] bits specify a sequence length when the RSPI in master mode performs sequential operations. The RSPI in master mode changes SPCMD0 to SPCMD7 registers to be referenced and the order in which they are referenced according to the sequence length that is set in the SPSLN[2:0] bits.

In slave mode, SPCMD0 is always referred to.

26.2.7 RSPI Sequence Status Register (SPSSR)

SPSSR indicates the sequence control status when the RSPI operates in master mode.
Any writing to SPSSR is ignored.

Address(es): RSPI0.SPSSR A006 8009h, RSPI1.SPSSR A006 8409h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	SPECM[2:0]			—	SPCP[2:0]		
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	SPCP[2:0]	RSPI Command Pointer	b2 b0 0 0 0: SPCMD0 0 0 1: SPCMD1 0 1 0: SPCMD2 0 1 1: SPCMD3 1 0 0: SPCMD4 1 0 1: SPCMD5 1 1 0: SPCMD6 1 1 1: SPCMD7	R
b3	—	Reserved	This bit is read as 0.	R
b6 to b4	SPECM[2:0]	RSPI Error Command	b6 b4 0 0 0: SPCMD0 0 0 1: SPCMD1 0 1 0: SPCMD2 0 1 1: SPCMD3 1 0 0: SPCMD4 1 0 1: SPCMD5 1 1 0: SPCMD6 1 1 1: SPCMD7	R
b7	—	Reserved	This bit is read as 0.	R

SPCP[2:0] Bits (RSPI Command Pointer)

The SPCP[2:0] bits indicate SPCMDm that is currently pointed to by the pointer during sequence control by the RSPI. For the RSPI's sequence control, refer to section 26.3.10.1, Master Mode Operation.

SPECM[2:0] Bits (RSPI Error Command)

The SPECM[2:0] bits indicate SPCMDm that is specified by the SPCP[2:0] bits when an error is detected during sequence control by the RSPI. The RSPI updates the SPECM[2:0] bits only when an error is detected. If both the SPSR.OVRF and SPSR.MODF bits are 0 and there is no error, the values of the SPECM[2:0] bits have no meaning. For the RSPI's error detection function, refer to section 26.3.8, Error Detection. For the RSPI's sequence control, refer to section 26.3.10.1, Master Mode Operation.

26.2.8 RSPI Bit Rate Register (SPBR)

SPBR controls the bit rate settings in master mode. Writing to SPBR is prohibited while both the SPCR.MSTR and SPCR.SPE bits are 1.

Address(es): RSPI0.SPBR A006 800Ah, RSPI1.SPBR A006 840Ah

When the RSPI is used in slave mode, the bit rate depends on the bit rate of the input clock (bit rate satisfying the electrical characteristics should be used) regardless of the settings of SPBR and the SPCMDm.BRDV[1:0] bits (bit rate division setting bits).
The bit rate is determined by combinations of the SPBR setting and the SPCMDm.BRDV[1:0] bit setting. 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 a BRDV[1:0] bit setting (0, 1, 2, 3) (m = 0 to 7). However, setting n = 0 (SPR[7:0] = 0) and N = 0 (BRDV[1:0] = 0) is prohibited.

$$\text{Bit rate} = \frac{f\,(\text{SERICLK})}{2 \times (n + 1)\, 2^{N}}$$

Table 26.3 lists examples of the relationship among the SPBR settings, the BRDV[1:0] settings, and bit rates.

Table 26.3 Relationship among SPBR Settings, BRDV[1:0] Settings, and Bit Rates

			Bit Rate	
SPBR (n)	**BRDV[1:0] Bits (N)**	**Division Ratio**	**SERICLK = 120 MHz**	**SERICLK = 150 MHz**
0	0	2	Setting prohibited	Setting prohibited
1	0	4	30.0 Mbps	37.5 Mbps
2	0	6	20.0 Mbps	25.0 Mbps
3	0	8	15.0 Mbps	18.8 Mbps
4	0	10	12.0 Mbps	15.0 Mbps
5	0	12	10.0 Mbps	12.5 Mbps
5	1	24	5.00 Mbps	6.25 Mbps
5	2	48	2.50 Mbps	3.13 Mbps
5	3	96	1.25 Mbps	1.56 Mbps
255	3	4096	29.3 kbps	36.6 kbps

Note: The settings must not exceed the range of electrical characteristics.

26.2.9 RSPI Data Control Register (SPDCR)

SPDCR controls data in the SPDR register.

Up to four frames can be transmitted or received in one round of transmission or reception activation (m = 0 to 7). The amount of data in each transfer is controlled by the combination of the SPCMDm.SPB[3:0] bits, the SPSCR.SPSLN[2:0] bits, and the SPDCR.SPFC[1:0] bits.

When changing the SPDCR.SPFC[1:0] bits while the SPCR.SPE bit is 1, the bits should be changed while the SPSR.IDLNF flag is 0.

Address(es): RSPI0.SPDCR A006 800Bh, RSPI1.SPDCR A006 840Bh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	SPLW	SPRDTD	—	—	SPFC[1:0]	
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	SPFC[1:0]	Number of Frames Specification	b1 b0 0 0: 1 frame 0 1: 2 frames 1 0: 3 frames 1 1: 4 frames	R/W
b3, b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b4	SPRDTD	RSPI Receive/Transmit Data Selection	0: SPDR values are read from the reception buffer 1: SPDR values are read from the transmission buffer (but only if the transmission buffer is empty)	R/W
b5	SPLW	RSPI Longword Access/ Word Access Specification	0: SPDR is accessed in words 1: SPDR is accessed in longwords	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

SPFC[1:0] Bits (Number of Frames Specification)

The SPFC[1:0] bits specify the number of frames that can be stored in SPDR (per transfer activation). Up to four frames can be transmitted or received in one round of transmission or reception, and the amount of data is determined by the combination of the SPSCR.SPSLN[2:0] bits, and the SPDCR.SPFC[1:0] bits. Furthermore, the setting of the SPFC[1:0] bits adjusts the number of frames for generation of RSPI reception buffer full interrupt, and start of transmission or generation of transmission buffer empty interrupts. Table 26.4 lists the frame configurations that can be stored in SPDR and examples of combinations of settings for transmission and reception. Combinations of settings other than those shown in the examples should not be made.

Table 26.4 Settable Combinations of SPSLN[2:0] Bits and SPFC[1:0] Bits

Setting	SPSLN[2:0]	SPFC[1:0]	Number of Frames in a Single Sequence	Number of Frames at which Reception Buffer Full Interrupt Occurs or Transmission Buffer Holding Data is Recognized
1-1	000b	00b	1	1
1-2	000b	01b	2	2
1-3	000b	10b	3	3
1-4	000b	11b	4	4
2-1	001b	01b	2	2
2-2	001b	11b	4	4
3	010b	10b	3	3
4	011b	11b	4	4
5	100b	00b	5	1
6	101b	00b	6	1
7	110b	00b	7	1
8	111b	00b	8	1

SPRDTD Bit (RSPI Receive/Transmit Data Selection)

The SPRDTD bit selects whether the SPDR reads values from the reception buffer or from the transmission buffer.
If reading is from the transmission buffer, the value written to SPDR register immediately beforehand is read.
When reading the transmission buffer, do so before writing of the number of frames set in the SPFC[1:0] bits is finished and after generation of the transmission buffer empty interrupt.
For details, refer to section 26.2.5, RSPI Data Register (SPDR).

SPLW Bit (RSPI Longword Access/Word Access Specification)

The SPLW bit specifies the access width for SPDR. Access to SPDR is in words when the SPLW bit is 0 and in longwords when the SPLW bit is 1.
Also, when the SPLW bit is 0, set the SPCMDm.SPB[3:0] bits (RSPI data length setting bits) to 8 to 16 bits. Setting these bits to 20, 24, or 32 bits is prohibited.

26.2.10 RSPI Clock Delay Register (SPCKD)

SPCKD sets a period from the beginning of SSLyi signal assertion to RSPCK oscillation (RSPCK delay) when the SPCMDm.SCKDEN bit is 1. While both the SPCR.MSTR and SPCR.SPE bits are 1, do not change the value of the SPCKD register (m = 0 to 7; y = 0, 1 (for all channels); i = 0 to 3).

Address(es): RSPI0.SPCKD A006 800Ch, RSPI1.SPCKD A006 840Ch

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	SCKDL[2:0]		
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	SCKDL[2:0]	RSPCK Delay Setting	b2 b1 b0 0 0 0: 1 RSPCK 0 0 1: 2 RSPCK 0 1 0: 3 RSPCK 0 1 1: 4 RSPCK 1 0 0: 5 RSPCK 1 0 1: 6 RSPCK 1 1 0: 7 RSPCK 1 1 1: 8 RSPCK	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

SCKDL[2:0] Bits (RSPCK Delay Setting)

The SCKDL[2:0] bits set an RSPCK delay value when the SPCMDm.SCKDEN bit is 1.
When using the RSPI in slave mode, set the SCKDL[2:0] bits to 000b.

26.2.11 RSPI 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 SSLyi signal during a serial transfer by the RSPI in master mode. Writing to SSLND is prohibited while both the SPCR.MSTR and SPCR.SPE bits are 1 (y = 0, 1 (for all channels); i = 0 to 3).

Address(es): RSPI0.SSLND A006 800Dh, RSPI1.SSLND A006 840Dh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	SLNDL[2:0]		
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	SLNDL[2:0]	SSL Negation Delay Setting	b2 b0 0 0 0: 1 RSPCK 0 0 1: 2 RSPCK 0 1 0: 3 RSPCK 0 1 1: 4 RSPCK 1 0 0: 5 RSPCK 1 0 1: 6 RSPCK 1 1 0: 7 RSPCK 1 1 1: 8 RSPCK	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

SLNDL[2:0] Bits (SSL Negation Delay Setting)

The SLNDL[2:0] bits set an SSL negation delay value when the RSPI is in master mode.
When using the RSPI in slave mode, set the SLNDL[2:0] bits to 000b.

26.2.12 RSPI Next-Access Delay Register (SPND)

SPND sets a non-active period (next-access delay) of the SSLyi signal after termination of a serial transfer when the SPCMDm.SPNDEN bit is 1. Writing to SPND is prohibited while both the SPCR.MSTR and SPCR.SPE bits are 1 (m = 0 to 7, y = 0, 1 (for all channels); i = 0 to 3).

Address(es): RSPI0.SPND A006 800Eh, RSPI1.SPND A006 840Eh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	SPNDL[2:0]		
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	SPNDL[2:0]	RSPI Next-Access Delay Setting	b2 b0 0 0 0: 1 RSPCK + 2 SERICLK 0 0 1: 2 RSPCK + 2 SERICLK 0 1 0: 3 RSPCK + 2 SERICLK 0 1 1: 4 RSPCK + 2 SERICLK 1 0 0: 5 RSPCK + 2 SERICLK 1 0 1: 6 RSPCK + 2 SERICLK 1 1 0: 7 RSPCK + 2 SERICLK 1 1 1: 8 RSPCK + 2 SERICLK	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

SPNDL[2:0] Bits (RSPI Next-Access Delay Setting)

The SPNDL[2:0] bits set a next-access delay when the SPCMDm.SPNDEN bit is 1.

When using the RSPI in slave mode, set the SPNDL[2:0] bits to 000b.

26.2.13 RSPI Control Register 2 (SPCR2)

The SPCR2 register controls the settings for operation of the RSPI.
Changing the value of SPPE, SPOE, or SCKASE bit in the SPCR2 register is prohibited while the setting of the SPCR.SPE bit is 1.

Address(es): RSPI0.SPCR2 A006 800Fh, RSPI1.SPCR2 A006 840Fh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	SCKASE	PTE	SPIIE	SPOE	SPPE
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SPPE	Parity Enable	0: Does not add the parity bit to transmit data and does not check the parity bit of receive data 1: Adds the parity bit to transmit data and checks the parity bit of receive data (when SPCR.TXMD = 0) Adds the parity bit to transmit data but does not check the parity bit of receive data (when SPCR.TXMD = 1)	R/W
b1	SPOE	Parity Mode	0: Selects even parity for use in transmission and reception 1: Selects odd parity for use in transmission and reception	R/W
b2	SPIIE	RSPI Idle Interrupt Enable	0: Disables the generation of idle interrupt requests 1: Enables the generation of idle interrupt requests	R/W
b3	PTE	Parity Self-Testing	0: Disables the self-diagnosis function of the parity circuit 1: Enables the self-diagnosis function of the parity circuit	R/W
b4	SCKASE	RSPCK Auto-Stop Function Enable	0: Disables the RSPCK auto-stop function 1: Enables the RSPCK auto-stop function	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

SPPE Bit (Parity Enable)

The SPPE bit enables or disables the parity function.
The parity bit is added to transmit data and parity checking is performed for receive data when the SPCR.TXMD bit is 0 and the SPCR2.SPPE bit is 1.
The parity bit is added to transmit data but parity checking is not performed for receive data when the SPCR.TXMD bit is 1 and the SPCR2.SPPE bit is 1.

SPOE Bit (Parity Mode)

The SPOE bit specifies odd or even parity.
When even parity is set, parity bit addition is performed so that the total number of 1-bits in the transmit/receive character plus the parity bit is even. Similarly, when odd parity is set, parity bit addition is performed so that the total number of 1-bits in the transmit/receive character plus the parity bit is odd.
The SPOE bit is valid only when the SPPE bit is 1.

SPIIE Bit (RSPI Idle Interrupt Enable)

The SPIIE bit enables or disables the generation of RSPI idle interrupt requests when the RSPI being in the idle state is detected and the SPSR.IDLNF flag is set to 0.

PTE Bit (Parity Self-Testing)

The PTE bit enables the self-diagnosis function of the parity circuit in order to check whether the parity function is operating correctly.

SCKASE Bit (RSPCK Auto-Stop Function Enable)

The SCKASE bit enables or disables the RSPCK auto-stop function. When this function is enabled, the RSPCK clock is stopped before an overrun error occurs when data is received in master mode. For details, see section 26.3.8.1, Overrun Error.

26.2.14 RSPI Command Registers 0 to 7 (SPCMD0 to SPCMD7)

SPCMDm registers control a transfer format for the RSPI in master mode. Each channel has eight RSPI command registers (m = 0 to 7).

Some of the bits in the SPCMD0 register is used to set a transfer format for the RSPI in slave mode. The RSPI in master mode refers to SPCMDm registers in sequence according to the settings in the SPSCR.SPSLN[2:0] bits, and executes the serial transfer that is set in respective SPCMDm registers.

SPCMDm registers should be set while the transmission buffers are empty (data for the next transfer is not set) and before setting of the data that are to be transmitted as a result of reference to the SPCMDm registers.

SPCMDm currently being referred to by the RSPI in master mode can be checked by means of the SPSSR.SPCP[2:0] bits. Writing to SPCMDm is prohibited while the SPCR.MSTR bit is 0 and the SPCR.SPE bit is 1.

Address(es): RSPI0.SPCMD0 A006 8010h, RSPI0.SPCMD1 A006 8012h, RSPI0.SPCMD2 A006 8014h, RSPI0.SPCMD3 A006 8016h, RSPI0.SPCMD4 A006 8018h, RSPI0.SPCMD5 A006 801Ah, RSPI0.SPCMD6 A006 801Ch, RSPI0.SPCMD7 A006 801Eh, RSPI1.SPCMD0 A006 8410h, RSPI1.SPCMD1 A006 8412h, RSPI1.SPCMD2 A006 8414h, RSPI1.SPCMD3 A006 8416h, RSPI1.SPCMD4 A006 8418h, RSPI1.SPCMD5 A006 841Ah, RSPI1.SPCMD6 A006 841Ch, RSPI1.SPCMD7 A006 841Eh

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB[3:0]				SSLKP	SSLy[2:0]			BRDV[1:0]		CPOL	CPHA
Value after reset:	0	0	0	0	0	1	1	1	0	0	0	0	1	1	0	1

Bit	Symbol	Bit Name	Description	R/W
b0	CPHA	RSPCK Phase Setting	0: Data sampling on odd edge, data variation on even edge 1: Data variation on odd edge, data sampling on even edge	R/W
b1	CPOL	RSPCK Polarity Setting	0: RSPCK is low when idle 1: RSPCK is high when idle	R/W
b3, b2	BRDV[1:0]	Bit Rate Division Setting	b3 b2 0 0: These bits select the base bit rate 0 1: These bits select the base bit rate divided by 2 1 0: These bits select the base bit rate divided by 4 1 1: These bits select the base bit rate divided by 8	R/W
b6 to b4	SSLy[2:0]	SSL Signal Assertion Setting	b6 b4 0 0 0: SSLy0 (y = 0, 1) 0 0 1: SSLy1 (y = 0, 1) 0 1 0: SSL02 0 1 1: SSL03 Settings other than above are prohibited.	R/W
b7	SSLKP	SSL Signal Level Keeping	0: Negates all SSL signals upon completion of transfer 1: Keeps the SSL signal level from the end of transfer until the beginning of the next access	R/W
b11 to b8	SPB[3:0]	RSPI Data Length Setting	b11 b8 0100 to 0111: 8 bits 1 0 0 0: 9 bits 1 0 0 1: 10 bits 1 0 1 0: 11 bits 1 0 1 1: 12 bits 1 1 0 0: 13 bits 1 1 0 1: 14 bits 1 1 1 0: 15 bits 1 1 1 1: 16 bits 0 0 0 0: 20 bits 0 0 0 1: 24 bits 0010, 0011: 32 bits	R/W
b12	LSBF	RSPI LSB First	0: MSB first 1: LSB first	R/W
b13	SPNDEN	RSPI Next-Access Delay Enable	0: A next-access delay of 1 RSPCK + 2 SERICLK 1: A next-access delay is equal to the setting of the RSPI next-access delay register (SPND)	R/W

Bit	Symbol	Bit Name	Description	R/W
b14	SLNDEN	SSL Negation Delay Setting Enable	0: An SSL negation delay of 1 RSPCK 1: An SSL negation delay is equal to the setting of the RSPI slave select negation delay register (SSLND)	R/W
b15	SCKDEN	RSPCK Delay Setting Enable	0: An RSPCK delay of 1 RSPCK 1: An RSPCK delay is equal to the setting of the RSPI clock delay register (SPCKD)	R/W

CPHA Bit (RSPCK Phase Setting)

The CPHA bit sets an RSPCK phase of the RSPI in master mode or slave mode. Data communications between RSPI modules require the same RSPCK phase setting between the modules.

CPOL Bit (RSPCK Polarity Setting)

The CPOL bit sets an RSPCK polarity of the RSPI in master mode or slave mode. Data communications between RSPI modules require the same RSPCK polarity setting between the modules.

BRDV[1:0] Bits (Bit Rate Division Setting)

The BRDV[1:0] bits are used to determine the bit rate. A bit rate is determined by combinations of the settings in the BRDV[1:0] bits and SPBR (refer to section 26.2.8, RSPI Bit Rate Register (SPBR)). The settings in SPBR determine the base bit rate. The settings in the BRDV[1:0] bits are used to select a bit rate which is obtained by dividing the base bit rate by 1, 2, 4, or 8. In SPCMDm register, different BRDV[1:0] bit settings can be specified. This permits the execution of serial transfers at a different bit rate for each command (m = 0 to 7).

SSLy[2:0] Bits (SSL Signal Assertion Setting)

The SSLy[2:0] bits control the SSLyi signal assertion when the RSPI performs serial transfers in master mode.
Setting the SSLy[2:0] bits controls the assertion for the SSLyi signal. When an SSLyi signal is asserted, its polarity is determined by the set value in the corresponding SSLP. When the SSLy[2:0] bits are set to 000b in multi-master mode, serial transfers are performed with all the SSL signals in the negated state (as the SSLy0 pin acts as input).
When using the RSPI in slave mode, set the SSLy[2:0] bits to 000b.
(y = 0, 1 (for all channels); i = 0 to 3)

SSLKP Bit (SSL Signal Level Keeping)

When the RSPI in master mode performs a serial transfer, the SSLKP bit specifies whether the SSLyi 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.
Setting the SSLKP bit to 1 enables a burst transfer. For details, refer to section 26.3.10.1, (4) Burst Transfer.
When using the RSPI in slave mode, the SSLKP bit should be set to 0.
(y = 0, 1 (for all channels); i = 0 to 3)

SPB[3:0] Bits (RSPI Data Length Setting)

The SPB[3:0] bits set a transfer data length for the RSPI in master mode or slave mode.

LSBF Bit (RSPI LSB First)

The LSBF bit sets the data format of the RSPI in master mode or slave mode to MSB first or LSB first.

SPNDEN Bit (RSPI Next-Access Delay Enable)

The SPNDEN bit sets the period from the time the RSPI in master mode terminates a serial transfer and sets the SSLyi signal inactive until the RSPI enables the SSLyi signal assertion for the next access (next-access delay) (y = 0, 1 (for all channels); i = 0 to 3). If the SPNDEN bit is 0, the RSPI sets the next-access delay to 1 RSPCK + 2 SERICLK. If the SPNDEN bit is 1, the RSPI inserts a next-access delay in compliance with the SPND setting.
When using the RSPI in slave mode, the SPNDEN bit should be set to 0.

SLNDEN Bit (SSL Negation Delay Setting Enable)

The SLNDEN bit sets the period from the time the RSPI in master mode stops RSPCK oscillation until the RSPI sets the SSLyi signal inactive (SSL negation delay). If the SLNDEN bit is 0, the RSPI sets the SSL negation delay to 1 RSPCK. If the SLNDEN bit is 1, the RSPI negates the SSL signal at an SSL negation delay in compliance with the SSLND setting.
When using the RSPI in slave mode, the SLNDEN bit should be set to 0.

SCKDEN Bit (RSPCK Delay Setting Enable)

The SCKDEN bit sets the period from the point when the RSPI in master mode activates the SSLyi signal until the RSPCK starts oscillation (RSPI clock delay). If the SCKDEN bit is 0, the RSPI sets the RSPCK delay to 1 RSPCK. If the SCKDEN bit is 1, the RSPI starts the oscillation of RSPCK at an RSPCK delay in compliance with the SPCKD setting.
When using the RSPI in slave mode, the SCKDEN bit should be set to 0.

26.3 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.

26.3.1 Overview of RSPI Operations

The RSPI is capable of synchronous serial transfers in slave mode (SPI operation), single-master mode (SPI operation), multi-master mode (SPI operation), slave mode (clock synchronous operation), and master mode (clock synchronous operation). A particular mode of the RSPI can be selected by using the MSTR, MODFEN, and SPMS bits in SPCR. Table 26.5 lists the relationship between RSPI modes and SPCR settings, and a description of each mode.

Table 26.5 Relationship between RSPI Modes and SPCR Settings and Description of Each Mode

Mode	Slave (SPI Operation)	Single-Master (SPI Operation)	Multi-Master (SPI Operation)	Slave (Clock Synchronous Operation)	Master (Clock Synchronous Operation)
MSTR bit setting	0	1	1	0	1
MODFEN bit setting	0 or 1	0	1	0	0
SPMS bit setting	0	0	0	1	1
RSPCKy signal	Input	Output	Output/Hi-Z	Input	Output
MOSIy signal	Input	Output	Output/Hi-Z	Input	Output
MISOy signal	Output/Hi-Z	Input	Input	Output	Input
SSLy0 signal	Input	Output	Input	Hi-Z*1	Hi-Z*1
SSLy1 to SSLy3 signals	Hi-Z*1	Output	Output/Hi-Z	Hi-Z*1	Hi-Z*1
SSL polarity modification function	Supported	Supported	Supported	—	—
Transfer rate	Up to SERICLK/8	Up to SERICLK/4	Up to SERICLK/4	Up to SERICLK/8	Up to SERICLK/4
Clock source	RSPCK input	On-chip baud rate generator	On-chip baud rate generator	RSPCK input	On-chip baud rate generator
Clock polarity	Two				
Clock phase	Two	Two	Two	One (CPHA = 1)	Two
First transfer bit	MSB/LSB				
Transfer data length	8 to 16, 20, 24, 32 bits				
Burst transfer	Possible (CPHA = 1)	Possible (CPHA = 0,1)	Possible (CPHA = 0,1)	—	—
RSPCK delay control	Not supported	Supported	Supported	Not supported	Supported
SSL negation delay control	Not supported	Supported	Supported	Not supported	Supported
Next-access delay control	Not supported	Supported	Supported	Not supported	Supported
Transfer activation method	SSL input active or RSPCK oscillation	Transmission buffer is written to at generation of a transmission buffer empty interrupt request	Transmission buffer is written to at generation of a transmission buffer empty interrupt request	RSPCK oscillation	Transmission buffer is written to at generation of a transmission buffer empty interrupt request
Sequence control	Not supported	Supported	Supported	Not supported	Supported
Transmission buffer empty detection	Supported				
Reception buffer full detection	Supported*2				
Overrun error detection	Supported*2	Supported*2, *4	Supported*2, *4	Supported*2	Supported*2
Parity error detection	Supported*2,*3				
Mode fault error detection	Supported (MODFEN = 1)	Not supported	Supported	Not supported	Not supported

Note 1. This function is not supported in this mode.
Note 2. When the SPCR.TXMD bit is 1, receiver buffer full detection, overrun error detection, and parity error detection are not performed.
Note 3. When the SPCR2.SPPE bit is 0, parity error detection is not performed.
Note 4. When the SPCR2.SCKASE bit is 1, overrun error detection does not proceed.
y = 0, 1(for all channels)

26.3.2 Controlling RSPI Pins

According to the SPCR.MSTR, MODFEN, SPMS, and SPPCR.SPOM bits, the RSPI can switch pin states. Table 26.6 lists the relationship between pin states and bit settings. Setting the SPPCR.SPOM bit to 0 selects CMOS output; setting it to 1 selects open-drain output. The I/O port settings should follow this relationship.

Table 26.6 Relationship between Pin States and Bit Settings

		Pin State*2	
Mode	**Pin**	**SPOM = 0**	**SPOM = 1**
Single-master mode (SPI operation) (MSTR = 1, MODFEN = 0, SPMS = 0)	RSPCKy	CMOS output	Open-drain output
	SSLy0 to SSLy3	CMOS output	Open-drain output
	MOSIy	CMOS output	Open-drain output
	MISOy	Input	Input
Multi-master mode (SPI operation) (MSTR = 1, MODFEN = 1, SPMS = 0)	RSPCKy*3	CMOS output/Hi-Z	Open-drain output/Hi-Z
	SSLy0	Input	Input
	SSLy1 to SSLy3*3	CMOS output/Hi-Z	Open-drain output/Hi-Z
	MOSIy*3	CMOS output/Hi-Z	Open-drain output/Hi-Z
	MISOy	Input	Input
Slave mode (SPI operation) (MSTR = 0, SPMS = 0)	RSPCKy	Input	Input
	SSLy0	Input	Input
	SSLy1 to SSLy3*5	Hi-Z*1	Hi-Z*1
	MOSIy	Input	Input
	MISOy*4	CMOS output/Hi-Z	Open-drain output/Hi-Z
Master mode (Clock synchronous operation) (MSTR = 1, MODFEN = 0, SPMS = 1)	RSPCKy	CMOS output	Open-drain output
	SSLy0 to SSLy3*5	Hi-Z*1	Hi-Z*1
	MOSIy	CMOS output	Open-drain output
	MISOy	Input	Input
Slave mode (Clock synchronous operation) (MSTR = 0, SPMS = 1)	RSPCKy	Input	Input
	SSLy0 to SSLy3*5	Hi-Z*1	Hi-Z*1
	MOSIy	Input	Input
	MISOy	CMOS output	Open-drain output

Note 1. This function is not supported in this mode.
Note 2. RSPI settings are not reflected in the multiplex pins for which the RSPI function is not selected.
Note 3. When SSLy0 is at the active level, the pin state is Hi-Z.
Note 4. When SSLy0 is at the non-active level or the SPCR.SPE bit is 0, the pin state is Hi-Z.
Note 5. These pins are available for use as I/O port pins.
y = 0, 1 (for all channels)

The RSPI in single-master mode (SPI operation) or multi-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 listed in Table 26.7.

Table 26.7 MOSI Signal Value Determination during SSL Negation Period

MOIFE Bit	MOIFV Bit	MOSIy Signal Value during SSL Negation Period
0	0, 1	Final data from previous transfer
1	0	Always low
1	1	Always high

Note: During the period of SSL negation, the RSPI function must be enabled (i.e. SPCR.SPE bit = 1) for control over the levels of the MOSIy signals.

26.3.3 RSPI System Configuration Examples

26.3.3.1 Single Master/Single Slave (with This LSI Acting as Master)

Figure 26.5 shows a single-master/single-slave RSPI system configuration example when this LSI is used as a master. In the single-master/single-slave configuration, the SSLy0 to SSLy3 output of this LSI (master) are not used. The SSL input of the SPI slave is fixed to the low level, and the SPI slave is always maintained in a select state.*1
This LSI (master) always drives RSPCKy and MOSIy. The SPI slave always drives the MISO.
(y = 0, 1 (for all channels))

Note 1. In the transfer format corresponding to the case where the SPCMDm.CPHA bit 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 SSLyi output of this LSI should be connected to the SSL input of the slave device.

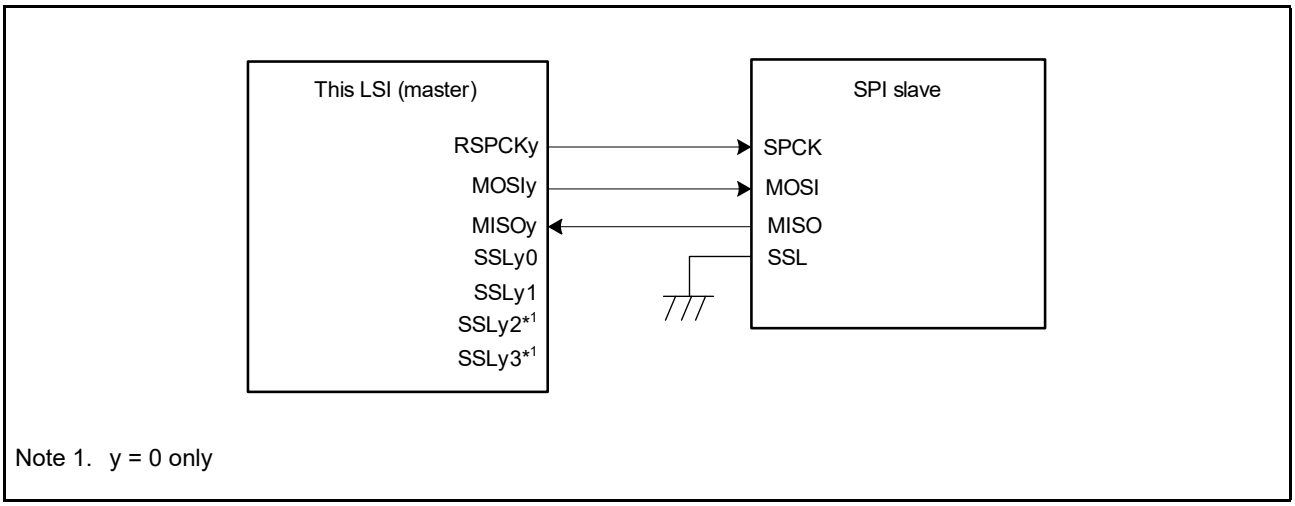


Figure 26.5 Single-Master/Single-Slave Configuration Example (This LSI = Master)

26.3.3.2 Single Master/Single Slave (with This LSI Acting as Slave)

Figure 26.6 shows a single-master/single-slave RSPI system configuration example when this LSI is used as a slave. When this LSI is to operate as a slave, the SSLy0 pin is used as SSL input. The SPI master always drives the RSPCK and MOSI. This LSI (slave) always drives the MISOy.*1

In the single-slave configuration in which the SPCMDm.CPHA bit is set to 1, the SSLy0 input of this LSI (slave) is fixed to the low level, this LSI (slave) is always maintained in a select state, and in this manner it is possible to execute serial transfer (Figure 26.7).

(y = 0, 1 (for all channels))

Note 1. When SSLy0 is at the non-active level, the pin state is Hi-Z.

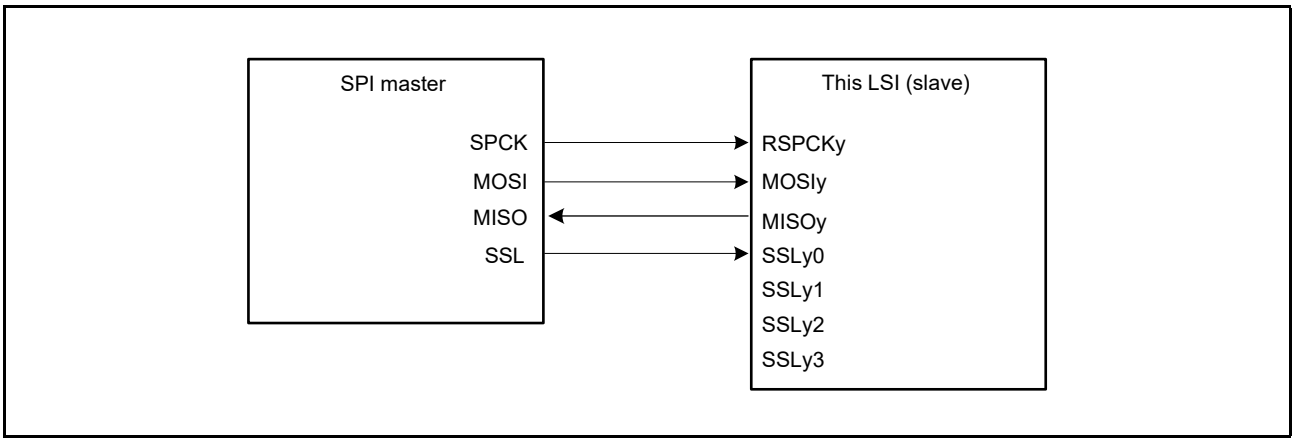


Figure 26.6 Single-Master/Single-Slave Configuration Example (This LSI = Slave, CPHA = 0)

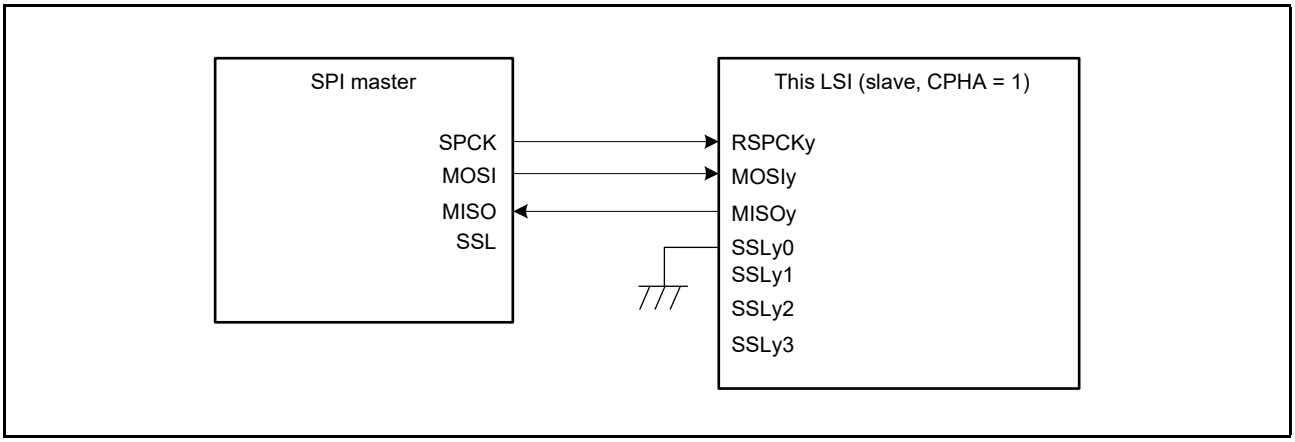


Figure 26.7 Single-Master/Single-Slave Configuration Example (This LSI = Slave, CPHA = 1)

26.3.3.3 Single Master/Multi-Slave (with This LSI Acting as Master)

Figure 26.8 shows a single-master/multi-slave RSPI system configuration example when this LSI is used as a master. In the example of Figure 26.8, the RSPI system is comprised of this LSI (master) and four slaves (SPI slave 0 to SPI slave 3).

The RSPCKy and MOSIy outputs of this LSI (master) are connected to the RSPCK and MOSI inputs of SPI slave 0 to SPI slave 3. The MISO outputs of SPI slave 0 to SPI slave 3 are all connected to the MISOy input of this LSI (master). SSLy0 to SSLy3 outputs of this LSI (master) are connected to the SSL inputs of SPI slave 0 to SPI slave 3, respectively. This LSI (master) always drives RSPCK, MOSI, and SSLy0 to SSLy3. Of the SPI slave 0 to SPI slave 3, the slave that receives low-level input into the SSL input drives MISO.

(y = 0, 1 (for all channels))

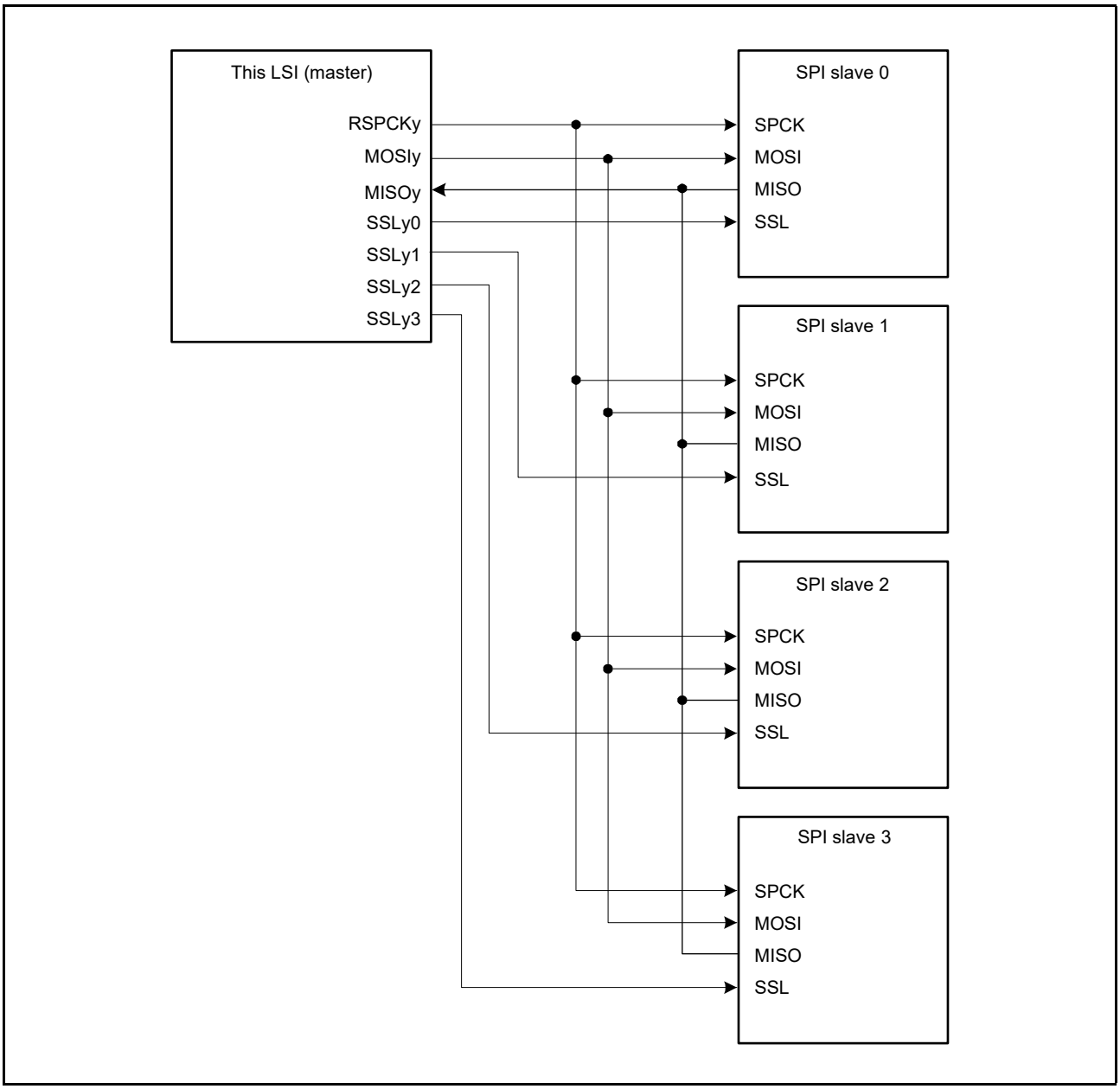


Figure 26.8 Single-Master/Multi-Slave Configuration Example (This LSI = Master)

26.3.3.4 Single Master/Multi-Slave (with This LSI Acting as Slave)

Figure 26.9 shows a single-master/multi-slave RSPI system configuration example when this LSI is used as a slave. In the example of Figure 26.9, the RSPI system is comprised of an SPI master and two LSIs (slave X and slave Y).
The SPCK and MOSI outputs of the SPI master are connected to the RSPCKy and MOSIy inputs of the LSIs (slave X and slave Y). The MISOy outputs of the LSIs (slave X and slave Y) are all connected to the MISO input of the SPI master. SSLX and SSLY outputs of the SPI master are connected to the SSLy0 inputs of the LSIs (slave X and slave Y), respectively.
The SPI master always drives SPCK, MOSI, SSLX, and SSLY. Of the LSIs (slave X and slave Y), the slave that receives low-level input into the SSLy0 input drives MISOy.
(y = 0, 1 (for all channels))

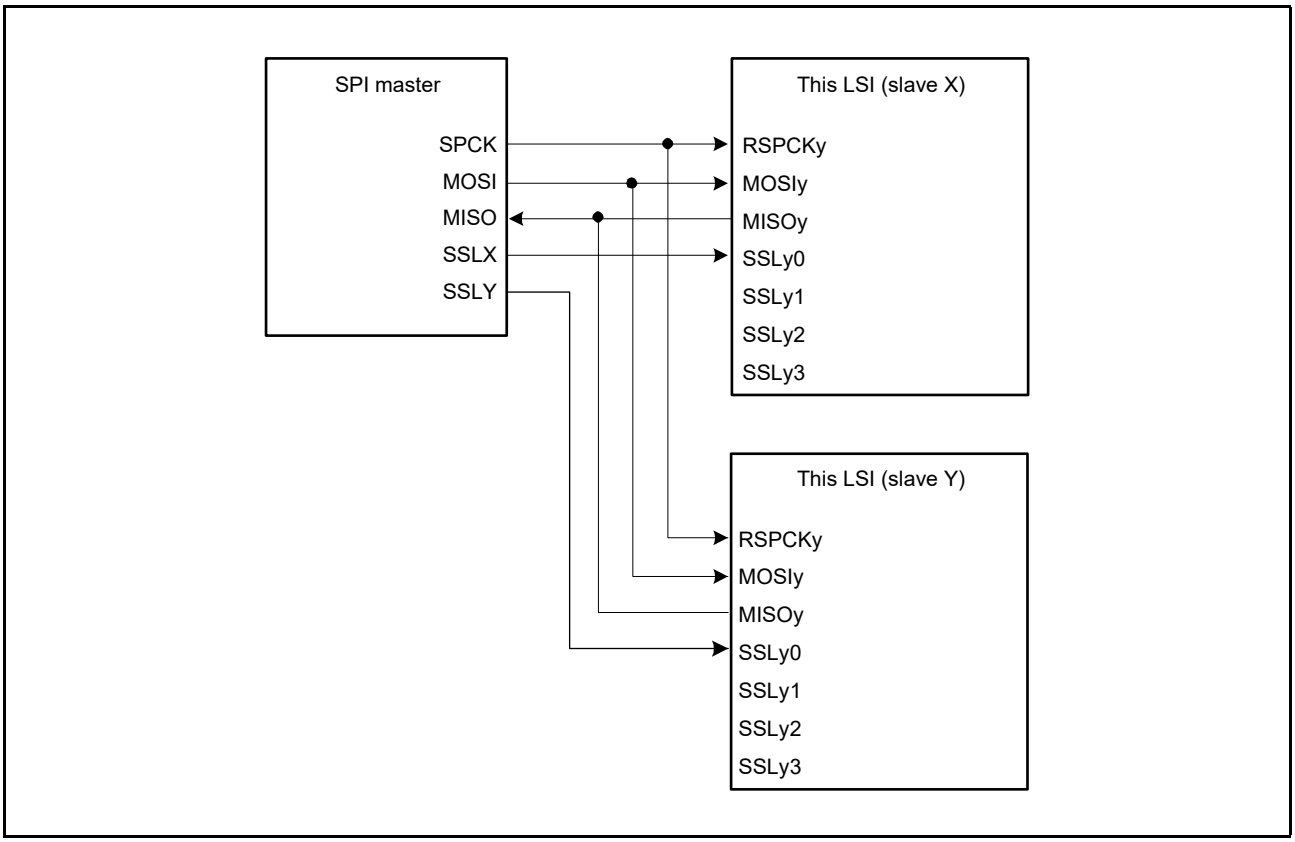


Figure 26.9 Single-Master/Multi-Slave Configuration Example (This LSI = Slave)

26.3.3.5 Multi-Master/Multi-Slave (with This LSI Acting as Master)

Figure 26.10 shows a multi-master/multi-slave RSPI system configuration example when this LSI is used as a master. In the example of Figure 26.10, the RSPI system is comprised of two LSIs (master X and master Y) and two SPI slaves (SPI slave 1 and SPI slave 2).

The RSPCKy and MOSIy outputs of the LSIs (master X and master Y) are connected to the RSPCK and MOSI inputs of SPI slaves 1 and 2. The MISO outputs of SPI slaves 1 and 2 are connected to the MISOy inputs of the LSIs (master X and master Y). Any generic port Y output from this LSI (master X) is connected to the SSLy0 input of this LSI (master Y). Any generic port X output of this LSI (master Y) is connected to the SSLy0 input of this LSI (master X). The SSLy1 and SSLy2 outputs of the LSIs (master X and master Y) are connected to the SSL inputs of the SPI slaves 1 and 2. In this configuration example, since the system can be comprised solely of SSLy0 input, and SSLy1 and SSLy2 outputs for slave connections, the SSLy3 output of this LSI is not required.

This LSI drives RSPCKy, MOSIy, SSLy1, and SSLy2 when the SSLy0 input level is high. When the SSLy0 input level is low, this LSI detects a mode fault error, sets RSPCKy, MOSIy, SSLy1, and SSLy2 to Hi-Z, and releases the RSPI bus right to the other master. Of the SPI slaves 1 and 2, the slave that receives low-level input into the SSL input drives MISO.

(y = 0, 1 (for all channels))

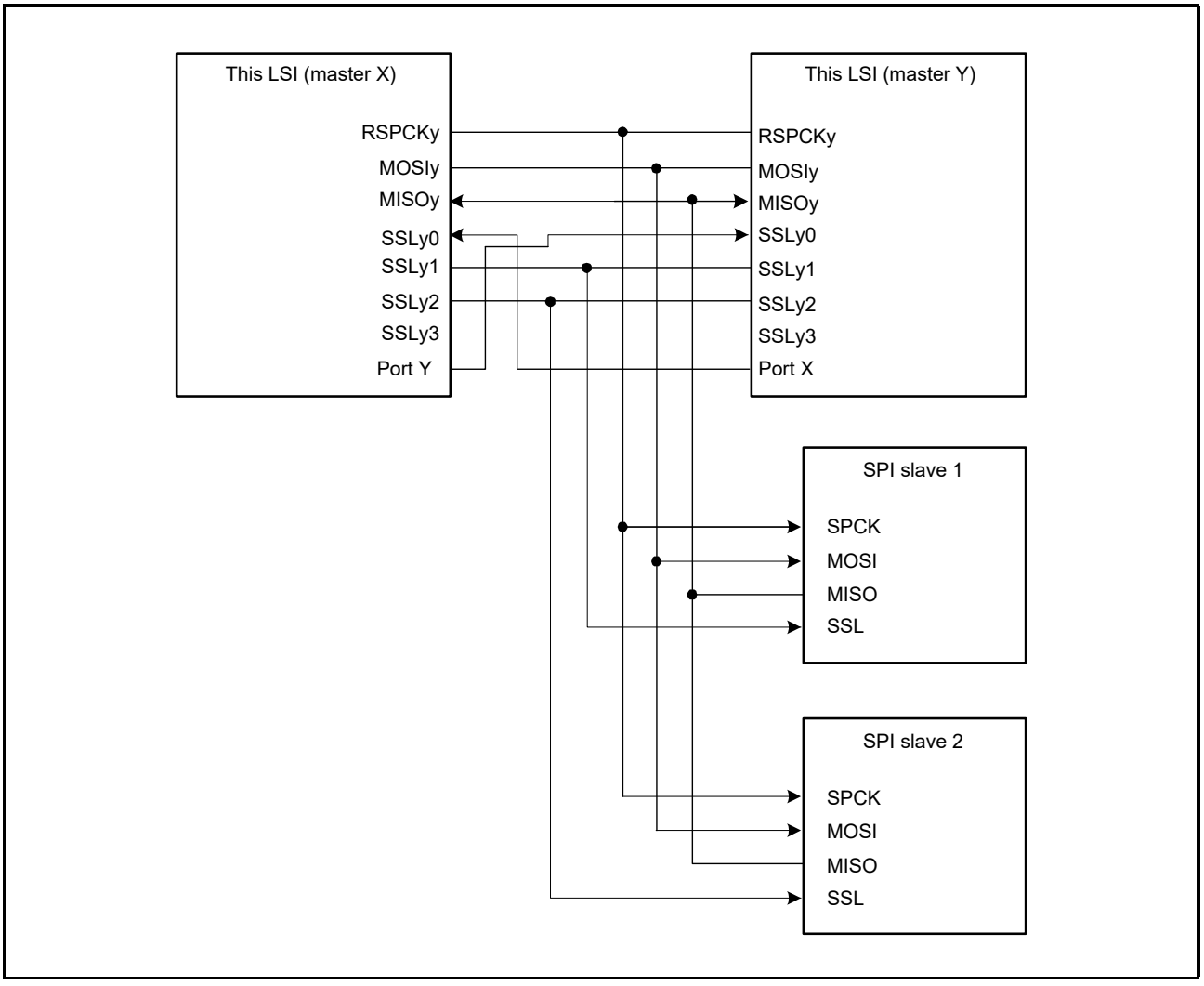


Figure 26.10 Multi-Master/Multi-Slave Configuration Example (This LSI = Master)

26.3.3.6 Master (Clock Synchronous Operation)/Slave (Clock Synchronous Operation) (with This LSI Acting as Master)

Figure 26.11 shows a master (clock synchronous operation)/slave (clock synchronous operation) RSPI system configuration example when this LSI is used as a master. In the master (clock synchronous operation)/slave (clock synchronous operation) configuration, SSLy0 to SSLy3 of this LSI (master) are not used.
This LSI (master) always drives the RSPCKy and MOSIy. The SPI slave always drives the MISO.
(y = 0, 1 (for all channels))

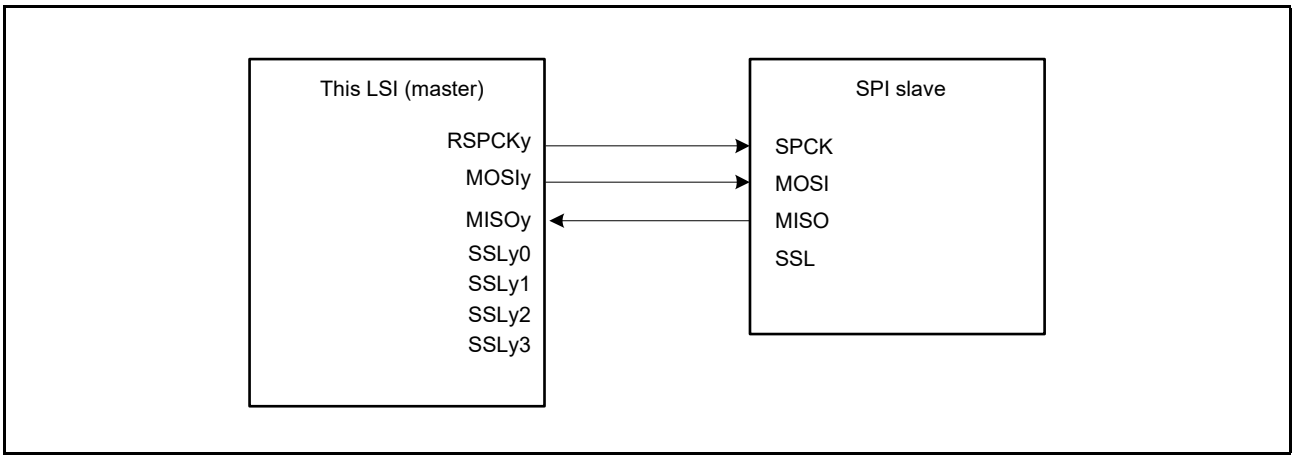


Figure 26.11 Master (Clock Synchronous Operation)/Slave (Clock Synchronous Operation) Configuration Example (This LSI = Master)

26.3.3.7 Master (Clock Synchronous Operation)/Slave (Clock Synchronous Operation) (with This LSI Acting as Slave)

Figure 26.12 shows a master (clock synchronous operation)/slave (clock synchronous operation) RSPI system configuration example when this LSI is used as a slave. When this LSI is to operate as a slave (clock synchronous operation), this LSI (slave) always drives the MISOy and the SPI master always drives the SPCK and MOSI. In addition, SSLy0 to SSLy3 of this LSI (slave) are not used.
Only in the single-slave configuration in which the SPCMDm.CPHA bit is set to 1, this LSI (slave) can execute serial transfer.
(y = 0, 1 (for all channels))

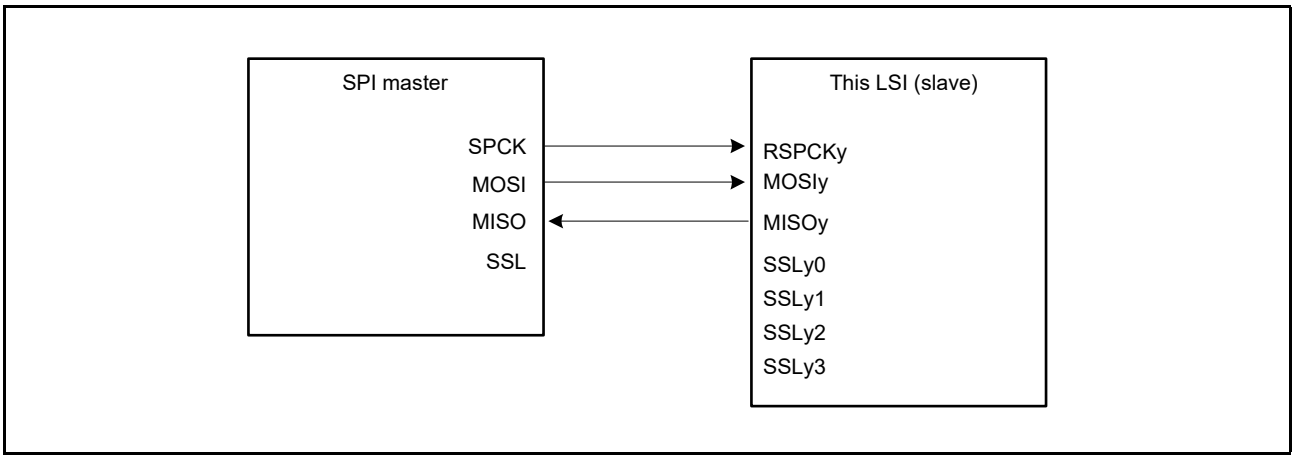


Figure 26.12 Master (Clock Synchronous Operation)/Slave (Clock Synchronous Operation) Configuration Example (This LSI = Slave, CPHA = 1)

26.3.4 Data Format

The RSPI's data format depends on the settings in RSPI command register m (SPCMDm) (m = 0 to 7) and the parity enable bit of RSPI control register 2 (SPCR2.SPPE). Regardless of whether the MSB or LSB is first, the RSPI treats the range from the LSB bit of the RSPI data register (SPDR) to the selected data length as transfer data.
The format of one frame of data before or after transfer is shown below.

(a) With Parity Disabled

When parity is disabled, transmission or reception of data proceeds with the length in bits selected in the RSPI data length setting bits in RSPI command register m (SPCMDm.SPB[3:0]).

(b) With Parity Enabled

When parity is enabled, transmission or reception of data proceeds with the length in bits selected in the RSPI data length setting bits in RSPI command register m (SPCMDm.SPB[3:0]). In this case, however, the last bit is a parity bit.

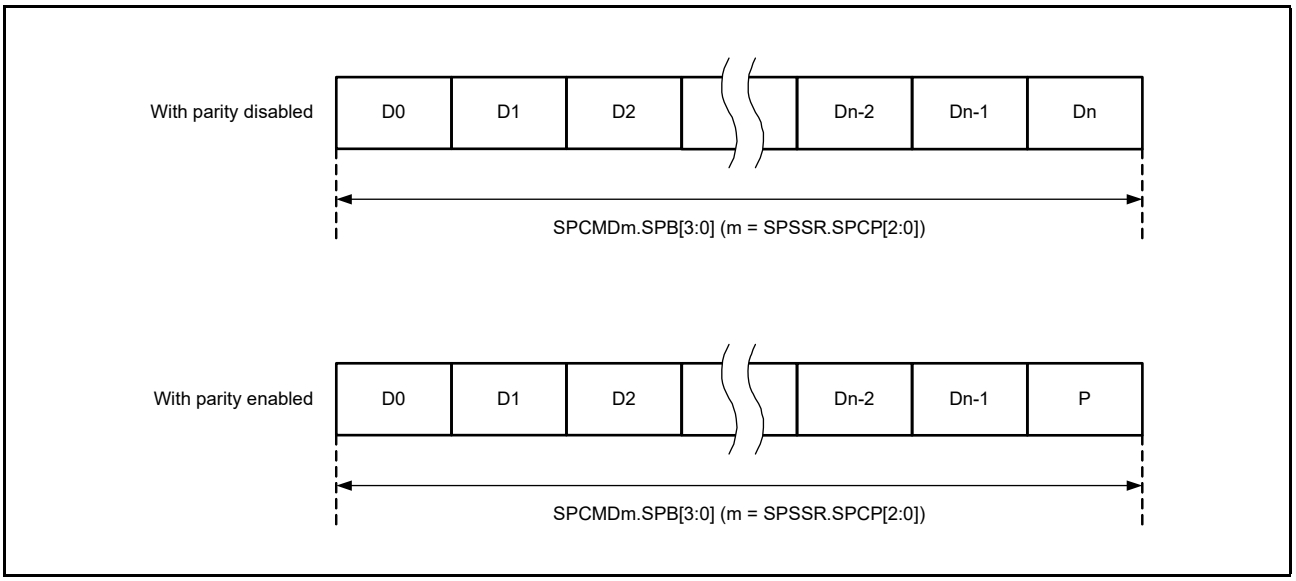


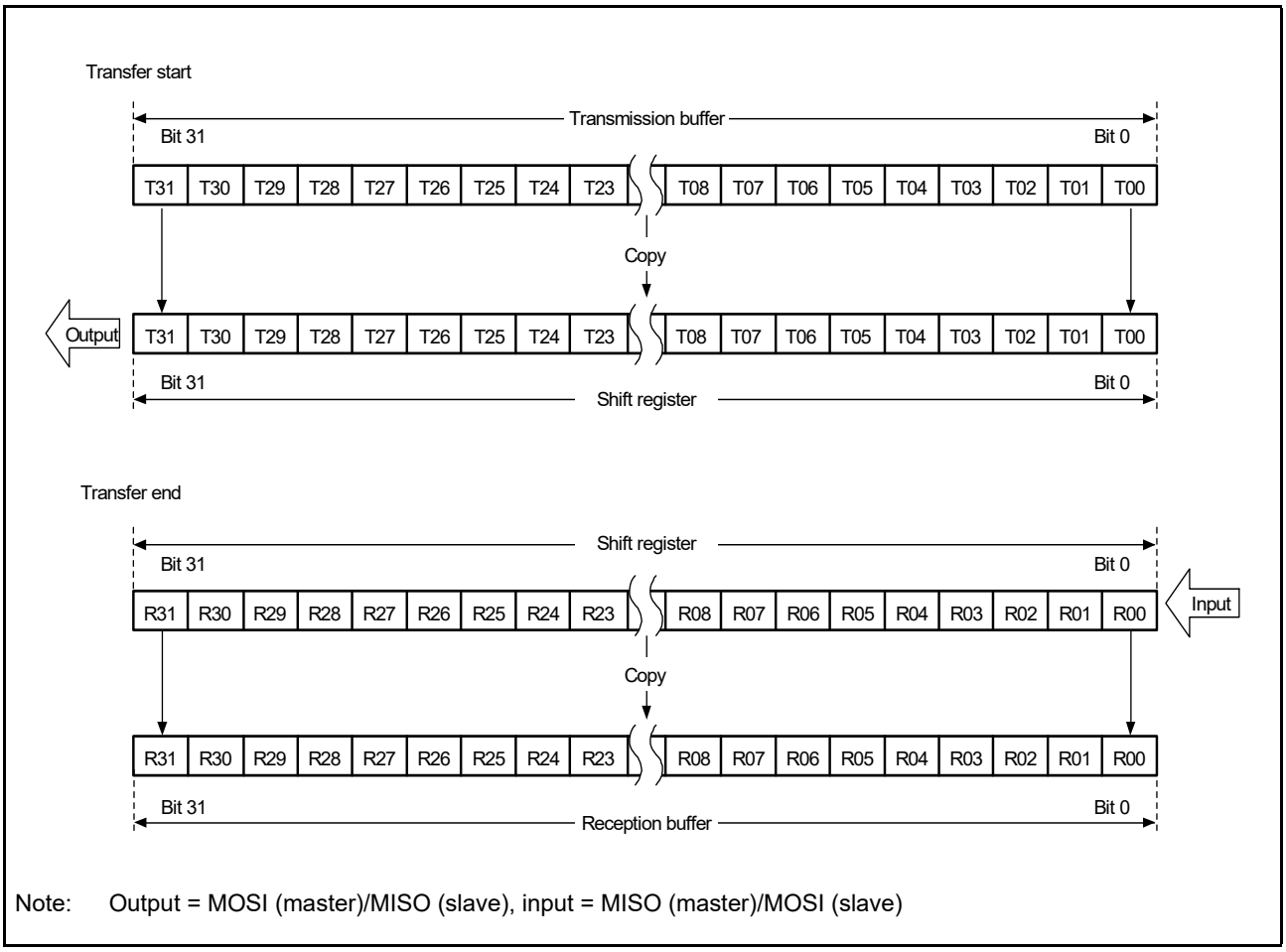
Figure 26.13 Outline of the Data Format (with Parity Disabled/Enabled)

26.3.4.1 When Parity is Disabled (SPCR2.SPPE = 0)

When parity is disabled, data for transmission are copied to the shift register with no prior processing. A description of the connection between the RSPI data register (SPDR) and the shift register in terms of the combination of MSB or LSB first and data length is given below.

(1) MSB First Transfer (32-Bit Data)

Figure 26.14 shows details of operations by the RSPI data register (SPDR) and the shift register in transfer with parity disabled, an RSPI data length of 32 bits, and MSB first selected.
In transmission, bits T31 to T00 from the current stage of the transmission buffer are copied to the shift register. Data for transmission are shifted out from the shift register in order from T31, through T30, and so on to T00.
In reception, received data are shifted in bit by bit through bit 0 of the shift register. When bits R31 to R00 have been collected after input of the required number of cycles of RSPCK, the value in the shift register is copied to the reception buffer.



Note: Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

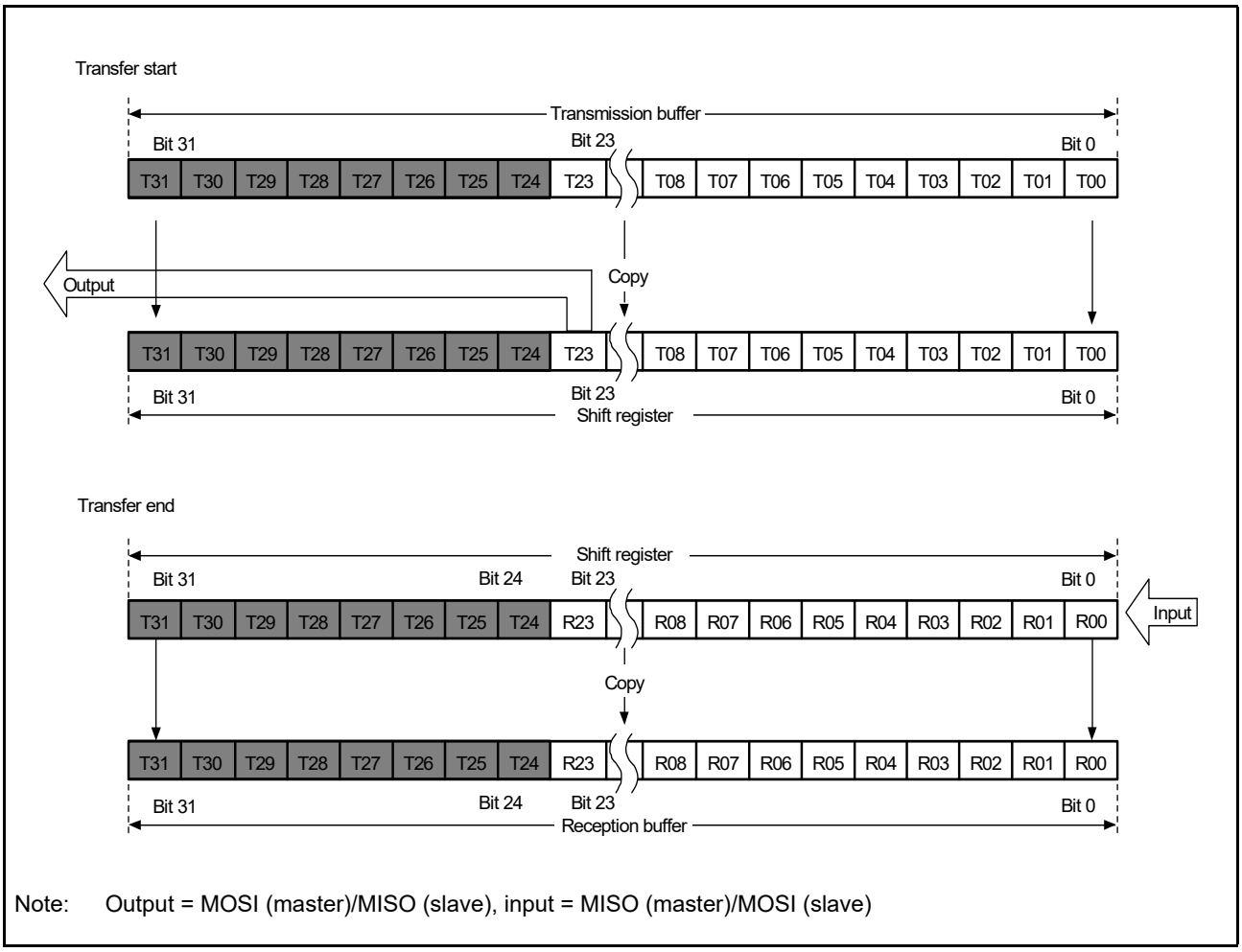
Figure 26.14 MSB First Transfer (32-Bit Data, Parity Disabled)

(2) MSB First Transfer (24-Bit Data)

Figure 26.15 shows details of operations by the RSPI data register (SPDR) and the shift register in transfer with parity disabled, 24 bits as the RSPI data length for an example that is not 32 bits, and MSB first selected.

In transmission, the lower-order 24 bits (T23 to T00) from the current stage of the transmission buffer are copied to the shift register. Data for transmission are shifted out from the shift register in order from T23, through T22, and so on to T00.

In reception, received data are shifted in bit by bit through bit 0 of the shift register. When bits R23 to R00 have been collected after input of the required number of cycles of RSPCK, the value in the shift register is copied to the reception buffer. At this time, the higher-order 8 bits of the transmission buffer are stored in the higher-order 8 bits of the reception buffer. Writing 0 to bits T31 to T24 at the time of transmission leads to 0 being inserted in the higher-order 8 bits of the reception buffer.



Note: Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

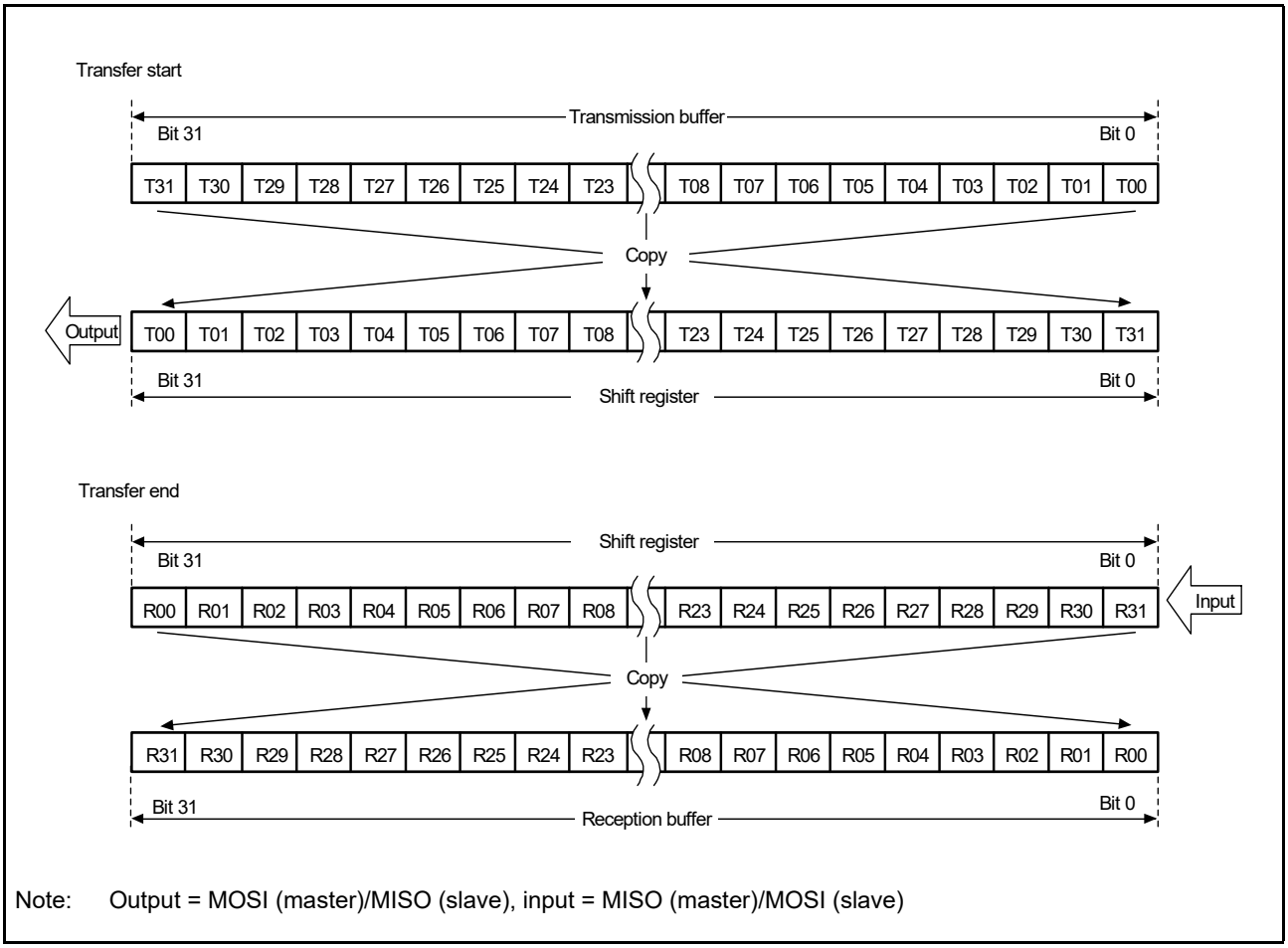
Figure 26.15 MSB First Transfer (24-Bit Data, Parity Disabled)

(3) LSB First Transfer (32-Bit Data)

Figure 26.16 shows details of operations by the RSPI data register (SPDR) and the shift register in transfer with parity disabled, an RSPI data length of 32 bits, and LSB first selected.

In transmission, bits T31 to T00 from the current stage of the transmission buffer are reordered bit by bit to obtain the order T00 to T31 for copying to the shift register. Data for transmission are shifted out from the shift register in order from T00, through T01, and so on to T31.

In reception, received data are shifted in bit by bit through bit 0 of the shift register. When bits R00 to R31 have been collected after input of the required number of cycles of RSPCK, the value in the shift register is copied to the reception buffer.



Note: Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

Figure 26.16 LSB First Transfer (32-Bit Data, Parity Disabled)

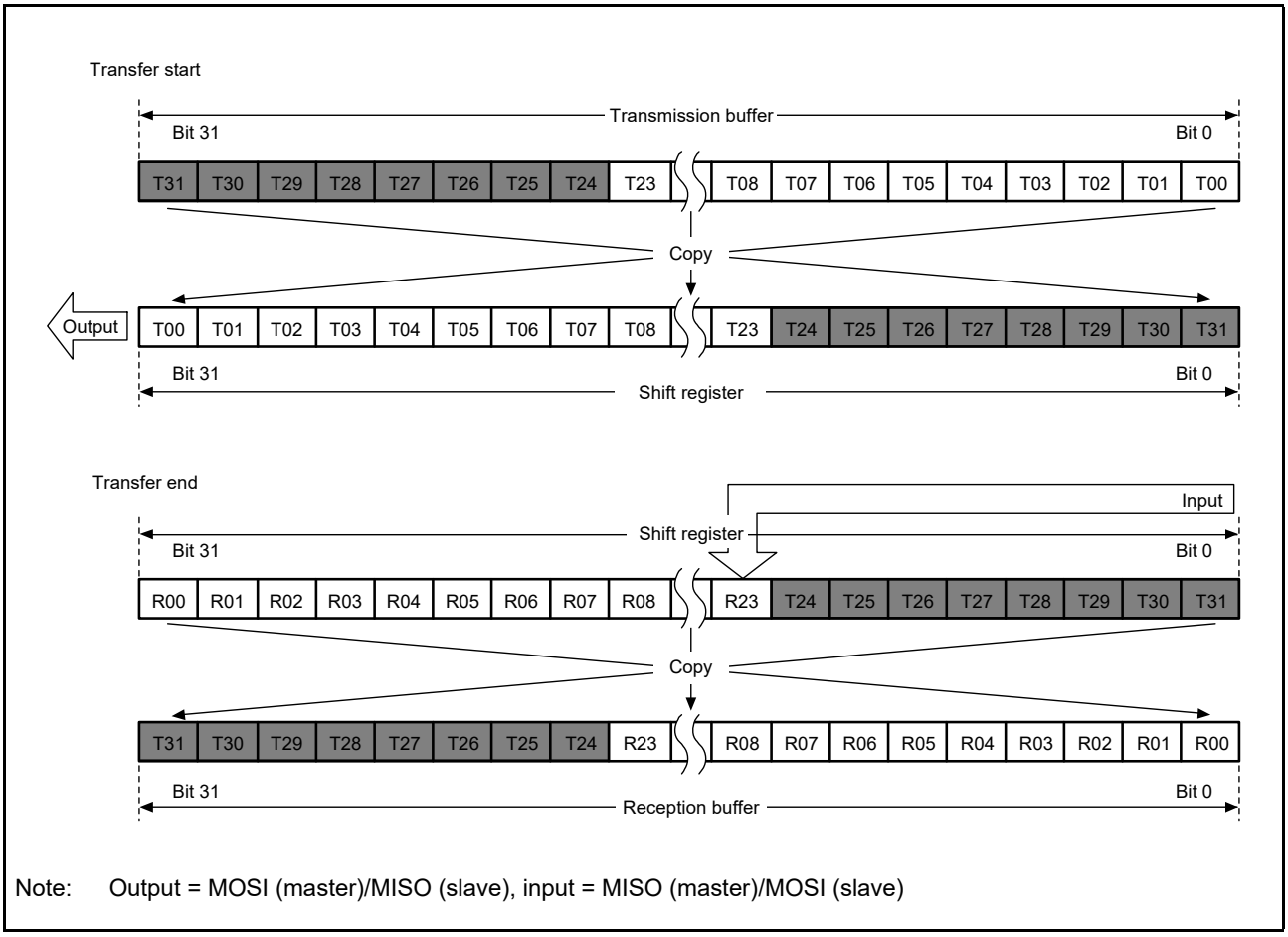
(4) LSB First Transfer (24-Bit Data)

Figure 26.17 shows details of operations by the RSPI data register (SPDR) and the shift register in transfer with parity disabled, 24 bits as the RSPI data length for an example that is not 32 bits, and LSB first selected.

In transmission, the lower-order 24 bits (T23 to T00) from the current stage of the transmission buffer are reordered bit by bit to obtain the order T00 to T23 for copying to the shift register. Data for transmission are shifted out from the shift register in order from T00, through T01, and so on to T23.

In reception, received data are shifted in bit by bit through bit 8 of the shift register. When bits R00 to R23 have been collected after input of the required number of cycles of RSPCK, the value in the shift register is copied to the reception buffer.

At this time, the higher-order 8 bits of the transmission buffer are stored in the higher-order 8 bits of the reception buffer. Writing 0 to bits T31 to T24 at the time of transmission leads to 0 being inserted in the higher-order 8 bits of the reception buffer.



Note: Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

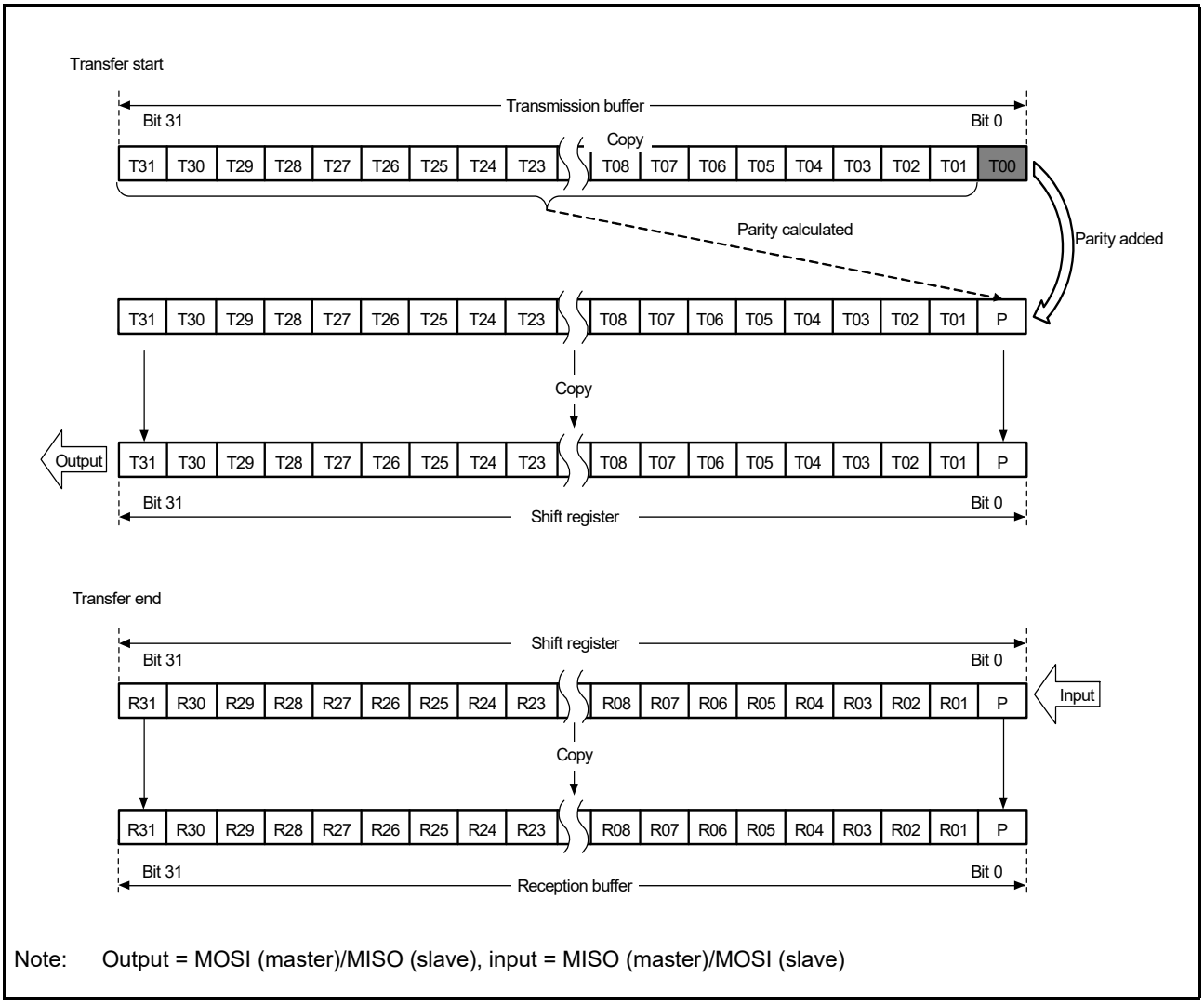
Figure 26.17 LSB First Transfer (24-Bit Data, Parity Disabled)

26.3.4.2 When Parity is Enabled (SPCR2.SPPE = 1)

When parity is enabled, the lowest-order bit of the data for transmission becomes a parity bit. Hardware calculates the value of the parity bit.

(1) MSB First Transfer (32-Bit Data)

Figure 26.18 shows details of operations by the RSPI data register (SPDR) and the shift register in transfer with parity enabled, an RSPI data length of 32 bits, and MSB first selected.
In transmission, the value of the parity bit (P) is calculated from bits T31 to T01. This replaces the final bit, T00, and the whole is copied to the shift register. Data are transmitted in the order T31, T30, …, T01, and P.
In reception, received data are shifted in bit by bit through bit 0 of the shift register. When bits R31 to P have been collected after input of the required number of cycles of RSPCK, the value in the shift register is copied to the reception buffer. On copying of data to the shift register, the data from R31 to P are checked by judging the parity.

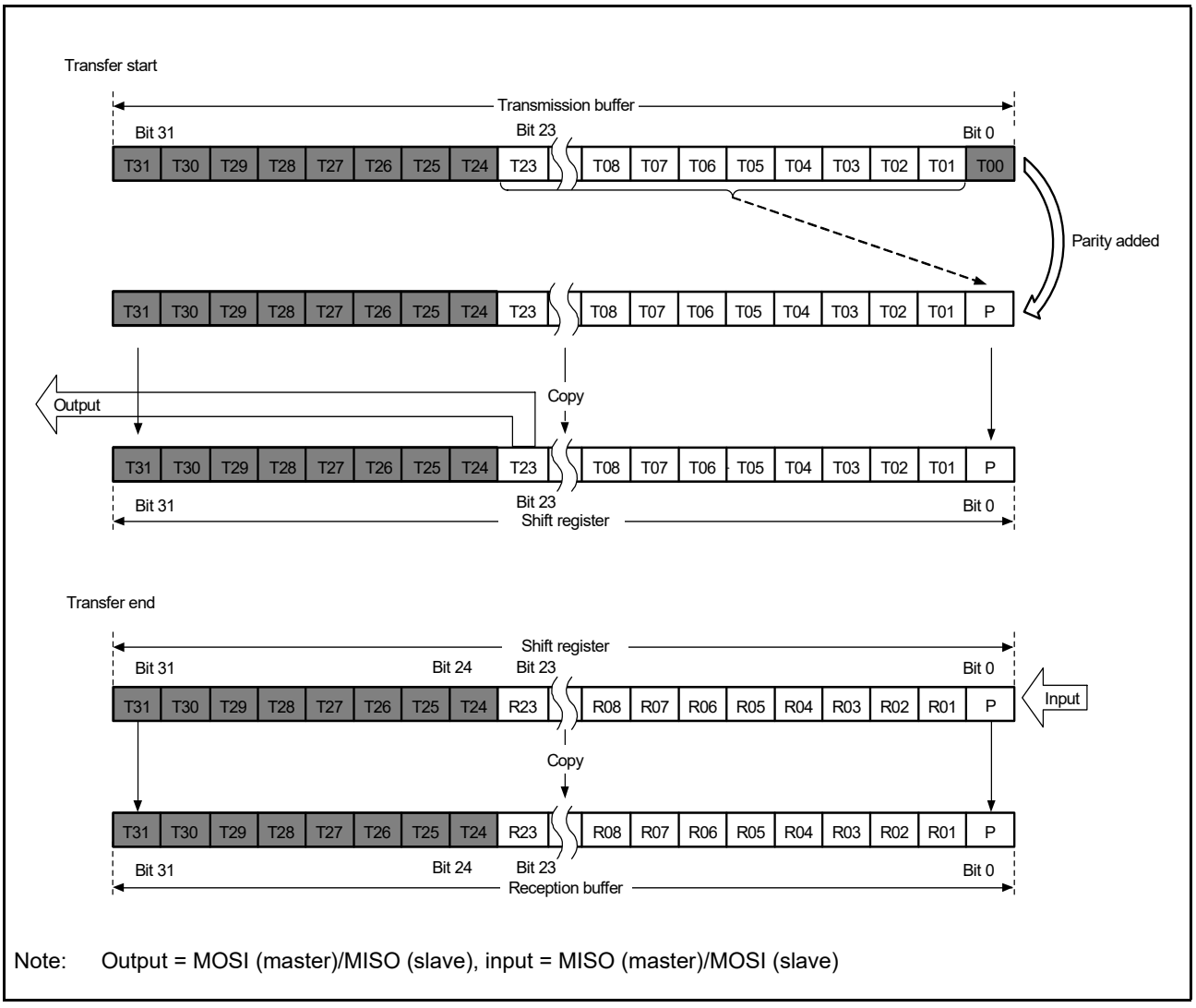


Note: Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

Figure 26.18 MSB First Transfer (32-Bit Data, Parity Enabled)

(2) MSB First Transfer (24-Bit Data)

Figure 26.19 shows details of operations by the RSPI data register (SPDR) and the shift register in transfer with parity enabled, 24 bits as the RSPI data length for an example that is not 32 bits, and MSB first selected.
In transmission, the value of the parity bit (P) is calculated from bits T23 to T01. This replaces the final bit, T00, and the whole is copied to the shift register. Data are transmitted in the order T23, T22, …, T01, and P.
In reception, received data are shifted in bit by bit through bit 0 of the shift register. When bits R23 to P have been collected after input of the required number of cycles of RSPCK, the value in the shift register is copied to the reception buffer. On copying of data to the shift register, the data from R23 to P are checked by judging the parity. At this time, the higher-order 8 bits of the transmission buffer are stored in the higher-order 8 bits of the reception buffer. Writing 0 to bits T31 to T24 at the time of transmission leads to 0 being inserted in the higher-order 8 bits of the reception buffer.



Note: Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

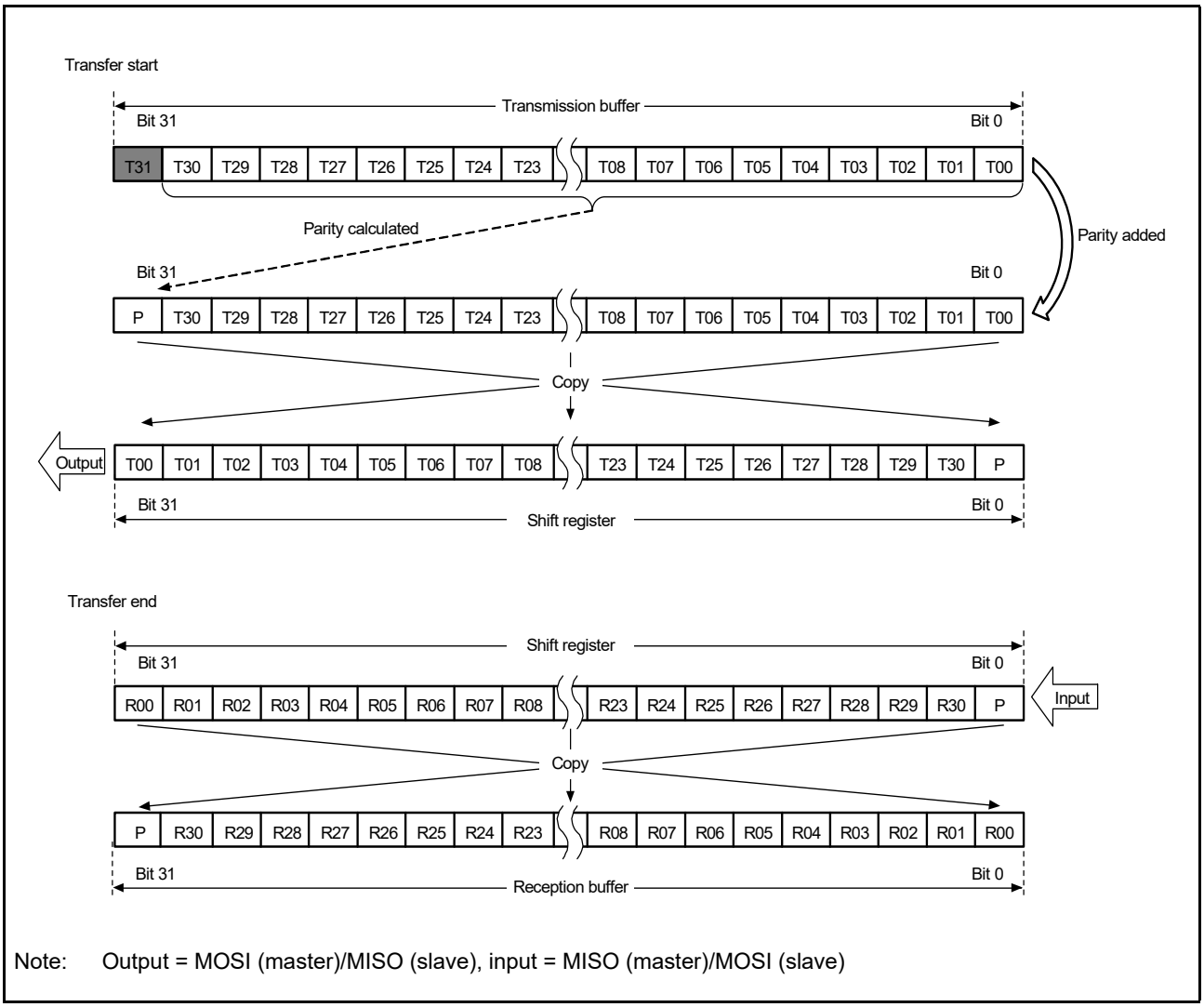
Figure 26.19 MSB First Transfer (24-Bit Data, Parity Enabled)

(3) LSB First Transfer (32-Bit Data)

Figure 26.20 shows details of operations by the RSPI data register (SPDR) and the shift register in transfer with parity enabled, an RSPI data length of 32 bits, and LSB first selected.

In transmission, the value of the parity bit (P) is calculated from bits T30 to T00. This replaces the final bit, T31, and the whole is copied to the shift register. Data are transmitted in the order T00, T01, …, T30, and P.

In reception, received data are shifted in bit by bit through bit 0 of the shift register. When bits R00 to P have been collected after input of the required number of cycles of RSPCK, the value in the shift register is copied to the reception buffer. On copying of data to the shift register, the data from R00 to P are checked by judging the parity.

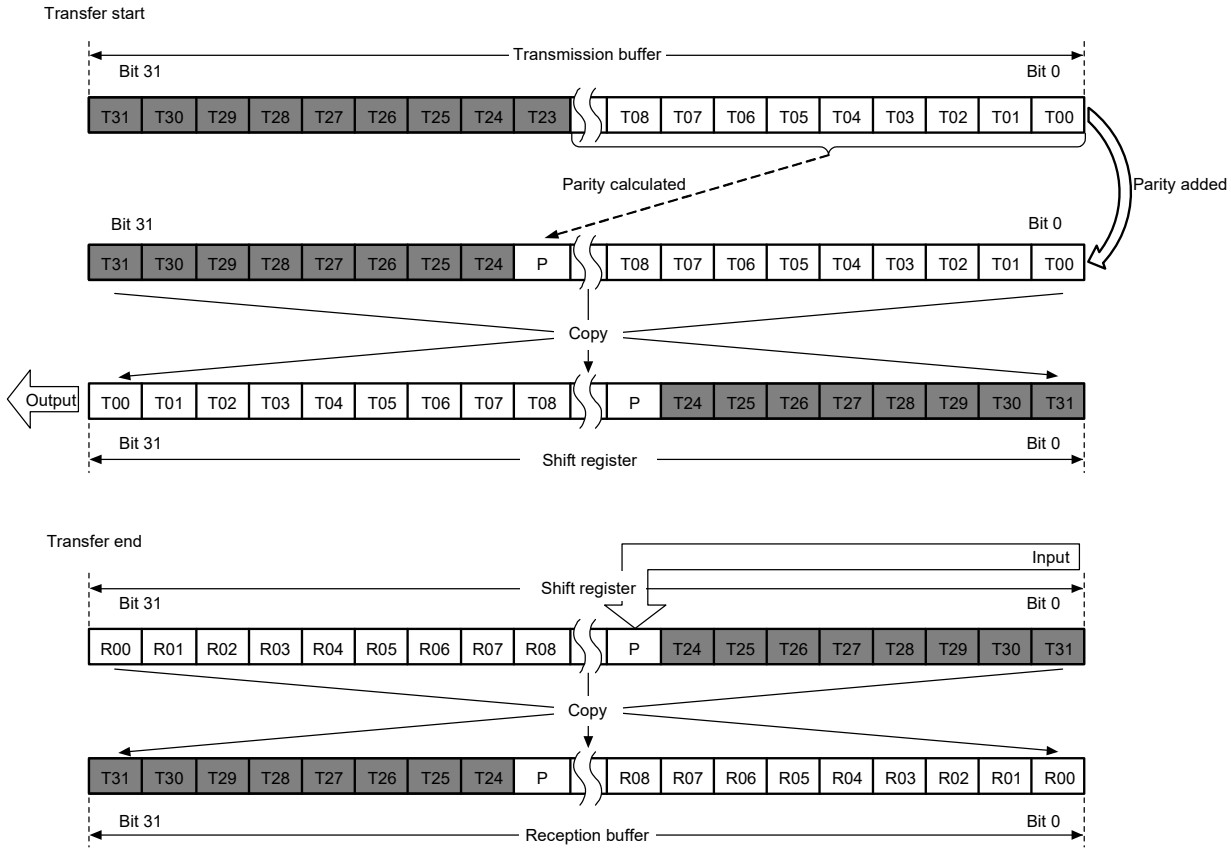


Note: Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

Figure 26.20 LSB First Transfer (32-Bit Data, Parity Enabled)

(4) LSB First Transfer (24-Bit Data)

Figure 26.21 shows details of operations by the RSPI data register (SPDR) and the shift register in transfer with parity enabled, 24 bits as the RSPI data length for an example that is not 32 bits, and LSB first selected.
In transmission, the value of the parity bit (P) is calculated from bits T22 to T00. This replaces the final bit, T23, and the whole is copied to the shift register. Data are transmitted in the order T00, T01, …, T22, and P.
In reception, received data are shifted in bit by bit through bit 8 of the shift register. When bits R00 to P have been collected after input of the required number of cycles of RSPCK, the value in the shift register is copied to the reception buffer. On copying of data to the shift register, the data from R00 to P are checked by judging the parity. At this time, the higher-order 8 bits of the transmission buffer are stored in the higher-order 8 bits of the reception buffer. Writing 0 to bits T31 to T24 at the time of transmission leads to 0 being inserted in the higher-order 8 bits of the reception buffer.



Note: Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

Figure 26.21 LSB First Transfer (24-Bit Data, Parity Enabled)

26.3.5 Transfer Format

26.3.5.1 CPHA = 0

Figure 26.22 shows a sample transfer format for the serial transfer of 8-bit data when the SPCMDm.CPHA bit is 0. Note that clock synchronous operation (the SPCR.SPMS bit is 1) should not be set when the RSPI operates in slave mode (SPCR.MSTR = 0) and the CPHA bit is 0. In Figure 26.22, RSPCKy (CPOL = 0) indicates the RSPCKy signal waveform when the SPCMDm.CPOL bit is 0; RSPCKy (CPOL = 1) indicates the RSPCKy signal waveform when the CPOL bit is 1. The sampling timing represents the timing at which the RSPI fetches serial transfer data into the shift register. The I/O directions of the signals depend on the RSPI settings. For details, refer to section 26.3.2, Controlling RSPI Pins.

When the SPCMDm.CPHA bit is 0, the driving of valid data to the MOSIy and MISOy signals commences at an SSLyi signal assertion timing. The first RSPCKy signal change timing that occurs after the SSLyi signal assertion becomes the first transfer data fetch timing. After this timing, data is sampled at every 1 RSPCK cycle. The change timing for MOSIy and MISOy signals is always 1/2 RSPCK cycles after the transfer data fetch timing. The CPOL bit setting does not affect the RSPCK signal operation timing; it only affects the signal polarity.

t1 denotes a period from an SSLyi signal assertion to RSPCKy oscillation (RSPCK delay). t2 denotes a period from the termination of RSPCKy oscillation to an SSLyi signal negation (SSL negation delay). t3 denotes a period in which SSLyi 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 RSPI system. For a description of t1, t2, and t3 when the RSPI of this LSI is in master mode, refer to section 26.3.10.1, Master Mode Operation.

(i = 0 to 3; m = 0 to 7; y = 0, 1 (for all channels))

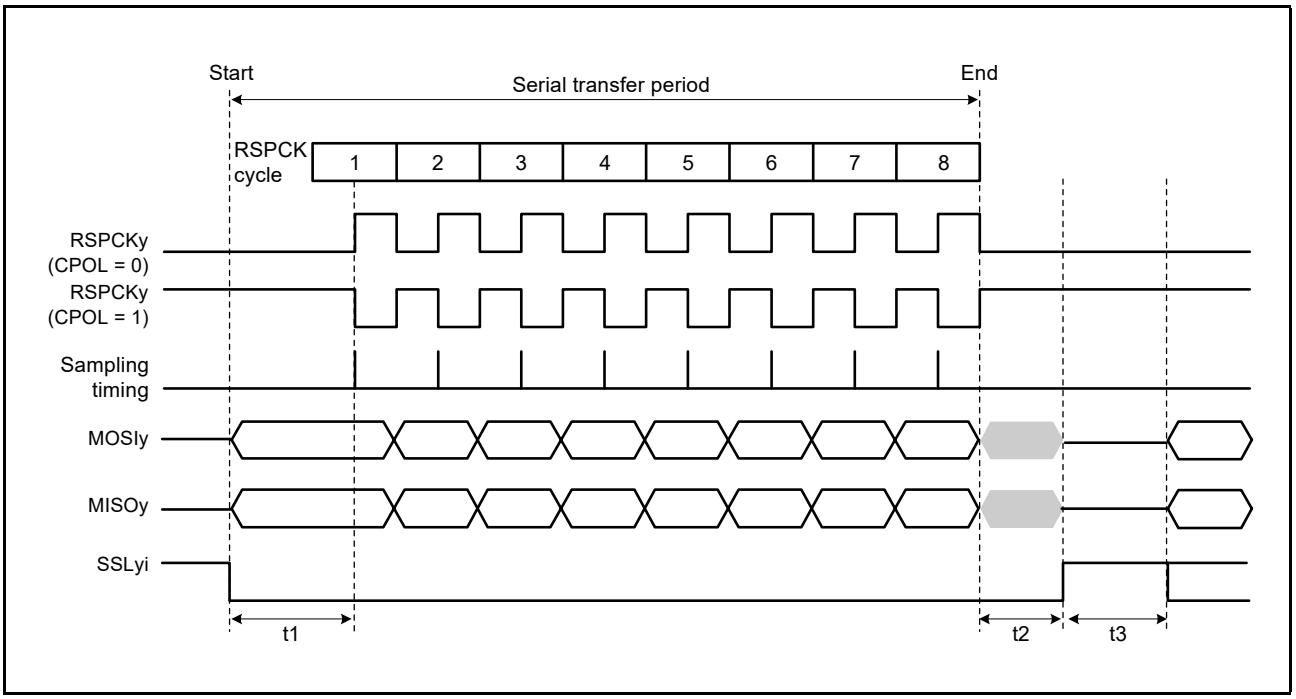


Figure 26.22 RSPI Transfer Format (CPHA = 0)

26.3.5.2 CPHA = 1

Figure 26.23 shows a sample transfer format for the serial transfer of 8-bit data when the SPCMDm.CPHA bit is 1. However, when the SPCR.SPMS bit is 1, the SSLyi signals are not used, and only the three signals RSPCKy, MOSIy, and MISOy handle communications. In Figure 26.23, RSPCK (CPOL = 0) indicates the RSPCKy signal waveform when the SPCMDm.CPOL bit is 0; RSPCK (CPOL = 1) indicates the RSPCKy signal waveform when the CPOL bit is 1. The sampling timing represents the timing at which the RSPI fetches serial transfer data into the shift register. The I/O directions of the signals depend on the RSPI mode (master or slave). For details, refer to section 26.3.2, Controlling RSPI Pins.

When the SPCMDm.CPHA bit is 1, the driving of invalid data to the MISOy signal commences at an SSLyi signal assertion timing. The output of valid data to the MOSIy and MISOy signals commences at the first RSPCKy signal change timing that occurs after the SSLyi signal assertion. After this timing, data is updated at every 1 RSPCK cycle. The transfer data fetch timing is always 1/2 RSPCK cycles after the data update timing. The SPCMDm.CPOL bit setting does not affect the RSPCKy signal operation timing; it only affects 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 the RSPI of this LSI is in master mode, refer to section 26.3.10.1, Master Mode Operation.

(i = 0 to 3; m = 0 to 7; y = 0, 1 (for all channels))

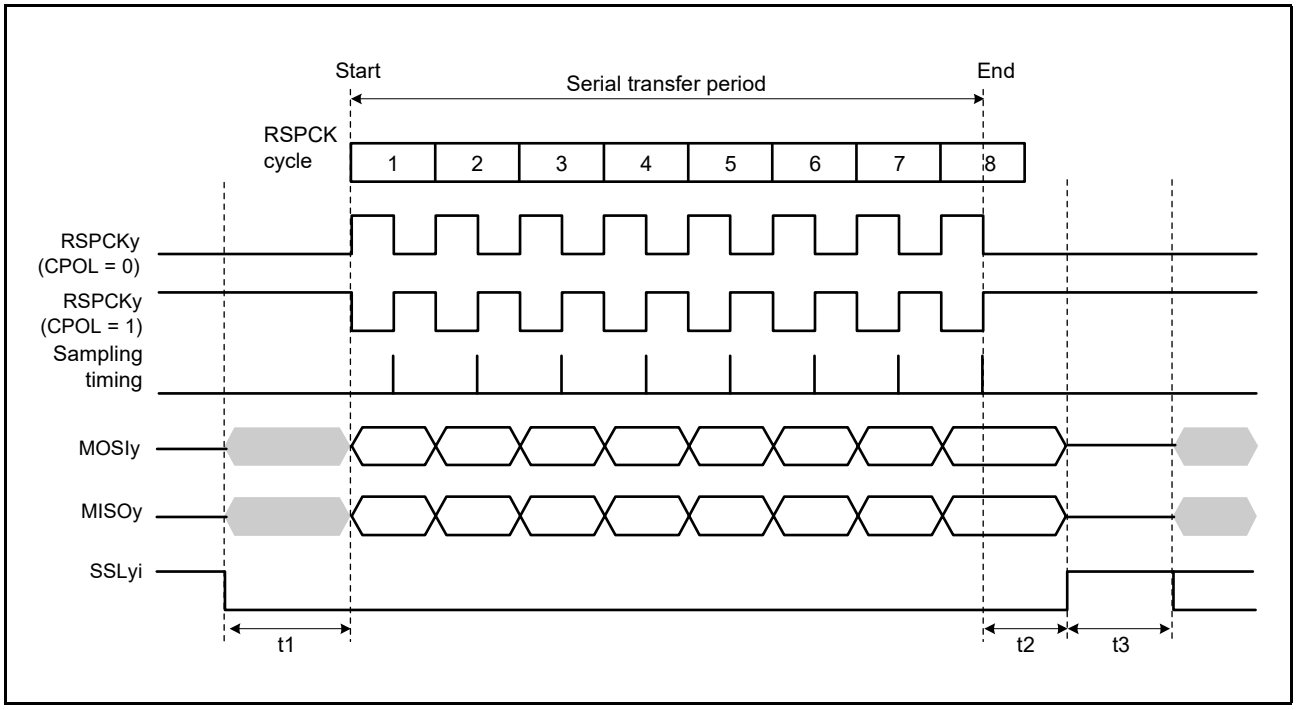


Figure 26.23 RSPI Transfer Format (CPHA = 1)

26.3.6 Communications Operating Mode

Full-duplex synchronous serial communications or transmit operations only can be selected by the communications operating mode select bit (SPCR.TXMD). The SPDR access shown in Figure 26.24 and Figure 26.25 indicates the condition of access to the SPDR register, where W denotes a write cycle.

26.3.6.1 Full-Duplex Synchronous Serial Communications (SPCR.TXMD = 0)

Figure 26.24 shows an example of operation when the communications operating mode select bit (SPCR.TXMD) is set to 0. In the example in Figure 26.24, the RSPI performs an 8-bit serial transfer in which the SPDCR.SPFC[1:0] bits are 00b, the SPCMDm.CPHA bit is 1, and the SPCMDm.CPOL bit is 0. The numbers given under the RSPCKy waveform represent the number of RSPCK cycles (i.e., the number of transferred bits) (m = 0 to 7; y = 0, 1 (for all channels)).

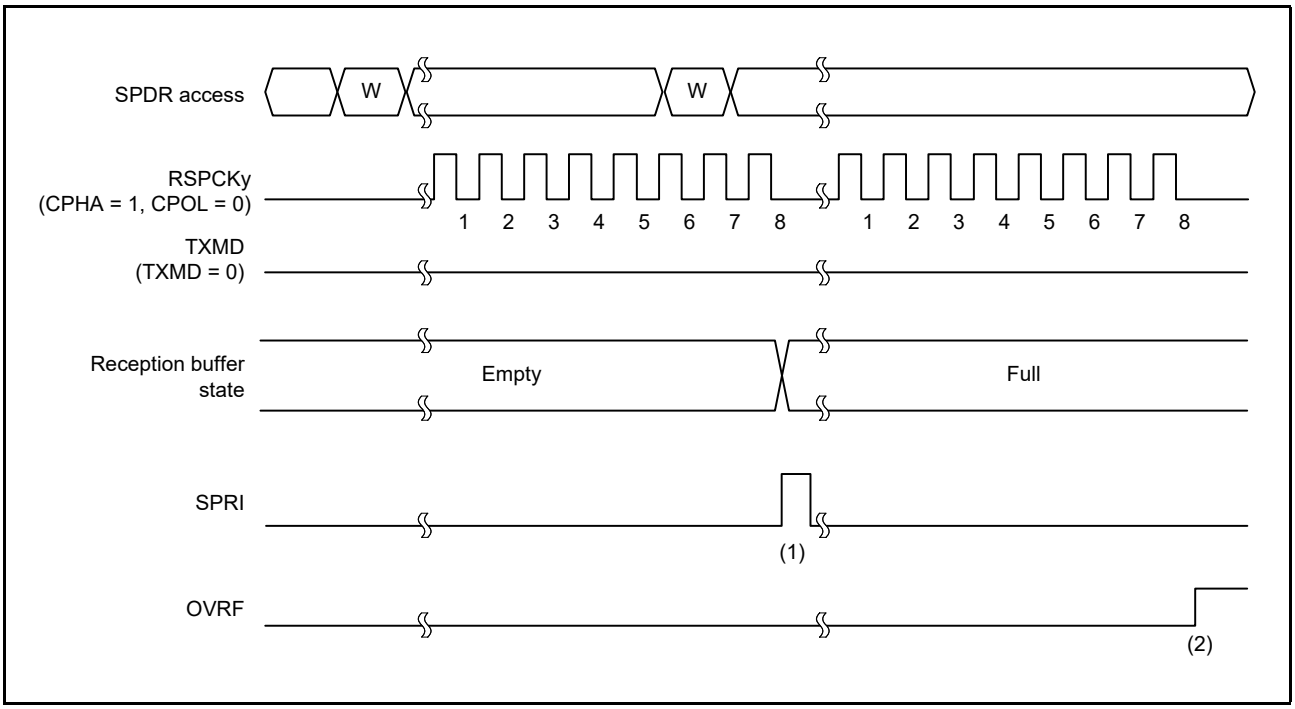


Figure 26.24 Operation Example of SPCR.TXMD = 0

The operation of the flags at timings shown in steps (1) and (2) in the figure is described below.

(1) When a serial transfer ends with the reception buffer of SPDR empty, the RSPI generates a reception buffer full interrupt request (SPRI) and copies the received data in the shift register to the reception buffer.

(2) When a serial transfer ends with the reception buffer of SPDR holding data that was received in the previous serial transfer, the RSPI sets the SPSR.OVRF flag to 1 and discards the received data in the shift register.

26.3.6.2 Transmit Operations Only (SPCR.TXMD = 1)

Figure 26.25 shows an example of operation when the communications operating mode select bit (SPCR.TXMD) is set to 1. In the example in Figure 26.25, the RSPI performs an 8-bit serial transfer in which the SPDCR.SPFC[1:0] bits are 00b, the SPCMDm.CPHA bit is 1, and the SPCMDm.CPOL bit is 0. The numbers given under the RSPCKy waveform represent the number of RSPCK cycles (i.e., the number of transferred bits) (m = 0 to 7; y = 0, 1 (for all channels)).

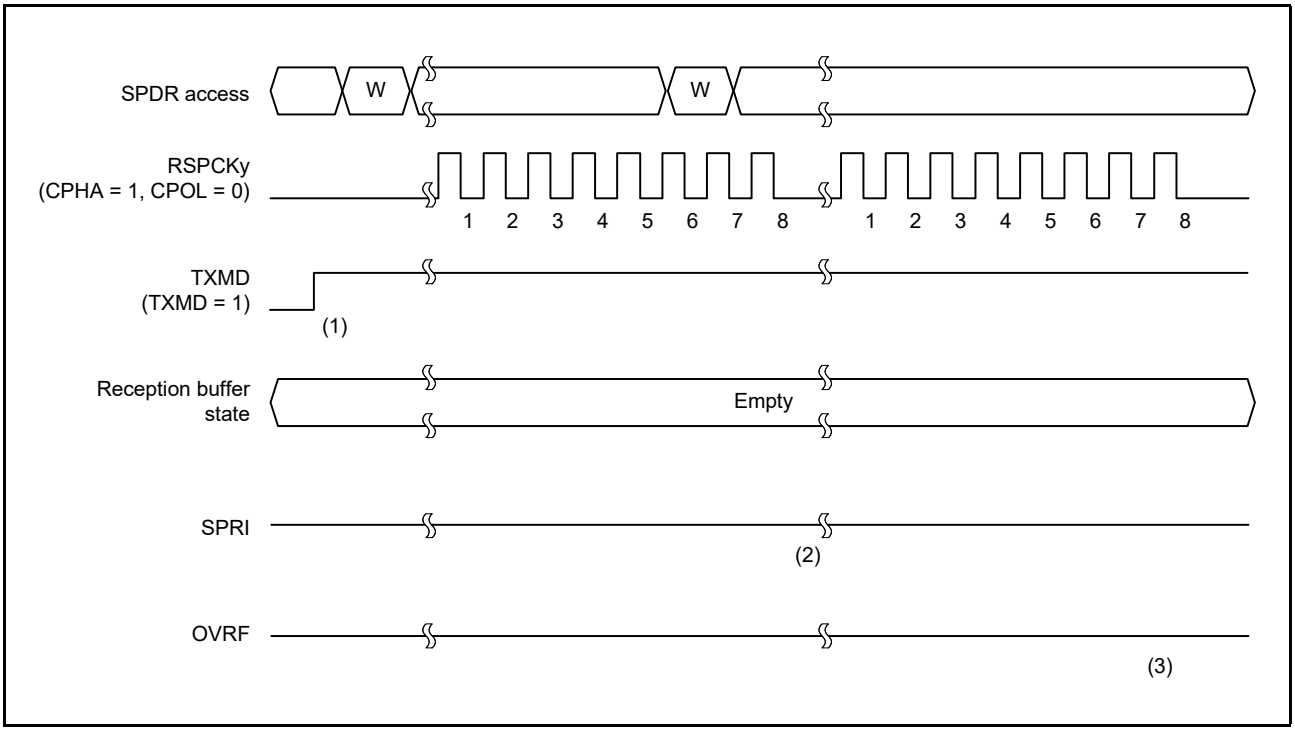


Figure 26.25 Operation Example of SPCR.TXMD = 1

The operation of the flags at timings shown in steps (1) to (3) in the figure is described below.

(1) Make sure there is no data left in the reception buffer and the SPSR.OVRF flag is 0 before entering the mode of transmit operations only (SPCR.TXMD = 1).

(2) When a serial transfer ends with the reception buffer of SPDR empty, if the mode of transmit operations only is selected (SPCR.TXMD = 1), the RSPI does not copy the data in the shift register to the reception buffer.

(3) Since the reception buffer of SPDR does not hold data that was received in the previous serial transfer, even when a serial transfer ends, the SPSR.OVRF flag retains the value of 0, and the data in the shift register is not copied to the reception buffer.

When performing transmit operations only (SPCR.TXMD = 1), the RSPI transmits transmit data but does not receive received data. Therefore, the SPSR.OVRF flag remains 0 at the timings of (1) to (3).

26.3.7 Transmission Buffer Empty/Reception Buffer Full Interrupts

Figure 26.26 shows an example of operation of the transmission buffer empty interrupt (SPTI) and the reception buffer full interrupt (SPRI). The SPDR register access shown in Figure 26.26 indicates the condition of access to the SPDR register, where W denotes a write cycle, and R a read cycle. In the example in Figure 26.26, the RSPI performs an 8-bit serial transfer in which the SPCR.TXMD bit is 0, the SPDCR.SPFC[1:0] bits are 00b, the SPCMDm.CPHA bit is 1, and the SPCMDm.CPOL bit is 0. The numbers given under the RSPCKy waveform represent the number of RSPCK cycles (i.e., the number of transferred bits) (m = 0 to 7; y = 0, 1 (for all channels)).

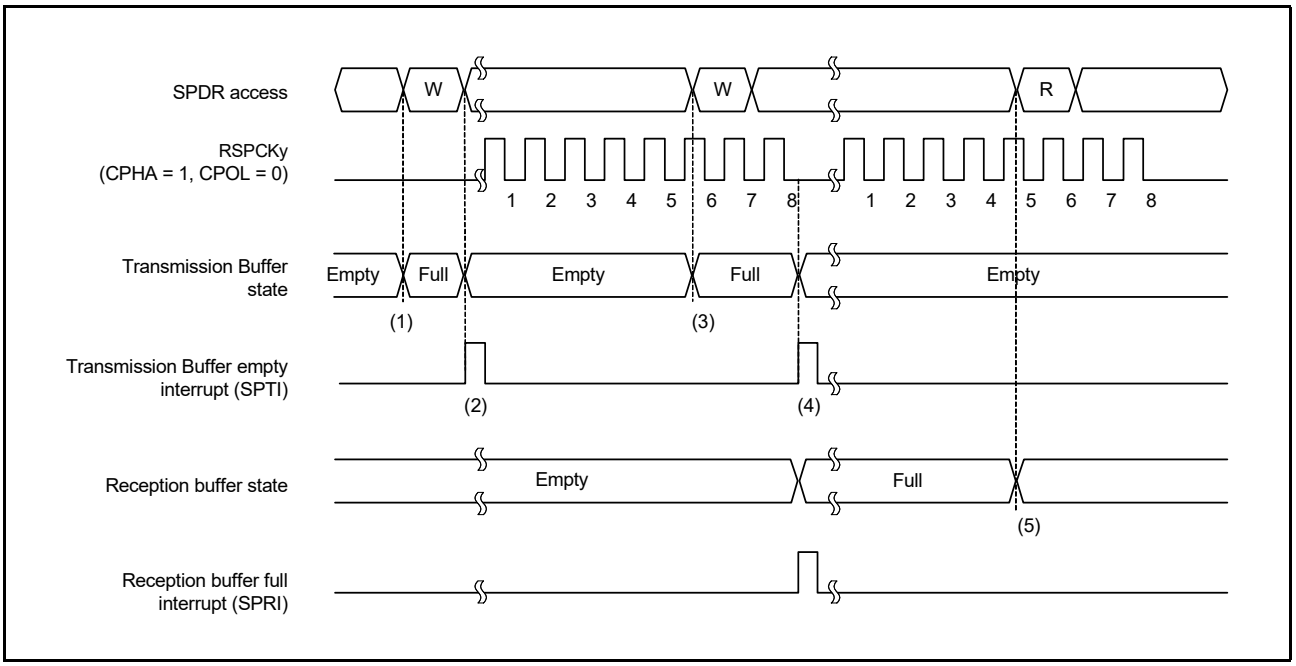


Figure 26.26 Operation Example of Transmission Buffer Empty Interrupt (SPTI) and Reception Buffer Full Interrupt (SPRI)

The operation of the interrupts at timings shown in steps (1) to (5) in the figure is described below.

1. When transmit data is written to SPDR when the transmission buffer of SPDR is empty (data for the next transfer is not set), the RSPI writes data to the transmission buffer.
2. If the shift register is empty, the RSPI copies the data in the transmission buffer to the shift register and generates a transmission buffer empty interrupt request (SPTI). How a serial transfer is started depends on the mode of the RSPI. For details, refer to section 26.3.10, SPI Operation, and section 26.3.11, Clock Synchronous Operation.
3. When transmit data is written to SPDR by the transmission buffer empty interrupt routine, the data is transferred to the transmission buffer. Because the data being transferred serially is stored in the shift register, the RSPI does not copy the data in the transmission buffer to the shift register.
4. When the serial transfer ends with the reception buffer of SPDR being empty, the RSPI copies the receive data in the shift register to the reception buffer and generates a reception buffer full interrupt request (SPRI). Since the shift register becomes empty upon completion of serial transfer, when the transmission buffer had been full before the serial transfer ended, the RSPI copies the data in the transmission buffer to the shift register. Even when received data is not copied from the shift register to the reception buffer in an overrun error status, upon completion of the serial transfer, the RSPI determines that the shift register is empty, thus data transfer from the transmission buffer to the shift register is enabled.
5. When SPDR is read by the reception buffer full interrupt routine, the receive data can be read.

If SPDR is written to when the transmission buffer holds data that has not yet been transmitted, the RSPI does not update the data in the transmission buffer. When writing to SPDR, make sure to use a transmission buffer empty interrupt request. To use a transmission buffer empty interrupt, set the SPTIE bit in SPCR to 1.
If the RSPI function is disabled (the SPCR.SPE bit being 0), set the SPTIE bit to 0.
When serial transfer ends with the reception buffer being full, the RSPI does not copy data from the shift register to the reception buffer, and detects an overrun error (refer to section 26.3.8, Error Detection). To prevent a receive data overrun error, read the received data using a reception buffer full interrupt request before the next serial transfer ends. To use an RSPI reception buffer full interrupt, set the SPCR.SPRIE bit to 1.
For the states of the transmit and reception buffers, transmission buffer empty and reception buffer interrupts or the corresponding IRQ status register (IRQSn) can be used to confirm the generation of interrupt requests. For the IRQ status register (IRQSn), see section 12.4.2.1, IRQ Status Register n (IRQSn) (n = 0 to 9).

26.3.8 Error Detection

In the normal RSPI serial transfer, the data written to the transmission buffer of SPDR is transmitted, and the received data can be read from the reception buffer of SPDR. If access is made to SPDR, depending on the status of the transmission/reception buffer or the status of the RSPI at the beginning or end of serial transfer, in some cases non-normal transfers can be executed.
If a non-normal transfer operation occurs, the RSPI detects the event as an overrun error, parity error, or mode fault error. Table 26.8 lists the relationship between non-normal transfer operations and the RSPI's error detection function.

Table 26.8 Relationship between Non-Normal Transfer Operations and RSPI Error Detection Function

	Occurrence Condition	RSPI Operation	Error Detection
1	SPDR is written when the transmission buffer is full.	• The contents of the transmission buffer are kept. • Missing write data.	None
2	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 transmitted.	None
3	SPDR is read when the reception buffer is empty.	Previously received data is output.	None
4	Serial transfer terminates when the reception buffer is full.	• The contents of the reception buffer are kept. • Missing receive data.	Overrun error
5	An incorrect parity bit is received when performing full-duplex synchronous serial communications with the parity function enabled.	The parity error flag is asserted.	Parity error
6	The SSLy0 input signal is asserted when the serial transfer is idle in multi-master mode.	• Driving of the RSPCKy, MOSIy, SSLy1 to SSLy3 output signals is stopped. • RSPI function is disabled.	Mode fault error
7	The SSLy0 input signal is asserted during serial transfer in multi-master mode.	• Serial transfer is suspended. • Missing transmit/receive data. • Driving of the RSPCKy, MOSIy, SSLy1 to SSLy3 output signals is stopped. • RSPI function is disabled.	Mode fault error
8	The SSLy0 input signal is negated during serial transfer in slave mode.	• Serial transfer is suspended. • Missing transmit/receive data. • Driving of the MISOy output signal is stopped. • RSPI function is disabled.	Mode fault error

y = 0, 1 (for all channels)

On operation 1 described in Table 26.8, the RSPI does not detect an error. To prevent data omission during the writing to SPDR, write operations to SPDR should be executed using a transmission buffer empty interrupt request.
Likewise, the RSPI does not detect an error on operation 2. In a serial transfer that was started before the shift register was updated, the RSPI sends the data that was received in the previous serial transfer, and does not treat the operation indicated in 2 as an error. Note that the received data from the previous serial transfer is retained in the reception buffer of SPDR, thus it can be correctly read (if SPDR is not read before the end of the serial transfer, an overrun error may occur).
Similarly, the RSPI does not detect an error on operation 3. To prevent extraneous data from being read, SPDR read operation should be executed using an RSPI reception buffer full interrupt request.
An overrun error shown in 4 is described in section 26.3.8.1, Overrun Error. A parity error shown in 5 is described in section 26.3.8.2, Parity Error. A mode fault error shown in 6 to 8 is described in section 26.3.8.3, Mode Fault Error. For the transmit and receive interrupts, refer to section 26.3.7, Transmission Buffer Empty/Reception Buffer Full Interrupts.

26.3.8.1 Overrun Error

If a serial transfer ends when the reception buffer of SPDR is full, the RSPI detects an overrun error, and sets the OVRF flag in SPSR to 1. When the OVRF flag is 1, the RSPI does not copy data from the shift register to the reception buffer so that the data prior to the occurrence of the error is retained in the reception buffer. To set the OVRF flag to 0, write 0 to the OVRF flag after the CPU has read SPSR with the OVRF flag set to 1.

Figure 26.27 shows an example of operation of the OVRF flag. The SPSR and SPDR accesses shown in Figure 26.27 indicate the condition of accesses to SPSR and SPDR, respectively, where W denotes a write cycle, and R a read cycle. In the example in Figure 26.27, the RSPI performs an 8-bit serial transfer in which the SPCMDm.CPHA bit is 1 and the SPCMDm.CPOL bit is 0. The numbers given under the RSPCKy waveform represent the number of RSPCK cycles (i.e., the number of transferred bits) (m = 0 to 7; y = 0, 1 (for all channels)).

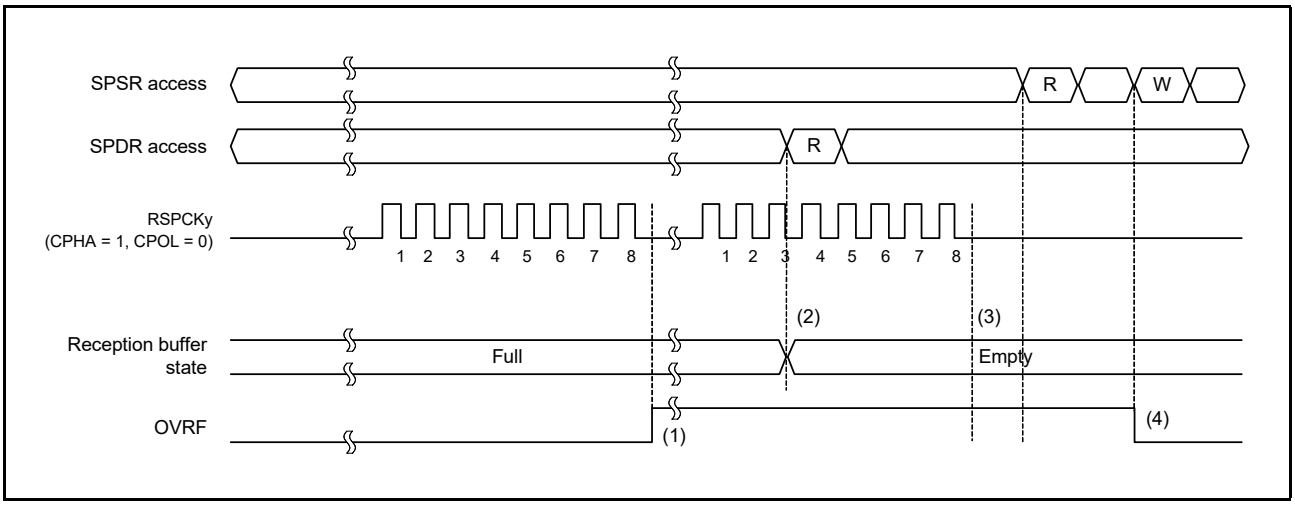


Figure 26.27 Operation Example of OVRF Flag

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 with the reception buffer full, the RSPI detects an overrun error, and sets the OVRF flag to 1. The RSPI does not copy the data in the shift register to the reception buffer. Even if the SPPE bit is 1, parity errors are not detected. In master mode, the RSPI copies the pointer value to SPCMDm register to the SPSSR.SPECM[2:0] bits.
2. When SPDR is read, the RSPI can read the data in the reception buffer. The reception buffer becoming empty does not set the OVRF flag to 0.
3. If the serial transfer ends with the OVRF flag being 1 (an overrun error occurs), the RSPI does not copy the data in the shift register to the reception buffer. A reception-buffer interrupt is not generated. Even if the SPPE bit is 1, parity errors are not detected. When in master mode, the RSPI does not update the SPSSR.SPECM[2:0] bits. When in an overrun error state and the RSPI does not copy the received data from the shift register to the reception buffer, upon termination of the serial transfer, the RSPI determines that the shift register is empty; in this manner, data transfer from the transmission buffer to the shift register is enabled.
4. If the value 0 is written to the OVRF flag after SPSR is read when the OVRF flag is 1, the OVRF flag is set to 0.

The occurrence of an overrun can be checked either by reading SPSR or by using an RSPI error interrupt and reading SPSR. When executing a serial transfer, measures should be taken to ensure the early detection of overrun errors, such as reading SPSR immediately after SPDR is read. When the RSPI is used in master mode, the pointer value to SPCMDm register at the occurrence of the error can be checked by reading the SPSSR.SPECM[2:0] bits.

If an overrun error occurs and the OVRF flag is set to 1, normal reception operations cannot be performed until the OVRF flag is set to 0.

When the RSPCK auto-stop function is enabled in master mode, an overrun error does not occur. Figure 26.28 and Figure 26.29 show the clock stop waveform when a serial transfer continues while the reception buffer is full in master mode.

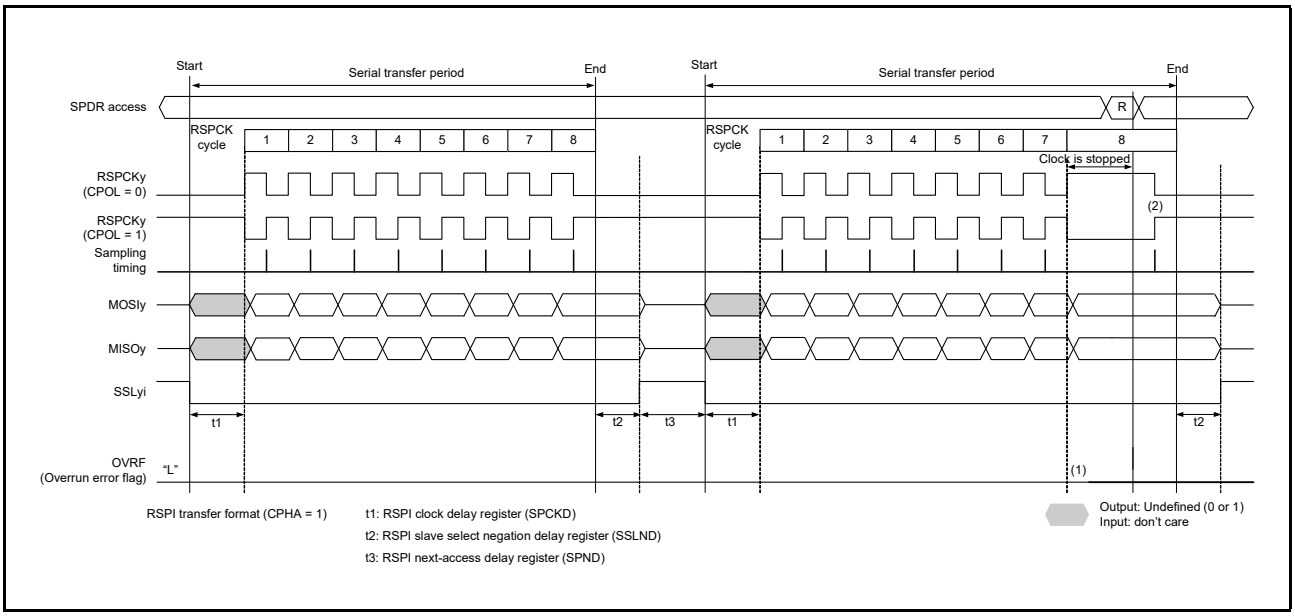


Figure 26.28 Clock Stop Waveform When a Serial Transfer Continues While the Reception Buffer is Full in Master Mode (CPHA = 1)

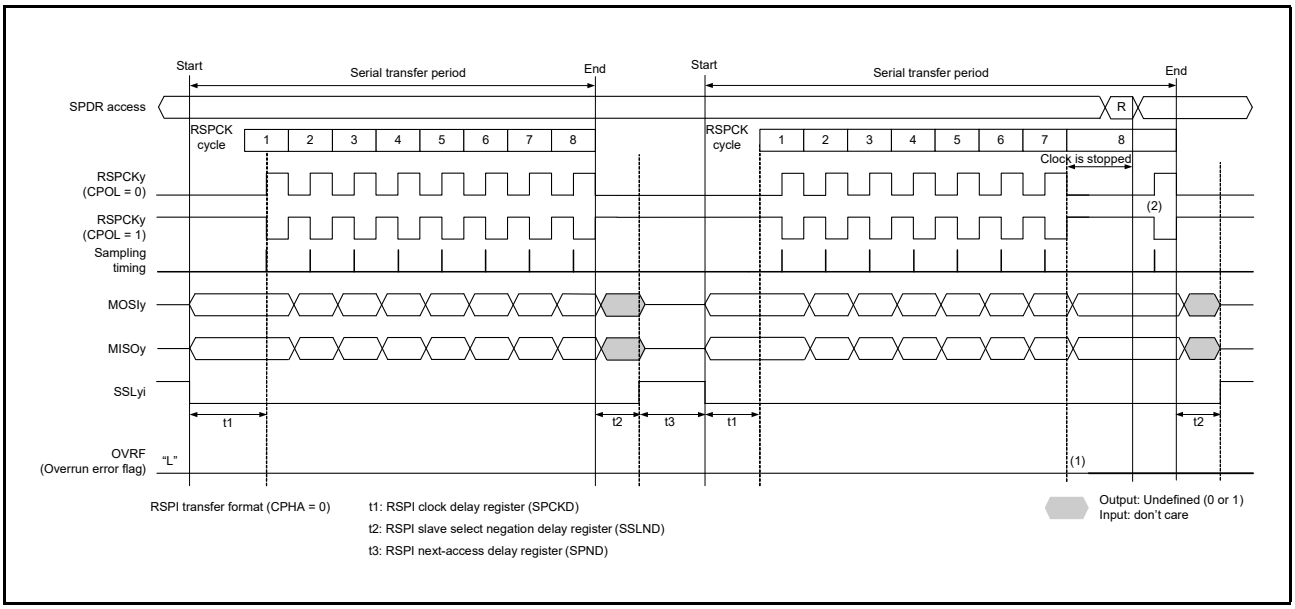


Figure 26.29 Clock Stop Waveform When a Serial Transfer Continues While the Reception Buffer is Full in Master Mode (CPHA = 0)

The operation of the flags at the timings shown in steps (1) and (2) in the figure is described below.

(1) When the reception buffer is full, an overrun error to stop the reception buffer does not occur.

(2) If SPDR is read while the clock is stopped, data in the reception buffer can be read. The RSPCK clock restarts after reading the reception buffer.

26.3.8.2 Parity Error

If full-duplex synchronous serial communications is performed with the SPCR.TXMD bit set to 0 and the SPCR2.SPPE bit set to 1, when serial transfer ends, the RSPI checks whether there are parity errors. Upon detecting a parity error in the received data, the RSPI sets the SPSR.PERF flag to 1. Since the RSPI does not copy the data in the shift register to the reception buffer when the SPSR.OVRF flag is set to 1, parity error detection is not performed for the received data. To set the PERF flag to 0, write 0 to the PERF flag after SPSR register is read with the PERF flag set to 1.
Figure 26.30 shows an example of operation of the OVRF and PERF flags. The SPSR access shown in Figure 26.30 indicates the condition of access to SPSR register, where W denotes a write cycle, and R a read cycle. In the example of Figure 26.30, full-duplex synchronous serial communications is performed while the SPCR.TXMD bit is 0 and the SPCR2.SPPE bit is 1. The RSPI performs an 8-bit serial transfer in which the SPCMDm.CPHA bit is 1 and the SPCMDm.CPOL bit is 0. The numbers given under the RSPCKy waveform represent the number of RSPCK cycles (i.e., the number of transferred bits) (m = 0 to 7; y = 0, 1 (for all channels)).

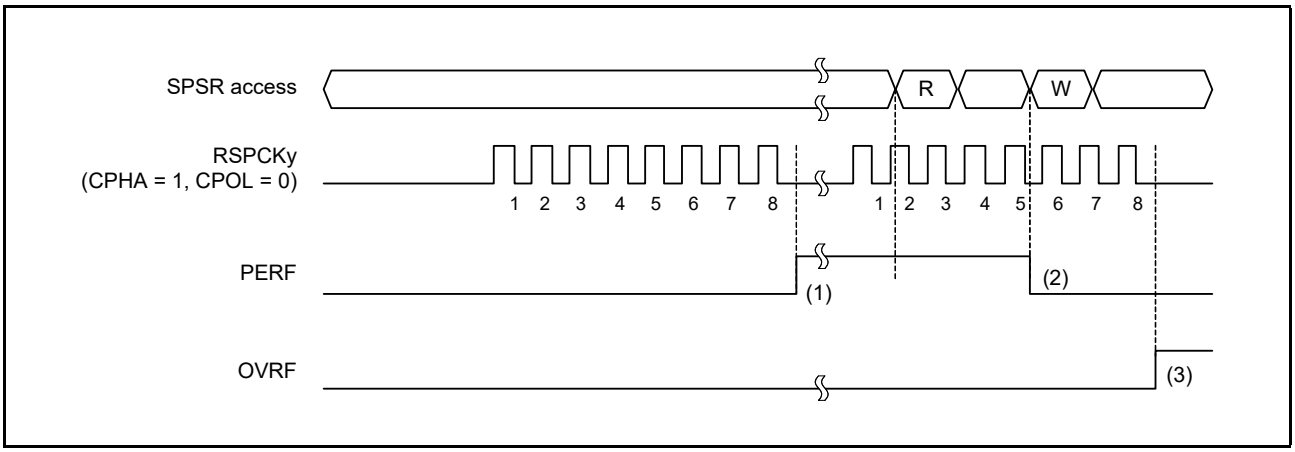


Figure 26.30 Operation Example of PERF Flag

The operation of the flags at the timing shown in steps (1) to (3) in the figure is described below.

1. If a serial transfer terminates with the RSPI not detecting an overrun error, the RSPI copies the data in the shift register to the reception buffer. The RSPI judges the received data at this timing, and sets the PERF flag to 1 if a parity error is detected. In master mode, the RSPI copies the pointer value to SPCMDm register to the SPSSR.SPECM[2:0] bits.
2. If the value 0 is written to the PERF flag after SPSR register is read when the PERF flag is 1, the PERF flag is set to 0.
3. When the RSPI detects an overrun error and serial transfer is terminated, the data in the shift register is not copied to the reception buffer. The RSPI does not perform parity error detection at this timing.

The occurrence of a parity error can be checked either by reading SPSR register or by using an RSPI error interrupt and reading SPSR register. When executing a serial transfer, measures should be taken to ensure the early detection of parity errors, such as reading SPSR. When the RSPI is used in master mode, the pointer value to SPCMDm register at the occurrence of the error can be checked by reading the SPSSR.SPECM[2:0] bits.

26.3.8.3 Mode Fault Error

The RSPI operates in multi-master mode when the SPCR.MSTR bit is 1, the SPCR.SPMS bit is 0, and the SPCR.MODFEN bit is 1. If the active level is input with respect to the SSLy0 input signal of the RSPI in multi-master mode, the RSPI detects a mode fault error irrespective of the status of the serial transfer, and sets the MODF bit in the RSPI status register (SPSR) to 1. Upon detecting the mode fault error, the RSPI copies the value of the pointer to SPCMDm to the SPSSR.SPECM[2:0] bits. The active level of the SSLy0 signal is determined by the SSL0P bit in the RSPI slave select polarity register (SSLP).

When the MSTR bit is 0, the RSPI operates in slave mode. The RSPI detects a mode fault error if the MODFEN bit in the RSPI in slave mode is 1, and the SPMS bit is 0, and if the SSLy0 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).

Upon detecting a mode fault error, the RSPI stops driving of the output signals and clears the SPCR.SPE bit to 0 (refer to section 26.3.9, Initializing RSPI). In the case of multi-master configuration, detection of a mode fault error is used to stop driving of the output signals and the RSPI function, which allows the master right to be released.

The occurrence of a mode fault error can be checked either by reading SPSR or by using an RSPI error interrupt and reading SPSR. Detecting mode-fault errors without utilizing the RSPI error interrupt requires polling of SPSR. When using the RSPI in master mode, the pointer value to SPCMDm register at the occurrence of the error can be checked by reading the SPSSR.SPECM[2:0] bits.

When the MODF bit is 1, writing of the value 1 to the SPE bit is ignored by the RSPI. To enable the RSPI function after the detection of a mode fault error, the MODF bit must be set to 0 (m = 0 to 7; y = 0, 1 (for all channels)).

26.3.9 Initializing RSPI

If the value 0 is written to the SPCR.SPE bit or the RSPI sets the SPE bit to 0 because of the detection of a mode fault error, the RSPI disables the RSPI function, and initializes some of the module functions. When a system reset is generated, the RSPI initializes all of the module functions. The following describes initialization by the clearing of the SPCR.SPE bit and initialization by a system reset.

26.3.9.1 Initialization by Clearing the SPE Bit

When the SPCR.SPE bit is set to 0, the RSPI 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 of the RSPI
- Initializing the transmission buffer of the RSPI

Initialization by the clearing of the SPE bit does not initialize the control bits of the RSPI. For this reason, the RSPI can be started in the same transfer mode as prior to the initialization if the SPE bit is set to 1 again.
The SPSR.OVRF and SPSR.MODF flags are not initialized, nor is the value of the RSPI sequence status register (SPSSR) initialized. For this reason, even after the RSPI is initialized, data from the reception buffer can be read in order to check the status of error occurrence during an RSPI transfer.
The transmission buffer is initialized to an empty state. Therefore, if the SPCR.SPTIE bit is set to 1 after RSPI initialization, a transmission buffer empty interrupt is generated. When the RSPI is initialized, in order to disable any transmission buffer empty interrupt, the value 0 should be written to the SPTIE bit simultaneously with the writing of the value 0 to the SPE bit. To disable any transmission buffer empty interrupt after a mode fault error is detected, use an error handling routine to write the value 0 to the SPTIE bit.

26.3.9.2 System Reset

The initialization by a system reset completely initializes the RSPI through the initialization of all bits for controlling the RSPI, initialization of the status bits, and initialization of data registers, in addition to the requirements described in section 26.3.9.1, Initialization by Clearing the SPE Bit. For details, see section 6, Reset.

26.3.10 SPI Operation

26.3.10.1 Master Mode Operation

The only difference between single-master mode operation and multi-master mode operation lies in mode fault error detection (refer to section 26.3.8, Error Detection). When operating in single-master mode, the RSPI does not detect mode fault errors whereas the RSPI running in multi-master mode does detect mode fault errors. This section explains operations that are common to single-master mode and multi-master mode.

(1) Starting a Serial Transfer

The RSPI updates the data in the transmission buffer (SPTX) when data is written to the RSPI data register (SPDR) with the RSPI transmission buffer being empty (data for the next transfer is not set). When the shift register is empty after the number of frames set in the SPDCR.SPFC[1:0] bits are written to the SPDR, the RSPI copies data from the transmission buffer to the shift register and starts serial transfer. Upon copying transmit data to the shift register, the RSPI changes the status of the shift register to “full”, and upon termination of serial transfer, it changes the status of the shift register to “empty”. The status of the shift register cannot be referenced.
For details on the RSPI transfer format, refer to section 26.3.5, Transfer Format. The polarity of the SSLyi output pins depends on the SSLP register settings (i = 0 to 3; y = 0, 1 (for all channels)).

(2) Terminating a Serial Transfer

Irrespective of the SPCMDm.CPHA bit the RSPI terminates the serial transfer after transmitting an RSPCKy edge corresponding to the final sampling timing. If free space is available in the reception buffer (SPRX), upon termination of serial transfer, the RSPI copies data from the shift register to the reception buffer of the RSPI data register (SPDR).
It should be noted that the final sampling timing varies depending on the bit length of transfer data. In master mode, the RSPI data length depends on the SPCMDm.SPB[3:0] bit setting. The polarity of the SSLyi output pin depends on the SSLP register settings. For details on the RSPI transfer format, refer to section 26.3.5, Transfer Format.
(i = 0 to 3; m = 0 to 7; y = 0, 1 (for all channels))

(3) Sequence Control

The transfer format that is employed in master mode is determined by SPSCR, SPCMDm, SPBR, SPCKD, SSLND, and SPND registers.

SPSCR is a register used to determine the sequence configuration for serial transfers that are executed by the RSPI in master mode. The following items are set in SPCMDm register: SSLyi pin 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, an RSPI 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, the RSPI makes up a sequence comprised of a part or all of SPCMDm register. The RSPI contains a pointer to the SPCMDm register that makes up the sequence. The value of this pointer can be checked by reading the SPSSR.SPCP[2:0] bits. When the SPCR.SPE bit is set to 1 and the RSPI function is enabled, the RSPI loads the pointer to the commands in SPCMD0, and incorporates the SPCMD0 settings into the transfer format at the beginning of serial transfer. The RSPI 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, the RSPI sets the pointer in SPCMD0, and in this manner the sequence is executed repeatedly.

(i = 0 to 3; m = 0 to 7; y = 0, 1 (for all channels))

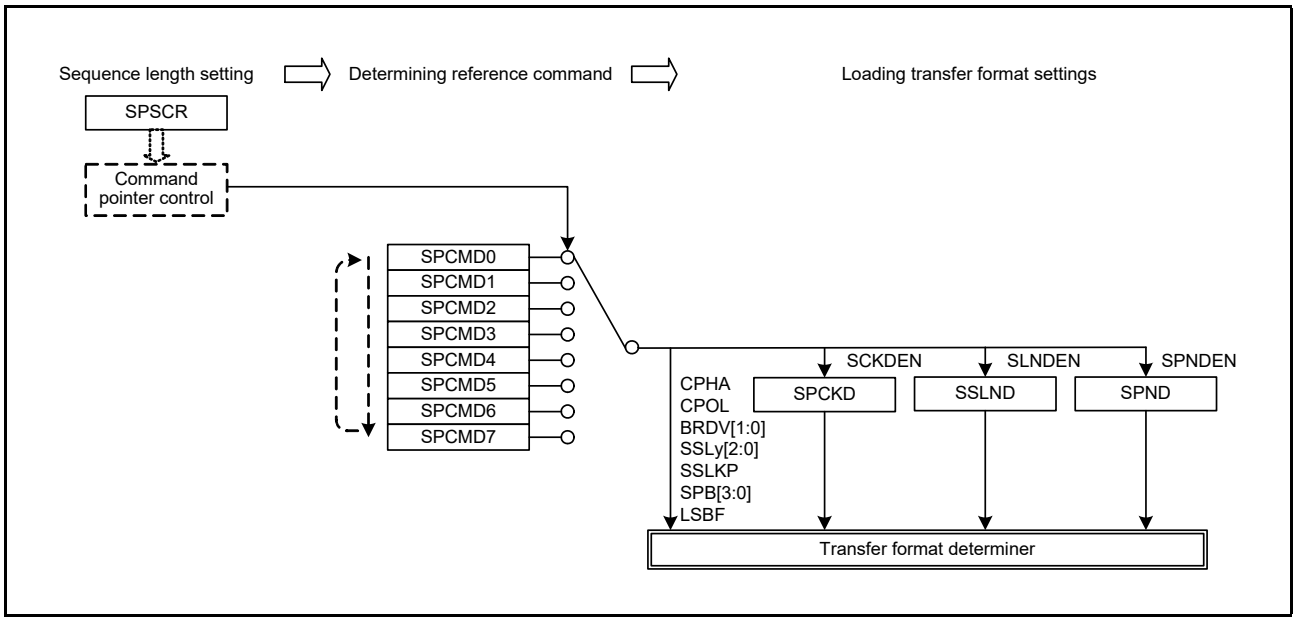


Figure 26.31 Procedure for Determining the Form of Serial Transfer in Master Mode

In this section, a frame is the combination of the data (SPDR) and the settings (SPCMDm).

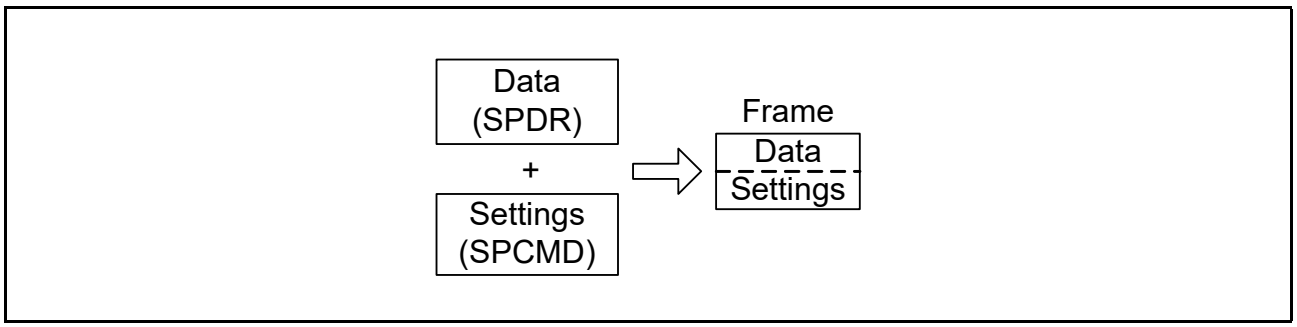


Figure 26.32 Concept of a Frame

Figure 26.33 shows the relationship between the command and the transmit and reception buffers in the sequence of operations specified by the settings in Table 26.4.

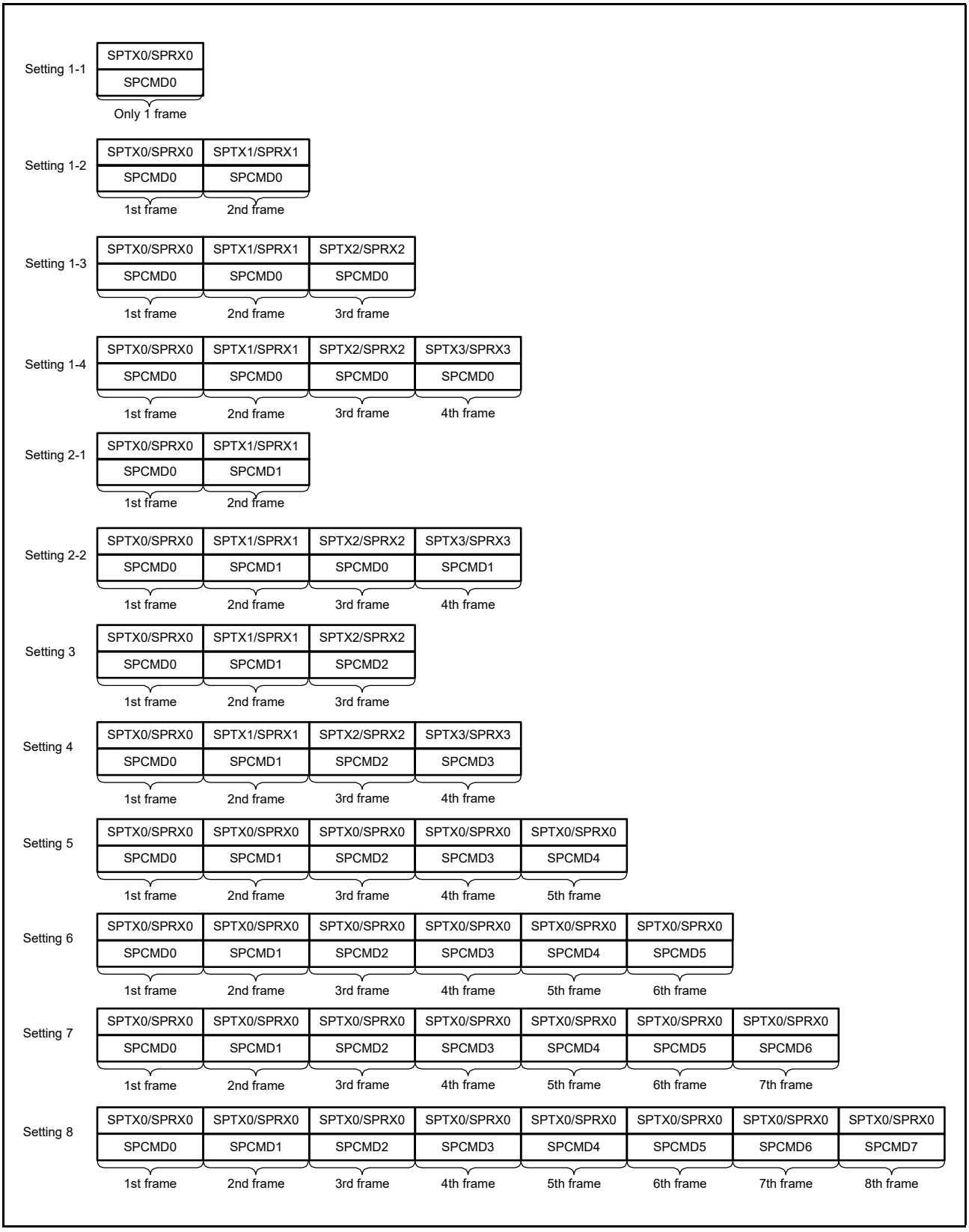


Figure 26.33 Correspondence between the RSPI Command Register and Transmission/Reception Buffers in Sequence Operations

(4) Burst Transfer

If the SPCMDm.SSLKP bit that the RSPI references during the current serial transfer is 1, the RSPI keeps the SSLyi signal level during the serial transfer until the beginning of the SSLyi signal assertion for the next serial transfer. If the SSLyi signal level for the next serial transfer is the same as the SSLyi signal level for the current serial transfer, the RSPI can execute continuous serial transfers while keeping the SSLyi signal assertion status (burst transfer).
Figure 26.34 shows an example of an SSLyi signal operation for the case where a burst transfer is implemented using SPCMD0 and SPCMD1 register settings. The text below explains the RSPI operations (1) to (7) as shown in Figure 26.34. It should be noted that the polarity of the SSLyi output signal depends on the SSLP register settings.

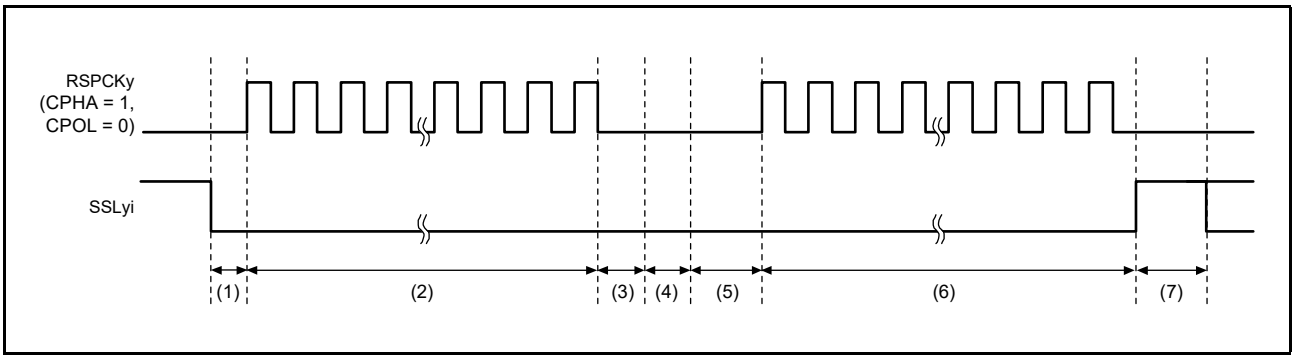


Figure 26.34 Example of Burst Transfer Operation Using SSLKP Bit

(1) Based on SPCMD0, the RSPI asserts the SSLyi signal and inserts RSPCK delays.
(2) The RSPI executes serial transfers according to SPCMD0.
(3) The RSPI inserts SSL negation delays.
(4) Since the SPCMD0.SSLKP bit is 1, the RSPI keeps the SSLyi 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 the transmit data is stored in the shift register for the next transfer.
(5) Based on SPCMD1, the RSPI asserts the SSLyi signal and inserts RSPCK delays.
(6) The RSPI executes serial transfers according to SPCMD1.
(7) Because the SPCMD1.SSLKP bit is 0, the RSPI negates the SSLyi signal. In addition, a next-access delay is inserted according to SPCMD1.

If the SSLyi signal output settings in the SPCMDm register in which 1 is assigned to the SSLKP bit are different from the SSLyi signal output settings in the SPCMDm register to be used in the next transfer, the RSPI switches the SSLyi signal status to SSLyi signal assertion ((5) in Figure 26.34) corresponding to the command for the next transfer. Note that if such an SSLyi signal switching occurs, the slaves that drive the MISOy signal compete, and collision of signal levels may occur.
The RSPI in master mode references the SSLyi signal operation within the module for the case where the SSLKP bit is not used. Even when the SPCMDm.CPHA bit is 0, the RSPI can accurately start serial transfers by using the SSLyi signal assertion for the next transfer that is detected internally.
(i = 0 to 3; m = 0 to 7; y = 0, 1 (for all channels))

(5) RSPCK Delay (t1)

The RSPCK delay value of the RSPI in master mode depends on the SPCMDm.SCKDEN bit setting and the SPCKD register setting. The RSPI determines the SPCMDm register to be referenced during serial transfer by pointer control, and determines an RSPCK delay value during serial transfer by using the SPCMDm.SCKDEN bit and SPCKD, as listed in Table 26.9. For a definition of RSPCK delay, refer to section 26.3.5, Transfer Format.

Table 26.9 Relationship among SCKDEN Bit, SPCKD, and RSPCK Delay Value

SPCMDm.SCKDEN Bit	SPCKD.SCKDL[2:0] Bits	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

(6) SSL Negation Delay (t2)

The SSL negation delay value of the RSPI in master mode depends on the SPCMDm.SLNDEN bit setting and the SSLND register setting. The RSPI determines the SPCMDm register to be referenced during serial transfer by pointer control, and determines an SSL negation delay value during serial transfer by using the SPCMDm.SLNDEN bit and SSLND, as listed in Table 26.10. For a definition of SSL negation delay, refer to section 26.3.5, Transfer Format.

Table 26.10 Relationship among SLNDEN Bit, SSLND, and SSL Negation Delay Value

SPCMDm.SLNDEN Bit	SSLND.SLNDL[2:0] Bits	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

(7) Next-Access Delay (t3)

The next-access delay value of the RSPI in master mode depends on the SPCMDm.SPNDEN bit setting and the SPND setting. The RSPI determines the SPCMDm register to be referenced during serial transfer by pointer control, and determines a next-access delay value during serial transfer by using the SPCMDm.SPNDEN bit and SPND, as listed in Table 26.11. For a definition of next-access delay, refer to section 26.3.5, Transfer Format.

Table 26.11 Relationship among SPNDEN Bit, SPND, and Next-Access Delay Value

SPCMDm.SPNDEN Bit	SPND.SPNDL[2:0] Bits	Next-Access Delay Value
0	000 to 111	1 RSPCK + 2 SERICLK
1	000	1 RSPCK + 2 SERICLK
	001	2 RSPCK + 2 SERICLK
	010	3 RSPCK + 2 SERICLK
	011	4 RSPCK + 2 SERICLK
	100	5 RSPCK + 2 SERICLK
	101	6 RSPCK + 2 SERICLK
	110	7 RSPCK + 2 SERICLK
	111	8 RSPCK + 2 SERICLK

(8) Initialization Flowchart

Figure 26.35 is a flowchart illustrating an example of initialization in SPI operation when the RSPI is used in master mode. For a description of how to set up the interrupt controller, DMAC, and I/O ports, refer to the descriptions given in the individual blocks.

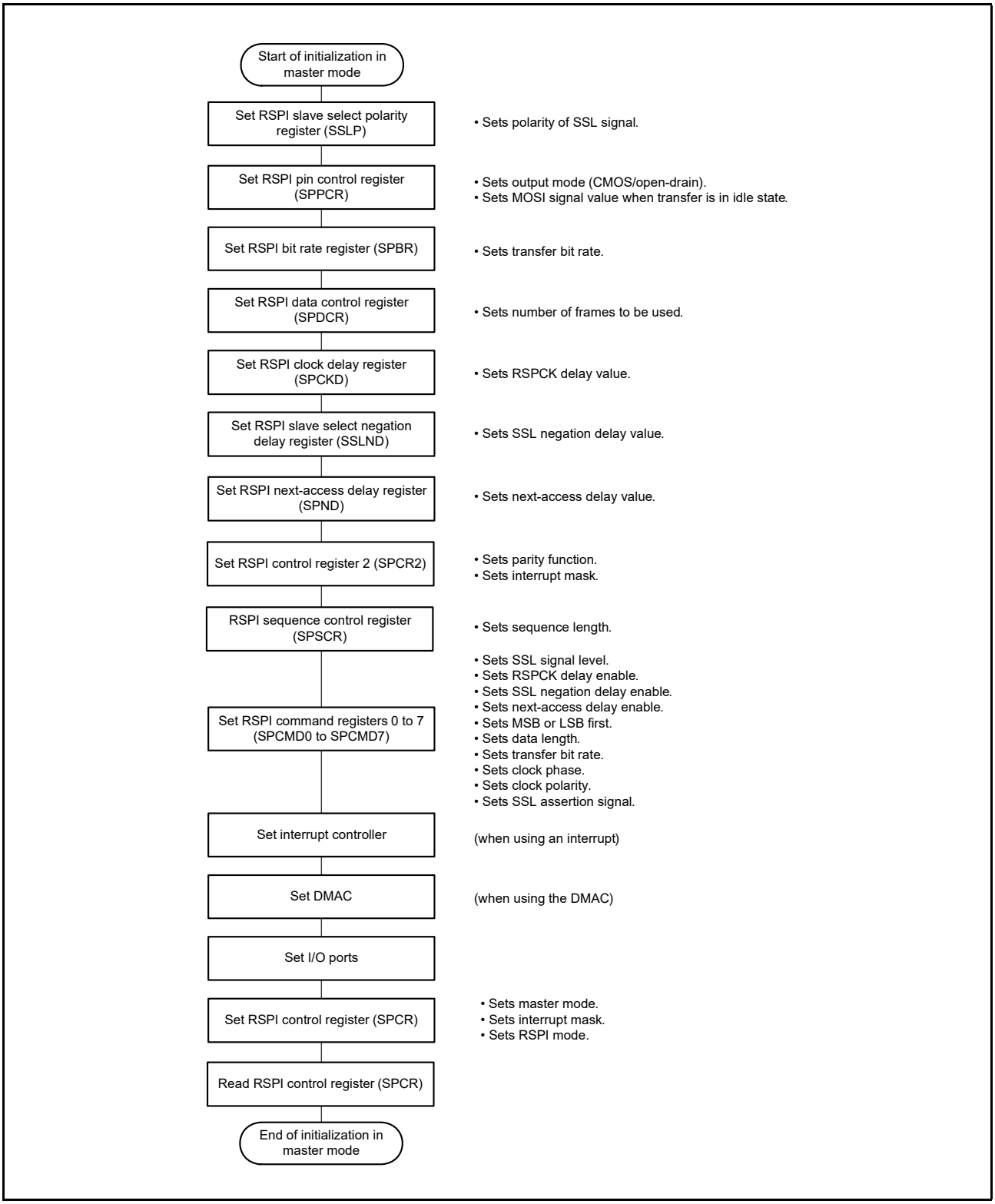


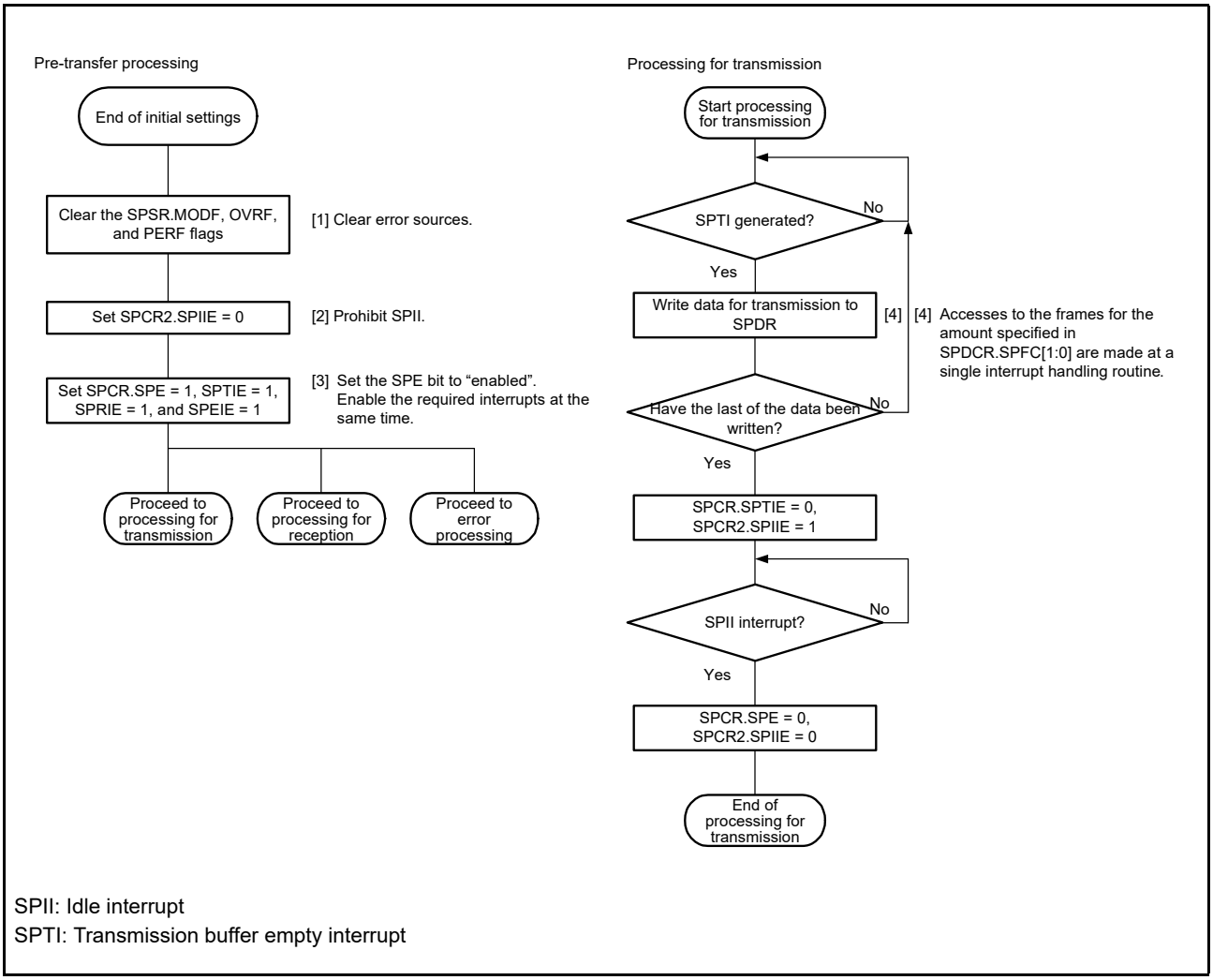
Figure 26.35 Example of Initialization Flowchart in Master Mode (SPI Operation)

(9) Software Processing Flow

Figure 26.36 to Figure 26.38 show examples of the flow of software processing.

(a) Transmit Processing Flow

When transmitting data, the CPU will be notified of the completion of data transmission after the last writing of data for transmission if the idle interrupt (SPII) is enabled.



SPII: Idle interrupt
SPTI: Transmission buffer empty interrupt

Figure 26.36 Flowchart in Master Mode (Transmission)

(b) Receive Processing Flow

The RSPI does not handle receive-only operation, so processing for transmission is required if reception is to proceed.

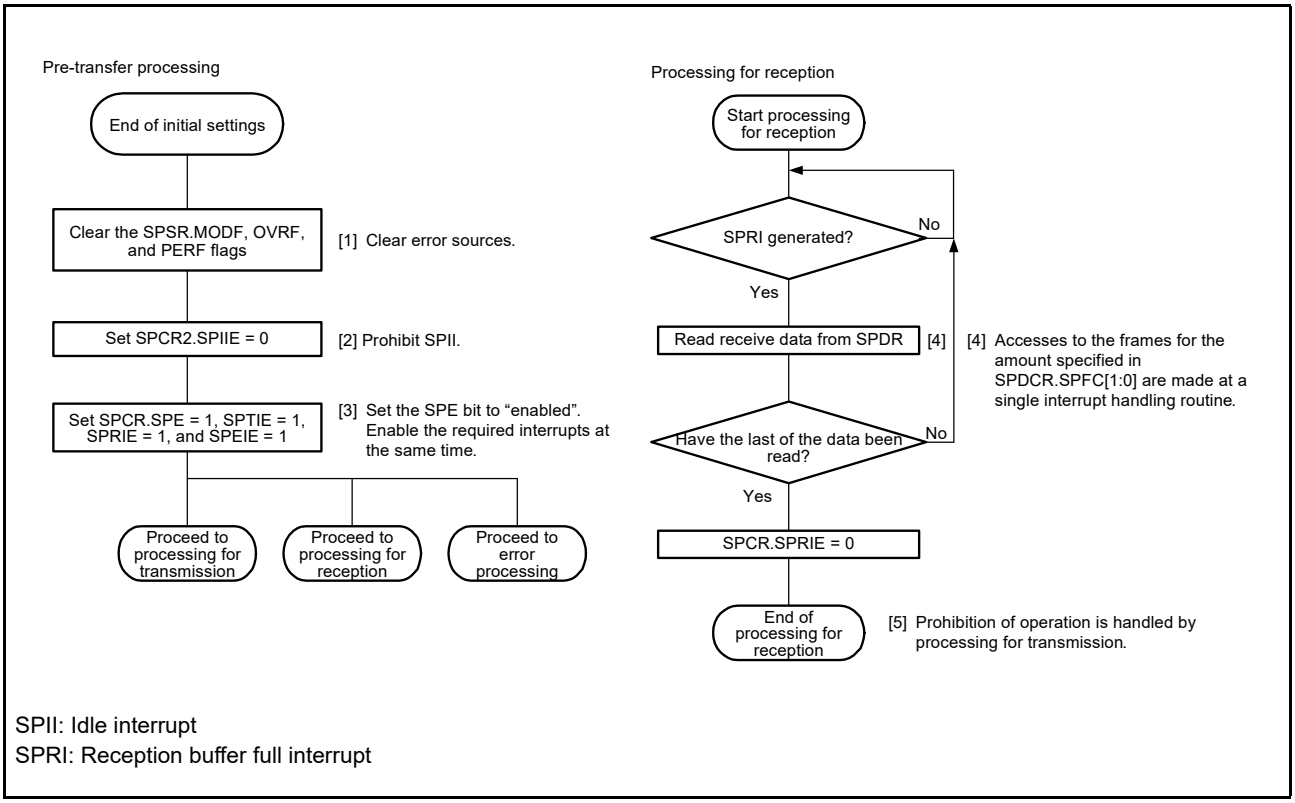
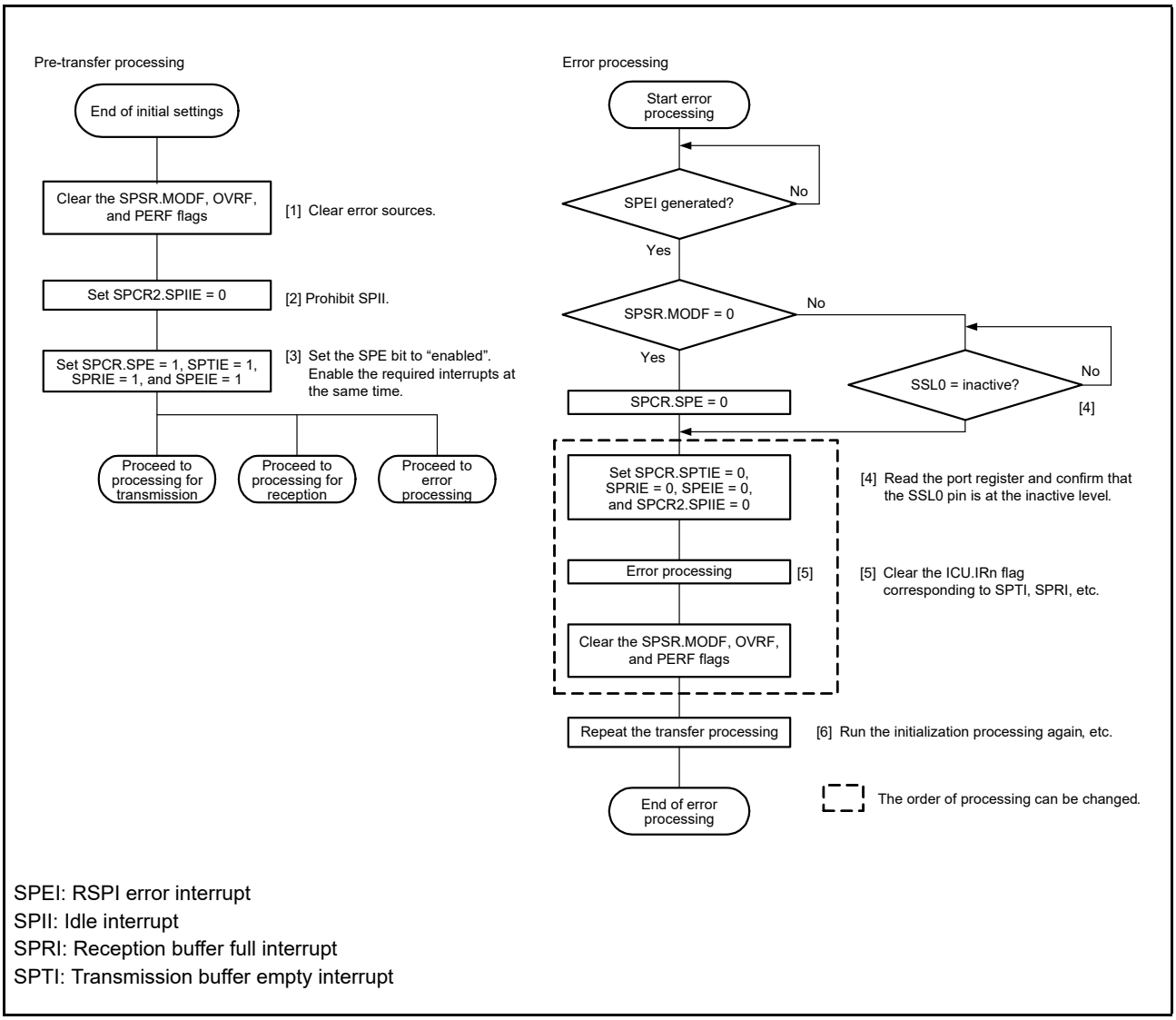


Figure 26.37 Flowchart in Master Mode (Reception)

(c) Flow of error processing

The RSPI has three types of error. When a mode fault error is generated, the SPCR.SPE bit is automatically cleared, stopping operations for transmission and reception. For errors from other sources, however, the SPCR.SPE bit is not cleared and operations for transmission and reception continue; accordingly, we recommend clearing of the SPCR.SPE bit to stop operations in the case of errors other than mode-fault errors. Not doing so will lead to updating of the SPSSR.SPECM[2:0] bits.

When an error occurs, clear the corresponding flag of the IRQ status register from within the error processing routine. If this is not done, the corresponding interrupt request flag in the IRQ status register may continue to indicate a transmission buffer empty interrupt (SPTI) or reception buffer full interrupt (SPRI) request. If the reception buffer full interrupt (SPRI) request is indicated, read the reception buffer and initialize the sequencer in the RSPI.



SPEI: RSPI error interrupt
SPII: Idle interrupt
SPRI: Reception buffer full interrupt
SPTI: Transmission buffer empty interrupt

Figure 26.38 Flowchart for Master Mode (Error Processing)

26.3.10.2 Slave Mode Operation

(1) Starting a Serial Transfer

If the SPCMD0.CPHA bit is 0, when detecting an SSLy0 input signal assertion, the RSPI needs to start driving valid data to the MISOy output signal. For this reason, when the CPHA bit is 0, the assertion of the SSLy0 input signal triggers the start of a serial transfer.
If the CPHA bit is 1, when detecting the first RSPCKy edge in an SSLy0 signal asserted condition, the RSPI needs to start driving valid data to the MISOy output signal. For this reason, when the CPHA bit is 1, the first RSPCKy edge in an SSLy0 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, the RSPI changes the status of the shift register to “full”, so that data cannot be copied from the transmission buffer to the shift register when serial transfer is in progress. If the shift register was full before the serial transfer started, the RSPI leaves the status of the shift register unchanged, in the full state.
Irrespective of CPHA bit setting, the timing at which the RSPI starts driving of the MISOy output signal is the SSLy0 signal assertion timing. The data which is output by the RSPI is either valid or invalid, depending on the CPHA bit setting.
For details on the RSPI transfer format, refer to section 26.3.5, Transfer Format. The polarity of the SSLy0 input signal depends on the setting of the SSLP.SSL0P bit.
(y = 0, 1 (for all channels))

(2) Terminating a Serial Transfer

Irrespective of the SPCMD0.CPHA bit, the RSPI terminates the serial transfer after detecting an RSPCKy edge corresponding to the final sampling timing. When free space is available in the reception buffer, upon termination of serial transfer the RSPI copies received data from the shift register to the reception buffer of the RSPI data register (SPDR). Upon termination of a serial transfer the RSPI changes the status of the shift register to “empty”, regardless of the reception buffer state. A mode fault error occurs if the RSPI detects an SSLy0 input signal negation from the beginning of serial transfer to the end of serial transfer (refer to section 26.3.8, Error Detection).
The final sampling timing changes depending on the bit length of transfer data. In slave mode, the RSPI data length depends on the SPCMD0.SPB[3:0] bit setting. The polarity of the SSLy0 input signal depends on the SSLP.SSL0P bit setting.
For details on the RSPI transfer format, refer to section 26.3.5, Transfer Format.
(y = 0, 1 (for all channels))

(3) Notes on Single-Slave Operations

If the SPCMD0.CPHA bit is 0, the RSPI starts serial transfers when it detects the assertion edge for an SSLy0 input signal. In the type of configuration shown in Figure 26.7 as an example, if the RSPI is used in single-slave mode, the SSLy0 signal is always fixed at the active state. Therefore, when the CPHA bit is set to 0, the RSPI cannot correctly start a serial transfer. To correctly execute transmit/receive operations by the RSPI in slave mode in a configuration in which the SSLy0 input signal is fixed at the active state, the CPHA bit should be set to 1. If there is a need for setting the CPHA bit to 0, the SSLy0 input signal should not be fixed.
(y = 0, 1 (for all channels))

(4) Burst Transfer

If the SPCMD0.CPHA bit is 1, continuous serial transfer (burst transfer) can be executed while retaining the assertion state for the SSLy0 input signal. If the CPHA bit is 1, the period from the first RSPCKy edge to the sampling timing for the reception of the final bit in an SSLy0 signal active state corresponds to a serial transfer period. Even when the SSLy0 input signal remains at the active level, the RSPI can accommodate burst transfers because it can detect the start of an access.

If the CPHA bit is 0, the second and subsequent serial transfers during burst transfer cannot be executed correctly.

(5) Initialization Flowchart

Figure 26.39 is a flowchart illustrating an example of initialization in SPI operation when the RSPI is used in slave mode. For a description of how to set up the interrupt controller, DMAC, and I/O ports, refer to the descriptions given in the individual blocks.

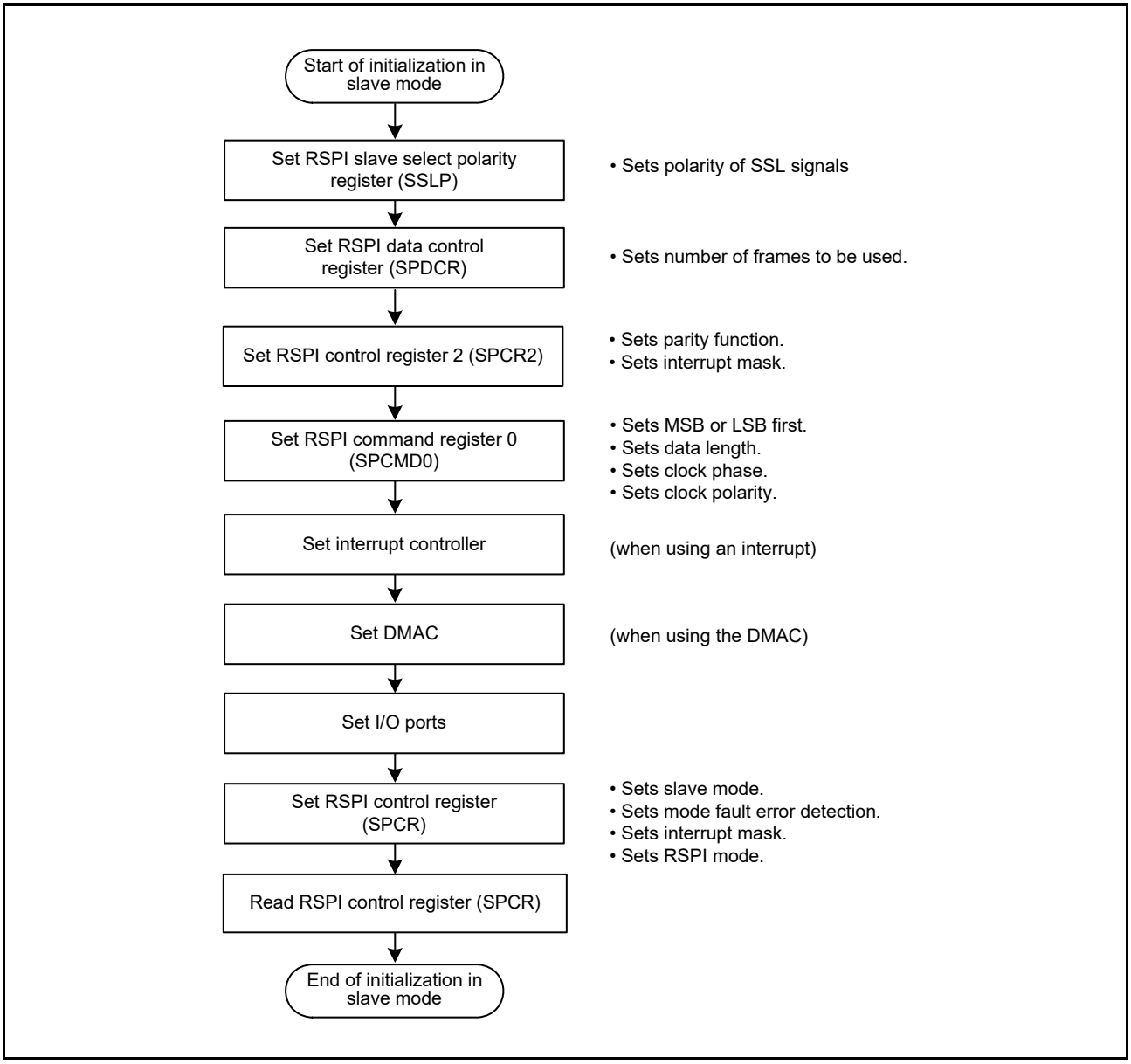


Figure 26.39 Example of Initialization Flowchart in Slave Mode (SPI Operation)

(6) Software Processing Flow

Figure 26.40 to Figure 26.42 show examples of the flow of software processing.

(a) Transmit Processing Flow

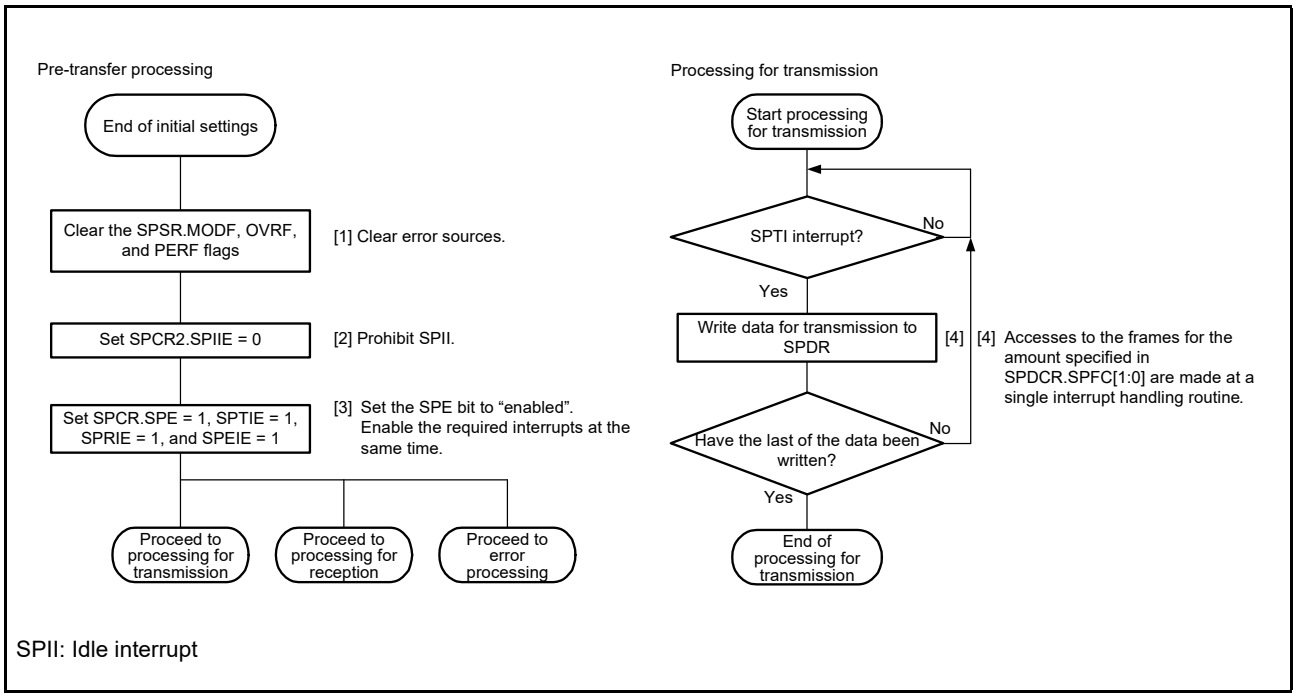


Figure 26.40 Flowchart in Slave Mode (Transmission)

(b) Receive Processing Flow

The RSPI does not handle receive-only operation, so processing for transmission is required if reception is to proceed.

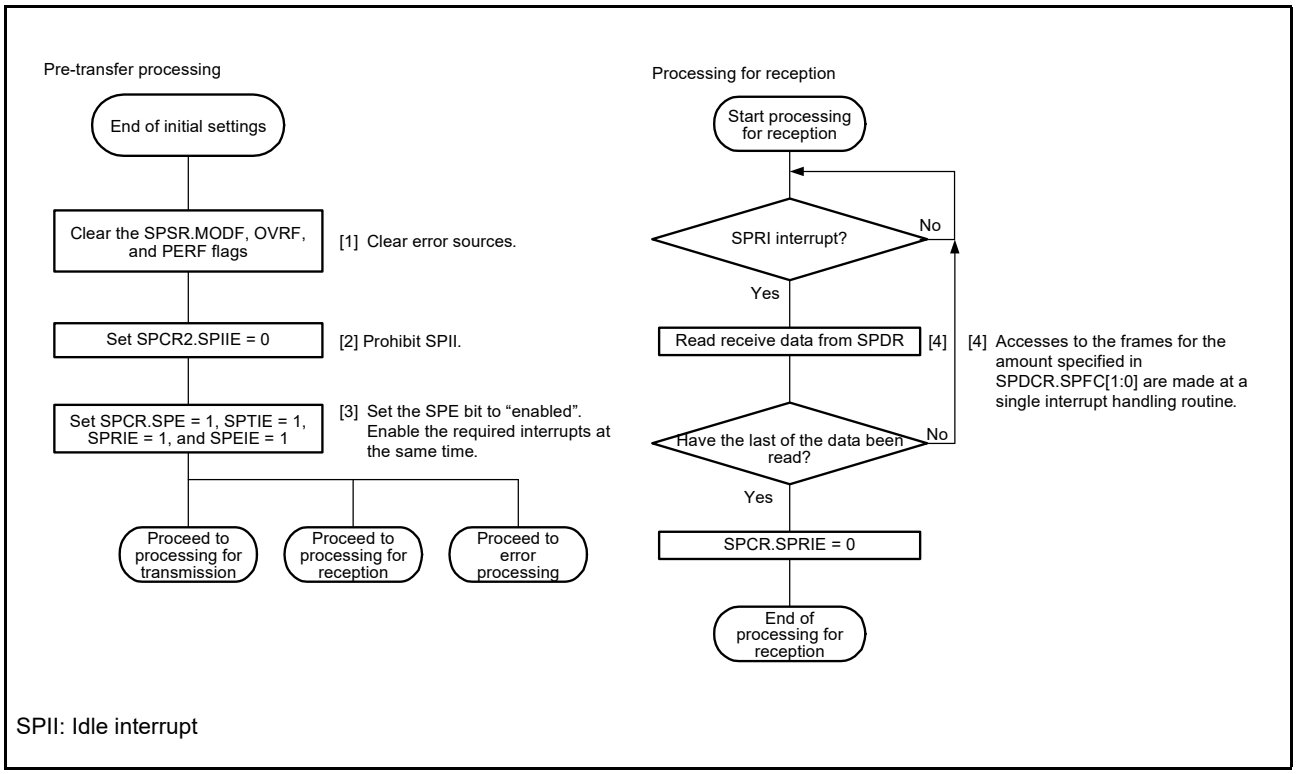
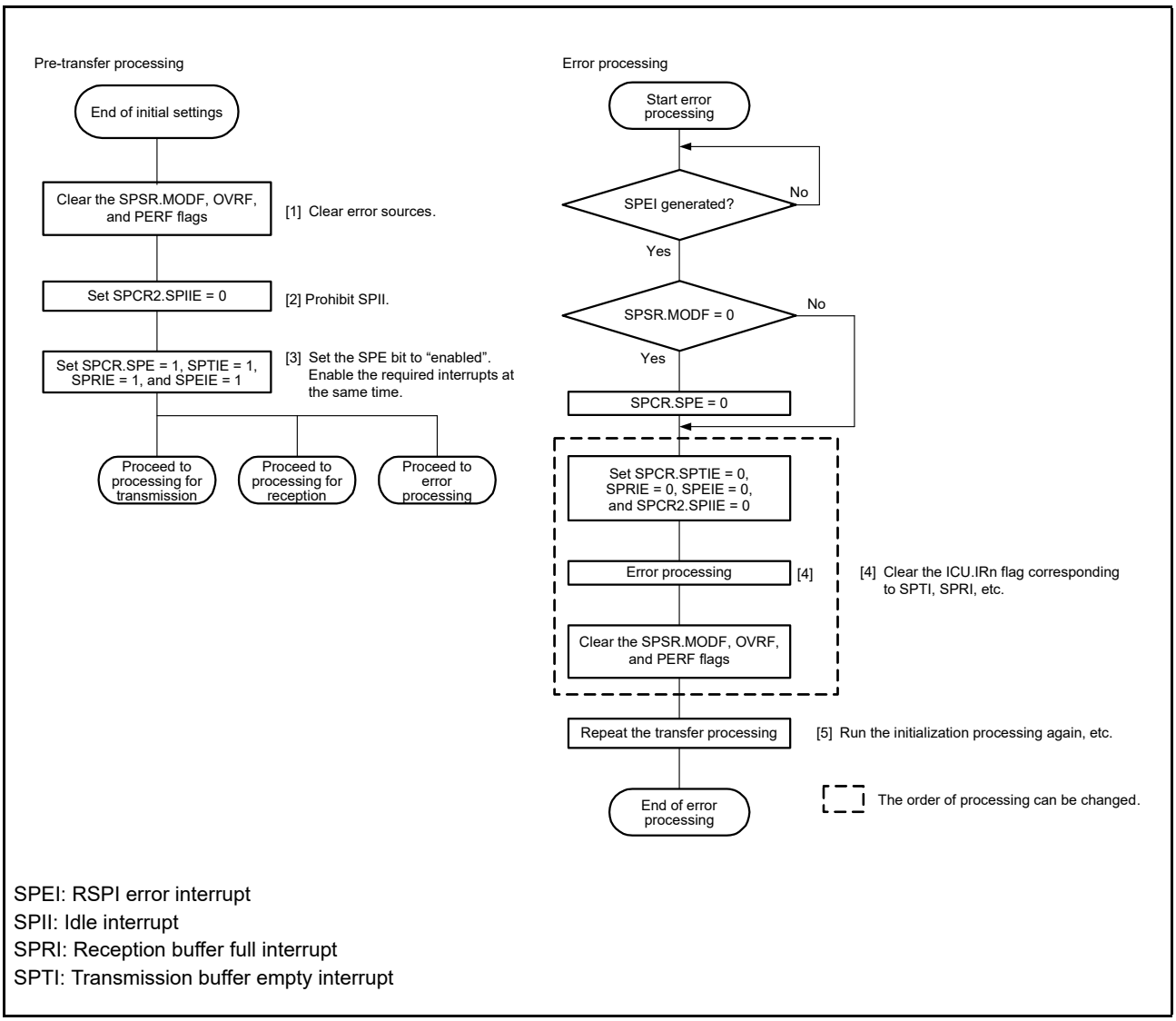


Figure 26.41 Flowchart in Slave Mode (Reception)

(c) Flow of error processing

In slave operation, even when a mode-fault error is generated, the SPSR.MODF flag can be cleared without de-asserting the pin.

When an error occurs, clear the corresponding flag of the IRQ status register (IRQSn) from within the error processing routine. If this is not done, the corresponding interrupt request flag in the IRQ status register (IRQSn) may continue to indicate a transmission buffer empty interrupt (SPTI) or reception buffer full interrupt (SPRI) request. If the reception buffer full interrupt (SPRI) request is indicated, read the reception buffer and initialize the sequencer in the RSPI.



SPEI: RSPI error interrupt
SPII: Idle interrupt
SPRI: Reception buffer full interrupt
SPTI: Transmission buffer empty interrupt

Figure 26.42 Flowchart for Slave Mode (Error Processing)

26.3.11 Clock Synchronous Operation

Setting the SPMS bit in the RSPI control register (SPCR) to 1 selects clock synchronous operation of the RSPI. In clock synchronous operation, the SSLyi pin is not used, and the three pins of RSPCKy, MOSIy, and MISOy handle communications. The SSLyi pin is available as I/O port pins.
Although clock synchronous operation does not require use of the SSLyi pin, operation of the module is the same as in SPI operation. That is, in both master and slave operations, communications can be performed with the same flow as in SPI operation. However, mode fault errors are not detected because the SSLyi pin is not used.
Furthermore, operation should not be performed if clock synchronous operation proceeds when the SPCMDm.CPHA bit is set to 0 in slave mode (SPCR.MSTR = 0).
(i = 0 to 3; m = 0 to 7; y = 0, 1 (for all channels))

26.3.11.1 Master Mode Operation

(1) Starting a Serial Transfer

The RSPI updates the data in the transmission buffer (SPTX) of SPDR when data is written to the RSPI data register (SPDR) with the transmission buffer being empty (data for the next transfer is not set). When the shift register is empty after the number of frames set in the SPDCR.SPFC[1:0] bits are written to the SPDR, the RSPI copies data from the transmission buffer to the shift register and starts serial transmission. Upon copying transmit data to the shift register, the RSPI changes the status of the shift register to “full”, and upon termination of serial transfer, it changes the status of the shift register to “empty”. The status of the shift register cannot be referenced.
For details on the RSPI transfer format, refer to section 26.3.5, Transfer Format.
However, transfer in clock synchronous operation is conducted without the SSLy0 output signal (y = 0, 1 (for all channels)).

(2) Terminating a Serial Transfer

The RSPI terminates the serial transfer after transmitting an RSPCKy edge corresponding to the sampling timing. If free space is available in the reception buffer (SPRX), upon termination of serial transfer, the RSPI copies data from the shift register to the reception buffer of the RSPI data register (SPDR).
It should be noted that the final sampling timing varies depending on the bit length of transfer data. In master mode, the RSPI data length depends on the SPCMDm.SPB[3:0] bit setting.
For details on the RSPI transfer format, refer to section 26.3.5, Transfer Format.
However, transfer in clock synchronous operation is conducted without the SSLy0 output signal (y = 0, 1 (for all channels)).

(3) Sequence Control

The transfer format employed in master mode is determined by SPSCR, SPCMDm, SPBR, SPCKD, SSLND, and SPND registers. Although the SSLyi signals are not output in clock synchronous operation, these settings are valid.
SPSCR is a register used to determine the sequence configuration for serial transfers that are executed by the RSPI in master mode. The following items are set in SPCMDm register: SSLyi output signal value, MSB/LSB first, data length, some of the bit rate settings, RSPCKy 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, an RSPI 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, the RSPI makes up a sequence comprised of a part or all of SPCMDm register. The RSPI contains a pointer to the SPCMDm register that makes up the sequence. The value of this pointer can be checked by reading the SPSSR.SPCP[2:0] bits. When the SPCR.SPE bit is set to 1 and the RSPI function is enabled, the RSPI loads the pointer to the commands in SPCMD0 register, and incorporates the SPCMD0 register setting into the transfer format at the beginning of serial transfer. The RSPI 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, the RSPI sets the pointer in SPCMD0 register, and in this manner the sequence is executed repeatedly.
(i = 0 to 3; m = 0 to 7; y = 0, 1 (for all channels))

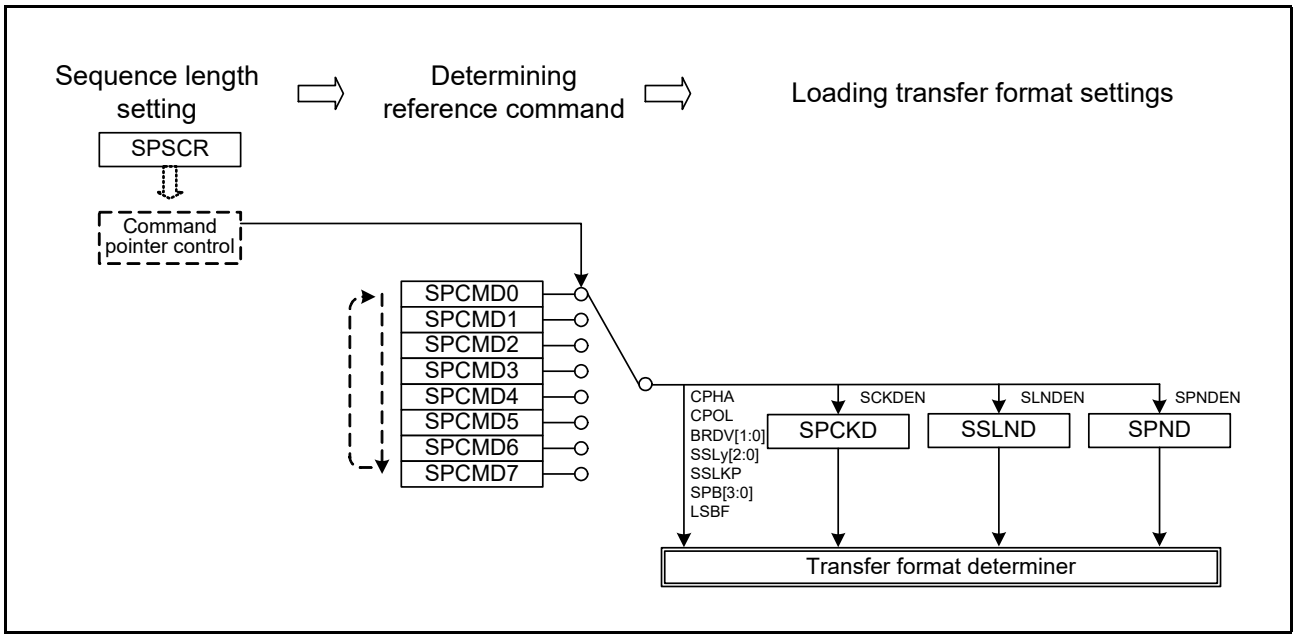


Figure 26.43 Procedure for Determining the Form of Serial Transmission in Master Mode

In this section, a frame is the combination of the data (SPDR) and the settings (SPCMDm) (m = 0 to 7).

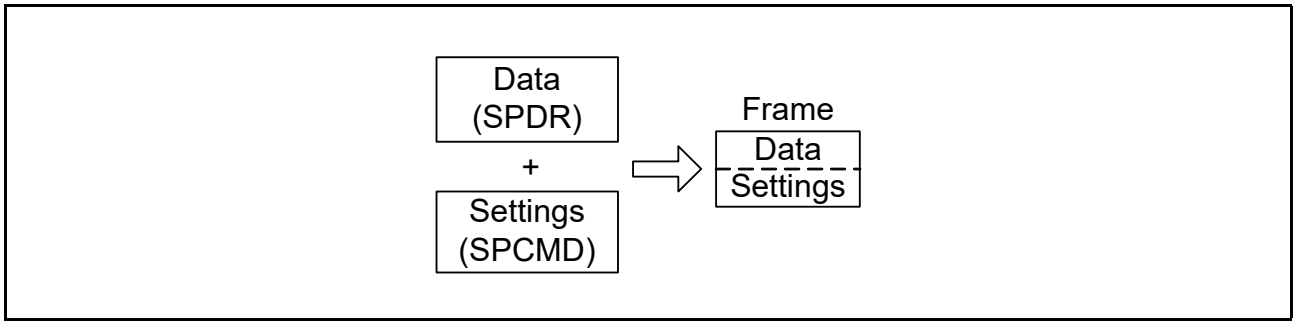


Figure 26.44 Concept of a Frame

Figure 26.45 shows the relationship between the command and the transmit and reception buffers in the sequence of operations specified by the settings in Table 26.4.

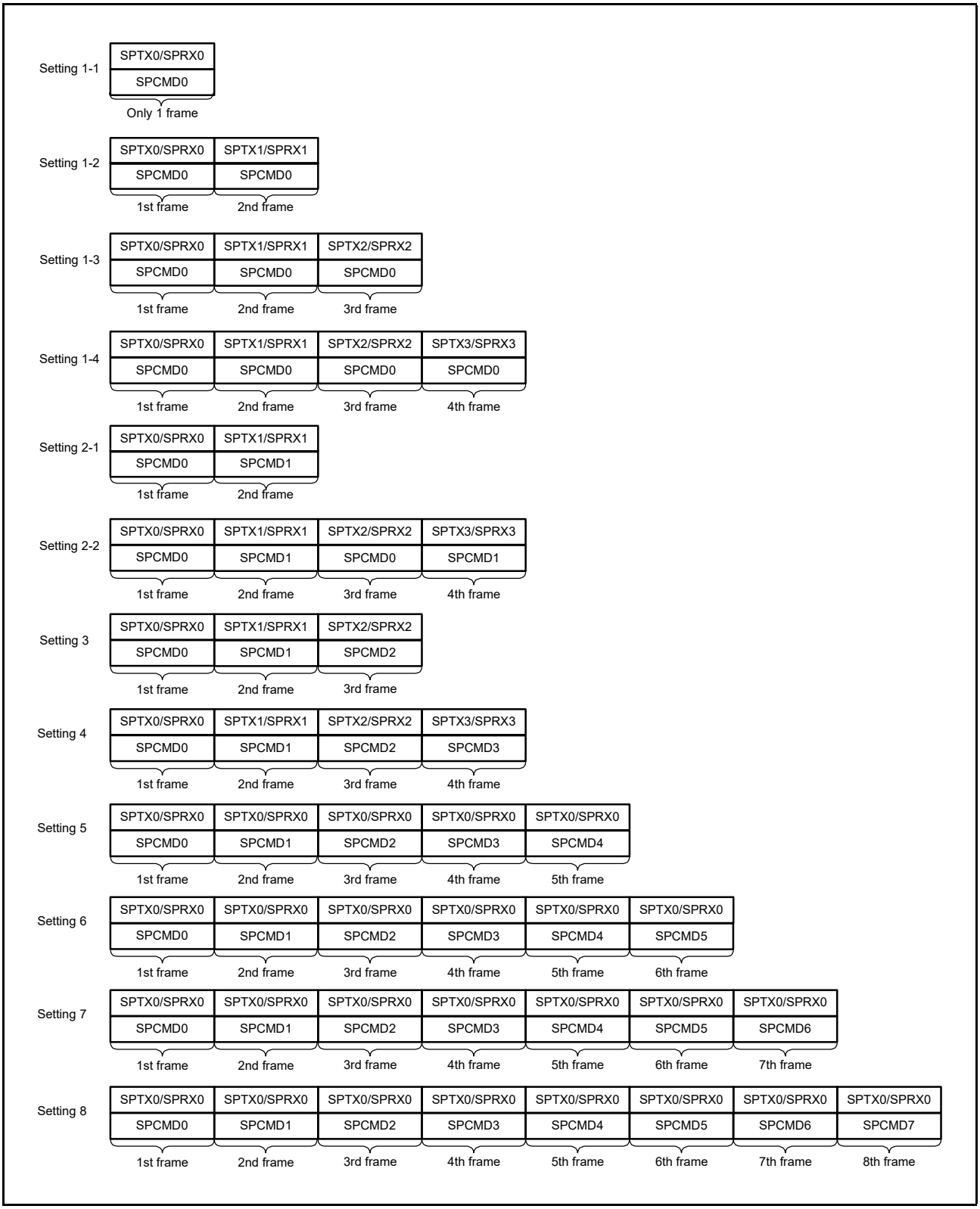


Figure 26.45 Correspondence between the RSPI Command Register and Transmission/Reception Buffers in Sequence Operations

(4) Initialization Flowchart

Figure 26.46 is a flowchart illustrating an example of initialization in clock synchronous operation when the RSPI is used in master mode. For a description of how to set up the interrupt controller, DMAC, and I/O ports, refer to the descriptions given in the individual blocks.

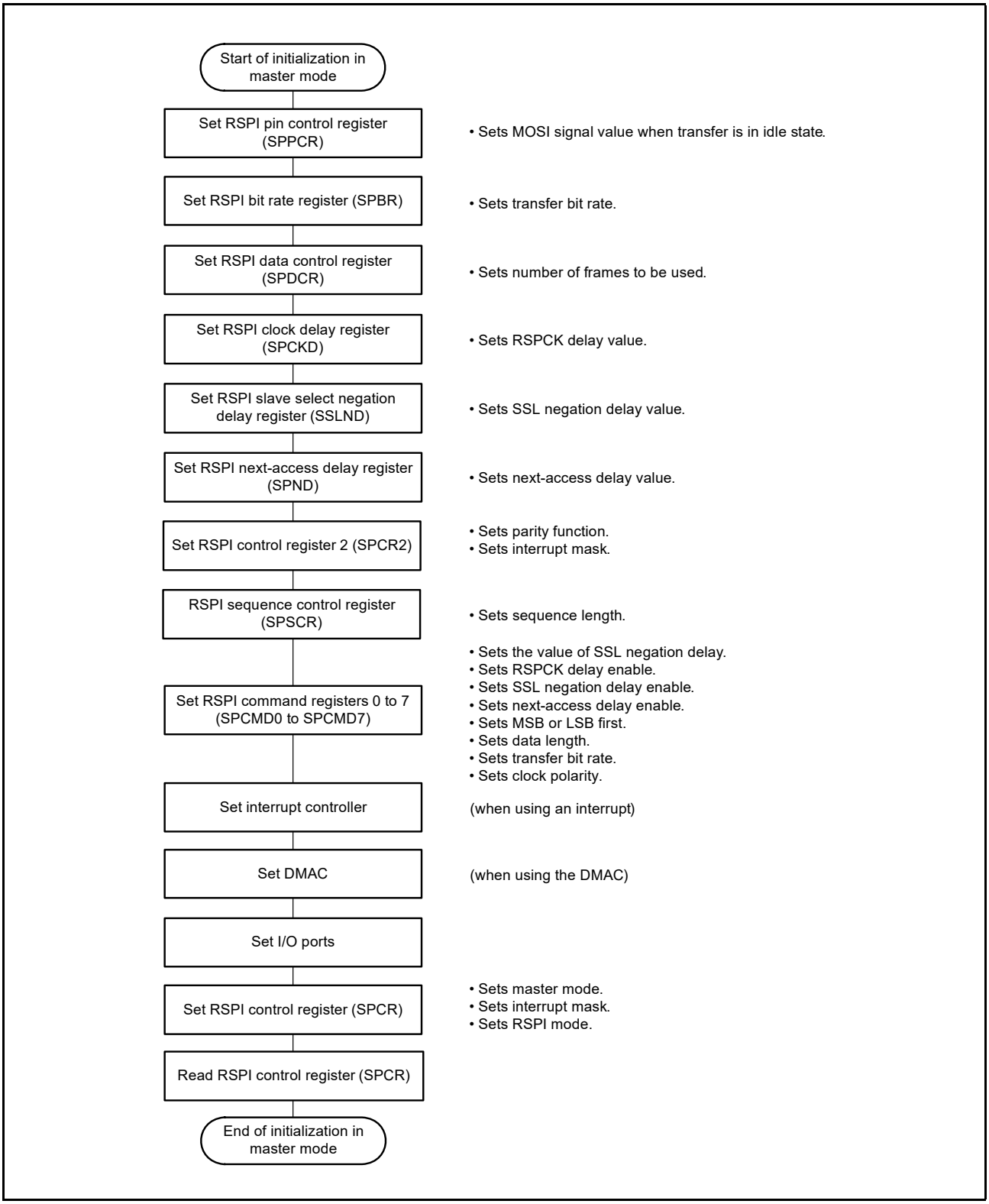


Figure 26.46 Example of Initialization Flowchart in Master Mode (Clock Synchronous Operation)

(5) Flow of Software Processing

Software processing during clock-synchronous master operation is the same as that for SPI master operation. For details, refer to section 26.3.10.1, (9) Software Processing Flow. Note that mode-fault errors will not occur.

26.3.11.2 Slave Mode Operation

(1) Starting a Serial Transfer

When the SPCR.SPMS bit is 1, the first RSPCKy edge triggers the start of a serial transfer in the RSPI.
When detecting the start of a serial transfer in a condition in which the shift register is empty, the RSPI changes the status of the shift register to “full”, so that data cannot be copied from the transmission buffer to the shift register when serial transfer is in progress. If the shift register was full before the serial transfer started, the RSPI keeps the status of the shift register unchanged, in the full state.
When the SPMS bit is 1, the RSPI always drives the MISOy output signal.
For details on the RSPI transfer format, refer to section 26.3.5, Transfer Format.
It should be noted that the SSL0 input signal is not used in clock synchronous operation (y = 0, 1 (for all channels)).

(2) Terminating a Serial Transfer

The RSPI terminates the serial transfer after detecting an RSPCKy edge corresponding to the final sampling timing. When free space is available in the reception buffer, upon termination of serial transfer the RSPI copies received data from the shift register to the reception buffer of the RSPI data register (SPDR). Upon termination of a serial transfer the RSPI changes the status of the shift register to “empty”. The final sampling timing changes depending on the bit length of transfer data. In slave mode, the RSPI data length depends on the SPCMD0.SPB[3:0] bit setting. For details on the RSPI transfer format, refer to section 26.3.5, Transfer Format. (y = 0, 1 (for all channels))

(3) Initialization Flowchart

Figure 26.47 is a flowchart illustrating an example of initialization in clock synchronous operation when the RSPI is used in slave mode. For a description of how to set up the interrupt controller, DMAC, and I/O ports, refer to the descriptions given in the individual blocks.

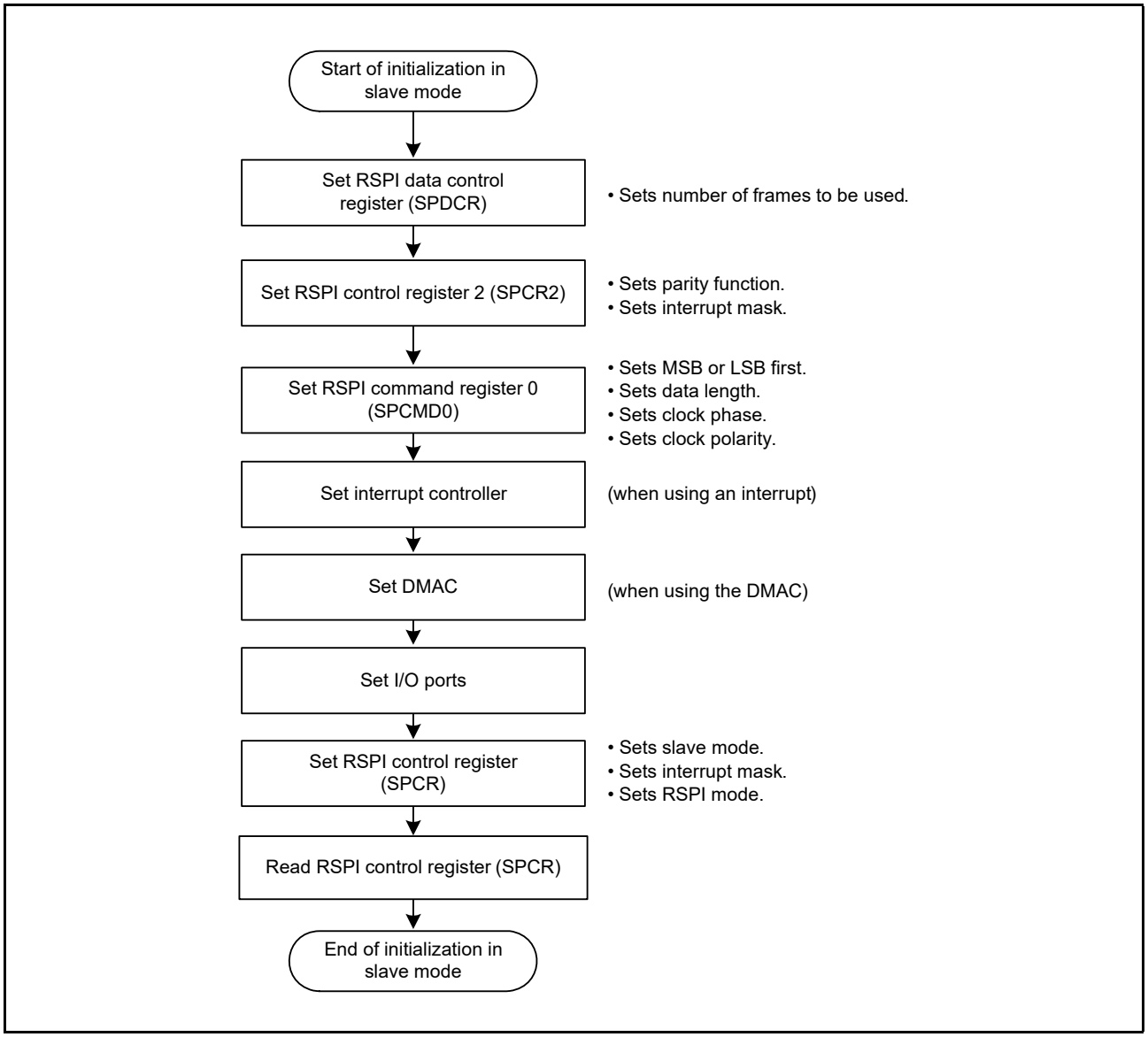


Figure 26.47 Example of Initialization Flowchart in Slave Mode (Clock Synchronous Operation)

(4) Flow of Software Processing

Software processing during clock-synchronous slave operation is the same as that for SPI slave operation. For details, refer to section 26.3.10.2, (6) Software Processing Flow. Note that mode-fault errors will not occur.

26.3.12 Loopback Mode

When 1 is written to the SPPCR.SPLP2 bit or SPPCR.SPLP bit, the RSPI shuts off the path between the MISOy pin and the shift register if the SPCR.MSTR bit is 1, and between the MOSIy pin and the shift register if the SPCR.MSTR bit is 0, and connects the input path and output path of the shift register. The RSPI does not shut off the path between the MOSIy pin and the shift register if the SPCR.MSTR bit is 1, and between the MISOy pin and the shift register if the SPCR.MSTR bit is 0. This is called loopback mode. When a serial transfer is executed in loopback mode, the transmit data for the RSPI or the inversed transmit data becomes the received data for the RSPI.
Table 26.12 lists the relationship among the SPLP2 and SPLP bits and the received data. Figure 26.48 shows the configuration of the shift register I/O paths for the case where the RSPI in master mode is set in loopback mode (SPPCR.SPLP2 = 0, SPPCR.SPLP = 1) (y = 0, 1 (for all channels)).

Table 26.12 SPLP2 and SPLP Bit Settings and Received Data

SPPCR.SPLP2 Bit	SPPCR.SPLP Bit	Received Data
0	0	Input data from the MOSIy pin or MISOy pin
0	1	Inversed transmit data
1	0	Transmit data
1	1	Transmit data

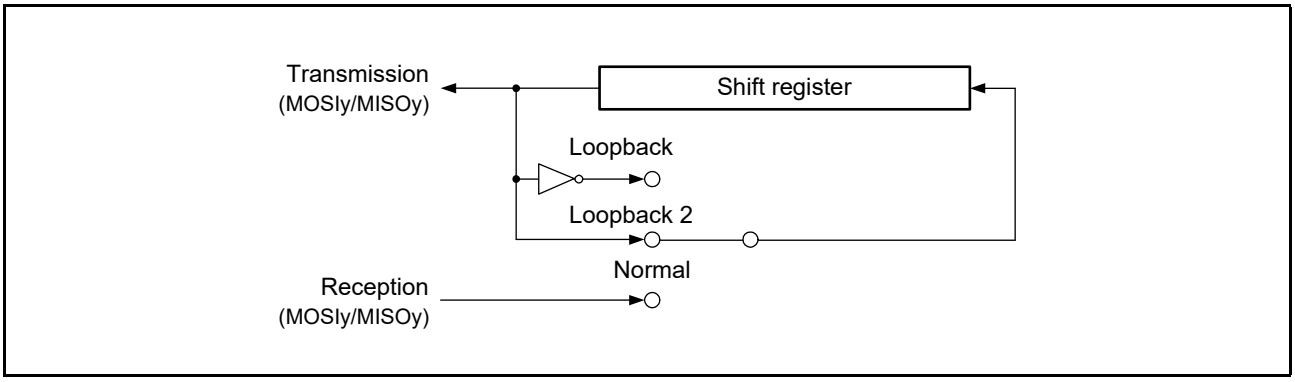


Figure 26.48 Configuration of Shift Register I/O Paths in Loopback Mode (Master Mode)

26.3.13 Self-Diagnosis of Parity Bit Function

The parity circuit consists of a parity bit adding unit used for transmit data and an error detecting unit used for received data. In order to detect defects in the parity bit adding unit and error detecting unit of the parity circuit, self-diagnosis is executed for the parity circuit following the flowchart shown in Figure 26.49.

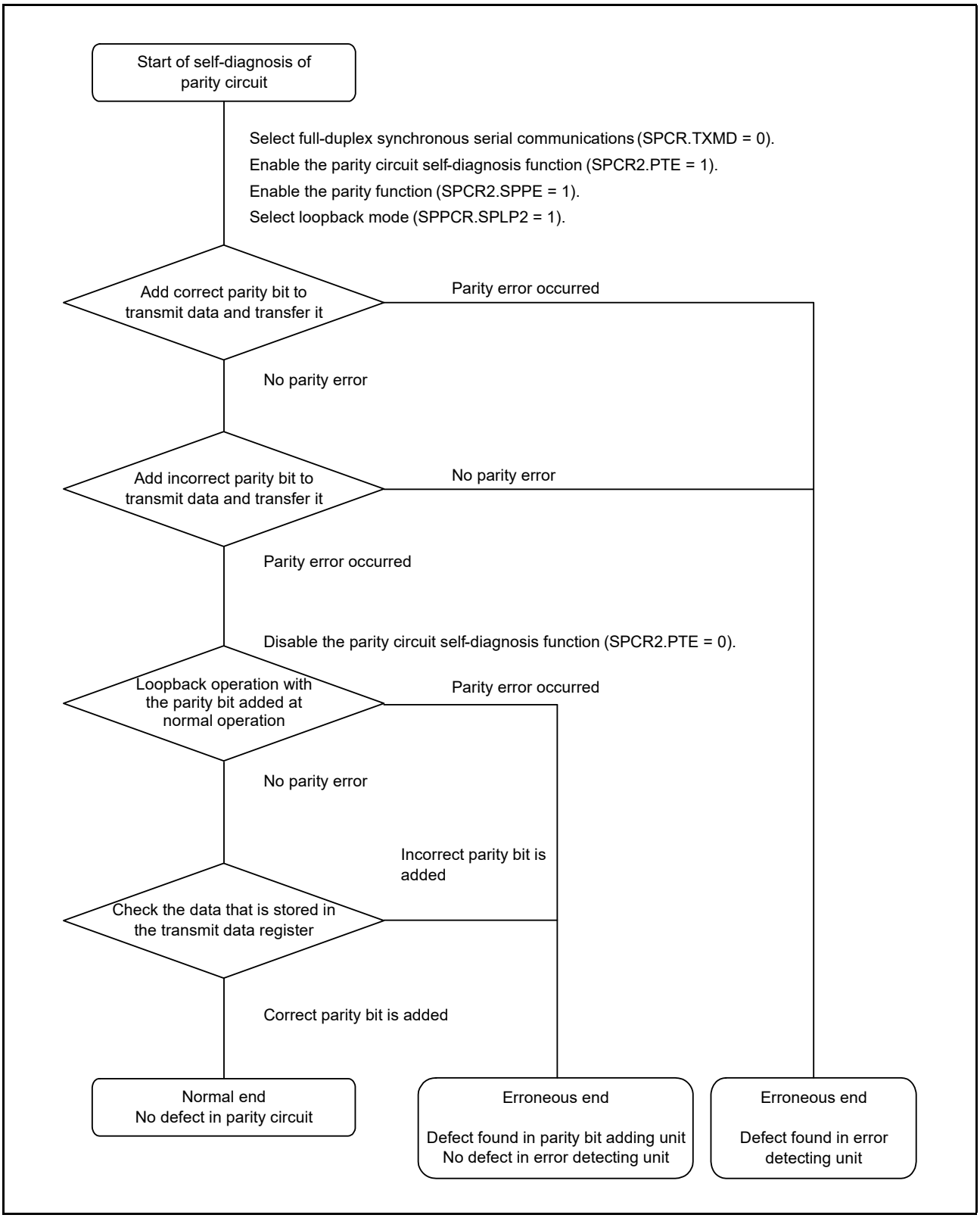


Figure 26.49 Flowchart for Self-Diagnosis of Parity Circuit

26.3.14 Interrupt Sources

The RSPI has interrupt sources of reception buffer full, transmission buffer empty, mode fault, overrun, parity error, and RSPI idle. In addition, the DMAC can be activated by the reception buffer full or transmission buffer empty interrupt to perform data transfer.

Since the common vector address is allocated to interrupt requests due to mode-fault, overrun, and parity errors, the actual interrupt source must be determined from the flags. Interrupt sources for the RSPI are listed in Table 26.13. An interrupt is generated on satisfaction of an interrupt condition in Table 26.13. Clear the reception buffer full and transmission buffer empty sources through data transfer.

When using the DMAC to perform data transmission/reception, the DMAC must be set up first to be in a status in which transfer is enabled before making the RSPI settings. For the method for setting the DMAC, refer to section 14, DMA Controller (DMACAa).

Table 26.13 Interrupt Sources of RSPI

Interrupt Source	Symbol	Interrupt Condition	DMAC/DTC Activation
Reception buffer full	SPRI	The reception buffer becomes full while the SPCR.SPRIE bit is 1.	Possible
Transmission buffer empty	SPTI	The transmission buffer becomes empty while the SPCR.SPTIE bit is 1.	Possible
RSPI errors (mode fault, overrun and parity error)	SPEI	The SPSR.MODF, OVRF, or PERF flag is set to 1 while the SPCR.SPEIE bit is 1.	Impossible
RSPI idle	SPII	The SPSR.IDLNF flag is set to 0 while the SPCR2.SPIIE bit is 1.	Impossible

26.4 Link Operation by Event Linking (only for RSPI channel 0)

The event link controller (ELC) is capable of producing the following event output signals. The event link output signal is output regardless of the setting of the corresponding interrupt enable bit (SPCR.SPEIE, SPCR.SPTIE or SPCR.SPRIE).

26.4.1 Reception Buffer Full Event Output

This event signal is output when received data have been transferred from the shift register to the SPDR on completion of serial transfer.

26.4.2 Transmission Buffer Empty Event Output

This event signal is output when data for transmission have been transferred from the transmission buffer to the shift register and when the value of the SPE bit has changed from 0 to 1.

26.4.3 Mode Fault, Overrun, or Parity Error Event Output

(1) Mode Fault

Table 26.14 lists the occurrence conditions of a mode fault event.

Table 26.14 Occurrence Conditions of Mode Fault Event

	SPCR.MODFEN Bit	SSLy0 Pin (y = 0, 1)	Remarks
Master (SPCR.MSTR bit = 1)	1	Active	When the setting of the SPCR.SPMS bit is 0 while the MSTR and SPCR.MODFEN bits are 1, mode fault error, overrun error, and parity error event output cannot be used. Do not set the ELSRn register to 52h.
Slave (SPCR.MSTR bit = 0)	1	Not active	Event is output only when the pin is deactivated during transmission.

(2) Overrun

The condition for this event signal being output in response to an overrun is completion of serial transfer while the reception buffer contains data that have not been read and the value of the SPCR.TXMD bit is 0, in which case the OVRF flag is set to 1.

(3) Parity Error

The condition for this event signal being output in response to a parity error is detection of a parity error on completion of serial transfer while the value of the TXMD bit in SPCR is 0 and the value of the SPPE bit in SPCR2 is 1.

26.4.4 RSPI Idle Event Output

(1) In Master Mode

In master mode, an event is output when the condition for setting the IDLNF flag (RSPI idle flag) to 0 is satisfied.

(2) In Slave Mode

In slave mode, an event is output when the SPE bit in the SPCR is set to 0 (RSPI is initialized).

26.4.5 Transmission-Completed Event Output

During both SPI operation and clock synchronous operation in master mode, an event is output under the condition for setting the IDLNF flag (RSPI idle flag) from 1 to 0.

Table 26.15 Conditions for Generation of a Transmission-Completed Event (Slave)

	Transmission Buffer State	Shift Register State	Others
SPI operation (SPCR.SPMS = 0)	Empty	Empty	Negation of SSL0 input
Clock synchronous operation (SPCR.SPMS = 1)	Empty	Empty	Edge detection of the last RSPCK

Regardless of whether the operation is in master mode or slave mode, an event is not output if 0 is written to the SPCR.SPE bit in transmission or the SPCR.SPE bit is cleared because of a mode fault error.

26.5 Usage Notes

26.5.1 Setting Module Stop Function

Module stop control register B (MSTPCRB) is used to enable or disable operation of the RSPI. The RSPI is stopped after a reset. The registers become accessible on release from the module-stop state. For details, refer to section 9, Low-Power Consumption Function.

26.5.2 Note on Low-Power Consumption Functions

Set the SPCR.SPE bit to 0 and terminate communications prior to entering module-stop state.

26.5.3 Notes on Starting Transfer

If the corresponding interrupt request flag in the IRQ status register (IRQSn) is 1 at the time transfer is to be started, the next interrupt request is generated after the start of transfer and an interrupt request is retained in the module. This can lead to unanticipated behavior of the interrupt request flag.
When the request flag for the RSPI interrupt request is 1 at the time transfer is to start, follow the procedure below to clear interrupt requests in the module or the IRQ status register (IRQSn) before enabling operations (by setting the SPCR.SPE bit to 1).

1. Confirm that transfer has stopped (i.e. that the SPCR.SPE bit is 0).
2. Set the relevant interrupt enable bit (the SPCR.SPTIE or SPCR.SPRIE bit) to 0.
3. Read the relevant interrupt enable bit (the SPCR.SPTIE or SPCR.SPRIE bit) and confirm that its value is 0.
4. Set the relevant interrupt request flag in the IRQ status register (IRQSn) to 0.

27. SPI Multi I/O Bus Controller (SPIBSC)

The SPI multi I/O bus controller outputs control signals to the serial flash memory connected to the SPI multi I/O bus space, thus enabling direct connection of the serial flash memory.
This LSI incorporates one channel of SPI multi I/O bus controller.

27.1 Overview

This module allows using direct reading or SPI operating mode to transmit and receive data with the serial flash memory connected to the SPI multi I/O bus space. The specification of the SPIBSC is listed in Table 27.1.

Table 27.1 SPIBSC Specifications

Item	Description
Serial flash interface	• One serial flash memory can be connected. • A data bus size of 1 bit, 2 bits, or 4 bits can be selected.
External address space read mode	• A maximum of 4 Gbytes of address space is supported. • The SPBSSL pin can be automatically controlled by access address monitoring. • Efficient data reception is possible due to the internal read cache (line size: 64 bits × 16 entries).
SPI operating mode	• Any read and write operations are available for a serial flash memory.
Bit rate	• The internal baud rate generator divides the frequency of PCLKA to generate SPBCLK. • The frequency division ratio of SPBCLK can be set in the range from 2 to 4080.
SPBSSL pin control	• The delay from the time the SPBSSL signal is activated to the time SPBCLK starts operating (clock delay) can be set. Setting range: 1 to 8 SPBCLK; Unit of the setting: 1 SPBCLK • The delay from the time SPBCLK stops to the time the SPBSSL output is inactivated (SPBSSL negation delay) can be set. Setting range: 1.5 to 8.5 SPBCLK; Unit of the setting: 1 SPBCLK • The wait of the SPBSSL output for the next access (next access delay) can be set. Setting range: 1 to 8 SPBCLK; Unit of the setting: 1 SPBCLK • The polarity of the SPBSSL signal can be changed.

Figure 27.1 is a block diagram of this module.

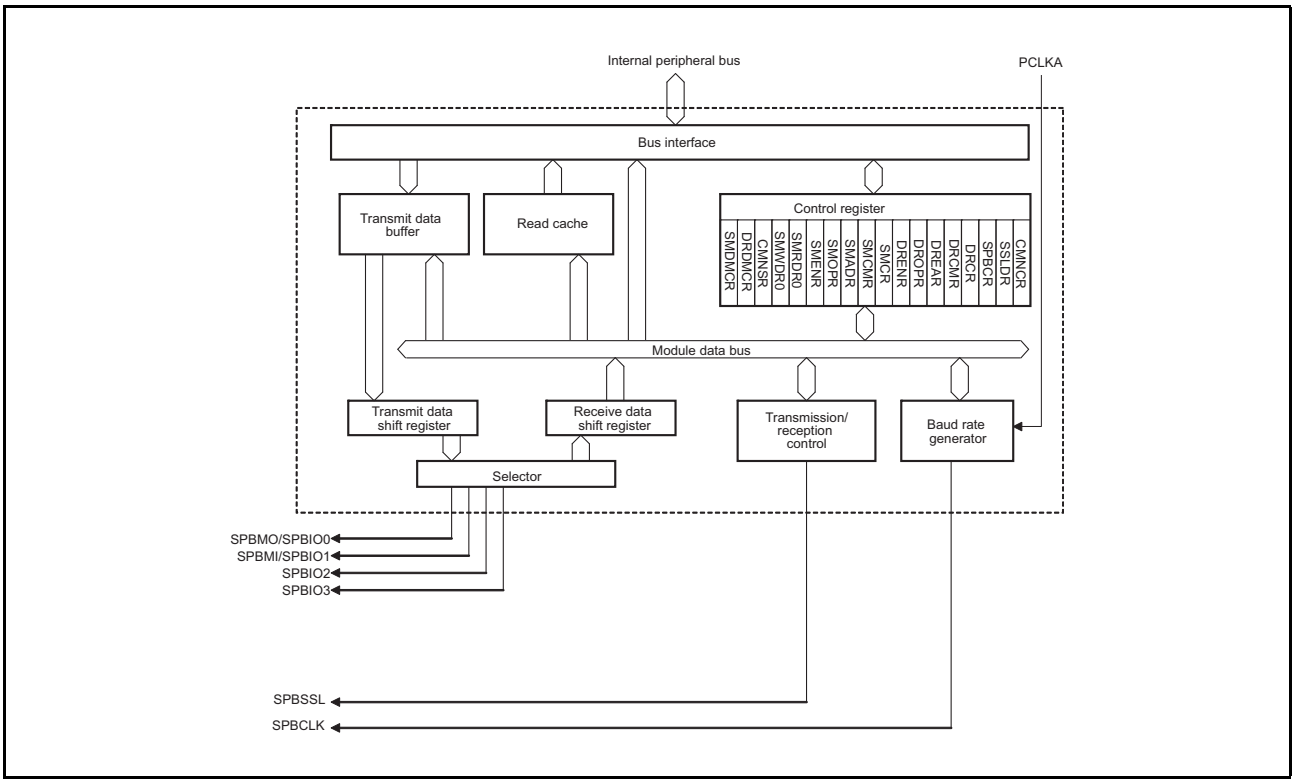


Figure 27.1 Block Diagram of SPIBSC

Table 27.2 lists the input/output pins of this module.

Table 27.2 Pin Configuration of the SPIBSC

Pin Name	Symbol	I/O	Function
Clock pin	SPBCLK	Output	Clock output
Slave select pin	SPBSSL	Output	Slave selection
Port data 0 pin	SPBMO/SPBIO0	I/O	Port master transmit data/data 0
Port data 1 pin	SPBMI/SPBIO1	I/O	Port master input data/data 1
Port data 2 pin	SPBIO2	I/O	Port data 2
Port data 3 pin	SPBIO3	I/O	Port data 3

27.2 Register Descriptions

27.2.1 Common Control Register (CMNCR)

The CMNCR register is a 32-bit register that controls the SPI multi I/O bus controller. The settings of this register are reflected both in external address space read mode and SPI operating mode.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Address(es): A000 5000h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	MD	—	—	—	—	—	—	SFDE	MOIIO3[1:0]		MOIIO2[1:0]		MOIIO1[1:0]		MOIIO0[1:0]	
Value after reset:	0	0	0	0	0	0	0	1	1	0	1	0	1	0	1	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IO3FV[1:0]		IO2FV[1:0]		—	—	IO0FV[1:0]		—	CPHAT	CPHAR	SSLP	CPOL	—	BSZ[1:0]	
Value after reset:	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	BSZ[1:0]	Data Bus Size	Specifies the number of serial flash memories to be connected. This bit must be set to 00b because only one memory can connect to this product. If another value is set, the operation cannot be guaranteed. b1 b0 0 0: 1 memory 0 1: Setting prohibited 1 X: Setting prohibited	R/W
b2	—	Reserved	This bit is always read as 0. The write value should always be 0.	R/W
b3	CPOL	SPBSSL Negation Period SPBCLK Output Direction	Sets the output level of the SPBCLK pin during inactive period of the SPBSSL signal. 0: The output of the SPBCLK pin is 0 during inactive period of the SPBSSL signal. 1: The output of the SPBCLK pin is 1 during inactive period of the SPBSSL signal.	R/W
b4	SSLP	SPBSSL Signal Polarity	Sets the polarity of SPBSSL signal. 0: Active low SPBSSL signal 1: Active high SPBSSL signal	R/W
b5	CPHAR	Input Latch	Sets the edge of the SPBCLK signal for the reception data. The CPHAT bit and this bit should be set according to the following table. 0: Data reception at odd edge 1: Data reception at even edge Settings of the CPHAT and CPHAR Bits	R/W

CPHAT	CPHAR	
0	0	Setting enabled
0	1	Setting enabled
1	0	Setting prohibited
1	1	Setting enabled

Bit	Symbol	Bit Name	Description	R/W
b6	CPHAT	Output Shift	Sets the edge of the SPBCLK signal for data transmission. This bit and the CPHAR bit should be set according to the description of the CPHAR bit. 0: Data transmission at even edge 1: Data transmission at odd edge	R/W
b7	—	Reserved	This bit is always read as 0. The write value should always be 0.	R/W
b9, b8	IO0FV[1:0]	SPBIO0 Fixed Value for 1-bit Size Input	Fixes the output value of the SPBIO0 pins for 1-bit size input. b9 b8 0 0: Output value 0 0 1: Output value 1 1 0: Output value is the last bit value of the previous transfer (or the state is Hi-Z, if Hi-Z was the state in the last bit period of the previous transfer). 1 1: Output value Hi-Z	R/W
b11, b10	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b13, b12	IO2FV[1:0]	SPBIO2 Fixed Value for 1-bit/2-bit Size	Fixes the output value of the SPBIO2 pins for 1-bit/2-bit size. b13 b12 0 0: Output value 0 0 1: Output value 1 1 0: Output value is the last bit value of the previous transfer (or the state is Hi-Z, if Hi-Z was the state in the last bit period of the previous transfer). 1 1: Output value Hi-Z	R/W
b15, b14	IO3FV[1:0]	SPBIO3 Fixed Value for 1-bit/2-bit Size	Fixes the output value of the SPBIO3 pins for 1-bit/2-bit size. b15 b14 0 0: Output value 0 0 1: Output value 1 1 0: Output value is the last bit value of the previous transfer (or the state is Hi-Z, if Hi-Z was the state in the last bit period of the previous transfer). 1 1: Output value Hi-Z	R/W
b17, b16	MOIIO0[1:0]	SPBIO0 Fixed Value for SPBSSL Idle	Fixes output values of the SPBIO0 pins during inactive period of the SPBSSL signal. b17 b16 0 0: Output value 0 0 1: Output value 1 1 0: Output value is the last bit value of the previous transfer (or the state is Hi-Z, if Hi-Z was the state in the last bit period of the previous transfer). 1 1: Output value Hi-Z	R/W
b19, b18	MOIIO1[1:0]	SPBIO1 Fixed Value for SPBSSL Idle	Fixes output values of the SPBIO1 pins during inactive period of the SPBSSL signal. b19 b18 0 0: Output value 0 0 1: Output value 1 1 0: Output value is the last bit value of the previous transfer (or the state is Hi-Z, if Hi-Z was the state in the last bit period of the previous transfer). 1 1: Output value Hi-Z	R/W
b21, b20	MOIIO2[1:0]	SPBIO2 Fixed Value for SPBSSL Idle	Fixes output values of the SPBIO2 pins during inactive period of the SPBSSL signal. b21 b20 0 0: Output value 0 0 1: Output value 1 1 0: Output value is the last bit value of the previous transfer (or the state is Hi-Z, if Hi-Z was the state in the last bit period of the previous transfer). 1 1: Output value Hi-Z	R/W

Bit	Symbol	Bit Name	Description	R/W
b23, b22	MOIIO3[1:0]	SPBIO3 Fixed Value for SPBSSL Idle	Fixes output values of the SPBIO3 pins during inactive period of the SPBSSL signal. b23 b22 0 0: Output value 0 0 1: Output value 1 1 0: Output value is the last bit value of the previous transfer (or the state is Hi-Z, if Hi-Z was the state in the last bit period of the previous transfer). 1 1: Output value Hi-Z	R/W
b24	SFDE	Data Swap Setting for Serial Flash Memory	Specifies whether or not swapping of data in serial flash memory is performed. 0: Swapping is not performed. 1: Swapping is performed in 8-bit units. For details, see section 27.3.4, Data Alignment.	R/W
b30 to b25	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b31	MD	Operating Mode Switch	Switches the operating modes. 0: External address space read mode 1: SPI operating mode	R/W

27.2.2 SSL Delay Register (SSLDR)

The SSLDR register is a 32-bit register that adjusts the timing between the SPBSSL signal and the SPBCLK signal. The settings of this register are reflected both in external address space read mode and SPI operating mode.
The settings of this register should be changed when the TEND flag in the CMNSR register is 1; otherwise, the operation cannot be guaranteed.

Address(es): A000 5004h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	SPNDL[2:0]		
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	SLNDL[2:0]			—	—	—	—	—	SCKDL[2:0]		
Value after reset:	0	0	0	0	0	1	1	1	0	0	0	0	0	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	SCKDL[2:0]	Clock Delay	Sets the period from the time the SPBSSL signal is activated to the time the clock is output from the SPBCLK signal (clock delay). b2 b0 000: 1 SPBCLK cycle 001: 2 SPBCLK cycles 010: 3 SPBCLK cycles 011: 4 SPBCLK cycles 100: 5 SPBCLK cycles 101: 6 SPBCLK cycles 110: 7 SPBCLK cycles 111: 8 SPBCLK cycles	R/W
b7 to b3	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b10 to b8	SLNDL[2:0]	SPBSSL Negation Delay	Sets the period from the time the last edge of the SPBCLK signal is transferred to the time the SPBSSL signal is inactivated (SPBSSL negation delay). b10 b8 000: 1.5 SPBCLK cycles 001: 2.5 SPBCLK cycles 010: 3.5 SPBCLK cycles 011: 4.5 SPBCLK cycles 100: 5.5 SPBCLK cycles 101: 6.5 SPBCLK cycles 110: 7.5 SPBCLK cycles 111: 8.5 SPBCLK cycles	R/W
b15 to b11	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b18 to b16	SPNDL[2:0]	Next Access Delay	Sets the period from transfer end to next transfer start (next access). b18 b16 000: 1 SPBCLK cycle 001: 2 SPBCLK cycles 010: 3 SPBCLK cycles 011: 4 SPBCLK cycles 100: 5 SPBCLK cycles 101: 6 SPBCLK cycles 110: 7 SPBCLK cycles 111: 8 SPBCLK cycles	R/W
b31 to b19	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W

27.2.3 Bit Rate Register (SPBCR)

The SPBCR register is a 32-bit register that sets the bit rate.
The settings of this register are reflected both in external address space read mode and SPI operating mode.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Address(es): A000 5008h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SPBR[7:0]								—	—	—	—	—	—	BRDV[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

Bit	Symbol	Bit Name	Description	R/W
b1, b0	BRDV[1:0]	Bit Rate Frequency Division	Sets the bit rate. The bit rate is determined by a combination of these bits with the SPBR[7:0] bits. The setting value of the SPBR bit determines the base bit rate. This bit is used to select a division ratio of the base bit rate from among no division, 2, 4, and 8. b1 b0 0 0: Base bit rate 0 1: Base bit rate divided by 2 1 0: Base bit rate divided by 4 1 1: Base bit rate divided by 8	R/W
b7 to b2	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b15 to b8	SPBR[7:0]	Bit Rate	Sets the bit rate. The bit rate is determined by a combination of these bits with the BRDV[1:0] bits. For details, see Table 27.3, Relationship between SPBR[7:0] and BRDV[1:0] Settings.	R/W
b31 to b16	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W

(1) Bit Rate

The SPBR[7:0] and BRDV[1:0] bits are used for setting the bit rate.
The following formula is used to calculate the bit rate when SPBR[7:0] ≠ 0:

Bit rate = PCLKA / ($2 \times n \times 2^N$)

n: SPBR[7:0] setting (1, …, 255)

N: BRDV[1:0] setting (0 to 3)

The following formula is used to calculate the bit rate when SPBR[7:0] = 0:

Bit rate = PCLKA / 2^N

Setting both the SPBR[7:0] and BRDV[1:0] bits to 0 is prohibited.

Table 27.3 Relationship between SPBR[7:0] and BRDV[1:0] Settings

			Bit Rate
SPBR[7:0] (n)	**BRDV[1:0] (N)**	**Division Ratio**	**PCLKA = 150 MHz**
0	0	1	Setting prohibited
0	1	2	75 Mbps
0	2	4	37.5 Mbps
0	3	8	18.75 Mbps
1	0	2	75 Mbps
2	0	4	37.5 Mbps
3	0	6	25 Mbps
4	0	8	18.75 Mbps
5	0	10	15 Mbps
6	0	12	12.5 Mbps
6	1	24	6.25 Mbps
6	2	48	3.13 Mbps
6	3	96	1.56 Mbps
255	3	4080	36.76 Kbps

Note: The bit rate should be set so that it will satisfy the AC characteristics of this module.

27.2.4 Data Read Control Register (DRCR)

The DRCR register is a 32-bit register that sets the operation in external address space read mode.
The bits except the SSLN bit should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Address(es): A000 500Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	SSLN	—	—	—	—	RBURST[3:0]			
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	RCF	RBE	—	—	—	—	—	—	—	SSLE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SSLE	SPBSSL Negation	Sets the inactivation conditions for the SPBSSL signal during read burst. The SPBSSL signal is inactivated for each access during normal read. 0: The SPBSSL signal is inactivated after transfer of data set in burst length. 1: The SPBSSL signal is inactivated when the accessed address is not continuous with the previously transferred address.	R/W
b7 to b1	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b8	RBE	Read Burst	Turns burst read ON or OFF. 0: Data is read according to the access size. 1: Read cache is enabled, and as many data units as the burst count specified in RBURST[3:0] bits is read.	R/W
b9	RCF	Read Cache Flush	When 1 is written to this bit, all the entries in the read cache are cleared. This bit is always read as 0. Note: After flushing the read cache by writing 1 to this bit, read the DRCR register before proceeding to read from the external address space.	R/W
b15 to b10	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b19 to b16	RBURST [3:0]	Read Data Burst Length	Sets the burst length (data unit count) when reading. This bit is enabled when the RBE bit of this register is set to 1. b19 b16 0000: 1 data unit 0001: 2 continuous data units : 1110: 15 continuous data units 1111: 16 continuous data units One data unit is 64 bits long.	R/W
b23 to b20	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b24	SSLN	SPBSSL Negation	Writing 1 to this bit when both the RBE and SSLE bits of this register are 1 inactivates the active SPBSSL signal. This bit is always read as 0. Note: To start next access after the SPBSSL signal is inactivated by this bit, read the SSLF bit of the CMNSR register = 0 to confirm that the SPBSSL signal has been inactivated.	R/W

Bit	Symbol	Bit Name	Description	R/W
b31 to b25	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W

27.2.5 Data Read Command Setting Register (DRCMR)

The DRCMR register is a 32-bit register that sets the commands issued in external address space read mode.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Address(es): A000 5010h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	CMD[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	OCMD[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	OCMD[7:0]	Optional Command	Sets the optional command.	R/W
b15 to b8	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b23 to b16	CMD[7:0]	Command	Sets the command. For details, see the details about the serial flash memory to be used.	R/W
b31 to b24	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W

27.2.6 Data Read Extended Address Setting Register (DREAR)

The DREAR register is an address setting register when the serial flash address is output in 32-bit mode.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Address(es): A000 5014h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	EAV[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	EAC[2:0]		
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	EAC[2:0]	32-Bit Extended External Address Valid Range	Sets the range of the external address to be used as serial flash address when the serial flash address is output in 32-bit mode. This setting is valid when the ADE[3] bit in DRENR is 1. b2 b0 000: External address bits [24:0] enabled 001: External address bits [25:0] enabled Other than above: Setting prohibited	R/W
b15 to b3	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b23 to b16	EAV[7:0]	32-Bit Extended Upper Address Fixed Value	The upper address of the external address specified by the EAC[2:0] bits of this register are set to these bits when the serial flash address is output in 32-bit mode. Bit 0 corresponds to the serial flash address bit [25], and bit 7 corresponds to the bit [32]. This setting is valid only when the ADE[3] bit in DRENR is 1. When EAC[2:0] are 000, serial flash address [32:25] are set to EAV[7:0]. When EAC[2:0] are 001, serial flash address [32:26] are set to EAV[7:1].	R/W
b31 to b24	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W

27.2.7 Data Read Option Setting Register (DROPR)

The DROPR register is a 32-bit register that sets the option data in external address space read mode.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Address(es): A000 5018h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	OPD3[7:0]								OPD2[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	OPD1[7:0]								OPD0[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	OPD0[7:0]	Option Data 0	Sets the option data 0.	R/W
b15 to b8	OPD1[7:0]	Option Data 1	Sets the option data 1.	R/W
b23 to b16	OPD2[7:0]	Option Data 2	Sets the option data 2.	R/W
b31 to b24	OPD3[7:0]	Option Data 3	Sets the option data 3.	R/W

Note: OPD3, OPD2, OPD1, and OPD0 are output in this order.

27.2.8 Data Read Enable Setting Register (DRENR)

The DRENR register is a 32-bit register that sets the bit size of the command, optional command, address, option data, and read data in external address space read mode and enables outputting them other than read data.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Address(es): A000 501Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	CDB[1:0]		OCDB[1:0]		—	—	ADB[1:0]		—	—	OPDB[1:0]		—	—	DRDB[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	DME	CDE	—	OCDE	ADE[3:0]				OPDE[3:0]				—	—	—	—
Value after reset:	0	1	0	0	0	1	1	1	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b3 to b0	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b7 to b4	OPDE[3:0]	Option Data Enable	Sets the option data to be output. Be sure to use the following setting; otherwise, the operation is not guaranteed. b7 b4 0000: Output disabled 1000: OPD3 1100: OPD3, OPD2 1110: OPD3, OPD2, OPD1 1111: OPD3, OPD2, OPD1, OPD0 Other than above: Setting prohibited	R/W
b11 to b8	ADE[3:0]	Address Enable	Sets the address to be output. Be sure to use the following setting; otherwise, the operation is not guaranteed. b11 b8 0000: Output disabled 0111: Address[23:0] 1111: Address[31:0] Other than above: Setting prohibited	R/W
b12	OCDE	Optional Command Enable	Sets the optional command to be output. 0: Optional command output disabled 1: Optional command output enabled	R/W
b13	—	Reserved	This bit is always read as 0. The write value should always be 0.	R/W
b14	CDE	Command Enable	Sets the command to be output. 0: Command output disabled 1: Command output enabled	R/W
b15	DME	Dummy Cycle Enable	Enables insertion of the dummy cycle before the read data. Note: A setting is prohibited for a transfer starting with a dummy cycle. 0: Dummy cycle insertion disabled 1: Dummy cycle insertion enabled	R/W
b17, b16	DRDB[1:0]	Data Read Bit Size	Sets the data read size in bit units. b17 b16 0 0: 1 bit 0 1: 2 bits 1 0: 4 bits 1 1: Setting prohibited	R/W

Bit	Symbol	Bit Name	Description	R/W
b19, b18	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b21, b20	OPDB[1:0]	Option Data Bit Size	Sets the option data size in bit units. b21 b20 0 0: 1 bit 0 1: 2 bits 1 0: 4 bits 1 1: Setting prohibited	R/W
b23, b22	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b25, b24	ADB[1:0]	Address Bit Size	Sets the address size in bit units. b25 b24 0 0: 1 bit 0 1: 2 bits 1 0: 4 bits 1 1: Setting prohibited	R/W
b27, b26	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b29, b28	OCDB[1:0]	Optional Command Bit Size	Sets the optional command size in bit units. b29 b28 0 0: 1 bit 0 1: 2 bits 1 0: 4 bits 1 1: Setting prohibited	R/W
b31, b30	CDB[1:0]	Command Bit Size	Sets the command size in bit units. For details, see the details about the serial flash memory to be used. b31 b30 0 0: 1 bit 0 1: 2 bits 1 0: 4 bits 1 1: Setting prohibited	R/W

27.2.9 SPI Mode Control Register (SMCR)

The SMCR register is a 32-bit register that sets the operation in SPI operating mode.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Address(es): A000 5020h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	SSLKP	—	—	—	—	—	SPIRE	SPIWE	SPIE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SPIE	SPI Data Transfer Enable	Data is transferred by setting this bit to 1. This bit is enabled only when the TEND bit in CMNSR is set to 1. The operation cannot be guaranteed when this bit is set to 1 with the TEND bit set to 0. This bit is always read as 0. Note: When the SPBSSL signal is inactive, the command, optional command, address, and option data that are set as the output by the DRENR register are output even if the SPIRE and SPIWE bits are set to 0. When the SPBSSL signal is active, follow the notes described in section 27.4.2, Notes on Starting Transfer from the SPBSSL Signal Retained State in SPI Operating Mode.	R/W
b1	SPIWE	Data Write Enable	Sets write operation in SPI operating mode. 0: Data writing disabled 1: Data writing enabled Note: When the transfer data bit size is set to 2 bits or 4 bits with the SPIDB[1:0] bits, the SPIRE and SPIWE bits should not be set to 1 at the same time.	R/W
b2	SPIRE	Data Read Enable	Sets read operation in SPI operating mode. 0: Data reading disabled 1: Data reading enabled Note: When the transfer data bit size is set to 2 bits or 4 bits with the SPIDB[1:0] bits, the SPIRE and SPIWE bits should not be set to 1 at the same time.	R/W
b7 to b3	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b8	SSLKP	SPBSSL Signal Level	Determines the status of the SPBSSL signal after the end of transfer. 0: SPBSSL signal is inactivated at the end of transfer. 1: SPBSSL signal level is maintained from the end of transfer to the start of next access.	R/W
b31 to b9	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W

27.2.10 SPI Mode Command Setting Register (SMCMR)

The SMCMR register is a 32-bit register that sets the commands issued in SPI operating mode.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Address(es): A000 5024h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	CMD[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	OCMD[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	OCMD[7:0]	Optional Command	Sets the optional command.	R/W
b15 to b8	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b23 to b16	CMD[7:0]	Command	Sets the command. For details, see the details about the serial flash memory to be used.	R/W
b31 to b24	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W

27.2.11 SPI Mode Address Setting Register (SMADR)

The SMADR register is a 32-bit register that sets the addresses of the serial flash memory in SPI operating mode.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Address(es): A000 5028h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ADRE[7:0]								ADR[23:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ADR[23:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b23 to b0	ADR[23:0]	Address	Sets the address of the serial flash memory.	R/W
b31 to b24	ADRE[7:0]	Address	Sets the value of bits 31 to 24 when the serial flash address is output in 32-bit units. This setting is valid when ADE[3] in SMENR is 1.	R/W

27.2.12 SPI Mode Option Setting Register (SMOPR)

The SMOPR register is a 32-bit register that sets the option data in SPI operating mode.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Address(es): A000 502Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	OPD3[7:0]								OPD2[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	OPD1[7:0]								OPD0[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	OPD0[7:0]	Option Data 0	Sets the option data 0.	R/W
b15 to b8	OPD1[7:0]	Option Data 1	Sets the option data 1.	R/W
b23 to b16	OPD2[7:0]	Option Data 2	Sets the option data 2.	R/W
b31 to b24	OPD3[7:0]	Option Data 3	Sets the option data 3.	R/W

Note: OPD3, OPD2, OPD1, and OPD0 are output in this order.

27.2.13 SPI Mode Enable Setting Register (SMENR)

The SMENR register is a 32-bit register that sets the bit size of the command, optional command, address, option data, and transfer data in SPI operating mode and enables various outputs. Disabling output of all of the command, optional command, address, option data, dummy cycle, and transfer data is prohibited. At least one of them except dummy cycle must be enabled.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Address(es): A000 5030h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	CDB[1:0]		OCDB[1:0]		—	—	ADB[1:0]		—	—	OPDB[1:0]		—	—	SPIDB[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	DME	CDE	—	OCDE	ADE[3:0]				OPDE[3:0]				SPIDE[3:0]			
Value after reset:	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b3 to b0	SPIDE[3:0]	Transfer Data Enable	Sets transfer data. The settings below must be used. Otherwise, the operation is not guaranteed. b3 b0 0000: Not transferred 1000: 8 bits transferred (enables data at address 0 of the SPI mode read/write data registers 0) 1100: 16 bits transferred (enables data at addresses 0 and 1 of the SPI mode read/write data registers 0) 1111: 32 bits transferred (enables data at addresses 0 to 3 of the SPI mode read/write data registers 0) Other than above: Setting prohibited	R/W
b7 to b4	OPDE[3:0]	Option Data Enable	Sets the option data to be output. Use only the settings given below. Otherwise, the operation cannot be guaranteed. b7 b4 0000: Output disabled 1000: OPD3 1100: OPD3, OPD2 1110: OPD3, OPD2, OPD1 1111: OPD3, OPD2, OPD1, OPD0 Other than above: Setting prohibited	R/W
b11 to b8	ADE[3:0]	Address Enable	Sets the address to be output. Use only the settings given below. Otherwise, the operation cannot be guaranteed. b11 b8 0000: Output disabled 0100: ADR[23:16] 0110: ADR[23:8] 0111: ADR[23:0] 1111: ADR[31:0] Other than above: Setting prohibited	R/W
b12	OCDE	Optional Command Enable	Sets the optional command to be output. 0: Optional command output disabled 1: Optional command output enabled	R/W
b13	—	Reserved	This bit is always read as 0. The write value should always be 0.	R/W
b14	CDE	Command Enable	Sets the command to be output. 0: Command output disabled 1: Command output enabled	R/W

Bit	Symbol	Bit Name	Description	R/W
b15	DME	Dummy Cycle Enable	Enables insertion of the dummy cycle before the read data. Note 1. Dummy cycle insertion is prohibited for write in SPI operating mode including the case in which a transfer ends with a dummy cycle. Note 2. A setting is prohibited for a transfer starting with a dummy cycle. 0: Dummy cycle insertion disabled 1: Dummy cycle insertion enabled	R/W
b17, b16	SPIDB[1:0]	Transfer Data Bit Size	Sets the transfer data size in bit units. b17 b16 0 0: 1 bit 0 1: 2 bits 1 0: 4 bits 1 1: Setting prohibited	R/W
b19, b18	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b21, b20	OPDB[1:0]	Option Data Bit Size	Sets the option data size in bit units. b21 b20 0 0: 1 bit 0 1: 2 bits 1 0: 4 bits 1 1: Setting prohibited	R/W
b23, b22	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b25, b24	ADB[1:0]	Address Bit Size	Sets the address size in bit units. b25 b24 0 0: 1 bit 0 1: 2 bits 1 0: 4 bits 1 1: Setting prohibited	R/W
b27, b26	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b29, b28	OCDB[1:0]	Optional Command Bit Size	Sets the optional command size in bit units. b29 b28 0 0: 1 bit 0 1: 2 bits 1 0: 4 bits 1 1: Setting prohibited	R/W
b31, b30	CDB[1:0]	Command Bit Size	For details, see the details about the serial flash memory to be used. b31 b30 0 0: 1 bit 0 1: 2 bits 1 0: 4 bits 1 1: Setting prohibited	R/W

27.2.14 SPI Mode Read Data Register 0 (SMRDR0)

The SMRDR0 register is a 32-bit register that stores the read data in SPI operating mode.
The setting of this register should be read when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.
The alignment of data depends on the access size. For details, see section 27.3.4, Data Alignment.
This register must be accessed in the same unit as the transfer size set in the SPIDE[3:0] bits in the SPI mode enable register (SMENR). It must also be accessed from its own address 0.

Address(es): A000 5038h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	RDATA0[31:0]															
Value after reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RDATA0[31:0]															
Value after reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	RDATA0 [31:0]	Read Data	Holds the data read in SPI operating mode.	R

The contents of this register are modified upon completion of reception in SPI operating mode. Be sure to read data when reception in SPI operating mode is completed.

27.2.15 SPI Mode Write Data Register 0 (SMWDR0)

The SMWDR0 register is a 32-bit register that sets the write data in SPI operating mode.
The setting of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.
The alignment of data depends on the access size. For details, see section 27.3.4, Data Alignment.
This register must be accessed in the same unit as the transfer size set in the SPIDE[3:0] bits in the SPI mode enable register (SMENR). It must also be accessed from its own address 0.

Address(es): A000 5040h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	WDATA0[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	WDATA0[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	WDATA0 [31:0]	Write Data	Holds the data to be written in SPI operating mode.	R/W

27.2.16 Common Status Register (CMNSR)

The CMNSR register is a 32-bit register that holds flags indicating the operating state.
The settings of this register are reflected both in external address space read mode and SPI operating mode.

Address(es): A000 5048h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SSLF	TEND
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b0	TEND	Transfer End Flag	Indicates whether the data transfer has ended. 0: Indicates that data transfer is in progress. 1: Indicates that data transfer has ended.	R
b1	SSLF	SPBSSL Pin Monitor	0: SPBSSL pin is inactive. 1: SPBSSL pin is active.	R
b31 to b2	—	Reserved	These bits are always read as 0.	R

27.2.17 Data Read Dummy Cycle Setting Register (DRDMCR)

The DRDMCR register is a 32-bit register that sets the size and number of dummy cycles to be inserted in external address space read mode.

The settings of this register are enabled when the DME bit in the data read enable setting register (DRENR) is 1.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Address(es): A000 5058h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DMDB[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	DMCYC[2:0]		
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	DMCYC [2:0]	Number of Dummy Cycles	Sets the number of dummy cycles to be inserted when the DME bit in the data read enable setting register (DRENR) is 1. b2 b0 000: 1 cycle 001: 2 cycles 010: 3 cycles 011: 4 cycles 100: 5 cycles 101: 6 cycles 110: 7 cycles 111: 8 cycles	R/W
b15 to b3	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b17, b16	DMDB [1:0]	Dummy Cycle Bit Size	Sets the dummy cycle size in bit units. The setting of these bits is combined with the setting of the IO0FV, IO2FV, and IO3FV bits in the common control register (CMNCR) to determine the state of the unused pins during the dummy cycles. The state of the used pins is Hi-Z. b17 b16 0 0: 1 bit 0 1: 2 bits 1 0: 4 bits 1 1: Setting prohibited	R/W
b31 to b18	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W

27.2.18 SPI Mode Dummy Cycle Setting Register (SMDMCR)

The SMDMCR register is a 32-bit register that sets the size and number of dummy cycles to be inserted in SPI operating mode.

The settings of this register are enabled when the DME bit in the SPI mode enable setting register (SMENR) is 1.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Address(es): A000 5060h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DMDB[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	DMCYC[2:0]		
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	DMCYC[2:0]	Number of Dummy Cycles	Sets the number of dummy cycles to be inserted when the DME bit in the SPI mode enable setting register (SMENR) is 1. b2 b0 000: 1 cycle 001: 2 cycles 010: 3 cycles 011: 4 cycles 100: 5 cycles 101: 6 cycles 110: 7 cycles 111: 8 cycles	R/W
b15 to b3	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b17, b16	DMDB [1:0]	Dummy Cycle Bit Size	Sets the dummy cycle size in bit units. The setting of these bits is combined with the setting of the IO0FV, IO2FV, and IO3FV bits in the common control register (CMNCR) to determine the state of the unused pins during the dummy cycles. The state of the used pins is Hi-Z. b17 b16 00: 1 bit 01: 2 bits 10: 4 bits 11: Setting prohibited	R/W
b31 to b18	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W

27.3 Operation

27.3.1 System Configuration

With this module, one serial flash memory can be directly connected (data size of 1, 2, and 4 bits).
Figure 27.2 shows a system configuration example.

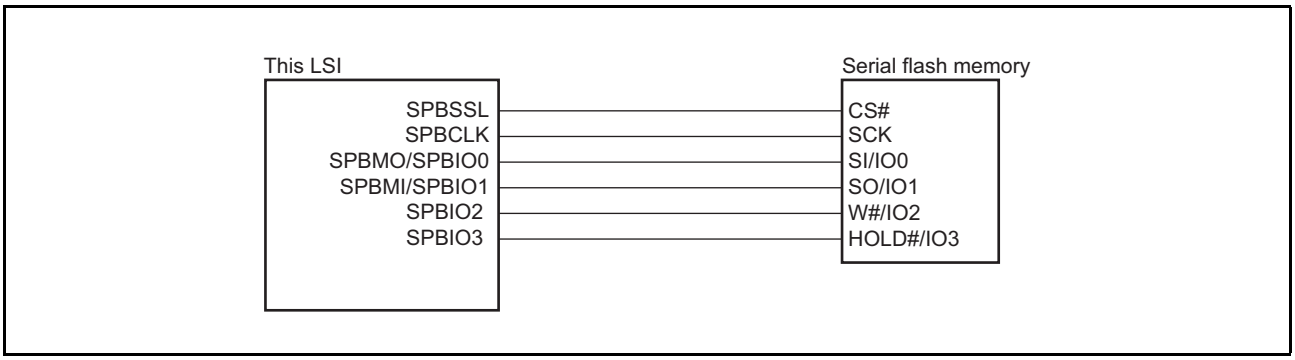


Figure 27.2 System Configuration Example with 4-Bit Data Size of a Serial Flash Memory Connected (BSZ[1:0] Bits in CMNCR = 00)

27.3.2 Address Map

In external address space read mode, the serial flash connected is assigned in the SPI multi I/O bus space. By the DREAR register setting, a maximum of 4 Gbytes can be accessed.

Table 27.4 Address Map

Number of Serial Flash Memories Connected	Internal Address	Max. Access Area
1	1000 0000h to 13FF FFFFh	4 Gbytes
	3000 0000h to 33FF FFFFh (mirror area)	

27.3.3 32-Bit Serial Flash Addresses

Since the SPI multi I/O bus space is 64 Mbytes, only a part of the 32-bit serial flash address area can be directly accessed. Here, the fixed value set in the DREAR register is used as the upper bit value of a 32-bit address.

To output serial flash addresses in 32 bits, set the ADE[3] bit in DRENR to 1, set the range of the external addresses used as the serial flash addresses to the EAC[2:0] bits in DREAR, and set the upper bit value of the 32-bit address as the fixed value to the EAV[7:0] bits in DREAR.

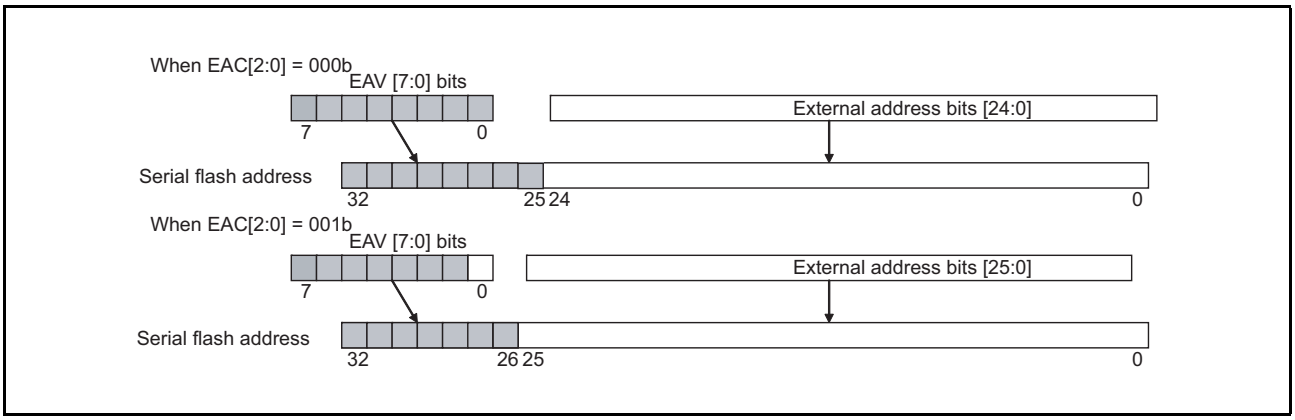


Figure 27.3 32-Bit Address Setting

Setting the ADE[3] bit in DRENR to 1 allows the serial flash address to be output using [31:0] bits.

When EAC[2:0] = 000b, external address bits [24:0] are valid; set the value for [32:25] bits to EAV[7:0].

When EAC[2:0] = 001b, external address bits [25:0] are valid; set the value for [32:26] bits to EAV[7:1].

When one serial flash memory is connected, address bits [31:0] are used.

27.3.4 Data Alignment

Data alignment can be set by using the SFDE bit in the common control register (CMNCR). Data alignment in data read mode and in SPI mode are shown in Figure 27.4 and Figure 27.5, respectively.

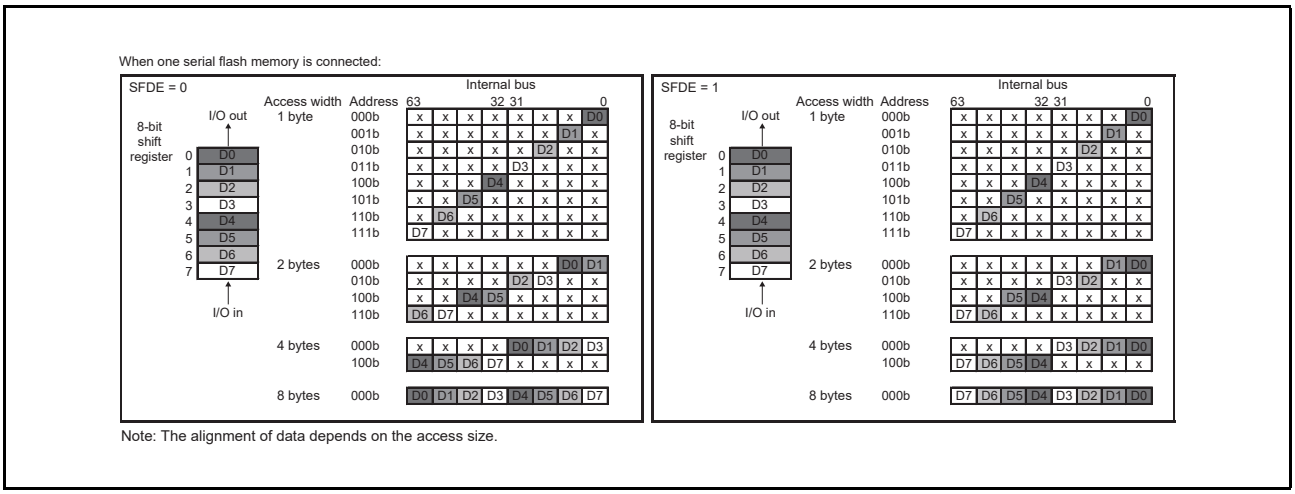


Figure 27.4 Data Alignment in External Address Space Read Mode

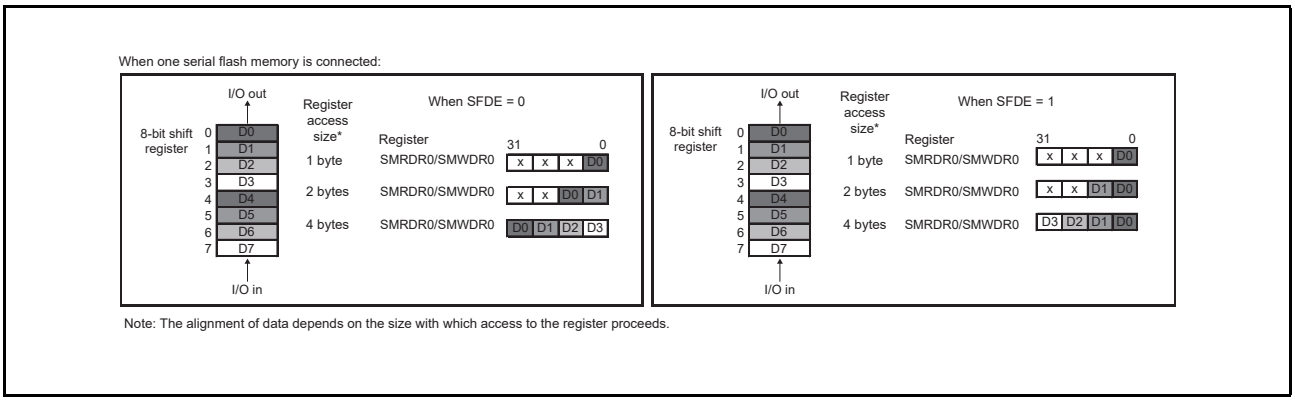


Figure 27.5 Data Alignment in SPI Operating Mode

27.3.5 Operating Modes

This module has two operating modes: external address space read mode and SPI operating mode.
In external address space read mode, a read access to the SPI multi I/O bus space is converted into SPI communication and data is received. After data acquisition, data is returned to the bus master that is the issuing source. For details, see section 27.3.6, External Address Space Read Mode.
In SPI operating mode, arbitrary SPI communication is carried out based on the register settings. For details, see section 27.3.8, SPI Operating Mode.

27.3.6 External Address Space Read Mode

A read access to the SPI multi I/O bus space can be converted into SPI communication in external address space read mode. Further, the commands, optional commands, option data, and dummy cycle issued for reading can be modified by the register settings.
In external address space read mode, either normal read operation or burst read operation can be selected. The transfer format is determined by the settings of the common control register (CMNCR), SSL delay register (SSLDR), bit rate setting register (SPBCR), data read control register (DRCR), data read command setting register (DRCMR), data read extended address setting register (DREAR), data read option setting register (DROPR), data read enable setting register (DRENR), and data read dummy cycle setting register (DRDMCR).

(1) Normal Read Operation

When the RBE bit in DRCR is set to 0, normal read operation is performed.
In the normal read operation, the data of 8 bits, 16 bits, and 32 bits are read for respectively a byte, a word, and a longword read access. After reading, the SPBSSL signal is inactivated.
The normal read operation timing is shown in Figure 27.6.
t1 is the time period from the time the SPBSSL signal is activated to the time the clock is output from the SPBCLK signal (clock delay). t2 is the time period from transmission of the last edge of the SPBCLK signal of a transfer to the time the SPBSSL signal is inactivated (SPBSSL negation delay). t3 is the time period from one transfer end to the next transfer start (next access). For details of t1, t2, and t3, see section 27.3.9, Transfer Format.

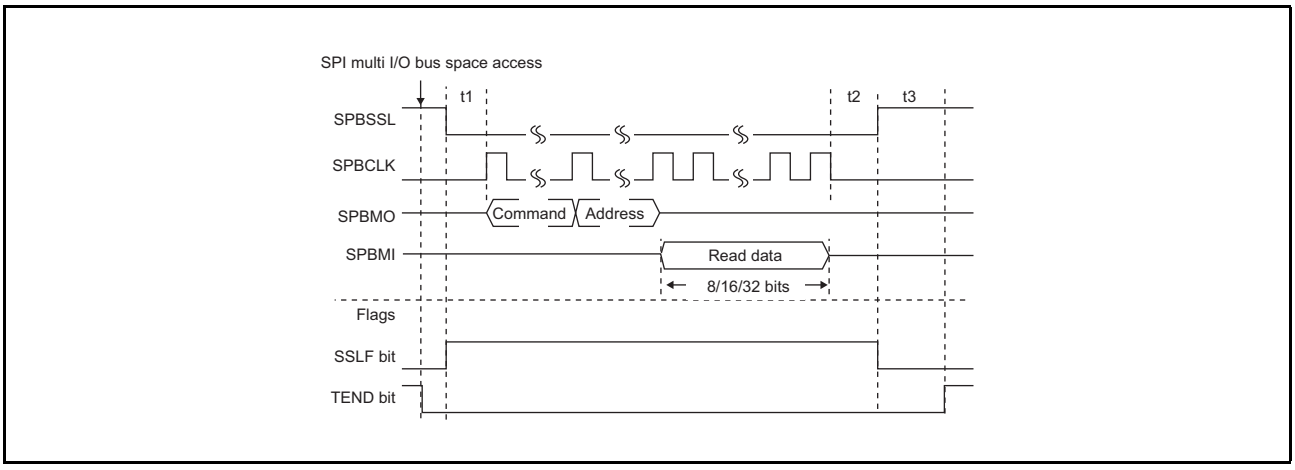


Figure 27.6 Normal Read Operation Timing

(2) Burst Read Operation

When the RBE bit in DRCR is set to 1, burst read operation is performed. Read cache is enabled in the burst read operation. For read cache operation, see section 27.3.7, Read Cache.

For reading bytes, words, or longwords, the read cache is first referred to for the data. When the read cache contains the data, the data is read from the read cache without accessing the serial flash memory. When the read cache does not contain the data, burst read operation is performed in the serial flash memory and the read data is stored in the read cache. The data transfer length at that time is 64 bits × RBURST[3:0] bits and the data is always read from the 64-bit boundary.

The status of the SPBSSL signal after data transfer can be selected by using the SSLE bit in DRCR. When the SSLE bit is set to 0, the SPBSSL signal is always inactivated after data transfer. For an operation performed when the SSLE bit is set to 1, see section 27.3.6, (3) Burst Read Operation with Automatic SPBSSL Inactivation, just below.

A pattern diagram of this operation and a burst read operation timing diagram when SSLE bit is set to 0 are shown in Figure 27.7 and Figure 27.8.

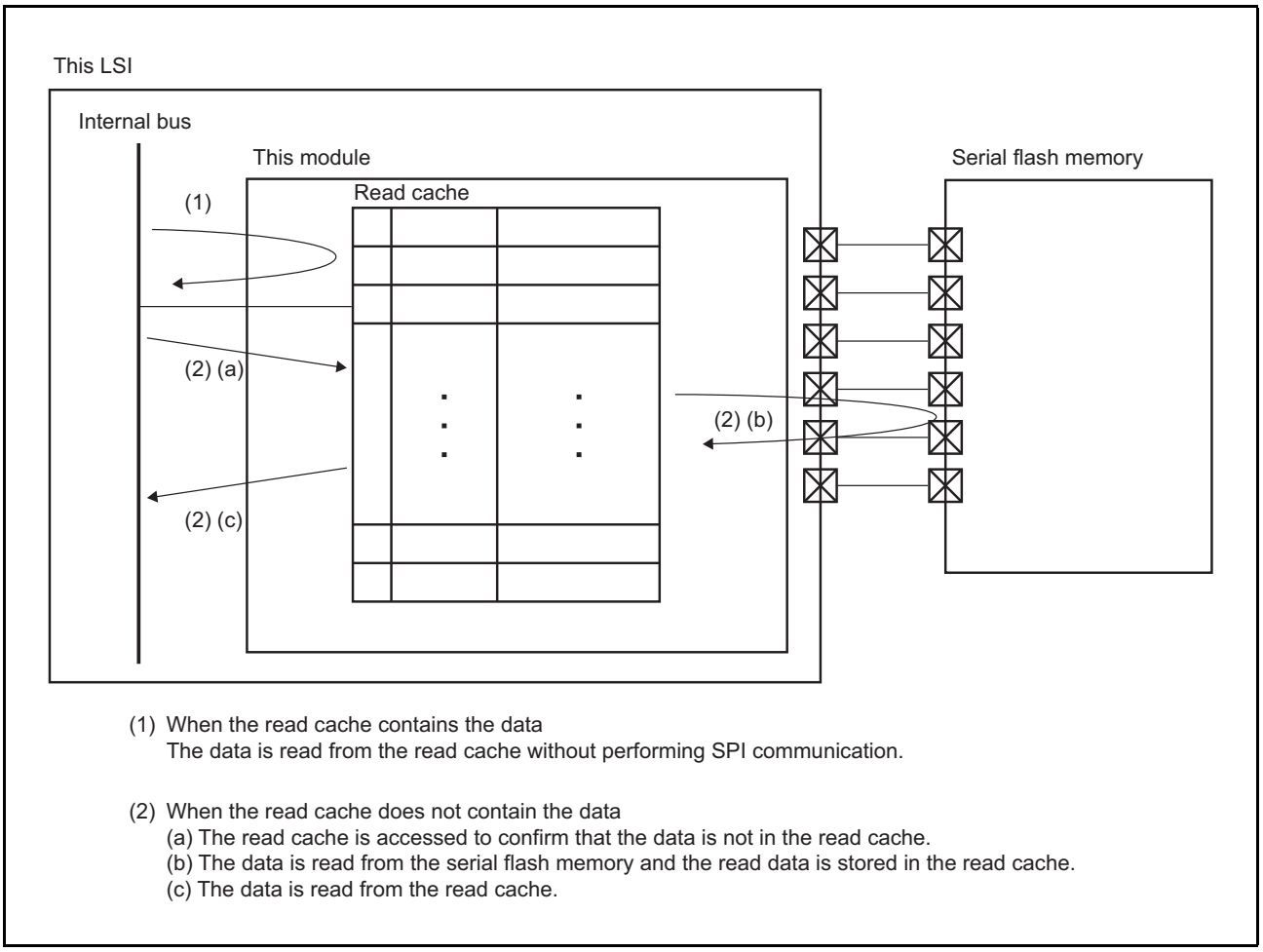


Figure 27.7 Burst Read Operation

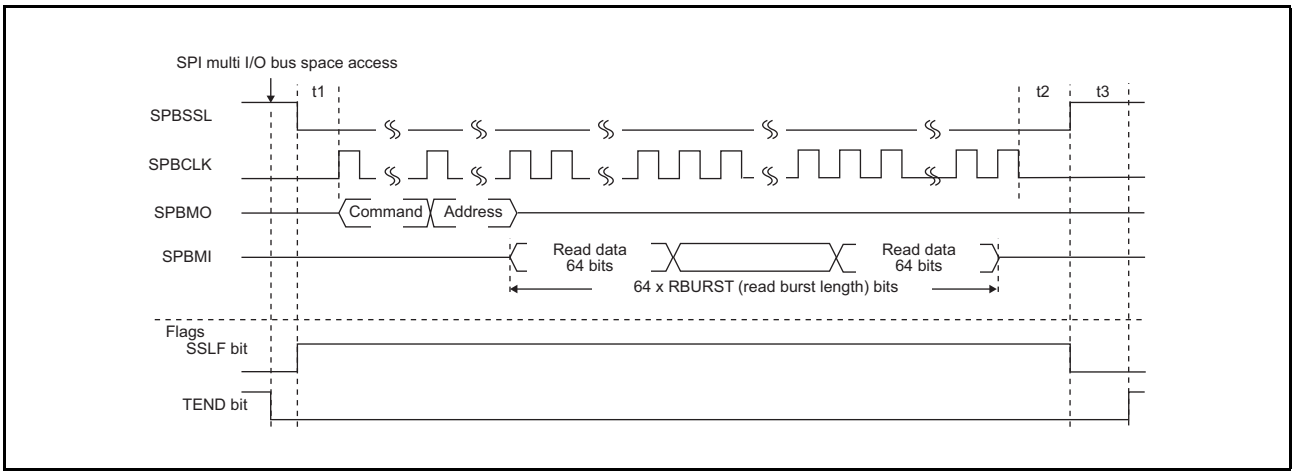


Figure 27.8 Burst Read Operation Timing (SSLE Bit = 0)

(3) Burst Read Operation with Automatic SPBSSL Inactivation

When SSLE bit in DRCR is set to 1, this module does not inactivate the SPBSSL signal after the burst read transfer. When accessing the next time, if the address is continuous with the previous read address, the burst read operation is performed without issuing the command, optional command, address, option data, or dummy cycle. If the address is not continuous with the previous read address, the SPBSSL signal is once inactivated and the burst read operation is performed after issuing the command, optional command, address, option data, or dummy cycle.
Burst read timing diagrams for continuous address and non-continuous address are shown in Figure 27.9 and Figure 27.10.

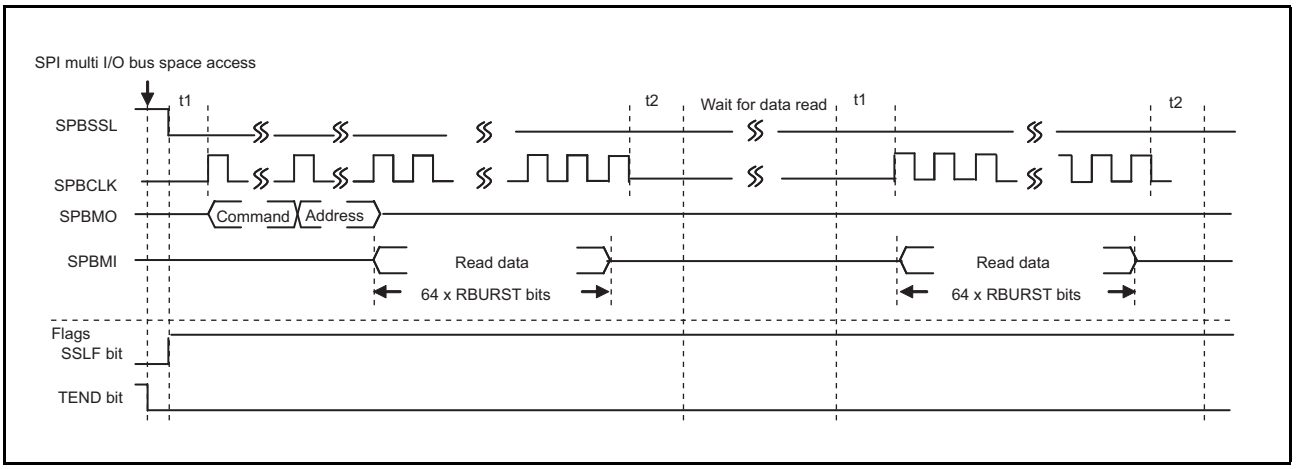


Figure 27.9 Burst Read Timing for Continuous Address (SSLE Bit = 1)

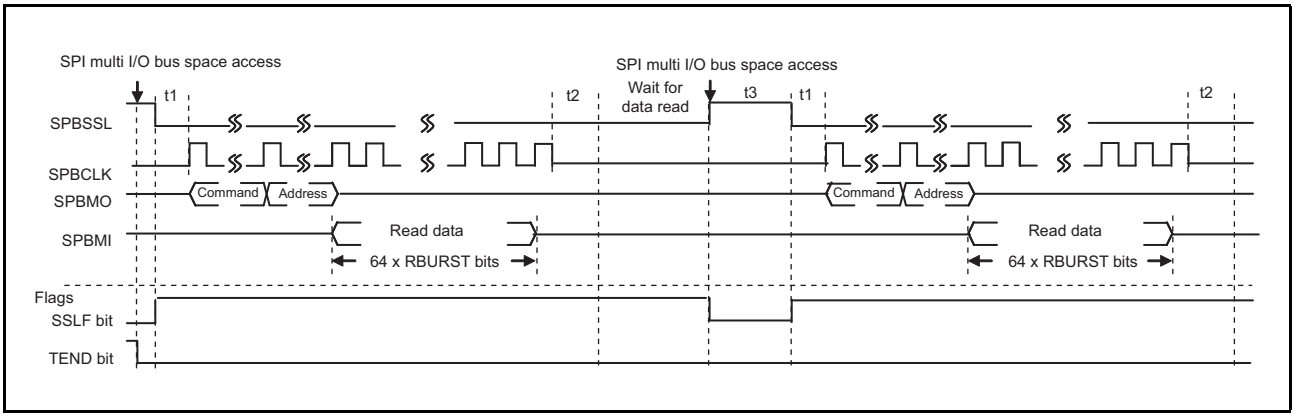


Figure 27.10 Burst Read Timing for Non-Continuous Address (SSLE Bit = 1)

For the next access after inactivation of the SPBSSL signal with the SSLN bit in DRCR with this operation, read SSLF = 0 in CMNSR to confirm that the SPBSSL signal has been inactivated.

(4) Initial Setting Flow

An example of an initial setting flow in external address space read mode is shown in Figure 27.11.

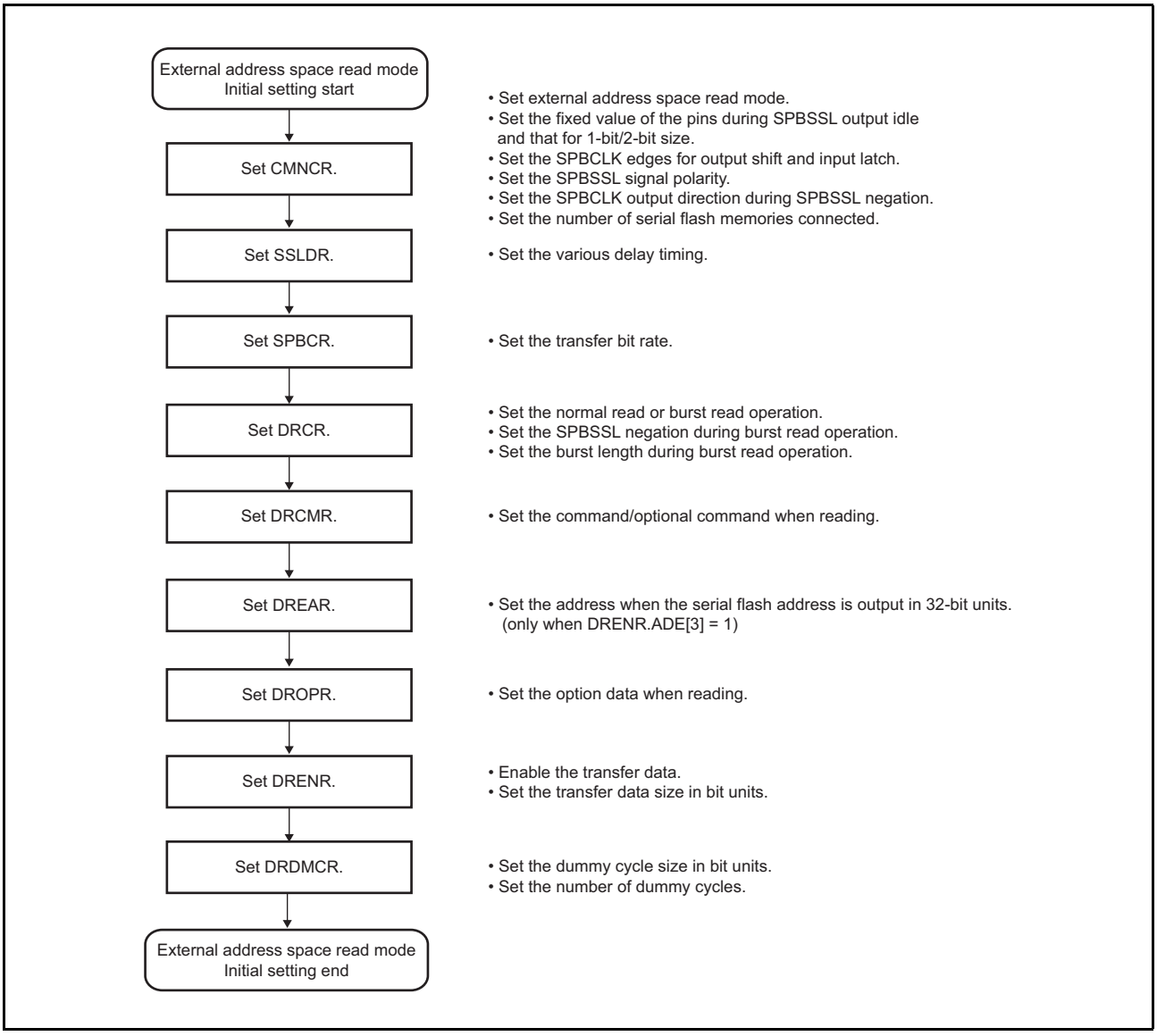


Figure 27.11 Example of Initial Setting Flow in External Address Space Read Mode

27.3.7 Read Cache

This module has a simple built-in read cache. The read cache can be used during external address space read mode and burst read operation. The read cache is configured with a line size of 64 bits and 16 entries.
Read cache configuration is shown in Figure 27.12.

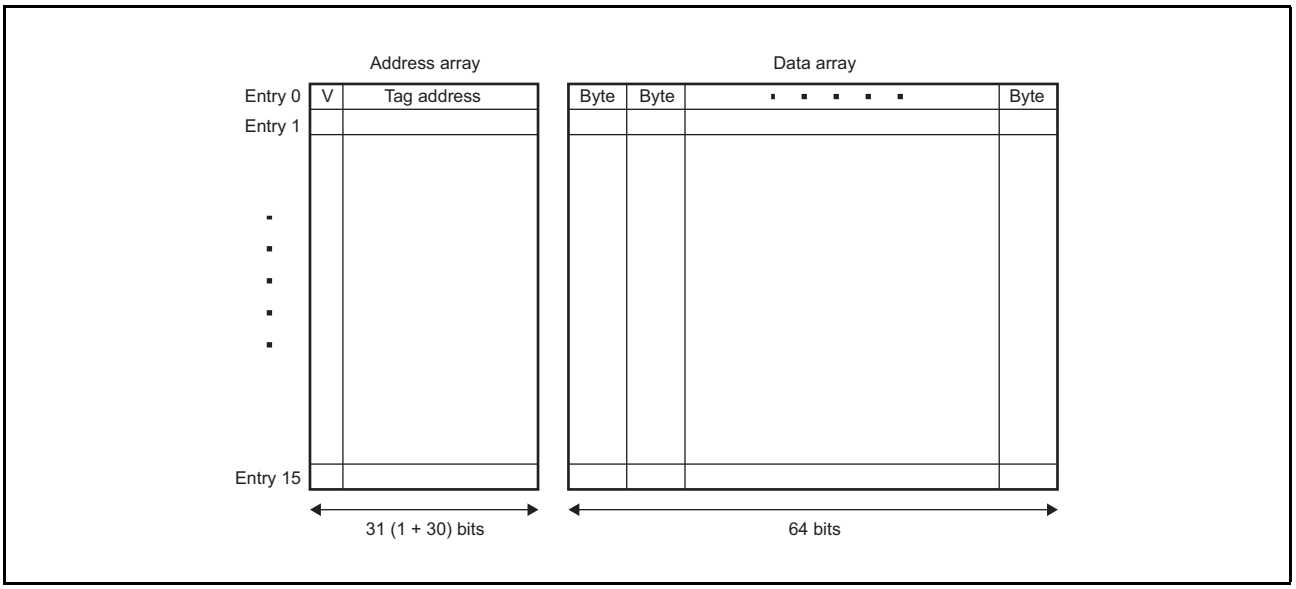


Figure 27.12 Read Cache Configuration

(1) Address Array

The V bit in Figure 27.12 indicates whether the entry data is valid. When the V bit is 1, the data is valid and when V bit is 0, the data is invalid.
The tag address bits hold the address used by the serial flash memory. An address is composed of bits 32 to 3. Address bits 23 to 3 are enabled when address output is 24 bits.
Address bits 31 to 3 are enabled when address output is 32 bits.

(2) Data Array

It retains the 64-bit read data. Registration in the read cache is performed in line units.

(3) Read Operation

If read data is hit in the cache, data is read from the read cache. In case of miss-hit, after the 64 × RBURST (read burst length) data is read from the serial flash memory and the read cache is updated, the data is returned to the bus master.

(4) Data Replacement

Data update is managed by the write pointer. In case of miss-hit of read data, the RBURST (read burst length) portion data is replaced starting at the entry specified by the write pointer. In other words, the data is replaced in the storage order of the data. Whether data is referred to or not will not affect the replacement order of data.

27.3.8 SPI Operating Mode

This module can carry out an arbitrary SPI operation by using the register settings.
The transfer format is determined by the settings of the common control register (CMNCR), SSL delay register (SSLDR), bit rate setting register (SPBCR), SPI mode control register (SMCR), SPI mode command setting register (SMCMR), SPI mode address setting register (SMADR), SPI mode option setting register (SMOPR), and SPI mode enable setting register (SMENR), SPI mode read data register (SMRDR), SPI mode write data register (SMWDR), and SPI mode dummy cycle setting register (SMDMCR).
SPI operating mode can be used for reading the status of the serial flash memory and writing to the serial flash memory. In this mode, one transfer refers to the operation from when the SPIE bit in SMCR is set to 1 to when the TEND bit is set to 1.

(1) Transfer Start

The transfer of data is started in the set transfer format by setting the SPIE bit in SMCR to 1. When write operation is enabled, the SPI mode write data register is transmitted to the serial flash memory. When read operation is enabled, data read from the serial flash memory is stored into the SPI mode read data register.
The SPI operation timing is shown in Figure 27.13.

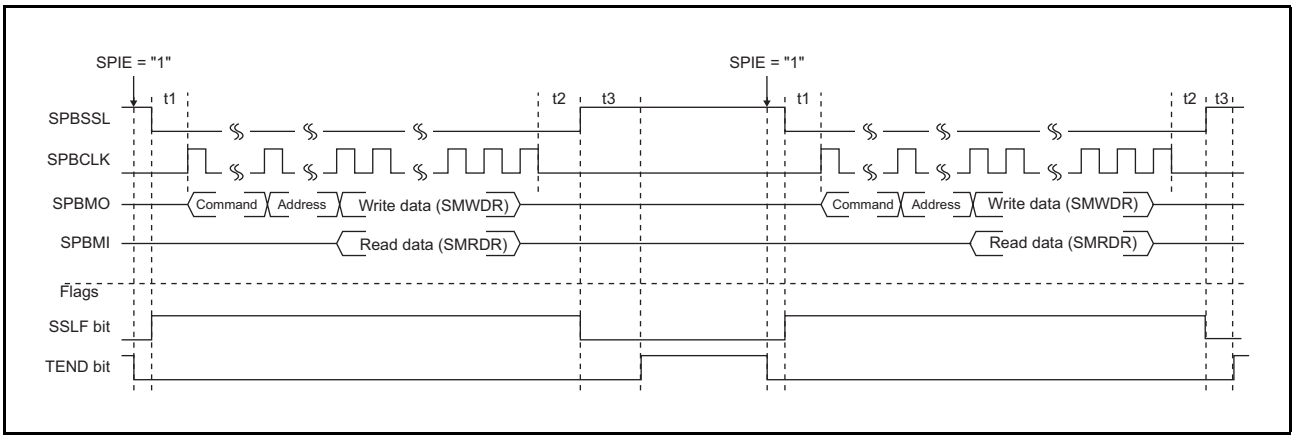


Figure 27.13 SPI Operation Timing

(2) Read/Write Enable

- Read operation: Data can be read by setting the SPIRE bit in SMCR to 1. The read data is stored into SMRDR.
- Write operation: Data can be written by setting the SPIWE bit in SMCR to 1. The data stored in SMWDR is output.

When the data size is set to 1 bit using the SPIDB[1:0] bits in SMENR, data can be transmitted and received by setting the SPIRE and SPIWE bits to 1. However, when the data size is set to 2 or 4 bits by using the SPIDB[1:0] bits, only one of the SPIRE and SPIWE bits should be enabled. The operation is not guaranteed if both the bits are enabled.

(3) Retention of SPBSSL Pin Activation

By setting the SSLKP bit in SMCR to 1, activation of the SPBSSL signal can be continued till the next transfer. With this function, the transfer can be carried out continuously with the SPBSSL signal kept in the active state.
The data transfer timing using the SSLKP bit is shown in Figure 27.14.

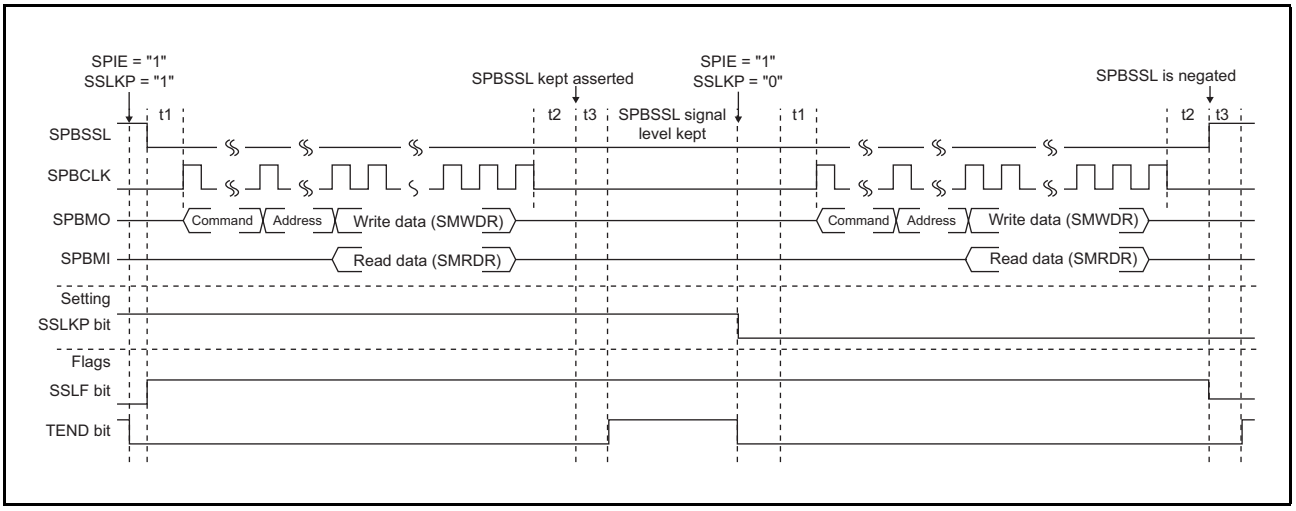


Figure 27.14 Data Transfer Timing using the SSLKP Bit

(4) Initial Setting Flow

An example of an initial setting flow in SPI operating mode is shown in Figure 27.15.

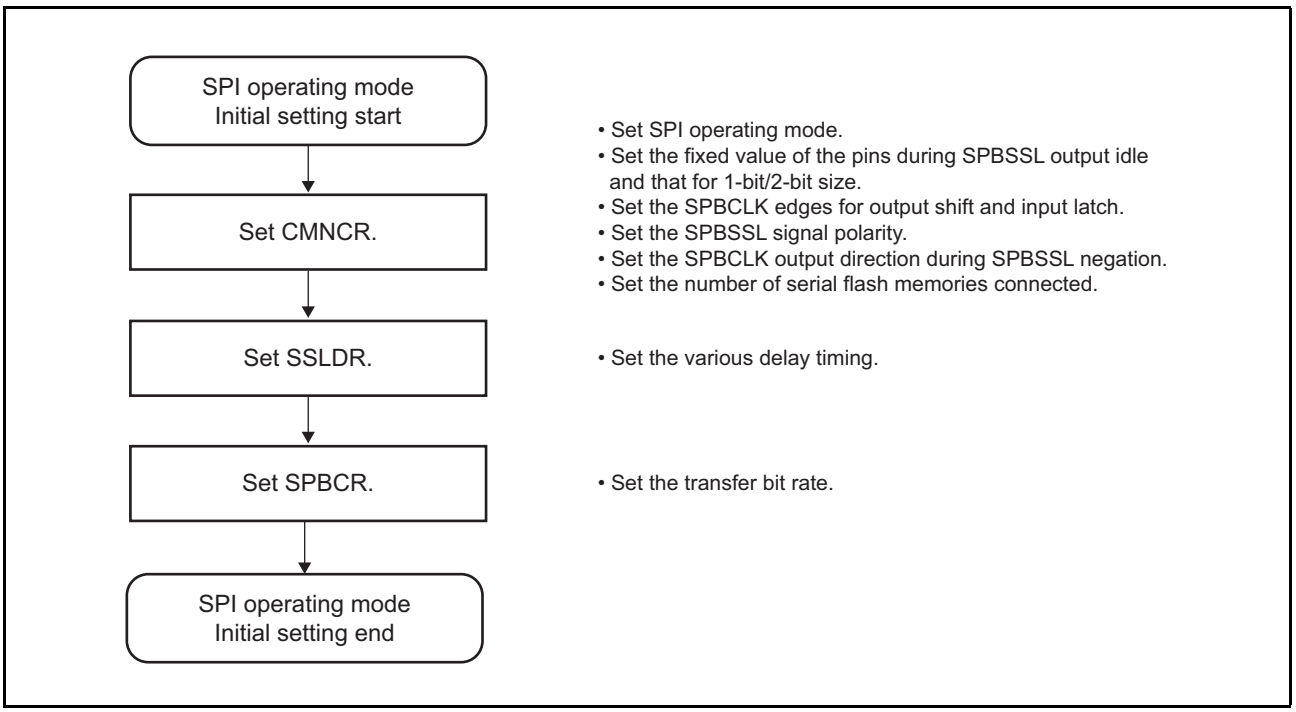


Figure 27.15 Example of Initial Setting Flow in SPI Operating Mode

(5) Data Transfer Setting Flow

An example of a data transfer setting flow in SPI operating mode is shown in Figure 27.16.

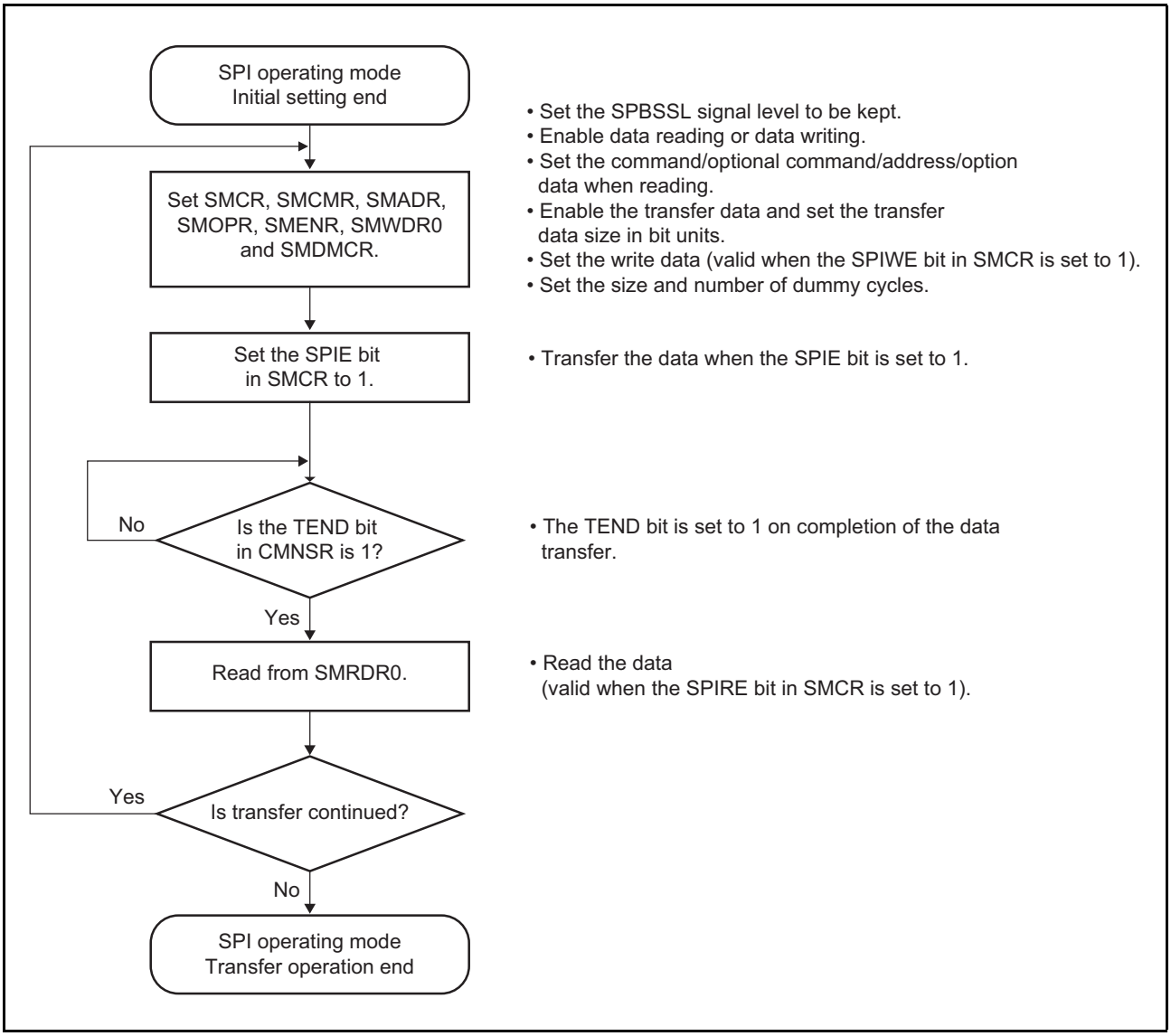


Figure 27.16 Example of a Data Transfer Setting Flow in SPI Operating Mode

27.3.9 Transfer Format

(1) SPBSSL Pin Enable Polarity Control

The enable polarity of the SPBSSL signal can be changed with the SSLP bit in CMNCR.

(2) SPBCLK Output

The output level of the SPBCLK signal while the SPBSSL signal is inactive can be set with the CPOL bit in CMNCR.

(3) Data Transmission and Reception Timing

Data is transmitted and received at either the odd or even edges. The data transmission timing can be set to the odd or even edge with the CPHAT bit in CMNCR. Similarly, the data reception timing can be set to the odd or even edge with the CPHAR bit in CMNCR.

(4) Delay Settings

t1 is the time period from the time the SPBSSL signal is activated to the clock output of the SPBCLK signal (clock delay). It can be set with the SCKDL[2:0] bits in SSLDR. t2 is the time period from the time the clock output of the SPBCLK signal is stopped to the time the SPBSSL signal is inactivated (SPBSSL negation delay). It can be set with the SLNDL[2:0] bits in SSLDR. t3 is the time period required to prevent SPBSSL signal activation for the next transfer after the end of the previous transfer (next access delay). It can be set with the SPNDL[2:0] bits in SSLDR.

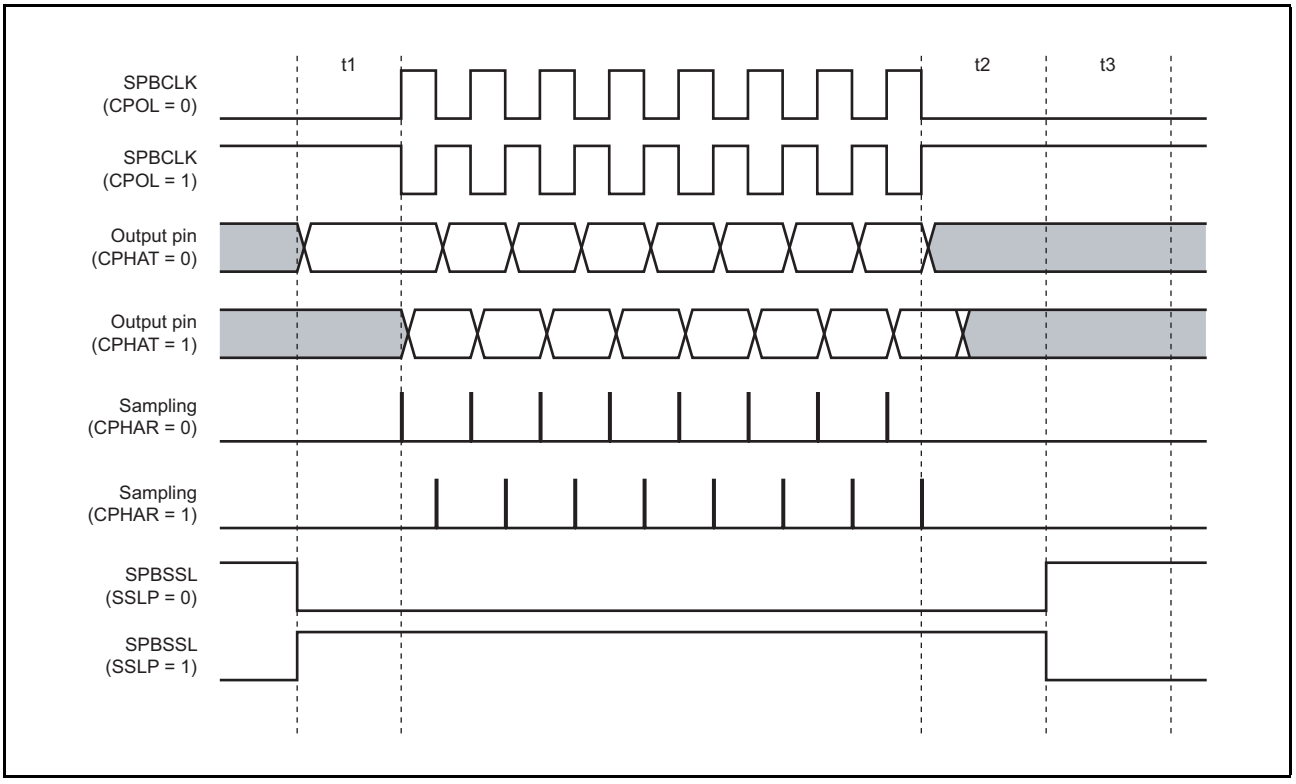


Figure 27.17 SDR Transfer Format

27.3.10 Data Format

This module can input and output data in the order of command, optional command, address, option data, dummy cycle, and data.

(1) Data Registers

Table 27.5 shows the input and output data.

Table 27.5 Data Registers

Data	External Address Space Read Mode	SPI Operating Mode
Command (8 bits)	CMD[7:0] bits in DRCMR	CMD[7:0] bits in SMCMR
Optional command (8 bits)	OCMD[7:0] bits in DRCMR	OCMD[7:0] bits in SMCMR
Address (32/24 bits)	32 bits: DREAR.EAV[6:1 to 0] bits + lower [25 to 24:0] bits of the read address. 24 bits: Lower [23:0] bits of the read address	32 bits: ADR[31:0] bits in SMADR 24 bits: ADR[23:0] bits in SMADR
Option data (8 bits × 4)	DROPR	SMOPR
Dummy cycle (1 to 8 cycles)	DRDMCR	SMDMCR (only when read)
Transfer data	Normal read: 8/16/32 bits Burst read: 64 × RBURST bits	Read: SMRDR0, SMRDR1 Write: SMWDR0, SMWDR1

(2) Data Enable

In external address space read mode, transfer enable of the command, optional command, address, option data, and dummy cycle can be controlled with the CDE, OCDE, ADE[3:0], OPDE[3:0], and DME bits in DRENR, respectively. The size and number of dummy cycles can be controlled with the data read dummy cycle setting register (DRDMCR). Similarly, in SPI operating mode, enable of the command, optional command, address, option data, dummy cycle, and transfer data can be controlled with the CDE, OCDE, ADE[3:0], OPDE[3:0], DME, and SPIDE[3:0] bits in SMENR, respectively. However, disabling all the above parameters is prohibited in SPI operating mode. At least one of them except dummy cycle must be enabled. The size and number of dummy cycles can be controlled with the SPI mode dummy cycle setting register (SMDMCR).

For the address and option data in external address space read mode; and the address, option data, and transfer data in SPI operating mode, the enable bit setting allowed is determined according to the transfer data size. For the allowed setting combinations of the enable bits and transfer data size, refer to the description of the pertinent register.

If these enable bits are disabled, the data is not output, and input and output of the next data is carried out. The command, optional command, address, and option data are always output. During dummy cycles, the state of the used pins is Hi-Z. In external address space read mode, data is always input; and in SPI operating mode, input and output of data is determined based on the settings of the SPIRE and SPIWE bits in SMCR.

There are some restrictions on dummy cycle insertion; refer to the description of the DME bits in DRENR and SMENR for details.

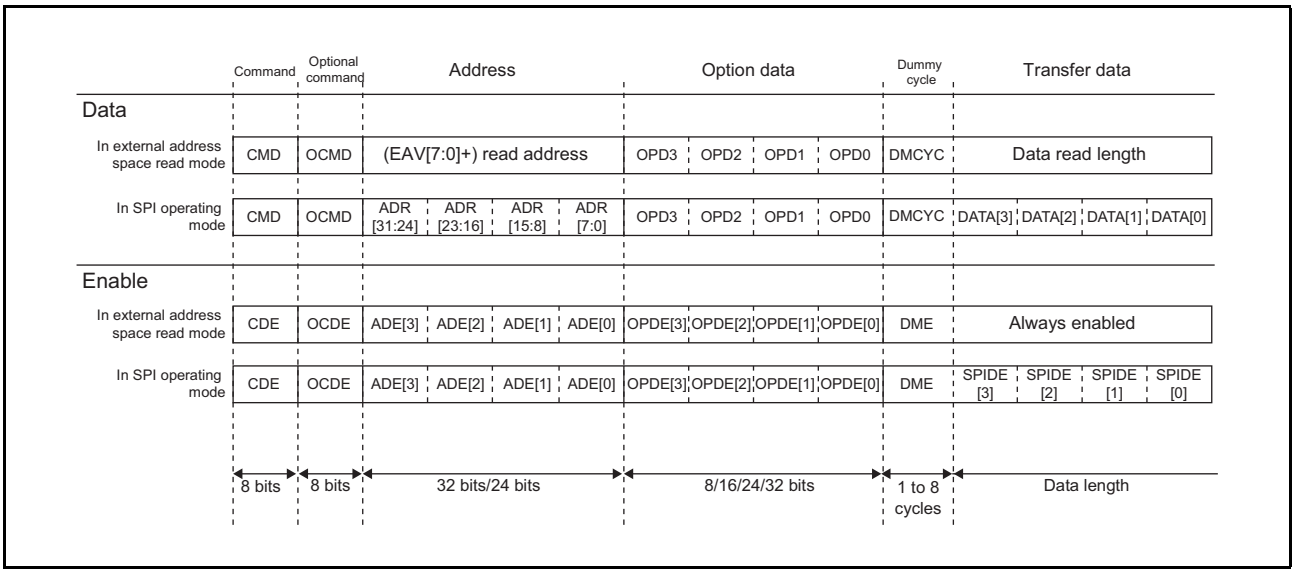


Figure 27.18 Data and Enable

(3) Bit Size

In external address space read mode, the size of the command, optional command, address, option data, and the read data in bit units is respectively controlled with the CDB[1:0], OCDB[1:0], ADB[1:0], OPDB[1:0], and DRDB[1:0] bits in DRENR. The size of the dummy cycle in bit units is also controlled with the DMDB[1:0] bits in DRDMCR.
Similarly, in SPI operating mode, the size of the command, optional command, address, option data, and read write data in bit units is controlled with the CDB[1:0], OCDB[1:0], ADB[1:0], OPDB[1:0], and SPIDB[1:0] bits in SMENR. The size of the dummy cycle in bit units is also controlled with the DMDB[1:0] bits in SMDMCR.

(a) 1-bit size

When the size is set to 1 bit, SPBMI pin will be the input pins and SPBMO pin will be the output pins. SPBIO2, and SPBIO3 pins are not used.
Figure 27.19 show the transfer format example.

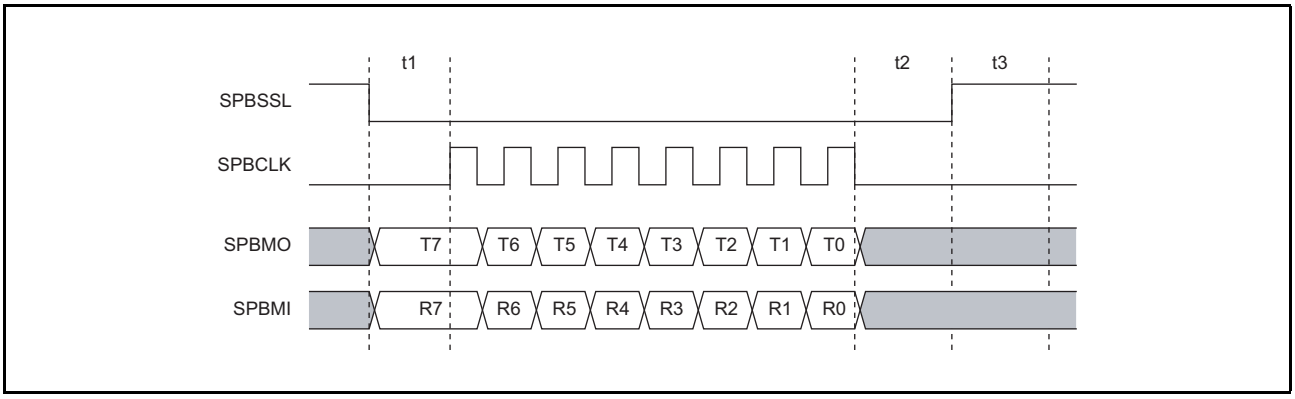


Figure 27.19 Transfer Format Example with 1-Bit Data Size and One Serial Flash Memory Connected

(b) 2-bit size

When the size is set to 2 bits, SPBIO0 and SPBIO1 pins will be either the input pins or the output pins. SPBIO2, and SPBIO3 pins are not used.
Figure 27.20 show the transfer format example.

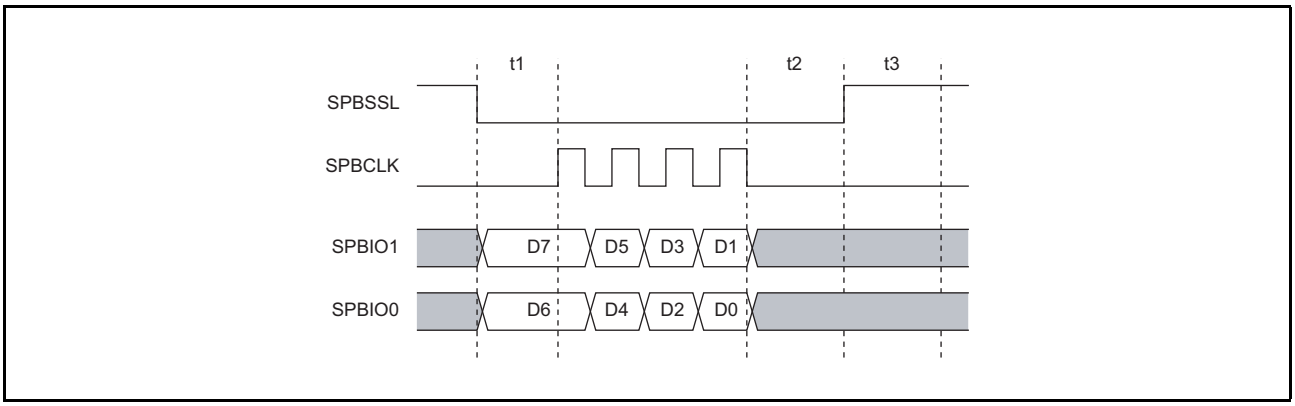


Figure 27.20 Transfer Format Example with 2-Bit Data Size and One Serial Flash Memory Connected

(c) 4-bit size

When the size is set to 4 bits, SPBIO0, SPBIO1, SPBIO2, and SPBIO3 pins will be either the input pins or the output pins.

Figure 27.21 show the transfer format examples.

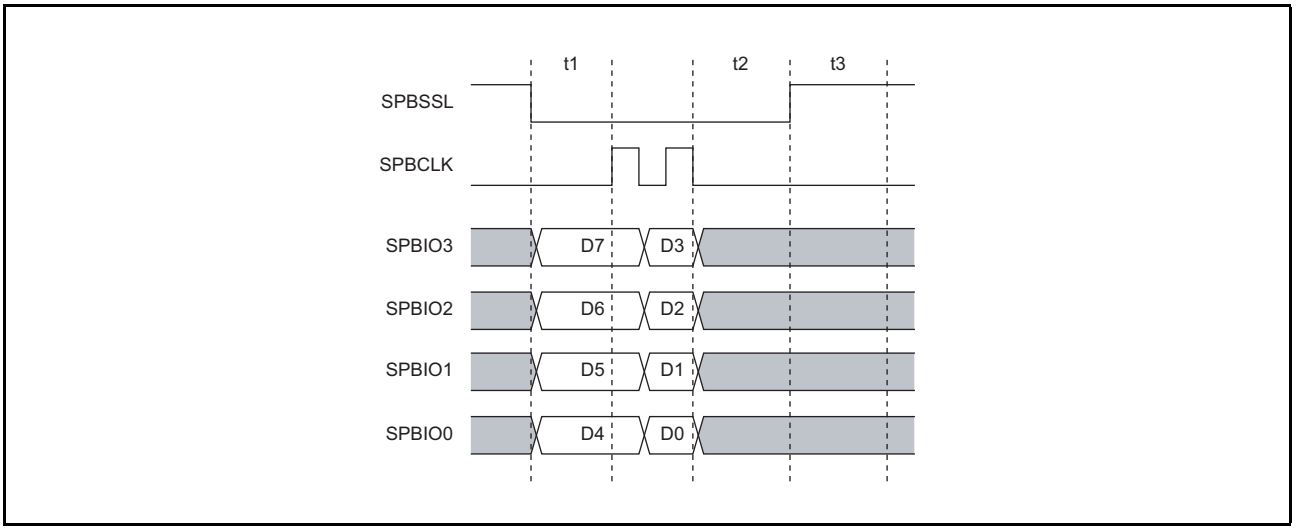


Figure 27.21 Transfer Format Example with 4-Bit Data Size and One Serial Flash Memory Connected

27.3.11 Data Pin Control

With this module, the status of pins can be automatically changed based on the data size to be used and the read/write settings. The inactivation status of the SPBSSL signal can be set with the MOIIO3, MOIIO2, MOIIO1, and MOIIO0 bits in CMNCR. The SPBSSL and SPBCLK pins are always output pins. The status of respective pins is specified in Table 27.6 to Table 27.9.

Table 27.6 Pin Status (1)

		SPBSSL Activation		
		Command, Optional Command, Address, Option Data		
Pin	**SPBSSL Inactivation**	**1-Bit Size**	**2-Bit Size**	**4-Bit Size**
SPBMO/SPBIO0	MOIIO0 bit value	Output	Output	Output
SPBMI/SPBIO1	MOIIO1 bit value	Hi-Z	Output	Output
SPBIO2	MOIIO2 bit value	IO2FV bit value	IO2FV bit value	Output
SPBIO3	MOIIO3 bit value	IO3FV bit value	IO3FV bit value	Output

Table 27.7 Pin Status (2)

	Transfer Data					
	External Address Space Read Mode			SPI Operating Mode		
				SPIRE Bit = 1, SPIWE Bit = 0		
Pin	**1-Bit Size**	**2-Bit Size**	**4-Bit Size**	**1-Bit Size**	**2-Bit Size**	**4-Bit Size**
SPBMO/SPBIO0	IO0FV bit value	Input	Input	IO0FV bit value	Input	Input
SPBMI/SPBIO1	Input	Input	Input	Input	Input	Input
SPBIO2	MOIIO2 bit value	MOIIO2 bit value	Input	MOIIO2 bit value	MOIIO2 bit value	Input
SPBIO3	MOIIO3 bit value	MOIIO3 bit value	Input	MOIIO3 bit value	MOIIO3 bit value	Input

Table 27.8 Pin Status (3)

	Transfer Data					
	SPI Operating Mode					
	SPIRE Bit = 0, SPIWE Bit = 1			SPIRE Bit = 1, SPIWE Bit = 1		
Pin	**1-Bit Size**	**2-Bit Size**	**4-Bit Size**	**1-Bit Size**	**2-Bit Size**	**4-Bit Size**
SPBMO/SPBIO0	Output	Output	Output	Output	Setting prohibited	Setting prohibited
SPBMI/SPBIO1	Hi-Z	Output	Output	Input	Setting prohibited	Setting prohibited
SPBIO2	MOIIO2 bit value	MOIIO2 bit value	Output	MOIIO2 bit value	Setting prohibited	Setting prohibited
SPBIO3	MOIIO3 bit value	MOIIO3 bit value	Output	MOIIO3 bit value	Setting prohibited	Setting prohibited

Table 27.9 Pin Status (4)

	Dummy Cycle		
Pin	**1-Bit Size**	**2-Bit Size**	**4-Bit Size**
SPBMO/SPBIO0	IO0FV bit value	Hi-Z	Hi-Z
SPBMI/SPBIO1	Hi-Z	Hi-Z	Hi-Z
SPBIO2	IO2FV bit value	IO2FV bit value	Hi-Z
SPBIO3	IO3FV bit value	IO3FV bit value	Hi-Z

27.3.12 SPBSSL Pin Control

Inactivation conditions of the SPBSSL signal are as follows.

(1) External Address Space Read Mode

(a) Normal read operation (RBE bit in DRCR = 0)

SPBSSL inactivated after the data transfer and t2 cycle are completed.

(b) Burst read without automatic SPBSSL inactivation (RBE bit in DRCR = 1, SSLE bit in DRCR = 0)

SPBSSL inactivated after the data transfer and t2 cycle are completed.

(c) Burst read with automatic SPBSSL inactivation (RBE bit in DRCR = 1, SSLE bit in DRCR = 1)

- SPBSSL inactivated after t2 cycle when the read address is not continuous with the previously read address
- SPBSSL inactivated after the SSLN bit in DRCR is set to 1

(2) SPI Operating Mode

(a) SPBSSL pin activation not retained (SSLKP bit in SMCR = 0)

SPBSSL inactivated after the data transfer and t2 cycle are completed.

(b) SPBSSL pin activation retained (SSLKP bit in SMCR = 1)

SPBSSL not inactivated.
When to be inactivated, data should be transferred after setting the SSLKP bit to 0.

27.3.13 Flags

This module has two flag bits SSLF and TEND in CMNSR. These bits are read-only bits.

(1) SSLF Bit

This bit indicates the SPBSSL pin status. The status is 1 when the SPBSSL signal is active, and the status is 0 when the SPBSSL signal is inactive.

(2) TEND Bit

This bit indicates whether transfer of data is in progress or the transfer of data has ended.
During t1 time period, data transfer, t2 time period, t3 time period, and waiting for read access by burst read and SPBSSL automatic inactivation, the TEND bit is read as 0 to indicate that the transfer of data is in progress.
When other than the above, the TEND bit is read as 1 to indicate that transfer of data has ended.

(3) Register Re-writing Timing

The status of the TEND bit determines the rewritable registers.
The registers which can be written to, except the SSLN bit in DRCR, should be modified when TEND = 1.
Read SMRDR0 when TEND = 1.
CMNSR can always be read.

27.4 Usage Notes

27.4.1 Notes on Transfer to Read Data in SPI Operating Mode

If the setting for the bit mode is for division by two or more in SPI operating mode, take note of the following points for caution when setting the SPI mode enable setting register (SMENR) to enable transfer only for reading data.
“Transfer only for reading data” indicates transfer to read data while the CDE, OCDE, ADE[3:0], and OPDE[3:0] bits in SMENR are all 0.

(1) Transfer to read data while the signal on the SPBSSL pin is inactivated

Set the SMENR.SPIDE[3:0] bits to 1100b or 1111b when transfer only for reading data is to proceed.
Transfer will not proceed normally if the setting of the SMENR.SPIDE[3:0] bits is 1000b.

(2) Transfer to read data while the signal on the SPBSSL pin is activated

When transfer only for reading data is to proceed, set the SMENR.SPIDE[3:0] bits to 1100b or 1111b, or end the immediately preceding transfer with reading data.
When the immediately preceding transfer is of a command, optional command, address, or option data, or is transfer for writing data, the subsequent transfer only for reading data will not proceed normally if the setting of the SMENR.SPIDE[3:0] bits is 1000b.

27.4.2 Notes on Starting Transfer from the SPBSSL Signal Retained State in SPI Operating Mode

Be sure to set the SPIWE bit in the SMCR register to 1 when the transfer of a command, optional command, address, or option data is started while the SPBSSL signal is active in SPI operating mode.

27.4.3 Note on Initialization

When using this module, do not set both SPBR[7:0] = 00h and BRDV[1:0] = 00b in the bit rate setting register (SPBCR).

28. CRC Operation Units (CRC)

Cyclic redundancy check (CRC) operation units generate CRC codes.

28.1 Overview

Table 28.1 describes the CRC operation unit specifications. Figure 28.1 shows a block diagram of a CRC operation unit.

Table 28.1 CRC Operation Unit (CRC) Specifications

Item	Specifications
Data subject to CRC operation	A CRC code can be generated for any data that is 8, 16, or 32 bits long.
CRC generation polynomial expression	One of the following polynomials can be selected: • 32-bit Ethernet CRC (32-Ethernet) $X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$ • 16-bit CCITT CRC (16-CCITT) $X^{16} + X^{12} + X^5 + 1$ • 8-bit SAE J1850 CRC (8-SAE J1850) $X^8 + X^4 + X^3 + X^2 + 1$ • 8-bit 0x2F CRC (8-0x2F) $X^8 + X^5 + X^3 + X^2 + X + 1$
Low-power consumption function	Module-stop state can be set.

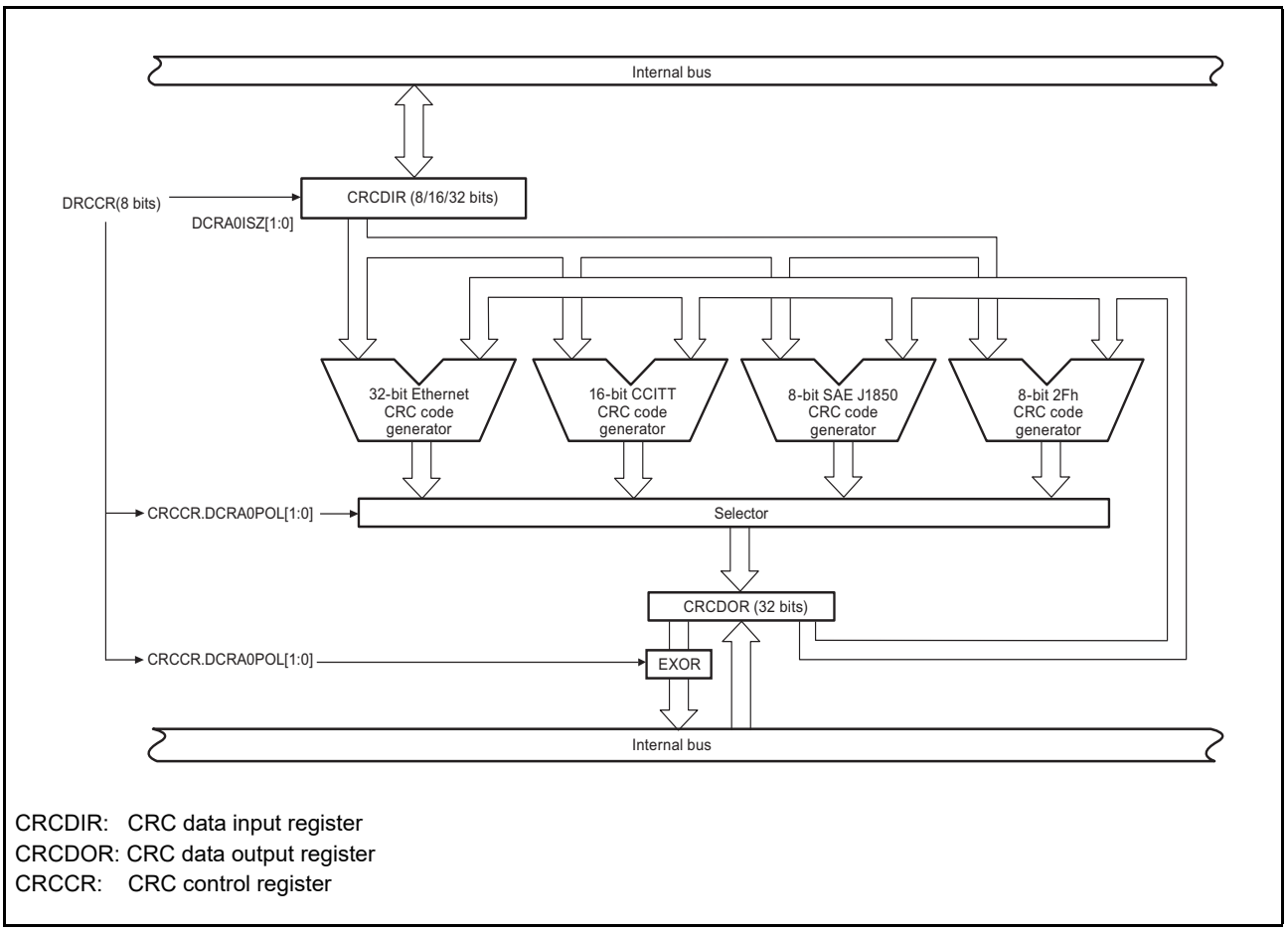


Figure 28.1 Block Diagram of a CRC Operation Unit (CRC)

28.2 Register Descriptions

28.2.1 CRC Data Input Register (CRCDIR)

The CRCDIR register stores the input data for CRC calculation. CRC calculation starts when data is written to this register.

The valid bit width used for CRC calculation must be set for CRCCR.DCRA0ISZ[1:0]. Before the first data is written to this register, the CRCDOR register must be initialized by writing the initial starting value. For details on initialization, see section 28.3.1, Initializing the CRC Data Output Register (CRCDOR).

Address(es): A007 C000h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	CRCDIR[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	CRCDIR[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	CRCDIR[31:0]	Input Data for CRC Calculation	The following bit widths are supported: • CRC input bit width of 32 bits: CRCDIR[31:0] • CRC input bit width of 16 bits: CRCDIR[15:0] • CRC input bit width of 8 bits: CRCDIR[7:0]	R/W

Byte order

The byte order of the CRCDIR register differs depending on the selected CRC generation method.

- If the 32-Ethernet CRC polynomial expression is used to generate a code (CRCCR.DCRA0POL[1:0] = 00b), the LSB-first order is used (LSB: least significant byte). For example, if the CRC input bit width is 8 bits (DCRA0ISZ[1:0] = 10b), the bits 7 to 0 of the CRCDIR register are the LSB.
- If the 16-CCITT CRC polynomial expression is used to generate a code (CRCCR.DCRA0POL[1:0] = 01b), the MSB-first order is used (MSB: most significant byte). For example, if the CRC input bit width is 8 bits (DCRA0ISZ[1:0] = 10b), the bits 7 to 0 of the CRCDIR register are the MSB.
- If the 8-SAE J1850 CRC polynomial expression is used to generate a code (CRCCR.DCRA0POL[1:0] = 10b), the MSB-first order is used (MSB: most significant byte). For example, if the CRC input bit width is 8 bits (DCRA0ISZ[1:0] = 10b), the bits 7 to 0 of the CRCDIR register are the MSB.
- If the 8-0x2F CRC polynomial expression is used to generate a code (CRCCR.DCRA0POL[1:0] = 11b), the MSB-first order is used (MSB: most significant byte). For example, if the CRC input bit width is 8 bits (DCRA0ISZ[1:0] = 10b), the bits 7 to 0 of the CRCDIR register are the MSB.

28.2.2 CRC Data Output Register (CRCDOR)

The CRCDOR register stores the CRC code calculated based on the selected CRC generation polynomial expression.

Address(es): A007 C004h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	CRCDOR[31:0]															
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	CRCDOR[31:0]															
Value after reset: *1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Note 1. After a reset, 32-bit Ethernet CRC is selected as the CRC generation polynomial expression. Therefore, when the bits are read, 0000 0000h is obtained as the result of EXOR operation.

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	CRCDOR[31:0]	Resulting CRC Code	• 32-Ethernet: CRCDOR[31:0] • 16-CCITT: CRCDOR[15:0] (Bits 31 to 16 are undefined.) • 8-SAE J1850/8-0x2F: CRCDOR[7:0] (Bits 31 to 8 are undefined.) The value read from this register is the result of EXOR operation with the following value: • 32-Ethernet: FFFF FFFFh • 16-CCITT: 0000h • 8-SAE J1850/8-0x2F: FFh	R/W

Note: This register must be initialized (by setting the initial starting value) before the first data for CRC calculation is written to the CRCDIR register. For details on initialization, see section 28.3.1, Initializing the CRC Data Output Register (CRCDOR).

CRCDOR[31:0] Bits

The CRC code calculated based on the CRC generation polynomial expression selected by CRCCR.DCRA0POL[1:0] is stored.

Resulting CRC code:

- If 32-Ethernet is used, the resulting CRC code is returned to CRCDOR[31:0].
- If 16-CCITT is used, the resulting CRC code is returned to CRCDOR[15:0].
  The settings of bits 31 to 16 are undefined.
- If 8-SAE J1850/8-0x2F is used, the resulting CRC code is returned to CRCDOR[7:0].
  The settings of bits 31 to 8 are undefined.

The value read from the bits is the result of EXOR operation with the following EXOR value:

EXOR value:

- 32-Ethernet: FFFF FFFFh
- 16-CCITT: 0000h
- 8-SAE J1850/8-0x2F: FFh

After a reset, 32-bit Ethernet is selected as the CRC generation polynomial expression. Therefore, when the bits are read, 0000 0000h is obtained as the result of EXOR operation of FFFF FFFFh (initial value stored in the bits) and FFFF FFFFh (EXOR value).

Example:

Assume that the value of the CRCDOR[31:0] bits is 5555 5555h as the CRC code produced by the 32-bit Ethernet CRC algorithm. In this case, when these bits are read, AAAA AAAAh is obtained as the result of EXOR operation with the EXOR value for this algorithm, i.e. FFFF FFFFh.

28.2.3 CRC Control Register (CRCCR)

The CRCCR register controls the CRC generation polynomial expression and the CRC input bit width.

Address(es): A007 C020h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	DCRA0ISZ[1:0]		—	—	DCRA0POL[1:0]	
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	DCRA0POL [1:0]	CRC Generation Mode Specification	Specify the mode in which to generate a CRC code. b1 b0 0 0: 32-Ethernet ($X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$) 0 1: 16-CCITT ($X^{16} + X^{12} + X^5 + 1$) 1 0: 8-SAE J1850 ($X^8 + X^4 + X^3 + X^2 + 1$) 1 1: 8-0x2F ($X^8 + X^5 + X^3 + X^2 + X + 1$)	R/W
b3, b2	—	Reserved	These bits are read as 0.	R
b5, b4	DCRA0ISZ [1:0]	CRC Input Bit Width Specification	Specify the CRC input bit width. b1 b0 0 0: 32 bits (CRCDIR[31:0]) 0 1: 16 bits (CRCDIR[15:0]) 1 0: 8 bits (CRCDIR[7:0]) 1 1: Setting prohibited	R/W
b7, b6	—	Reserved	These bits are read as 0.	R

Note 1. If the CRC generation mode (CRCCR.DCRA0POL) is changed or the CRC input bit width (CRCCR.DCRA0ISZ) is changed, the CRCDOR register must be initialized (by setting the initial starting value). For details, see section 28.3.1, Initializing the CRC Data Output Register (CRCDOR).

Note 2. The CRC input bit width (CRCCR.DCRA0ISZ[1:0]) must be set according to the block unit of data for CRC calculation. The CRC input bit width must not be changed during CRC calculation. The CRC input bit width can be changed after the final CRC calculation result is read from the CRCDOR register. In this case, before the next data for CRC calculation is written to the CRCDIR register, the CRCDOR register must be initialized (by setting the initial starting value).

28.3 Operation

The CRC operation unit calculates and generates the CRC code for a block of a specific length. Data for which the CRC is to be calculated can be set in the CRC data input register (CRCDIR) in 8-, 16-, or 32- bit units. When data are written to the CRC data input register (CRCDIR), CRC calculation based on the selected CRC generation polynomial expression starts. Before writing the first value to the CRC data input register (CRCDIR), the CRCDOR register must be initialized by setting it to its initial value.
The following shows an overview of using the CRC operation unit.

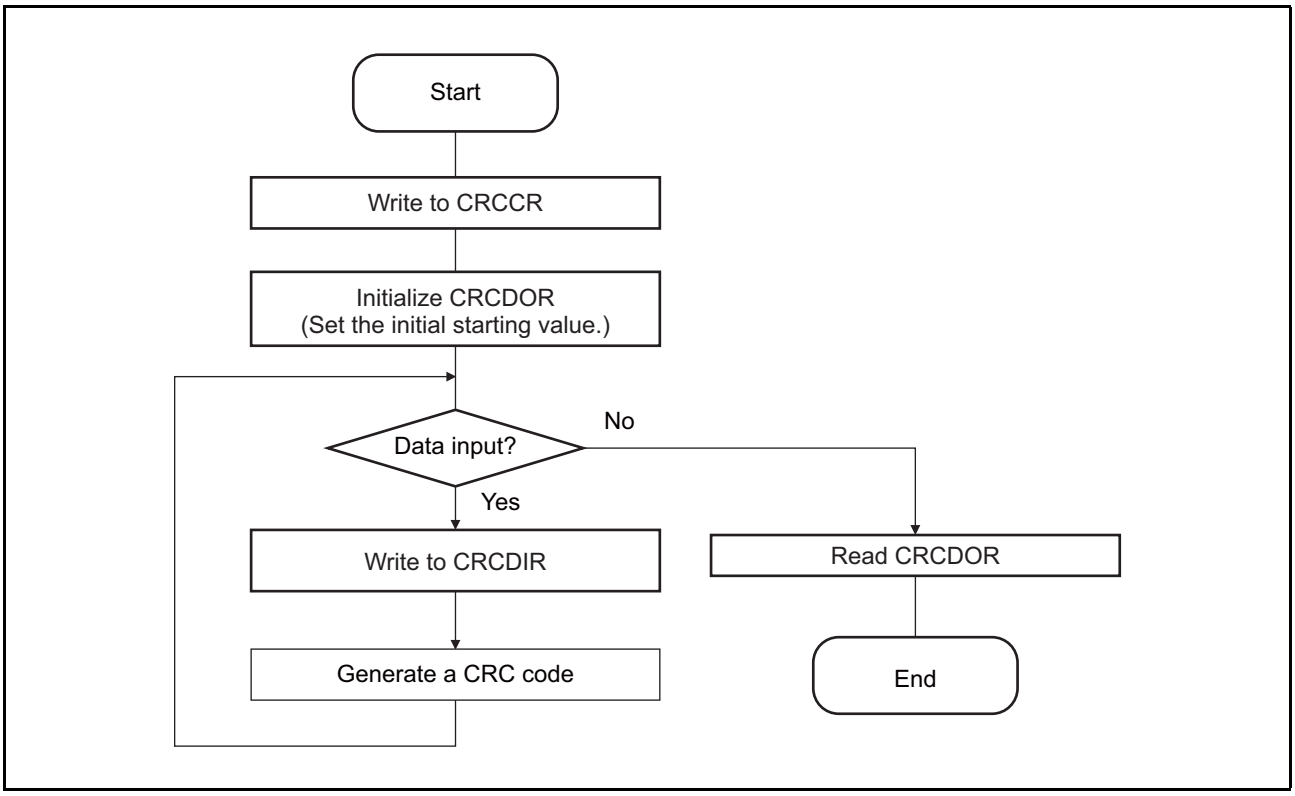


Figure 28.2 Use of the CRC Operation Unit

Note: If the CRC generation polynomial expression is changed by changing the value of CRCCR.DCRA0POL[1:0], the CRC data output register (CRCDOR) must be re-initialized (by setting the initial starting value).

28.3.1 Initializing the CRC Data Output Register (CRCDOR)

Before the first data is written to the CRC data input register (CRCDIR), the CRC data output register (CRCDOR) must be initialized by setting the initial starting value. Table 28.2 lists the initial starting value for each CRC generation polynomial expression.

Table 28.2 also shows the value obtained from the CRCDOR register after the initial starting value is set, as well as the EXOR value used for EXOR operation during a read. For details on the value obtained from the CRCDOR register, see section 28.2.2, CRC Data Output Register (CRCDOR).

Table 28.2 Initial Starting Value for Each CRC Generation Polynomial Expression

CRC Generation Polynomial Expression	Initial Starting Value	EXOR Value	Value Obtained from CRCDOR after the Initial Starting Value Is Set
32-Ethernet (DCRA0POL[1:0] = 00b)	FFFF FFFFh	FFFF FFFFh	0000 0000h
16-CCITT (DCRA0POL[1:0] = 01b)	0000 FFFFh	0000 0000h	0000 FFFFh
8-SAE J1850 (DCRA0POL[1:0] = 10b)	0000 00FFh	0000 00FFh	0000 0000h
8-0x2F (DCRA0POL[1:0] = 11b)	0000 00FFh	0000 00FFh	0000 0000h

29. Error Control Module (ECM)

This section describes an error control module (ECM).

29.1 Overview

The error control module (ECM) collects error output signals coming from individual peripheral modules. It also generates error interrupts and internal reset signals. Table 29.1 describes the ECM specifications and Figure 29.1 shows the block diagram of ECM.

Table 29.1 Specifications of ECM

Item	Description
Safety processing	ECM can handle the following processing in response to error signal inputs from individual modules. • Error flag set ECM has flags that indicate the state of error occurrence for individual error sources. • ECM maskable interrupt generation Maskable interrupt generation can be controlled (enabled or disabled) for individual error sources. • ECM non-maskable interrupt generation Non-maskable interrupt generation can be controlled (enabled or disabled) for individual error sources. • ECM reset (Internal reset) Internal reset generation can be controlled (enabled or disabled) for individual error sources.
Error status	ECM incorporates error status registers, whose error flag values can be used to check whether the corresponding error sources have been generated.
Debug, self-diagnosis	• Pseudo errors can be generated for debug and self-diagnosis. The operation during injection of pseudo errors is identical to that for the occurrence of real errors. All configurations for the interrupt, or internal reset apply in the same way. In addition, extended pseudo errors can be used for error detection for functional safely.
Delay timer timeout	ECM has a function of outputting an error signal or resetting ECM when the delay timer starts at the same time with generation of an ECM maskable interrupt or an ECM non-maskable interrupt and a delay timer overflow occurs because the delay timer cannot be stopped during the interrupt processing.
Others	ECM has a duplexed structure (master and checker) for redundancy.

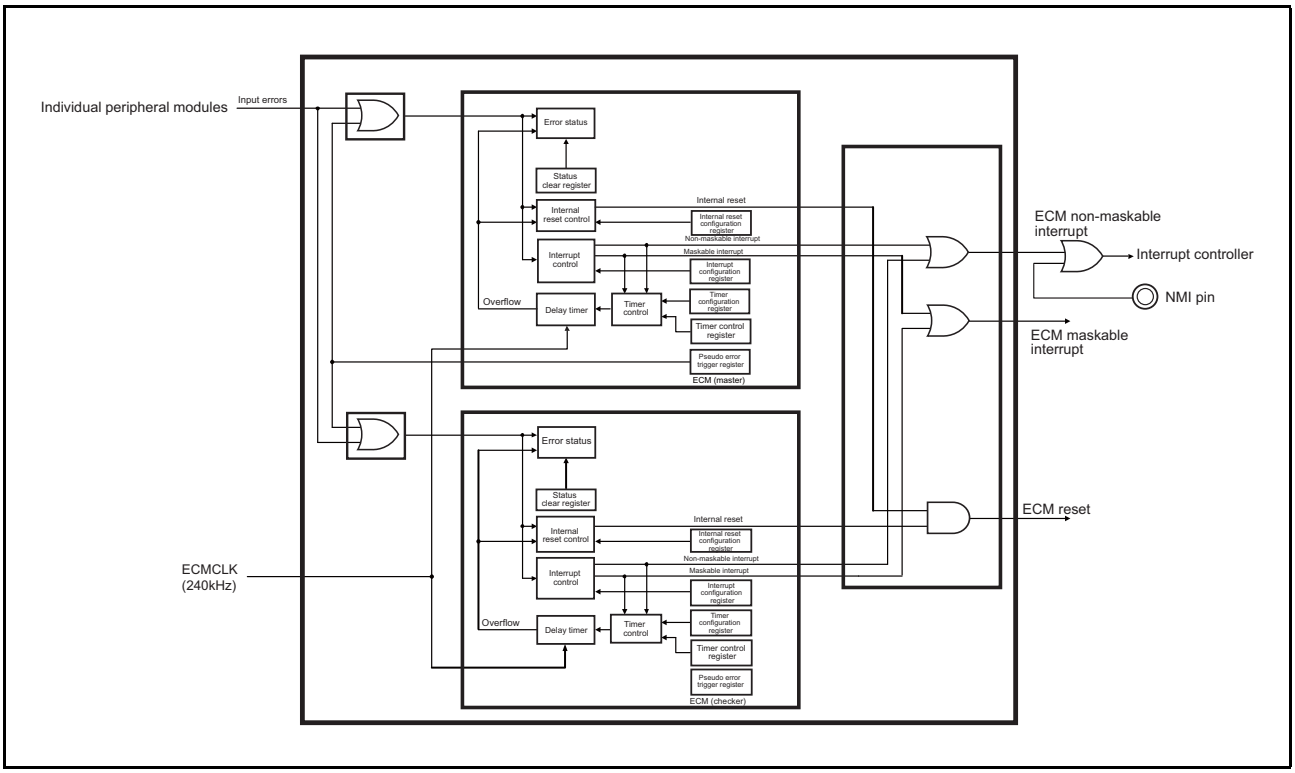


Figure 29.1 Block Diagram of ECM

Table 29.2 shows the error input to ECM.

Table 29.2 ECM Error Input (1 / 2)

Error Source Number	Module	Function
1	WDTA	WDTA underflow/refresh error (for Cortex-R4)
2	—	Reserved
3	IWDTa	IWDTa underflow/refresh error
4	—	Reserved
5	Cortex-R4 cache	1-bit or 2-bit ECC error in the instruction cache (Tag RAM)
6		1-bit or 2-bit ECC error in the instruction cache (Data RAM)
7		1-bit ECC error in the data cache (Tag/Dirty RAM)
8		2-bit ECC error in the data cache (Tag/Dirty RAM)
9		1-bit ECC error in the data cache (Data RAM)
10		2-bit ECC error in the data cache (Data RAM)
11	Cortex-R4 RAM	1-bit ECC error in the ATCM
12		2-bit ECC error in the ATCM
13		1-bit ECC error in the BTCM
14		2-bit ECC error in the BTCM
15	Extended internal SRAM	1-bit ECC error in the IRAM or DRAM
16		2-bit ECC error in the IRAM or DRAM
17	—	Reserved
18		Reserved
19		Reserved
20	Clock monitor circuit (CLMA)	Main clock oscillation stop detection
21		CLMA0 oscillation stop detection (PLL0)
22		CLMA1 oscillation stop detection (PLL1)
23		CLMA2 oscillation stop detection (LOCO)
24	12-bit A/D converter (S12ADCa)	Unit 0 overwrite interrupt
25		Unit 1 overwrite interrupt
26	—	Reserved
27		Reserved
28		Reserved
29		Reserved
30		Reserved
31		Reserved
32	Data operation circuit (DOC)	DOC operation error
33	Internal bus	Bus error
34	—	Reserved
35	—	Extended pseudo error 35*1
36	—	Extended pseudo error 36*1
37	—	Extended pseudo error 37*1
38	—	Extended pseudo error 38*1
39	—	Extended pseudo error 39*1
40	—	Extended pseudo error 40*1
41	—	Extended pseudo error 41*1
42 to 92	—	Reserved

Table 29.2 ECM Error Input (2 / 2)

Error Source Number	Module	Function
93	Error control module (ECM)	Compare error
94		Delay timer overflow error
95		Reserved
96		Reserved

Note 1. A pseudo error can be generated if the corresponding bit in the ECM pseudo error trigger register (ECMPEn) is set by software. For details, see section 29.3.1, Pseudo Error Generation.

29.2 Registers

29.2.1 ECM Master/Checker Error Source Status Register 0 (ECMmESSTR0 (m = M or C))

The ECMmESSTR0 (m = M or C) register is a flag register that indicates whether individual error sources occurred. This register is cleared with the corresponding bits in the ECM error source status clear trigger register 0 (ECMESSTC0) set, or with the RES# pin reset. This register is not cleared by occurrence of other reset sources.

Address(es): ECMMESSTR0: A007 D008h
ECMCESSTR0: A007 D048h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ECMm SSE031	—	—	—	—	—	—	ECMm SSE024	ECMm SSE023	ECMm SSE022	ECMm SSE021	ECMm SSE020	ECMm SSE019	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ECMm SSE015	ECMm SSE014	ECMm SSE013	ECMm SSE012	ECMm SSE011	ECMm SSE010	ECMm SSE009	ECMm SSE008	ECMm SSE007	ECMm SSE006	ECMm SSE005	ECMm SSE004	—	ECMm SSE002	—	ECMm SSE000
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMmSSE000	Error Source Status 1	Indicates occurrence of a WDTA underflow/refresh error (for Cortex-R4) (error source 1). 0: Error not occurred 1: Error occurred	R
b1	—	Reserved	This bit is read as 0.	R
b2	ECMmSSE002	Error Source Status 3	Indicates occurrence of an IWDTa underflow/refresh error (error source 3). 0: Error not occurred 1: Error occurred	R
b3	—	Reserved	This bit is read as 0.	R
b4	ECMmSSE004	Error Source Status 5	Indicates occurrence of a 1-bit ECC error (with correction) or a 2-bit ECC error (without correction) in the instruction cache (Tag RAM) (error source 5). 0: Error not occurred 1: Error occurred	R
b5	ECMmSSE005	Error Source Status 6	Indicates occurrence of a 1-bit ECC error (with correction) or a 2-bit ECC error (without correction) in the instruction cache (Data RAM) (error source 6). 0: Error not occurred 1: Error occurred	R
b6	ECMmSSE006	Error Source Status 7	Indicates occurrence of a 1-bit ECC error in the data cache (Tag/Dirty RAM) (error source 7). 0: Error not occurred 1: Error occurred	R
b7	ECMmSSE007	Error Source Status 8	Indicates occurrence of a 2-bit ECC error in the data cache (Tag/Dirty RAM) (error source 8). 0: Error not occurred 1: Error occurred	R
b8	ECMmSSE008	Error Source Status 9	Indicates occurrence of a 1-bit ECC error in the data cache (Data RAM) (error source 9) 0: Error not occurred 1: Error occurred	R
b9	ECMmSSE009	Error Source Status 10	Indicates occurrence of a 2-bit ECC error in the data cache (Data RAM) (error source 10). 0: Error not occurred 1: Error occurred	R

Bit	Symbol	Bit Name	Description	R/W
b10	ECMmSSE010	Error Source Status 11	Indicates occurrence of a 1-bit ECC error in the ATCM (error source 11). 0: Error not occurred 1: Error occurred	R
b11	ECMmSSE011	Error Source Status 12	Indicates occurrence of a 2-bit ECC error in the ATCM (error source 12). 0: Error not occurred 1: Error occurred	R
b12	ECMmSSE012	Error Source Status 13	Indicates occurrence of a 1-bit ECC error in the BTCM (error source 13). 0: Error not occurred 1: Error occurred	R
b13	ECMmSSE013	Error Source Status 14	Indicates occurrence of a 2-bit ECC error in the BTCM (error source 14). 0: Error not occurred 1: Error occurred	R
b14	ECMmSSE014	Error Source Status 15	Indicates occurrence of a 1-bit ECC error in the IRAM or DRAM (error source 15). 0: Error not occurred 1: Error occurred	R
b15	ECMmSSE015	Error Source Status 16	Indicates occurrence of a 2-bit ECC error in the IRAM or DRAM (error source 16). 0: Error not occurred 1: Error occurred	R
b18 to b16	—	Reserved	These bits are read as 0.	R
b19	ECMmSSE019	Error Source Status 20	Indicates occurrence of an error of main clock oscillation stop detection (error source 20). 0: Error not occurred 1: Error occurred	R
b20	ECMmSSE020	Error Source Status 21	Indicates occurrence of an error of CLMA0 oscillation stop detection (PLL0) (error source 21). 0: Error not occurred 1: Error occurred	R
b21	ECMmSSE021	Error Source Status 22	Indicates occurrence of an error of CLMA1 oscillation stop detection (PLL1) (error source 22). 0: Error not occurred 1: Error occurred	R
b22	ECMmSSE022	Error Source Status 23	Indicates occurrence of an error of CLMA2 oscillation stop detection (LOCO) (error source 23). 0: Error not occurred 1: Error occurred	R
b23	ECMmSSE023	Error Source Status 24	Indicates occurrence of a 12-bit A/D converter unit 0 overwrite interrupt (error source 24). 0: Error not occurred 1: Error occurred	R
b24	ECMmSSE024	Error Source Status 25	Indicates occurrence of a 12-bit A/D converter unit 1 overwrite interrupt (error source 25). 0: Error not occurred 1: Error occurred	R
b30 to b25	—	Reserved	These bits are read as 0.	R
b31	ECMmSSE031	Error Source Status 32	Indicates occurrence of a DOC operation error (error source 32). 0: Error not occurred 1: Error occurred	R

29.2.2 ECM Master/Checker Error Source Status Register 1 (ECMmESSTR1 (m = M or C))

The ECMmESSTR1 (m = M or C) register is a flag register that indicates whether individual error sources occurred. This register is cleared with the corresponding bits in the ECM error source status clear trigger register 1 (ECMESSTC1) set, or with the RES# pin reset. This register is not cleared by occurrence of other reset sources.

Address(es): ECMMESSTR1: A007 D00Ch
ECMCESSTR1: A007 D04Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	ECMm SSE108	ECMm SSE107	ECMm SSE106	ECMm SSE105	ECMm SSE104	ECMm SSE103	ECMm SSE102	—	ECMm SSE100
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMmSSE100	Error Source Status 33	Indicates occurrence of a bus error (error source 33). 0: Error not occurred 1: Error occurred	R
b1	—	Reserved	This bit is read as 0.	R
b2	ECMmSSE102	Error Source Status 35	Indicates occurrence of extended pseudo error 35 (error source 35)*1. 0: Error not occurred 1: Error occurred	R
b3	ECMmSSE103	Error Source Status 36	Indicates occurrence of extended pseudo error 36 (error source 36)*1. 0: Error not occurred 1: Error occurred	R
b4	ECMmSSE104	Error Source Status 37	Indicates occurrence of extended pseudo error 37 (error source 37)*1. 0: Error not occurred 1: Error occurred	R
b5	ECMmSSE105	Error Source Status 38	Indicates occurrence of extended pseudo error 38 (error source 38)*1. 0: Error not occurred 1: Error occurred	R
b6	ECMmSSE106	Error Source Status 39	Indicates occurrence of extended pseudo error 39 (error source 39)*1. 0: Error not occurred 1: Error occurred	R
b7	ECMmSSE107	Error Source Status 40	Indicates occurrence of extended pseudo error 40 (error source 40)*1. 0: Error not occurred 1: Error occurred	R
b8	ECMmSSE108	Error Source Status 41	Indicates occurrence of extended pseudo error 41 (error source 41)*1. 0: Error not occurred 1: Error occurred	R
b31 to b9	—	Reserved	These bits are read as 0.	R

Note 1. An error occurs when the ECMPE1.ECMPE102 to ECMPE1.ECMPE108 bit is set by software.

29.2.3 ECM Master/Checker Error Source Status Register 2 (ECMmESSTR2 (m = M or C))

The ECMmESSTR2 (m = M or C) register is a flag register that indicates whether individual error sources occurred. Bits 29 and 28 of this register are cleared with the corresponding bits in the ECM error source status clear trigger register 2 (ECMESSTC2) set, or with the RES# pin reset. This register is not cleared by occurrence of other reset sources.

Address(es): ECMMESSTR2: A007 D010h
ECMCESSTR2: A007 D050h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	ECMm SSE229	ECMm SSE228	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b27 to b0	—	Reserved	These bits are read as 0.	R
b28	ECMmSSE228	Error Source Status 93	Indicates occurrence of an ECM compare error (error source 93). 0: ECM compare error not occurred 1: ECM compare error occurred	R
b29	ECMmSSE229	Error Source Status 94	Indicates occurrence of a delay timer overflow (error source 94). 0: Delay timer overflow not occurred 1: Delay timer overflow occurred	R
b31, b30	—	Reserved	These bits are read as 0.	R

29.2.4 ECM Master/Checker Protection Command Register (ECMmPCMD0 (m = M or C))

The ECMmPCMD0 (m = M or C) register controls writing to the protected registers. For details, see section 29.3.2, Writing to Protected Registers.

Address(es): ECMMPCMD0: A007 D014h
ECMCPCMD0: A007 D054h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	Undefined															

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	ECMm0REG[7:0]							
Value after reset:	Undefined															

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	ECMm0REG7 to ECMm0REG0	Specific Instruction Sequence Write	Writes the specific instruction sequence to enable writing to the ECMm register (m = M or C).	W
b31 to b8	—	Reserved	The write value should be 0.	W

29.2.5 ECM Maskable Interrupt Configuration Register 0 (ECMMICFG0)

The ECMMICFG0 register controls generation of ECM maskable interrupts due to occurrence of individual error sources.

This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.

Writing to this register is protected by the specific instruction sequence. For details, see section 29.3.2, Writing to Protected Registers.

Address(es): A007 D084h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ECMMIE031	—	—	—	—	—	—	ECMMIE024	ECMMIE023	ECMMIE022	ECMMIE021	ECMMIE020	ECMMIE019	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ECMMIE015	ECMMIE014	ECMMIE013	ECMMIE012	ECMMIE011	ECMMIE010	ECMMIE009	ECMMIE008	ECMMIE007	ECMMIE006	ECMMIE005	ECMMIE004	—	ECMMIE002	—	ECMMIE000
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMMIE000	ECM Maskable Interrupt Generation Control 1	Enables or disables a maskable interrupt due to occurrence of a WDT overflow/refresh error (for Cortex-R4) (error source 1). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	ECMMIE002	ECM Maskable Interrupt Generation Control 3	Enables or disables a maskable interrupt due to occurrence of an IWDTa overflow/refresh error (error source 3). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b4	ECMMIE004	ECM Maskable Interrupt Generation Control 5	Enables or disables a maskable interrupt due to occurrence of a 1-bit ECC error (with correction) or a 2-bit ECC error (without correction) in the instruction cache (Tag RAM) (error source 5). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b5	ECMMIE005	ECM Maskable Interrupt Generation Control 6	Enables or disables a maskable interrupt due to occurrence of a 1-bit ECC error (with correction) or a 2-bit ECC error (without correction) in the instruction cache (Data RAM) (error source 6). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b6	ECMMIE006	ECM Maskable Interrupt Generation Control 7	Enables or disables a maskable interrupt due to occurrence of a 1-bit ECC error in the data cache (Tag/Dirty RAM) (error source 7). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b7	ECMMIE007	ECM Maskable Interrupt Generation Control 8	Enables or disables a maskable interrupt due to occurrence of a 2-bit ECC error in the data cache (Tag/Dirty RAM) (error source 8). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W

Bit	Symbol	Bit Name	Description	R/W
b8	ECMMIE008	ECM Maskable Interrupt Generation Control 9	Enables or disables a maskable interrupt due to occurrence of a 1-bit ECC error in the data cache (Data RAM) (error source 9). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b9	ECMMIE009	ECM Maskable Interrupt Generation Control 10	Enables or disables a maskable interrupt due to occurrence of a 2-bit ECC error in the data cache (Data RAM) (error source 10). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b10	ECMMIE010	ECM Maskable Interrupt Generation Control 11	Enables or disables a maskable interrupt due to occurrence of a 1-bit ECC error in the ATCM (error source 11). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b11	ECMMIE011	ECM Maskable Interrupt Generation Control 12	Enables or disables a maskable interrupt due to occurrence of a 2-bit ECC error in the ATCM (error source 12). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b12	ECMMIE012	ECM Maskable Interrupt Generation Control 13	Enables or disables a maskable interrupt due to occurrence of a 1-bit ECC error in the BTCM (error source 13). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b13	ECMMIE013	ECM Maskable Interrupt Generation Control 14	Enables or disables a maskable interrupt due to occurrence of a 2-bit ECC error in the BTCM (error source 14). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b14	ECMMIE014	ECM Maskable Interrupt Generation Control 15	Enables or disables a maskable interrupt due to occurrence of a 1-bit ECC error in the IRAM or DRAM (error source 15). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b15	ECMMIE015	ECM Maskable Interrupt Generation Control 16	Enables or disables a maskable interrupt due to occurrence of a 2-bit ECC error in the IRAM or DRAM (error source 16). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b18 to b16	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b19	ECMMIE019	ECM Maskable Interrupt Generation Control 20	Enables or disables a maskable interrupt due to occurrence of an error of main clock oscillation stop detection (error source 20). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b20	ECMMIE020	ECM Maskable Interrupt Generation Control 21	Enables or disables a maskable interrupt due to occurrence of an error of CLMA0 oscillation stop detection (PLL0) (error source 21). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b21	ECMMIE021	ECM Maskable Interrupt Generation Control 22	Enables or disables a maskable interrupt due to occurrence of an error of CLMA1 oscillation stop detection (PLL1) (error source 22). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b22	ECMMIE022	ECM Maskable Interrupt Generation Control 23	Enables or disables a maskable interrupt due to occurrence of an error of CLMA2 oscillation stop detection (LOCO) (error source 23). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W

Bit	Symbol	Bit Name	Description	R/W
b23	ECMMIE023	ECM Maskable Interrupt Generation Control 24	Enables or disables a maskable interrupt due to occurrence of a 12-bit A/D converter unit 0 overwrite interrupt (error source 24). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b24	ECMMIE024	ECM Maskable Interrupt Generation Control 25	Enables or disables a maskable interrupt due to occurrence of a 12-bit A/D converter unit 1 overwrite interrupt (error source 25). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b30 to b25	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b31	ECMMIE031	ECM Maskable Interrupt Generation Control 32	Enables or disables a maskable interrupt due to occurrence of a DOC operation error (error source 32). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W

29.2.6 ECM Maskable Interrupt Configuration Register 1 (ECMMICFG1)

The ECMMICFG1 register controls generation of ECM maskable interrupts due to occurrence of individual error sources.

This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.

Writing to this register is protected by the specific instruction sequence. For details, see section 29.3.2, Writing to Protected Registers.

Address(es): A007 D088h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	ECMMIE108	ECMMIE107	ECMMIE106	ECMMIE105	ECMMIE104	ECMMIE103	ECMMIE102	—	ECMMIE100
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMMIE100	ECM Maskable Interrupt Generation Control 33	Enables or disables a maskable interrupt due to occurrence of a bus error (error source 33). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	ECMMIE102	ECM Maskable Interrupt Generation Control 35	Enables or disables a maskable interrupt due to occurrence of extended pseudo error 35 (error source 35). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b3	ECMMIE103	ECM Maskable Interrupt Generation Control 36	Enables or disables a maskable interrupt due to occurrence of extended pseudo error 36 (error source 36). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b4	ECMMIE104	ECM Maskable Interrupt Generation Control 37	Enables or disables a maskable interrupt due to occurrence of extended pseudo error 37 (error source 37). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b5	ECMMIE105	ECM Maskable Interrupt Generation Control 38	Enables or disables a maskable interrupt due to occurrence of extended pseudo error 38 (error source 38). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b6	ECMMIE106	ECM Maskable Interrupt Generation Control 39	Enables or disables a maskable interrupt due to occurrence of extended pseudo error 39 (error source 39). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b7	ECMMIE107	ECM Maskable Interrupt Generation Control 40	Enables or disables a maskable interrupt due to occurrence of extended pseudo error 40 (error source 40). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b8	ECMMIE108	ECM Maskable Interrupt Generation Control 41	Enables or disables a maskable interrupt due to occurrence of extended pseudo error 41 (error source 41). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b31 to b9	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

29.2.7 ECM Maskable Interrupt Configuration Register 2 (ECMMICFG2)

The ECMMICFG2 register controls generation of ECM maskable interrupts due to occurrence of individual error sources.

This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.

Writing to this register is protected by the specific instruction sequence. For details, see section 29.3.2, Writing to Protected Registers.

Address(es): A007 D08Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	ECMMIE228	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b27 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b28	ECMMIE228	ECM Maskable Interrupt Generation Control 93	Enables or disables a maskable interrupt due to occurrence of an ECM compare error (error source 93). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b31 to b29	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

29.2.8 ECM Non-maskable Interrupt Configuration Register 0 (ECMNMICFG0)

The ECMNMICFG0 register controls generation of ECM non-maskable interrupts due to occurrence of individual error sources.

This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.

Writing to this register is protected by the specific instruction sequence. For details, see section 29.3.2, Writing to Protected Registers.

Address(es): A007 D090h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ECMNMIE031	—	—	—	—	—	—	ECMNMIE024	ECMNMIE023	ECMNMIE022	ECMNMIE021	ECMNMIE020	ECMNMIE019	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ECMNMIE015	ECMNMIE014	ECMNMIE013	ECMNMIE012	ECMNMIE011	ECMNMIE010	ECMNMIE009	ECMNMIE008	ECMNMIE007	ECMNMIE006	ECMNMIE005	ECMNMIE004	—	ECMNMIE002	—	ECMNMIE000
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMNMIE000	ECM Non-maskable Interrupt Generation Control 1	Enables or disables a non-maskable interrupt due to occurrence of a WDT overflow/refresh error (for Cortex-R4) (error source 1). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	ECMNMIE002	ECM Non-maskable Interrupt Generation Control 3	Enables or disables a non-maskable interrupt due to occurrence of an IWDTa overflow/refresh error (error source 3). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b4	ECMNMIE004	ECM Non-maskable Interrupt Generation Control 5	Enables or disables a non-maskable interrupt due to occurrence of a 1-bit ECC error (with correction) and 2-bit ECC error (without correction) in the instruction cache (Tag RAM) (error source 5). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b5	ECMNMIE005	ECM Non-maskable Interrupt Generation Control 6	Enables or disables a non-maskable interrupt due to occurrence of a 1-bit ECC error (with correction) and 2-bit ECC error (without correction) in the instruction cache (Data RAM) (error source 6). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b6	ECMNMIE006	ECM Non-maskable Interrupt Generation Control 7	Enables or disables a non-maskable interrupt due to occurrence of a 1-bit ECC error in the data cache (Tag/Dirty RAM) (error source 7). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b7	ECMNMIE007	ECM Non-maskable Interrupt Generation Control 8	Enables or disables a non-maskable interrupt due to occurrence of a 2-bit ECC error in the data cache (Tag/Dirty RAM) (error source 8). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W

Bit	Symbol	Bit Name	Description	R/W
b8	ECMNMIE008	ECM Non-maskable Interrupt Generation Control 9	Enables or disables a non-maskable interrupt due to occurrence of a 1-bit ECC error in the data cache (Data RAM) (error source 9). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b9	ECMNMIE009	ECM Non-maskable Interrupt Generation Control 10	Enables or disables a non-maskable interrupt due to occurrence of a 2-bit ECC error in the data cache (Data RAM) (error source 10). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b10	ECMNMIE010	ECM Non-maskable Interrupt Generation Control 11	Enables or disables a non-maskable interrupt due to occurrence of a 1-bit ECC error in the ATCM (error source 11). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b11	ECMNMIE011	ECM Non-maskable Interrupt Generation Control 12	Enables or disables a non-maskable interrupt due to occurrence of a 2-bit ECC error in the ATCM (error source 12). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b12	ECMNMIE012	ECM Non-maskable Interrupt Generation Control 13	Enables or disables a non-maskable interrupt due to occurrence of a 1-bit ECC error in the BTCM (error source 13). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b13	ECMNMIE013	ECM Non-maskable Interrupt Generation Control 14	Enables or disables a non-maskable interrupt due to occurrence of a 2-bit ECC error in the BTCM (error source 14). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b14	ECMNMIE014	ECM Non-maskable Interrupt Generation Control 15	Enables or disables a non-maskable interrupt due to occurrence of a 1-bit ECC error in the IRAM or DRAM (error source 15). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b15	ECMNMIE015	ECM Non-maskable Interrupt Generation Control 16	Enables or disables a non-maskable interrupt due to occurrence of a 2-bit ECC error in the IRAM or DRAM (error source 16). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b18 to b16	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b19	ECMNMIE019	ECM Non-maskable Interrupt Generation Control 20	Enables or disables a non-maskable interrupt due to occurrence of an error of main clock oscillation stop detection (error source 20). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b20	ECMNMIE020	ECM Non-maskable Interrupt Generation Control 21	Enables or disables a non-maskable interrupt due to occurrence of an error of CLMA0 oscillation stop detection (PLL0) (error source 21). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b21	ECMNMIE021	ECM Non-maskable Interrupt Generation Control 22	Enables or disables a non-maskable interrupt due to occurrence of an error of CLMA1 oscillation stop detection (PLL1) (error source 22). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b22	ECMNMIE022	ECM Non-maskable Interrupt Generation Control 23	Enables or disables a non-maskable interrupt due to occurrence of an error of CLMA2 oscillation stop detection (LOCO) (error source 23). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W

Bit	Symbol	Bit Name	Description	R/W
b23	ECMNMIE023	ECM Non-maskable Interrupt Generation Control 24	Enables or disables a non-maskable interrupt due to occurrence of a 12-bit A/D converter unit 0 overwrite interrupt (error source 24). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b24	ECMNMIE024	ECM Non-maskable Interrupt Generation Control 25	Enables or disables a non-maskable interrupt due to occurrence of a 12-bit A/D converter unit 1 overwrite interrupt (error source 25). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b30 to b25	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b31	ECMNMIE031	ECM Non-maskable Interrupt Generation Control 32	Enables or disables a non-maskable interrupt due to occurrence of a DOC operation error (error source 32). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W

29.2.9 ECM Non-maskable Interrupt Configuration Register 1 (ECMNMICFG1)

The ECMNMICFG1 register controls generation of ECM non-maskable interrupts due to occurrence of individual error sources.

This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.

Writing to this register is protected by the specific instruction sequence. For details, see section 29.3.2, Writing to Protected Registers.

Address(es): A007 D094h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	ECMNMIE108	ECMNMIE107	ECMNMIE106	ECMNMIE105	ECMNMIE104	ECMNMIE103	ECMNMIE102	—	ECMNMIE100
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMNMIE100	ECM Non-maskable Interrupt Generation Control 33	Enables or disables a non-maskable interrupt due to occurrence of a bus error (error source 33). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	ECMNMIE102	ECM Non-maskable Interrupt Generation Control 35	Enables or disables a non-maskable interrupt due to occurrence of extended pseudo error 35 (error source 35). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b3	ECMNMIE103	ECM Non-maskable Interrupt Generation Control 36	Enables or disables a non-maskable interrupt due to occurrence of extended pseudo error 36 (error source 36). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b4	ECMNMIE104	ECM Non-maskable Interrupt Generation Control 37	Enables or disables a non-maskable interrupt due to occurrence of extended pseudo error 37 (error source 37). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b5	ECMNMIE105	ECM Non-maskable Interrupt Generation Control 38	Enables or disables a non-maskable interrupt due to occurrence of extended pseudo error 38 (error source 38). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b6	ECMNMIE106	ECM Non-maskable Interrupt Generation Control 39	Enables or disables a non-maskable interrupt due to occurrence of extended pseudo error 39 (error source 39). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b7	ECMNMIE107	ECM Non-maskable Interrupt Generation Control 40	Enables or disables a non-maskable interrupt due to occurrence of extended pseudo error 40 (error source 40). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b8	ECMNMIE108	ECM Non-maskable Interrupt Generation Control 41	Enables or disables a non-maskable interrupt due to occurrence of extended pseudo error 41 (error source 41). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b31 to b9	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

29.2.10 ECM Non-maskable Interrupt Configuration Register 2 (ECMNMICFG2)

The ECMNMICFG2 register controls generation of ECM non-maskable interrupts due to occurrence of individual error sources.

This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.

Writing to this register is protected by the specific instruction sequence. For details, see section 29.3.2, Writing to Protected Registers.

Address(es): A007 D098h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	ECMNMIE228	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b27 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b28	ECMNMIE228	ECM Non-maskable Interrupt Generation Control 93	Enables or disables a non-maskable interrupt due to occurrence of an ECM compare error (error source 93). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b31 to b29	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

29.2.11 ECM Internal Reset Configuration Register 0 (ECMIRCFG0)

The ECMIRCFG0 register controls generation of internal resets (ECM resets) due to occurrence of individual error sources.

This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.

Writing to this register is protected by the specific instruction sequence. For details, see section 29.3.2, Writing to Protected Registers.

Address(es): A007 D09Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ECMIRE031	—	—	—	—	—	—	ECMIRE024	ECMIRE023	ECMIRE022	ECMIRE021	ECMIRE020	ECMIRE019	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ECMIRE015	ECMIRE014	ECMIRE013	ECMIRE012	ECMIRE011	ECMIRE010	ECMIRE009	ECMIRE008	ECMIRE007	ECMIRE006	ECMIRE005	ECMIRE004	—	ECMIRE002	—	ECMIRE000
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b0	ECMIRE000	ECM Internal Reset Generation Control 1	Enables or disables generation of an ECM reset due to occurrence of a WDT overflow/refresh error (for Cortex-R4) (error source 1). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	ECMIRE002	ECM Internal Reset Generation Control 3	Enables or disables generation of an ECM reset due to occurrence of an IWDTa overflow/refresh error (error source 3). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b4	ECMIRE004	ECM Internal Reset Generation Control 5	Enables or disables generation of an ECM reset due to occurrence of a 1-bit ECC error (with correction) or a 2-bit ECC error (without correction) in the instruction cache (Tag RAM) (error source 5). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b5	ECMIRE005	ECM Internal Reset Generation Control 6	Enables or disables generation of an ECM reset due to occurrence of a 1-bit ECC error (with correction) or a 2-bit ECC error (without correction) in the instruction cache (Data RAM) (error source 6). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b6	ECMIRE006	ECM Internal Reset Generation Control 7	Enables or disables generation of an ECM reset due to occurrence of a 1-bit ECC error in the data cache (Tag/Dirty RAM) (error source 7). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b7	ECMIRE007	ECM Internal Reset Generation Control 8	Enables or disables generation of an ECM reset due to occurrence of a 2-bit ECC error in the data cache (Tag/Dirty RAM) (error source 8). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W

Bit	Symbol	Bit Name	Description	R/W
b8	ECMIRE008	ECM Internal Reset Generation Control 9	Enables or disables generation of an ECM reset due to occurrence of a 1-bit ECC error in the data cache (Data RAM) (error source 9). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b9	ECMIRE009	ECM Internal Reset Generation Control 10	Enables or disables generation of an ECM reset due to occurrence of a 2-bit ECC error in the data cache (Data RAM) (error source 10). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b10	ECMIRE010	ECM Internal Reset Generation Control 11	Enables or disables generation of an ECM reset due to occurrence of a 1-bit ECC error in the ATCM (error source 11). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b11	ECMIRE011	ECM Internal Reset Generation Control 12	Enables or disables generation of an ECM reset due to occurrence of a 2-bit ECC error in the ATCM (error source 12). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b12	ECMIRE012	ECM Internal Reset Generation Control 13	Enables or disables generation of an ECM reset due to occurrence of a 1-bit ECC error in the BTCM (error source 13). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b13	ECMIRE013	ECM Internal Reset Generation Control 14	Enables or disables generation of an ECM reset due to occurrence of a 2-bit ECC error in the BTCM (error source 14). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b14	ECMIRE014	ECM Internal Reset Generation Control 15	Enables or disables generation of an ECM reset due to occurrence of a 1-bit ECC error in the IRAM or DRAM (error source 15). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b15	ECMIRE015	ECM Internal Reset Generation Control 16	Enables or disables generation of an ECM reset due to occurrence of a 2-bit ECC error in the IRAM or DRAM (error source 16). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b18 to b16	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b19	ECMIRE019	ECM Internal Reset Generation Control 20	Enables or disables generation of an ECM reset due to occurrence of an error of main clock oscillation stop detection (error source 20). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b20	ECMIRE020	ECM Internal Reset Generation Control 21	Enables or disables generation of an ECM reset due to occurrence of an error of CLMA0 oscillation stop detection (PLL0) (error source 21). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b21	ECMIRE021	ECM Internal Reset Generation Control 22	Enables or disables generation of an ECM reset due to occurrence of an error of CLMA1 oscillation stop detection (PLL1) (error source 22). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b22	ECMIRE022	ECM Internal Reset Generation Control 23	Enables or disables generation of an ECM reset due to occurrence of an error of CLMA2 oscillation stop detection (LOCO) (error source 23). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W

Bit	Symbol	Bit Name	Description	R/W
b23	ECMIRE023	ECM Internal Reset Generation Control 24	Enables or disables generation of an ECM reset due to occurrence of a 12-bit A/D converter unit 0 overwrite interrupt (error source 24). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b24	ECMIRE024	ECM Internal Reset Generation Control 25	Enables or disables generation of an ECM reset due to occurrence of a 12-bit A/D converter unit 1 overwrite interrupt (error source 25). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b30 to b25	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b31	ECMIRE031	ECM Internal Reset Generation Control 32	Enables or disables generation of an ECM reset due to occurrence of a DOC operation error (error source 32). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W

29.2.12 ECM Internal Reset Configuration Register 1 (ECMIRCFG1)

The ECMIRCFG1 register controls generation of internal resets (ECM resets) due to occurrence of individual error sources.

This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.

Writing to this register is protected by the specific instruction sequence. For details, see section 29.3.2, Writing to Protected Registers.

Address(es): A007 D0A0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	ECMIRE108	ECMIRE107	ECMIRE106	ECMIRE105	ECMIRE104	ECMIRE103	ECMIRE102	—	ECMIRE100
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMIRE100	ECM Internal Reset Generation Control 33	Enables or disables generation of an ECM reset due to occurrence of a bus error (error source 33). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	ECMIRE102	ECM Internal Reset Generation Control 35	Enables or disables generation of an ECM reset due to occurrence of extended pseudo error 35 (error source 35). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b3	ECMIRE103	ECM Internal Reset Generation Control 36	Enables or disables generation of an ECM reset due to occurrence of extended pseudo error 36 (error source 36). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b4	ECMIRE104	ECM Internal Reset Generation Control 37	Enables or disables generation of an ECM reset due to occurrence of extended pseudo error 37 (error source 37). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b5	ECMIRE105	ECM Internal Reset Generation Control 38	Enables or disables generation of an ECM reset due to occurrence of extended pseudo error 38 (error source 38). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b6	ECMIRE106	ECM Internal Reset Generation Control 39	Enables or disables generation of an ECM reset due to occurrence of extended pseudo error 39 (error source 39). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b7	ECMIRE107	ECM Internal Reset Generation Control 40	Enables or disables generation of an ECM reset due to occurrence of extended pseudo error 40 (error source 40). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b8	ECMIRE108	ECM Internal Reset Generation Control 41	Enables or disables generation of an ECM reset due to occurrence of extended pseudo error 41 (error source 41). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b31 to b9	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

29.2.13 ECM Internal Reset Configuration Register 2 (ECMIRCFG2)

The ECMIRCFG2 register controls generation of internal resets (ECM resets) due to occurrence of individual error sources.

This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.

Writing to this register is protected by the specific instruction sequence. For details, see section 29.3.2, Writing to Protected Registers.

Address(es): A007 D0A4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	ECMIRE229	ECMIRE228	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b27 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b28	ECMIRE228	ECM Internal Reset Generation Control 93	Enables or disables generation of an ECM reset due to occurrence of an ECM compare error (error source 93). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b29	ECMIRE229	ECM Internal Reset Generation Control 94	Enables or disables generation of an ECM reset due to occurrence of a delay timer overflow (error source 94). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b31, b30	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

29.2.14 ECM Error Source Status Clear Trigger Register 0 (ECMESSTC0)

The ECMESSTC0 register controls clearing of individual error statuses. Setting individual bits in this register to 1 can clear the corresponding error statuses held in the ECMmESSTR0 (m = M or C) register.

This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.

Writing to this register is protected by the specific instruction sequence. For details, see section 29.3.2, Writing to Protected Registers.

Address(es): A007 D0B4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ECMCLSSE031	—	—	—	—	—	—	ECMCLSSE024	ECMCLSSE023	ECMCLSSE022	ECMCLSSE021	ECMCLSSE020	ECMCLSSE019	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ECMCLSSE015	ECMCLSSE014	ECMCLSSE013	ECMCLSSE012	ECMCLSSE011	ECMCLSSE010	ECMCLSSE009	ECMCLSSE008	ECMCLSSE007	ECMCLSSE006	ECMCLSSE005	ECMCLSSE004	—	ECMCLSSE002	—	ECMCLSSE000
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMCLSSE000	ECM Error Status Clear 1	Clears the error status of WDT overflow/refresh error (for Cortex-R4) (error source 1) and the ECMmESSTR0.ECMmSSE000 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b1	—	Reserved	The write value should be 0.	W
b2	ECMCLSSE002	ECM Error Status Clear 3	Clears the error status of IWDTa overflow/refresh error (error source 3) and the ECMmESSTR0.ECMmSSE002 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b3	—	Reserved	The write value should be 0.	W
b4	ECMCLSSE004	ECM Error Status Clear 5	Clears the error status of a 1-bit ECC error (with correction) or a 2-bit ECC error (without correction) in the instruction cache (Tag RAM) (error source 5) and the ECMmESSTR0.ECMmSSE004 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b5	ECMCLSSE005	ECM Error Status Clear 6	Clears the error status of a 1-bit ECC error (with correction) or a 2-bit ECC error (without correction) in the instruction cache (Data RAM) (error source 6) and the ECMmESSTR0.ECMmSSE005 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b6	ECMCLSSE006	ECM Error Status Clear 7	Clears the error status of 1-bit ECC error in the data cache (Tag/Dirty RAM) (error source 7) and the ECMmESSTR0.ECMmSSE006 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b7	ECMCLSSE007	ECM Error Status Clear 8	Clears the error status of 2-bit ECC error in the data cache (Tag/Dirty RAM) (error source 8) and the ECMmESSTR0.ECMmSSE007 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W

Bit	Symbol	Bit Name	Description	R/W
b8	ECMCLSSE008	ECM Error Status Clear 9	Clears the error status of 1-bit ECC error in the data cache (Data RAM) (error source 9) and the ECMmESSTR0.ECMmSSE008 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b9	ECMCLSSE009	ECM Error Status Clear 10	Clears the error status of 2-bit ECC error in the data cache (Data RAM) (error source 10) and the ECMmESSTR0.ECMmSSE009 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b10	ECMCLSSE010	ECM Error Status Clear 11	Clears the error status of 1-bit ECC error in the ATCM (error source 11) and the ECMmESSTR0.ECMmSSE010 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b11	ECMCLSSE011	ECM Error Status Clear 12	Clears the error status of 2-bit ECC error in the ATCM (error source 12) and the ECMmESSTR0.ECMmSSE011 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b12	ECMCLSSE012	ECM Error Status Clear 13	Clears the error status of 1-bit ECC error in the BTCM (error source 13) and the ECMmESSTR0.ECMmSSE012 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b13	ECMCLSSE013	ECM Error Status Clear 14	Clears the error status of 2-bit ECC error in the BTCM (error source 14) and the ECMmESSTR0.ECMmSSE013 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b14	ECMCLSSE014	ECM Error Status Clear 15	Clears the error status of 1-bit ECC error in the IRAM or DRAM (error source 15) and the ECMmESSTR0.ECMmSSE014 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b15	ECMCLSSE015	ECM Error Status Clear 16	Clears the error status of 2-bit ECC error in the IRAM or DRAM (error source 16) and the ECMmESSTR0.ECMmSSE015 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b18 to b16	—	Reserved	The write value should be 0.	W
b19	ECMCLSSE019	ECM Error Status Clear 20	Clears the error status of main clock oscillation stop detection (error source 20) and the ECMmESSTR0.ECMmSSE019 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b20	ECMCLSSE020	ECM Error Status Clear 21	Clears the error status of CLMA0 oscillation stop detection (PLL0) (error source 21) and the ECMmESSTR0.ECMmSSE020 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b21	ECMCLSSE021	ECM Error Status Clear 22	Clears the error status of CLMA1 oscillation stop detection (PLL1) (error source 22) and the ECMmESSTR0.ECMmSSE021 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b22	ECMCLSSE022	ECM Error Status Clear 23	Clears the error status of CLMA2 oscillation stop detection (LOCO) (error source 23) and the ECMmESSTR0.ECMmSSE022 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W

Bit	Symbol	Bit Name	Description	R/W
b23	ECMCLSSE023	ECM Error Status Clear 24	Clears the error status of a 12-bit A/D converter unit 0 overwrite interrupt (error source 24) and the ECMmESSTR0.ECMmSSE023 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b24	ECMCLSSE024	ECM Error Status Clear 25	Clears the error status of a 12-bit A/D converter unit 1 overwrite interrupt (error source 25) and the ECMmESSTR0.ECMmSSE024 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b30 to b25	—	Reserved	The write value should be 0.	W
b31	ECMCLSSE031	ECM Error Status Clear 32	Clears the error status of DOC operation error (error source 32) and the ECMmESSTR0.ECMmSSE031 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W

29.2.15 ECM Error Source Status Clear Trigger Register 1 (ECMESSTC1)

The ECMESSTC1 register controls clearing of individual error statuses. Setting individual bits in this register to 1 can clear the corresponding error statuses held in the ECMmESSTR1 (m = M or C) register.
This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.
Writing to this register is protected by the specific instruction sequence. For details, see section 29.3.2, Writing to Protected Registers.

Address(es): A007 D0B8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	ECMCLSSE108	ECMCLSSE107	ECMCLSSE106	ECMCLSSE105	ECMCLSSE104	ECMCLSSE103	ECMCLSSE102	—	ECMCLSSE100
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMCLSSE100	ECM Error Status Clear 33	Clears the error status of bus error (error source 33) and the ECMmESSTR1.ECMmSSE100 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b1	—	Reserved	The write value should be 0.	W
b2	ECMCLSSE102	ECM Error Status Clear 35	Clears the error status of extended pseudo error 35 (error source 35) and the ECMmESSTR1.ECMmSSE102 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b3	ECMCLSSE103	ECM Error Status Clear 36	Clears the error status of extended pseudo error 36 (error source 36) and the ECMmESSTR1.ECMmSSE103 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b4	ECMCLSSE104	ECM Error Status Clear 37	Clears the error status of extended pseudo error 37 (error source 37) and the ECMmESSTR1.ECMmSSE104 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b5	ECMCLSSE105	ECM Error Status Clear 38	Clears the error status of extended pseudo error 38 (error source 38) and the ECMmESSTR1.ECMmSSE105 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b6	ECMCLSSE106	ECM Error Status Clear 39	Clears the error status of extended pseudo error 39 (error source 39) and the ECMmESSTR1.ECMmSSE106 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b7	ECMCLSSE107	ECM Error Status Clear 40	Clears the error status of extended pseudo error 40 (error source 40) and the ECMmESSTR1.ECMmSSE107 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b8	ECMCLSSE108	ECM Error Status Clear 41	Clears the error status of extended pseudo error 41 (error source 41) and the ECMmESSTR1.ECMmSSE108 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b31 to b9	—	Reserved	The write value should be 0.	W

29.2.16 ECM Error Source Status Clear Trigger Register 2 (ECMESSTC2)

The ECMESSTC2 register controls clearing of individual error statuses. Setting individual bits in this register to 1 can clear the corresponding error statuses held in the ECMmESSTR2 (m = M or C) register.
This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.
Writing to this register is protected by the specific instruction sequence. For details, see section 29.3.2, Writing to Protected Registers.

Address(es): A007 D0BCh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	ECMCLSSE229	ECMCLSSE228	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b27 to b0	—	Reserved	The write value should be 0.	W
b28	ECMCLSSE228	ECM Error Status Clear 93	Clears the error status of ECM compare error (error source 93) and the ECMmESSTR2.ECMmSSE228 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b29	ECMCLSSE229	ECM Error Status Clear 94	Clears the error status of a delay timer overflow (error source 94) and the ECMmESSTR2.ECMmSSE229 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b31, b30	—	Reserved	The write value should be 0.	W

29.2.17 ECM Protection Command Register (ECMPCMD1)

The ECMPCMD1 register controls writing to protected common registers. For details, see section 29.3.2, Writing to Protected Registers.

Address(es): A007 D0C0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	Undefined															

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	ECM1REG[7:0]							
Value after reset:	Undefined															

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	ECM1REG[7:0]	Specific Instruction Sequence Write	Write the specific instruction sequence to enable writing to common registers.	W
b31 to b8	—	Reserved	The write value should be 0.	W

29.2.18 ECM Protection Status Register (ECMPS)

The ECMPS register indicates the status of whether writing to the protected register has been correctly done. For details, see section 29.3.2, Writing to Protected Registers.

Address(es): A007 D0C4h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	ECMPRERR
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMPRERR	ECM Protection Status	Indicates whether writing to the write protected register has been correctly done. 0: Writing was successful. 1: Writing failed.	R
b7 to b1	—	Reserved	These bits are read as 0.	R

29.2.19 ECM Pseudo Error Trigger Register 0 (ECMPE0)

The ECMPE0 register is a control register that issues a pseudo error for self-diagnosis. When a pseudo error is issued, an interrupt set by ECM or an ECM reset occurs in the same way as when an actual error source occurs.
This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.
Writing to this register is protected by the specific instruction sequence. For details, see section 29.3.2, Writing to Protected Registers.

Address(es): A007 D0C8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ECMPE031	—	—	—	—	—	—	ECMPE024	ECMPE023	ECMPE022	ECMPE021	ECMPE020	ECMPE019	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ECMPE015	ECMPE014	ECMPE013	ECMPE012	ECMPE011	ECMPE010	ECMPE009	ECMPE008	ECMPE007	ECMPE006	ECMPE005	ECMPE004	—	ECMPE002	—	ECMPE000
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMPE000	ECM Pseudo Error Trigger 1	Generates a pseudo WDT overflow/refresh error (for Cortex-R4) (error source 1). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b1	—	Reserved	The write value should be 0.	W
b2	ECMPE002	ECM Pseudo Error Trigger 3	Generates a pseudo IWDTa overflow/refresh error (error source 3). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b3	—	Reserved	The write value should be 0.	W
b4	ECMPE004	ECM Pseudo Error Trigger 5	Generates a pseudo 1-bit ECC error (with correction) or 2-bit ECC error (without correction) in the instruction cache (Tag RAM) (error source 5). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b5	ECMPE005	ECM Pseudo Error Trigger 6	Generates a pseudo 1-bit ECC error (with correction) or 2-bit ECC error (without correction) in the instruction cache (Data RAM) (error source 6). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b6	ECMPE006	ECM Pseudo Error Trigger 7	Generates a pseudo 1-bit ECC error in the data cache (Tag/Dirty RAM) (error source 7). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b7	ECMPE007	ECM Pseudo Error Trigger 8	Generates a pseudo 2-bit ECC error in the data cache (Tag/Dirty RAM) (error source 8). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b8	ECMPE008	ECM Pseudo Error Trigger 9	Generates a pseudo 1-bit ECC error in the data cache (Data RAM) (error source 9). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b9	ECMPE009	ECM Pseudo Error Trigger 10	Generates a pseudo 2-bit ECC error in the data cache (Data RAM) (error source 10). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W

Bit	Symbol	Bit Name	Description	R/W
b10	ECMPE010	ECM Pseudo Error Trigger 11	Generates a pseudo 1-bit ECC error in the ATCM (error source 11). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b11	ECMPE011	ECM Pseudo Error Trigger 12	Generates a pseudo 2-bit ECC error in the ATCM (error source 12). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b12	ECMPE012	ECM Pseudo Error Trigger 13	Generates a pseudo 1-bit ECC error in the BTCM (error source 13). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b13	ECMPE013	ECM Pseudo Error Trigger 14	Generates a pseudo 2-bit ECC error in the BTCM (error source 14). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b14	ECMPE014	ECM Pseudo Error Trigger 15	Generates a pseudo 1-bit ECC error in the IRAM or DRAM (error source 15). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b15	ECMPE015	ECM Pseudo Error Trigger 16	Generates a pseudo 2-bit ECC error in the IRAM or DRAM (error source 16). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b18 to b16	—	Reserved	The write value should be 0.	W
b19	ECMPE019	ECM Pseudo Error Trigger 20	Generates pseudo detection of main clock oscillation stoppage (error source 20). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b20	ECMPE020	ECM Pseudo Error Trigger 21	Generates pseudo detection of CLMA0 oscillation stoppage (PLL0) (error source 21). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b21	ECMPE021	ECM Pseudo Error Trigger 22	Generates pseudo detection of CLMA1 oscillation stoppage (PLL1) (error source 22). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b22	ECMPE022	ECM Pseudo Error Trigger 23	Generates pseudo detection of CLMA2 oscillation stoppage (LOCO) (error source 23). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b23	ECMPE023	ECM Pseudo Error Trigger 24	Generates a pseudo 12-bit A/D converter unit 0 overwrite interrupt (error source 24). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b24	ECMPE024	ECM Pseudo Error Trigger 25	Generates a pseudo 12-bit A/D converter unit 1 overwrite interrupt (error source 25). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b30 to b25	—	Reserved	The write value should be 0.	W
b31	ECMPE031	ECM Pseudo Error Trigger 32	Generates a pseudo DOC operation error (error source 32). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W

29.2.20 ECM Pseudo Error Trigger Register 1 (ECMPE1)

The ECMPE1 register is a control register that issues a pseudo error for self-diagnosis. When a pseudo error is issued, an interrupt set by ECM or an ECM reset occurs in the same way as when an actual error source occurs.
This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.
Writing to this register is protected by the specific instruction sequence. For details, see section 29.3.2, Writing to Protected Registers.
Bits 2 to 8 are trigger bits for extended pseudo errors. For details, see section 29.3.1, Pseudo Error Generation.

Address(es): A007 D0CCh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	ECMPE108	ECMPE107	ECMPE106	ECMPE105	ECMPE104	ECMPE103	ECMPE102	—	ECMPE100
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMPE100	ECM Pseudo Error Trigger 33	Generates a pseudo bus error (error source 33). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b1	—	Reserved	The write value should be 0.	W
b2	ECMPE102	ECM Pseudo Error Trigger 35	Generates a pseudo “extended pseudo error 35” (error source 35). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b3	ECMPE103	ECM Pseudo Error Trigger 36	Generates a pseudo “extended pseudo error 36” (error source 36). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b4	ECMPE104	ECM Pseudo Error Trigger 37	Generates a pseudo “extended pseudo error 37” (error source 37). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b5	ECMPE105	ECM Pseudo Error Trigger 38	Generates a pseudo “extended pseudo error 38” (error source 38). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b6	ECMPE106	ECM Pseudo Error Trigger 39	Generates a pseudo “extended pseudo error 39” (error source 39). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b7	ECMPE107	ECM Pseudo Error Trigger 40	Generates a pseudo “extended pseudo error 40” (error source 40). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b8	ECMPE108	ECM Pseudo Error Trigger 41	Generates a pseudo “extended pseudo error 41” (error source 41). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b31 to b9	—	Reserved	The write value should be 0.	W

29.2.21 ECM Pseudo Error Trigger Register 2 (ECMPE2)

The ECMPE2 register is a control register that issues a pseudo error for self-diagnosis. When a pseudo error is issued, an interrupt set by ECM or an ECM reset occurs in the same way as when an actual error source occurs.
This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.
Writing to this register is protected by the specific instruction sequence. For details, see section 29.3.2, Writing to Protected Registers.

Address(es): A007 D0D0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	ECMPE229	ECMPE228	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b27 to b0	—	Reserved	The write value should be 0.	W
b28	ECMPE228	ECM Pseudo Error Trigger 93	Generates a pseudo EMC error (error source 93). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b29	ECMPE229	ECM Pseudo Error Trigger 94	Generates a pseudo delay timer overflow (error source 94). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b31, b30	—	Reserved	The write value should be 0.	W

29.2.22 ECM Delay Timer Control Register (ECMDTMCTL)

The ECMDTMCTL register controls the delay timer.
This register is a common register. Writing to this register is protected by the specific instruction sequence. For details, see section 29.3.2, Writing to Protected Registers.

Address(es): A007 D0D4h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	DTMSTACNTCLK	—	—	DTMSTP	DTMSTA
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	DTMSTA	Delay Timer Start	Sets the operation of the delay timer. 0: Delay timer operation disabled 1: Delay timer operation enabled	R/W
b1	DTMSTP	Delay Timer Stop	Writing 1 to this bit initializes the delay timer counter, causing the delay timer to stop. Simultaneously, the DTMSTA bit is set to 0.	W
b3, b2	—	Reserved	These bits are read as 0.	R
b4	DTMSTACNTCLK	Delay Timer Status	The value of the DTMSTA bit is applied to the operating status of the delay timer. If the DTMSTA bit is modified once, modifying the DTMSTA bit is disabled until the setting value of the DTMSTACNTCLK bit.	R
b7 to b5	—	Reserved	These bits are read as 0.	R

29.2.23 ECM Delay Timer Register (ECMDTMR)

The ECMDTMR register is a 16-bit counter register for the delay timer. The 16-bit counter counts up using ECMCLK (240 kHz). Changing the DTMSTA bit in the ECM delay timer control register from 1 (delay timer operation enabled) to 0 (delay timer operation disabled) initializes the 16-bit counter. This register can only be read.

Address(es): A007 D0D8h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ECMTDMR[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

29.2.24 ECM Delay Timer Compare Register (ECMDTMCMP)

The ECMDTMCMP register is a compare register used to set the overflow cycle of the delay timer. A delay timer overflow signal is generated to set the ECMmSSE229 bit when the value of this register matches with the value of the delay timer counter. Writing data to this register has to be conducted while the delay timer is stopped.
This register is a common register. Writing to this register is protected by the specific instruction sequence. For details, see section 29.3.2, Writing to Protected Registers.
Write this register after confirming that the read value of the CMPW bit is 0.

Address(es): A007 D0DCh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CMPW
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ECMDTMCMP[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	ECMDTMCMP [15:0]	Delay Timer Compare	Sets the overflow cycle of the delay timer. (overflow cycle) = (N + 1) × (ECMCLK cycle (240 kHz)) N: Set value For details, see section 29.3.3, Timeout Function for Interrupt Processing by Using the Delay Timer.	R/W
b16	CMPW	Compare Write	Indicates whether writing to the delay timer compare bit is enabled. Writing the compare value is possible when this bit is 0. 0: Writing to ECMDTMCMP[15:0] enabled 1: Writing to ECMDTMCMP[15:0] disabled	R
b31 to b17	—	Reserved	These bits are read as 0.	R

29.2.25 ECM Delay Timer Configuration Register 0 (ECMDTMCFG0)

The ECMDTMCFG0 register controls delay timer operation in response to an ECM maskable interrupt. Whether to enable delay timer operation when an ECM maskable interrupt is generated can be set.
This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.
Writing to this register is protected by the specific instruction sequence. For details, see section 29.3.2, Writing to Protected Registers.

Address(es): A007 D0E0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ECMTE031	—	—	—	—	—	—	ECMTE024	ECMTE023	ECMTE022	ECMTE021	ECMTE020	ECMTE019	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ECMTE015	ECMTE014	ECMTE013	ECMTE012	ECMTE011	ECMTE010	ECMTE009	ECMTE008	ECMTE007	ECMTE006	ECMTE005	ECMTE004	—	ECMTE002	—	ECMTE000
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMTE000	ECM Delay Timer Start Control 1	Enables delay timer operation in response to an ECM maskable interrupt caused by a WDT overflow/refresh error (for Cortex-R4) (error source 1). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	ECMTE002	ECM Delay Timer Start Control 3	Enables delay timer operation in response to an ECM maskable interrupt caused by an IWDTa overflow/refresh error (error source 3). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b4	ECMTE004	ECM Delay Timer Start Control 5	Enables delay timer operation in response to an ECM maskable interrupt caused by a 1-bit ECC error (with correction) or 2-bit ECC error (without correction) in the instruction cache (Tag RAM) (error source 5). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b5	ECMTE005	ECM Delay Timer Start Control 6	Enables delay timer operation in response to an ECM maskable interrupt caused by a 1-bit ECC error (with correction) or 2-bit ECC error (without correction) in the instruction cache (Data RAM) (error source 6). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b6	ECMTE006	ECM Delay Timer Start Control 7	Enables delay timer operation in response to an ECM maskable interrupt caused by a 1-bit ECC error in the data cache (Tag/Dirty RAM) (error source 7). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b7	ECMTE007	ECM Delay Timer Start Control 8	Enables delay timer operation in response to an ECM maskable interrupt caused by a 2-bit ECC error in the data cache (Tag/Dirty RAM) (error source 8). 0: Delay timer start disabled 1: Delay timer start enabled	R/W

Bit	Symbol	Bit Name	Description	R/W
b8	ECMTE008	ECM Delay Timer Start Control 9	Enables delay timer operation in response to an ECM maskable interrupt caused by a 1-bit ECC error in the data cache (Data RAM) (error source 9). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b9	ECMTE009	ECM Delay Timer Start Control 10	Enables delay timer operation in response to an ECM maskable interrupt caused by a 2-bit ECC error in the data cache (Data RAM) (error source 10). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b10	ECMTE010	ECM Delay Timer Start Control 11	Enables delay timer operation in response to an ECM maskable interrupt caused by a 1-bit ECC error in the ATCM (error source 11). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b11	ECMTE011	ECM Delay Timer Start Control 12	Enables delay timer operation in response to an ECM maskable interrupt caused by a 2-bit ECC error in the ATCM (error source 12). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b12	ECMTE012	ECM Delay Timer Start Control 13	Enables delay timer operation in response to an ECM maskable interrupt caused by a 1-bit ECC error in the BTCM (error source 13). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b13	ECMTE013	ECM Delay Timer Start Control 14	Enables delay timer operation in response to an ECM maskable interrupt caused by a 2-bit ECC error in the BTCM (error source 14). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b14	ECMTE014	ECM Delay Timer Start Control 15	Enables delay timer operation in response to an ECM maskable interrupt caused by a 1-bit ECC error in the IRAM or DRAM (error source 15). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b15	ECMTE015	ECM Delay Timer Start Control 16	Enables delay timer operation in response to an ECM maskable interrupt caused by a 2-bit ECC error in the IRAM or DRAM (error source 16). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b18 to b16	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b19	ECMTE019	ECM Delay Timer Start Control 20	Enables delay timer operation in response to an ECM maskable interrupt caused by main clock oscillation stop detection (error source 20). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b20	ECMTE020	ECM Delay Timer Start Control 21	Enables delay timer operation in response to an ECM maskable interrupt caused by CLMA0 oscillation stop detection (PLL0) (error source 21). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b21	ECMTE021	ECM Delay Timer Start Control 22	Enables delay timer operation in response to an ECM maskable interrupt caused by CLMA1 oscillation stop detection (PLL1) (error source 22). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b22	ECMTE022	ECM Delay Timer Start Control 23	Enables delay timer operation in response to an ECM maskable interrupt caused by CLMA2 oscillation stop detection (LOCO) (error source 23). 0: Delay timer start disabled 1: Delay timer start enabled	R/W

Bit	Symbol	Bit Name	Description	R/W
b23	ECMTE023	ECM Delay Timer Start Control 24	Enables delay timer operation in response to an ECM maskable interrupt caused by a 12-bit A/D converter unit 0 overwrite interrupt (error source 24). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b24	ECMTE024	ECM Delay Timer Start Control 25	Enables delay timer operation in response to an ECM maskable interrupt caused by a 12-bit A/D converter unit 1 overwrite interrupt (error source 25). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b30 to b25	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b31	ECMTE031	ECM Delay Timer Start Control 32	Enables delay timer operation in response to an ECM maskable interrupt caused by a DOC operation error (error source 32). 0: Delay timer start disabled 1: Delay timer start enabled	R/W

29.2.26 ECM Delay Timer Configuration Register 1 (ECMDTMCFG1)

The ECMDTMCFG1 register controls delay timer operation in response to an ECM maskable interrupt. Whether to enable delay timer operation when an ECM maskable interrupt is generated can be set.
This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.
Writing to this register is protected by the specific instruction sequence. For details, see section 29.3.2, Writing to Protected Registers.

Address(es): A007 D0E4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	ECMTE108	ECMTE107	ECMTE106	ECMTE105	ECMTE104	ECMTE103	ECMTE102	—	ECMTE100
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMTE100	ECM Delay Timer Start Control 33	Enables delay timer operation in response to an ECM maskable interrupt caused by a bus error (error source 33). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	ECMTE102	ECM Delay Timer Start Control 35	Enables delay timer operation in response to an ECM maskable interrupt caused by extended pseudo error 35 (error source 35). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b3	ECMTE103	ECM Delay Timer Start Control 36	Enables delay timer operation in response to an ECM maskable interrupt caused by extended pseudo error 36 (error source 36). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b4	ECMTE104	ECM Delay Timer Start Control 37	Enables delay timer operation in response to an ECM maskable interrupt caused by extended pseudo error 37 (error source 37). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b5	ECMTE105	ECM Delay Timer Start Control 38	Enables delay timer operation in response to an ECM maskable interrupt caused by extended pseudo error 38 (error source 38). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b6	ECMTE106	ECM Delay Timer Start Control 39	Enables delay timer operation in response to an ECM maskable interrupt caused by extended pseudo error 39 (error source 39). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b7	ECMTE107	ECM Delay Timer Start Control 40	Enables delay timer operation in response to an ECM maskable interrupt caused by extended pseudo error 40 (error source 40). 0: Delay timer start disabled 1: Delay timer start enabled	R/W

Bit	Symbol	Bit Name	Description	R/W
b8	ECMTE108	ECM Delay Timer Start Control 41	Enables delay timer operation in response to an ECM maskable interrupt caused by extended pseudo error 41 (error source 41). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b31 to b9	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

29.2.27 ECM Delay Timer Configuration Register 2 (ECMDTMCFG2)

The ECMDTMCFG2 register controls delay timer operation in response to an ECM maskable interrupt. Whether to enable delay timer operation when an ECM maskable interrupt is generated can be set.
This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.
Writing to this register is protected by the specific instruction sequence. For details, see section 29.3.2, Writing to Protected Registers.

Address(es): A007 D0E8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	ECMTE228	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b27 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b28	ECMTE228	ECM Delay Timer Start Control 93	Enables delay timer operation in response to an ECM maskable interrupt caused by an ECM compare error (error source 93). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b31 to b29	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

29.2.28 ECM Delay Timer Configuration Register 3 (ECMDTMCFG3)

The ECMDTMCFG3 register controls delay timer operation in response to an ECM non-maskable interrupt. Whether to enable delay timer operation when an ECM non-maskable interrupt is generated can be set.
This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.
Writing to this register is protected by the specific instruction sequence. For details, see section 29.3.2, Writing to Protected Registers.

Address(es): A007 D0ECh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ECMTE331	—	—	—	—	—	—	ECMTE324	ECMTE323	ECMTE322	ECMTE321	ECMTE320	ECMTE319	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ECMTE315	ECMTE314	ECMTE313	ECMTE312	ECMTE311	ECMTE310	ECMTE309	ECMTE308	ECMTE307	ECMTE306	ECMTE305	ECMTE304	—	ECMTE302	—	ECMTE300
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMTE300	ECM Delay Timer Start Control 1	Enables delay timer operation in response to an ECM non-maskable interrupt caused by a WDT overflow/ refresh error (for Cortex-R4) (error source 1). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	ECMTE302	ECM Delay Timer Start Control 3	Enables delay timer operation in response to an ECM non-maskable interrupt caused by an IWDTa overflow/ refresh error (error source 3). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b4	ECMTE304	ECM Delay Timer Start Control 5	Enables delay timer operation in response to an ECM non-maskable interrupt caused by a 1-bit ECC error (with correction) or 2-bit ECC error (without correction) in the instruction cache (Tag RAM) (error source 5). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b5	ECMTE305	ECM Delay Timer Start Control 6	Enables delay timer operation in response to an ECM non-maskable interrupt caused by a 1-bit ECC error (with correction) or 2-bit ECC error (without correction) in the instruction cache (Data RAM) (error source 6). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b6	ECMTE306	ECM Delay Timer Start Control 7	Enables delay timer operation in response to an ECM non-maskable interrupt caused by a 1-bit ECC error in the data cache (Tag/Dirty RAM) (error source 7). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b7	ECMTE307	ECM Delay Timer Start Control 8	Enables delay timer operation in response to an ECM non-maskable interrupt caused by a 2-bit ECC error in the data cache (Tag/Dirty RAM) (error source 8). 0: Delay timer start disabled 1: Delay timer start enabled	R/W

Bit	Symbol	Bit Name	Description	R/W
b8	ECMTE308	ECM Delay Timer Start Control 9	Enables delay timer operation in response to an ECM non-maskable interrupt caused by a 1-bit ECC error in the data cache (Data RAM) (error source 9). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b9	ECMTE309	ECM Delay Timer Start Control 10	Enables delay timer operation in response to an ECM non-maskable interrupt caused by a 2-bit ECC error in the data cache (Data RAM) (error source 10). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b10	ECMTE310	ECM Delay Timer Start Control 11	Enables delay timer operation in response to an ECM non-maskable interrupt caused by a 1-bit ECC error in the ATCM (error source 11). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b11	ECMTE311	ECM Delay Timer Start Control 12	Enables delay timer operation in response to an ECM non-maskable interrupt caused by a 2-bit ECC error in the ATCM (error source 12). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b12	ECMTE312	ECM Delay Timer Start Control 13	Enables delay timer operation in response to an ECM non-maskable interrupt caused by a 1-bit ECC error in the BTCM (error source 13). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b13	ECMTE313	ECM Delay Timer Start Control 14	Enables delay timer operation in response to an ECM non-maskable interrupt caused by a 2-bit ECC error in the BTCM (error source 14). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b14	ECMTE314	ECM Delay Timer Start Control 15	Enables delay timer operation in response to an ECM non-maskable interrupt caused by a 1-bit ECC error in the IRAM or DRAM (error source 15). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b15	ECMTE315	ECM Delay Timer Start Control 16	Enables delay timer operation in response to an ECM non-maskable interrupt caused by a 2-bit ECC error in the IRAM or DRAM (error source 16). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b18 to b16	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b19	ECMTE319	ECM Delay Timer Start Control 20	Enables delay timer operation in response to an ECM non-maskable interrupt caused by main clock oscillation stop detection (error source 20). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b20	ECMTE320	ECM Delay Timer Start Control 21	Enables delay timer operation in response to an ECM non-maskable interrupt caused by CLMA0 oscillation stop detection (PLL0) (error source 21). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b21	ECMTE321	ECM Delay Timer Start Control 22	Enables delay timer operation in response to an ECM non-maskable interrupt caused by CLMA1 oscillation stop detection (PLL1) (error source 22). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b22	ECMTE322	ECM Delay Timer Start Control 23	Enables delay timer operation in response to an ECM non-maskable interrupt caused by CLMA2 oscillation stop detection (LOCO) (error source 23). 0: Delay timer start disabled 1: Delay timer start enabled	R/W

Bit	Symbol	Bit Name	Description	R/W
b23	ECMTE323	ECM Delay Timer Start Control 24	Enables delay timer operation in response to an ECM non-maskable interrupt caused by a 12-bit A/D converter unit 0 overwrite interrupt (error source 24). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b24	ECMTE324	ECM Delay Timer Start Control 25	Enables delay timer operation in response to an ECM non-maskable interrupt caused by a 12-bit A/D converter unit 1 overwrite interrupt (error source 25). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b30 to b25	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b31	ECMTE331	ECM Delay Timer Start Control 32	Enables delay timer operation in response to an ECM non-maskable interrupt caused by a DOC operation error (error source 32). 0: Delay timer start disabled 1: Delay timer start enabled	R/W

29.2.29 ECM Delay Timer Configuration Register 4 (ECMDTMCFG4)

The ECMDTMCFG4 register controls delay timer operation in response to an ECM non-maskable interrupt. Whether to enable delay timer operation when an ECM non-maskable interrupt is generated can be set.
This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.
Writing to this register is protected by the specific instruction sequence. For details, see section 29.3.2, Writing to Protected Registers.

Address(es): A007 D0F0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	ECMTE408	ECMTE407	ECMTE406	ECMTE405	ECMTE404	ECMTE403	ECMTE402	—	ECMTE400
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMTE400	ECM Delay Timer Start Control 33	Enables delay timer operation in response to an ECM non-maskable interrupt caused by a bus error (error source 33). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	ECMTE402	ECM Delay Timer Start Control 35	Enables delay timer operation in response to an ECM non-maskable interrupt caused by extended pseudo error 35 (error source 35). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b3	ECMTE403	ECM Delay Timer Start Control 36	Enables delay timer operation in response to an ECM non-maskable interrupt caused by extended pseudo error 36 (error source 36). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b4	ECMTE404	ECM Delay Timer Start Control 37	Enables delay timer operation in response to an ECM non-maskable interrupt caused by extended pseudo error 37 (error source 37). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b5	ECMTE405	ECM Delay Timer Start Control 38	Enables delay timer operation in response to an ECM non-maskable interrupt caused by extended pseudo error 38 (error source 38). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b6	ECMTE406	ECM Delay Timer Start Control 39	Enables delay timer operation in response to an ECM non-maskable interrupt caused by extended pseudo error 39 (error source 39). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b7	ECMTE407	ECM Delay Timer Start Control 40	Enables delay timer operation in response to an ECM non-maskable interrupt caused by extended pseudo error 40 (error source 40). 0: Delay timer start disabled 1: Delay timer start enabled	R/W

Bit	Symbol	Bit Name	Description	R/W
b8	ECMTE408	ECM Delay Timer Start Control 41	Enables delay timer operation in response to an ECM non-maskable interrupt caused by extended pseudo error 41 (error source 41). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b31 to b9	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

29.2.30 ECM Delay Timer Configuration Register 5 (ECMDTMCFG5)

The ECMDTMCFG5 register controls delay timer operation in response to an ECM non-maskable interrupt. Whether to enable delay timer operation when an ECM non-maskable interrupt is generated can be set.
This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.
Writing to this register is protected by the specific instruction sequence. For details, see section 29.3.2, Writing to Protected Registers.

Address(es): A007 D0F4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	ECMTE528	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b27 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b28	ECMTE528	ECM Delay Timer Start Control 93	Enables delay timer operation in response to an ECM non-maskable interrupt caused by an ECM compare error (error source 93). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b31 to b29	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

29.2.31 ECM Mask Control Register (ECMMCNT)

The ECMMCNT register is used for mask control for ECM compare errors (error source 93).

Address(es): A00B 0A80h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MSKM	MSKC
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	MSKC	ECM Compare Error Mask for Checker	Masks the ECM compare error source (error source 93) on the ECM checker. 0: ECM compare error not masked 1: ECM compare error masked	R/W
b1	MSKM	ECM Compare Error Mask for Master	Masks the ECM compare error source (error source 93) on the ECM master. 0: ECM compare error not masked 1: ECM compare error masked	R/W
b31 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

29.3 Operations

29.3.1 Pseudo Error Generation

ECM can generate individual pseudo error sources for self-diagnosis. ECM generates a pseudo error when the corresponding bit in the ECM pseudo error trigger register (ECMPEn) is set to 1. When a pseudo error is generated, ECM operates in the same way as when a real error occurs, and the settings for error source mask, ECM interrupt, ECM reset, and delay timer all apply in the same way.
Error sources 35 to 41 can be used as extended pseudo errors to detect errors for functional safety as shown below. When software detects an error, it can set a bit in the ECMPEn register to generate a corresponding pseudo error to use a function such as ECM interrupt, ECM reset, or delay timer.

Table 29.3 Example Assignment of Errors for Functional Safety to Error Sources

Error Source Number	Example Functions to be Assigned to Extended Pseudo Errors 35 to 41
35	Access violation to the Cortex-R4 protected area
37	PWM cycle/Duty error
38	ADC unit 0 range over error
39	ADC unit 1 range over error
40	ADC unit 0 pin-level self-diagnosis error
41	ADC unit 1 pin-level self-diagnosis error

Note: In addition to the above list, error source number 36 is available.

29.3.2 Writing to Protected Registers

Write protected registers are protected from incorrect write access due to erroneous program execution, etc.

29.3.2.1 Protection Unlock Sequence

Write access to a write protected register is only possible within the following sequence.

1. Write the fixed value 0000 00A5h to the ECM protection command register (ECMPCMD1) or the ECM master/checker protection command register (ECMmPCMD0) (m = M or C). If the register to be written is the common register, write the fixed value to the ECM protection command register (ECMPCMD1). If the register to be written is not the common register, write the fixed value to the ECM master/checker protection command register (ECMmPCMD0). See descriptions of each register in section 29.2, Registers, to find out whether the target register is a common one or not.
2. Write the desired setting value to the protected, of the ECM common, ECM master, and ECM checker according to the following sequence:
   - Write the desired setting value.
   - Write the inversed value of the desired setting value.
   - Write the desired setting value again.
3. Check the desired setting value has been successfully written to the protected register by checking that the ECMPRERR bit of the ECM protection status register (ECMPS) is 0.

In case of any access to another register between step 1 to step 3 of the above sequence, the protection mechanism behaves as follows.

- If that register belongs to the ECM, the write to the protected register fails (the ECMPRERR bit of the ECM protection status register becomes 1). The sequence has to be reexecuted from step 1.
- If that register does not belong to the ECM, the sequence is not disrupted and the write to the protected register is conducted successfully.

In case the protection unlock sequence is interrupted, the protection mechanism behaves as follows.

- Interrupts during protection sequence
  If an interrupt is acknowledged within the protection sequence and the interrupt process does not access any ECM register, the protection sequence is not disrupted and the write to the protected register can be successfully completed after returning from the interrupt process.

In case a break occurs in the protection unlock sequence, the protection mechanism behaves as follows.

- Breaks during protection sequence
  If the CPU goes into the break state during the protection sequence and any ECM register is not accessed, the protection sequence is not disrupted and the write to the protected register can be successfully completed after returning from the break.
  If the CPU goes into the break state during the protection sequence and any ECM register is accessed, the protection sequence is disrupted and the write to the protected register is not performed after returning from the break.
  Therefore, be careful not to let the CPU go into the break state during the protection sequence.

29.3.3 Timeout Function for Interrupt Processing by Using the Delay Timer

ECM can start the delay timer at the same time as an ECM maskable or non-maskable interrupt request due to occurrence of an error source is issued, to manage timeout of the interrupt processing time. If ECM cannot stop the delay timer (by setting the ECMDTMCTL.DTMSTP bit to 1) during interrupt processing and the count value of the delay timer matches with the value of the delay timer compare register, ECM can generate a delay timer overflow (error source 94) and ECM reset. Specify the setting of the ECM reset when the delay timer overflow (error source 94) occurs in the ECMIRCFG2 register.
The delay timer always starts counting up from 0 by using ECMCLK (240 kHz).
Specify the setting of the overflow cycle of the delay timer in the ECMDTMCMP register.

(Overflow cycle) = (Value set in ECMDTMCMP. ECMDTMCMP[15:0] + 1) × ECMCLK cycle (240 kHz)

Note: The delay timer continues count operation even when a break occurs.

29.4 Usage Notes

29.4.1 Notes Regarding ECMCLK

Counting by the delay timer is of cycles of ECMCKL, a signal generated by the low-speed on-chip oscillator.
After release from the reset state, counting does not proceed if the delay timer is started while the low-speed on-chip oscillator is stopped.
To enable the delay timer function, enable the low-speed on-chip oscillator beforehand and wait for the LOCO oscillation stabilization time.
For details on the low-speed on-chip oscillator, see section 7.2.5, Low-Speed On-Chip Oscillator Control Register (LOCOCR).

30. 12-Bit A/D Converter (S12ADCa)

30.1 Overview

This LSI incorporates two units of a 12-bit successive approximation A/D converter. In unit 0, up to eight analog input channels and temperature sensor output are selectable. In unit 1, up to 8 analog input channels are selectable.
The 12-bit A/D converter converts a maximum of 8 analog input channels and temperature sensor output (unit 0) and a maximum of 8 analog input channels (unit 1), which have been selected, into a 12-bit digital value through successive approximation.
The A/D converter has three operating modes: single scan mode in which the analog inputs of up to 8 (unit 0) and 8 (unit 1) channels arbitrarily selected are converted for only once in ascending channel order; continuous scan mode in which the analog inputs of up to 8 (unit 0) and 8 (unit 1) channels arbitrarily selected are continuously converted in ascending channel order; and group scan mode in which up to 8 (unit 0) and 8 (unit 1) channels of the analog inputs are arbitrarily divided into two groups (group A and group B) and converted in ascending channel order in each group.
In group scan mode, the conditions for scanning start of group A and group B (synchronous trigger) can be independently selected, thus allowing A/D conversion of group A and group B to be started independently. When group A priority control is selected along with operation as described above, if a request to start scanning for group A is received during A/D conversion for group B, the conversion operation for group B is discontinued and the conversion for group A starts, which is given priority.
In double trigger mode, one analog input channel arbitrarily selected is converted in single scan mode or group scan mode (group A), and the resulting data of A/D conversion started by the first and second synchronous triggers are stored into different registers (duplication of A/D conversion data).
The temperature sensor output can be selected at the same time as the analog input of channels. A/D conversion of the analog input of channels and the temperature sensor output is performed in that order.
Self-diagnosis, pin-level self-diagnosis, detection of guidewire malfunction, overwrite checking of data registers, and automatic clearing of data registers are provided as security functions.
Table 30.1 lists the specifications of the 12-bit A/D converter and Table 30.2 indicates the functions of the 12-bit A/D converter. Figure 30.1 shows a block diagram of the 12-bit A/D converter (unit 0) and Figure 30.2 shows a block diagram of the 12-bit A/D converter (unit 1).

Table 30.1 Specifications of 12-Bit A/D Converter (1 / 2)

Item	Specifications
Number of units	Two units
Input channels	Unit 0: Up to 8 channels Unit 1: Up to 8 channels
Extended analog function	Temperature sensor output
A/D conversion method	Successive approximation method
Resolution	12 bits
Conversion time	VREFH0, VREFH1 = 3.0 to 3.6 V Unit 0: 0.483 µs per channel (when A/D conversion clock ADCLK = 60 MHz) Unit 1: 0.833 µs per channel (when A/D conversion clock ADCLK = 60 MHz) VREFH0, VREFH1 = 2.5 to 3.0 V Unit 0: 0.883 µs per channel (when A/D conversion clock ADCLK = 60 MHz) Unit 1: 0.833 µs per channel (when A/D conversion clock ADCLK = 60 MHz)
A/D conversion clock	Peripheral module clock PCLKH and A/D conversion clock ADCLK (PCLKF or PCLKG)*1 can be set so that the frequency division ratio should be one of the following. PCLKH to ADCLK frequency division ratio = 1:1, 1:2, 1:4, 1:8 ADCLK is set using the clock generation circuit.
Data registers	• 16 registers for analog input (8 for unit 0 and 8 for unit 1), 1 for A/D-converted data duplication in double trigger mode in each unit • One register for temperature sensor output (in unit 0 only) • One register for self-diagnosis (per unit) • The results of A/D conversion are stored in 12-bit A/D data registers. • 8-, 10-, and 12-bit accuracy output for the results of A/D conversion • The value obtained by adding up A/D-converted results is stored as a value in the number of bit for conversion accuracy + 2 bits in the A/D data registers in A/D-converted value addition mode. • Double trigger mode (selectable in single scan and group scan modes): The first piece of A/D-converted analog-input data on one selected channel is stored in the data register for the channel, and the second piece is stored in the duplication register.
Operating modes	• Single scan mode: A/D conversion is performed only once on the analog inputs of up to 8 (unit 0) and 8 (unit 1) channels arbitrarily selected and on the temperature sensor output (in unit 0 only). • Continuous scan mode: A/D conversion is performed repeatedly on the analog inputs of up to 8 (unit 0) and 8 (unit 1) channels arbitrarily selected and on the temperature sensor output (in unit 0 only). • Group scan mode: Analog inputs of up to 8 (unit 0) and 8 (unit 1) channels arbitrarily selected and the temperature sensor output (in unit 0 only) are divided into group A and group B, and A/D conversion of the selected analog input is performed only once on a group basis. The conditions for scanning start of group A and group B (synchronous trigger) can be independently selected, thus allowing A/D conversion of group A and group B to be started independently. • Group scan mode (when group A is given priority): If a group A trigger is input during A/D conversion on group B, the A/D conversion on group B is stopped and A/D conversion is performed on group A. Restarting (rescanning) for A/D conversion on group B after completion of A/D conversion on group A can only be set when the ratio between the frequency division settings for PCLKH and ADCLK = 1:1.
Conditions for A/D conversion start	• Software trigger • Synchronous trigger Trigger by the event link controller (ELC), and 16-bit timer pulse unit (TPUa). • Asynchronous trigger A/D conversion can be triggered by the external trigger pins, ADTRG0 (unit 0) and ADTRG1 (unit 1).

Table 30.1 Specifications of 12-Bit A/D Converter (2 / 2)

Item	Specifications
Function	• Sample-and-hold function • Channel-dedicated sample-and-hold function (4ch: in unit 0 only) • Variable sampling state count (which can be set for each channel) • Self-diagnosis of 12-bit A/D converter • Selectable A/D-converted value addition mode or average mode • Analog input disconnection detection function (Precharge function/discharge function) • Double trigger mode (duplication of A/D conversion data) • Switching function of 8-, 10-, and 12-bit conversion*2 • Automatic clear function of A/D data registers • Compare Function (which compares the compare register and data register) • Pin-level self-diagnosis function • Overwrite checking function of the A/D data register
Interrupt source	• In the modes except double trigger mode and group scan mode, A/D scan end interrupt (S12ADI) request can be generated on completion of single scan. • In double trigger mode, A/D scan end interrupt (S12ADI) request can be generated on completion of double scan. • In group scan mode, an A/D scan end interrupt (S12ADI) request can be generated on completion of group A scan, whereas an A/D scan end interrupt for group B (S12GBADI) request can be generated on completion of group B scan. • When double trigger mode is selected in group scan mode, A/D scan end interrupt (S12ADI) request can be generated on completion of double scan of group A, whereas A/D scan end interrupt specially for group B (S12GBADI) request can be generated on completion of group B scan. • A compare interrupt (S12CMPI) can be generated in response to the results of detection using compare function. • An AD error interrupt request (S12ADE) can be generated by an error in overwrite checking for data register. • The S12ADI or S12GBADI interrupt can activate the DMA controller (DMAC).
Event link (ELC) function	• An ELC event is generated on completion of scans other than group B scan in group scan mode. • Scan can be started by a trigger output by ELC.
Low-power consumption function	• Module-stop state can be specified.*3

Note 1. Peripheral module clock PCLKH is fixed to 60 MHz. A/D conversion clock ADCLK is set according to the setting of the SCKCR.PCKF[1:0] bits and the SCKCR.PCKG[1:0] bits in unit 0 and unit 1, respectively.
Note 2. When A/D conversion accuracy is modified, A/D conversion time is also changed. See section 30.3.6 Analog Input Sampling and Scan Conversion Time, for details.
Note 3. See section 9, Low-Power Consumption Function, for details.

Table 30.2 Functions of 12-Bit A/D Converter

Item			Unit 0 (S12ADC0)	Unit 1 (S12ADC1)
Analog input channel			AN000 to AN007 Temperature sensor output	AN100 to AN107 Extended input
Conditions for A/D conversion start	Software	Software trigger	Enabled	Enabled
	External trigger	Trigger input pin	ADTRG0	ADTRG1
	Synchronous trigger (trigger from TPUa)*1	Compare match with or input capture to TPU0.TGRA, compare match with or input capture to TPU1.TGRA, compare match with or input capture to TPU2.TGRA, compare match with or input capture to TPU3.TGRA, or compare match with or input capture to TPU4.TGRA.	TPTRGAN_0	TPTRGAN_0
		Compare match with or input capture to TPU0.TGRA	TPTRG0AN_0	TPTRG0AN_0
	Synchronous trigger (trigger from ELC)	ELC trigger	ELCTRG0	ELCTRG1
Channel-dedicated sample-and-hold function	Target channel		AN000 to AN003	—
Interrupt	Interrupt request to the CPU		S12ADI0 S12GBADI0 S12CMPI0 S12ADE0	S12ADI1 S12GBADI1 S12CMPI1 S12ADE1
	Start request to DMAC		S12ADI0 S12GBADI0	S12ADI1 S12GBADI1
	Event output to ELC		S12ADI0	S12ADI1
Setting of module stop function*2, *3			MSTPCRC.MSTPCRC5 bit	MSTPCRC.MSTPCRC4 bit

Note 1. For the settings to output synchronous triggers, see the sections related to A/D converter startup of the responding modules.
Note 2. See section 9, Low-Power Consumption Function for details.
Note 3. Wait for 1 µs or longer to start A/D conversion after release from the module-stop state.

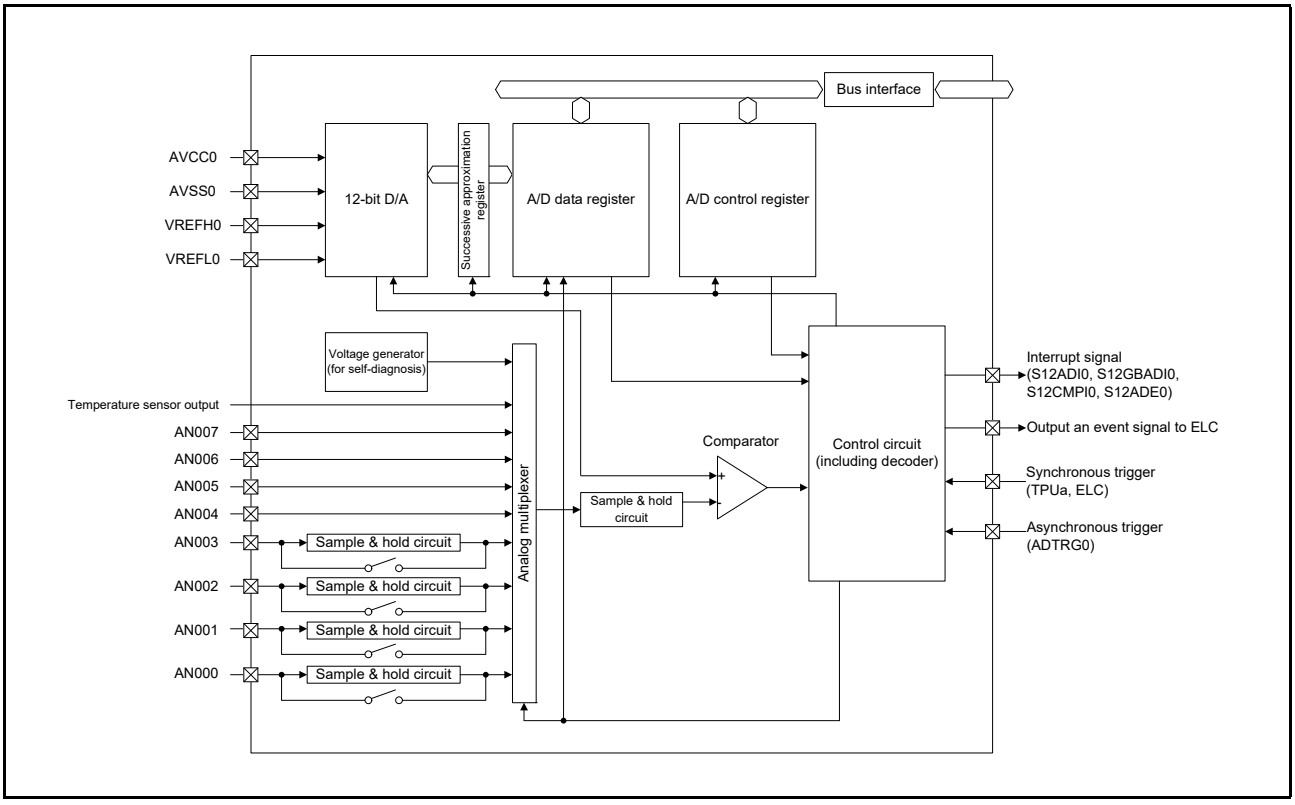


Figure 30.1 Block Diagram of 12-Bit A/D Converter (Unit 0)

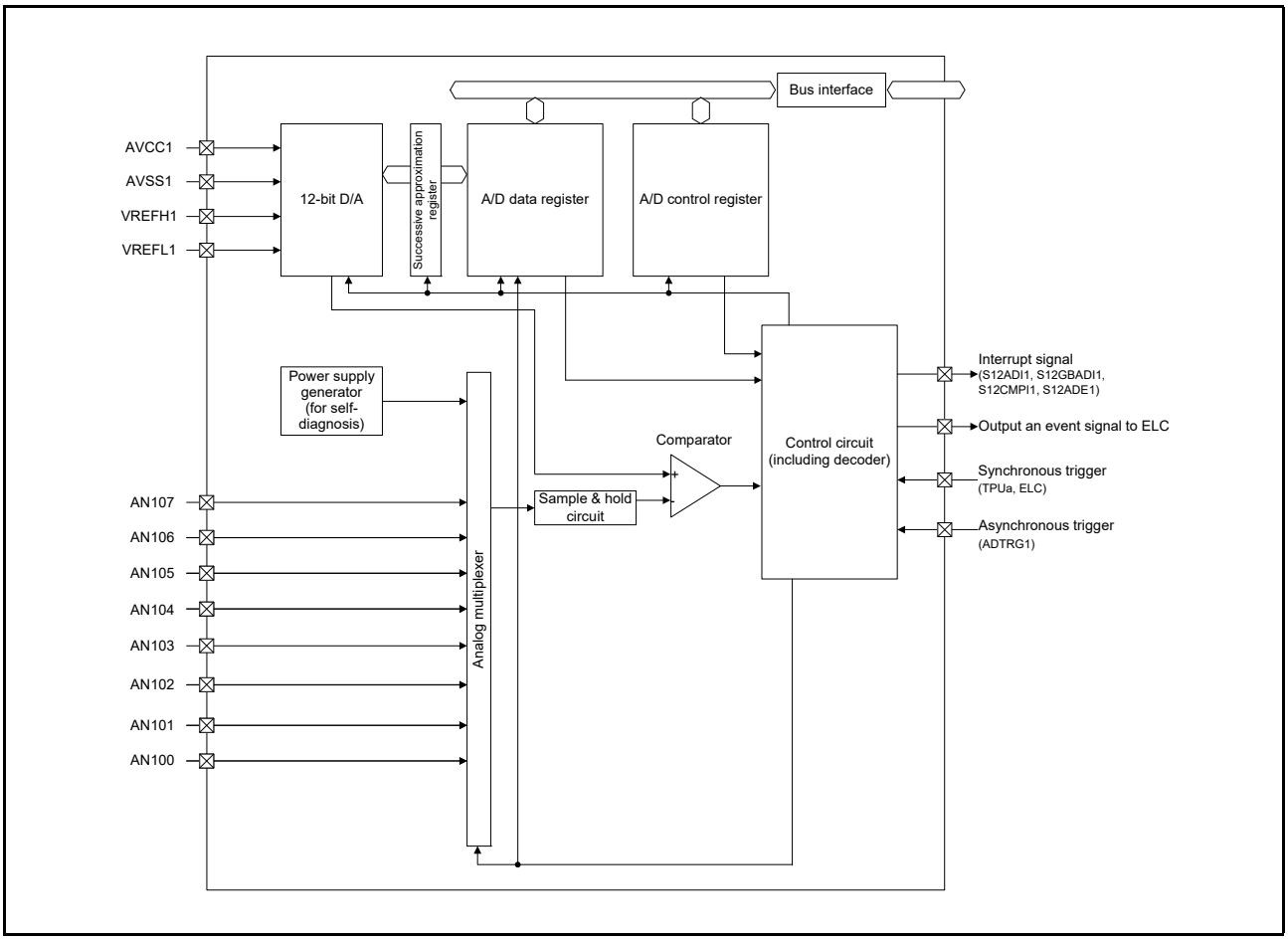


Figure 30.2 Block Diagram of 12-Bit A/D Converter (Unit 1)

Table 30.3 lists the I/O pins of the 12-bit A/D converter.

Table 30.3 I/O Pins of 12-Bit A/D Converter

Unit	Pin Name	I/O	Function
Unit 0	AVCC0	Input	Analog block power supply pin
	AVSS0	Input	Analog block ground pin
	VREFH0	Input	Reference power supply pin
	VREFL0	Input	Reference power supply ground pin
	AN000 to AN007	Input	Analog input pins 0 to 7
	ADTRG0	Input	External trigger input pin for starting A/D conversion
Unit 1	AVCC1	Input	Analog block power supply pin
	AVSS1	Input	Analog block ground pin
	VREFH1	Input	Reference power supply pin
	VREFL1	Input	Reference power supply ground pin
	AN100 to AN107	Input	Analog input pins 8 to 15
	ADTRG1	Input	External trigger input pin for starting A/D conversion

30.2 Register Descriptions

30.2.1 A/D Data Registers y (ADDRy), A/D Data Duplication Register (ADDBLDR), A/D Temperature Sensor Data Register (ADTSDR)

The ADDRy registers (y = 0 to 7 in unit 0; y = 0 to 7 in unit 1) are 16-bit read-only registers for storing the result of A/D conversion. Register ADDBLDR is a 16-bit read-only register for storing the result of A/D conversion in response to the second trigger in double trigger mode. Register ADTSDR is a 16-bit read-only register for storing the A/D conversion result of temperature sensor output.
The formats for data in the ADDRy, ADDBLDR, ADDBLDRA, ADDBLDRB, and ADTSDR registers vary according to the following conditions.

- The setting of the A/D data register format select bit (ADCER.ADRFT) (determining whether the data are flush-left or flush-right in the registers)
- The setting of the A/D conversion-accuracy selection bits (ADCER.ADPRC[1:0]) (8-, 10-, or 12-bit is selectable.)
- The setting of the addition frequency select bits (ADADC.ADC[1:0]) (once, twice, three times, or four times is selectable.)
- The setting of the average mode enable bit (ADADC.AVEE) (Addition or average is selectable.)

The data formats for each given condition are shown below.

(1) When A/D-converted value addition/average mode is not selected

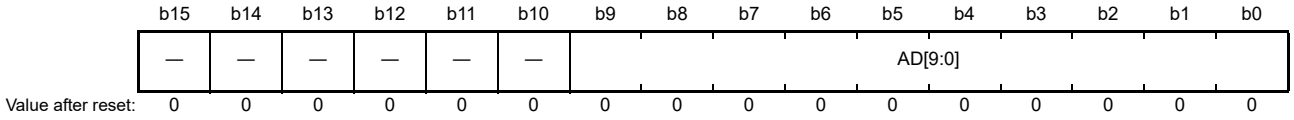
- The settings for flush-right data with 12-bit accuracy

Address(es): S12ADC0.ADDBLDR A008 C018h, S12ADC0.ADTSDR A008 C01Ah, S12ADC0.ADDR0 A008 C020h, S12ADC0.ADDR1 A008 C022h, S12ADC0.ADDR2 A008 C024h, S12ADC0.ADDR3 A008 C026h, S12ADC0.ADDR4 A008 C028h, S12ADC0.ADDR5 A008 C02Ah, S12ADC0.ADDR6 A008 C02Ch, S12ADC0.ADDR7 A008 C02Eh, S12ADC1.ADDBLDR A008 C418h, S12ADC1.ADDR0 A008 C420h, S12ADC1.ADDR1 A008 C422h, S12ADC1.ADDR2 A008 C424h, S12ADC1.ADDR3 A008 C426h, S12ADC1.ADDR4 A008 C428h, S12ADC1.ADDR5 A008 C42Ah, S12ADC1.ADDR6 A008 C42Ch, S12ADC1.ADDR7 A008 C42Eh

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	AD[11:0]											
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

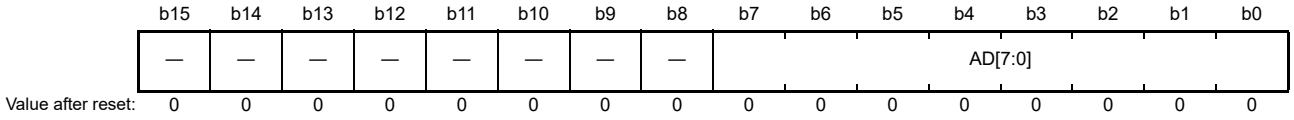
Bit	Symbol	Bit Name	Description	R/W
b11 to b0	AD[11:0]	Converted Value 11 to 0	12-bit A/D-converted value	R
b15 to b12	—	Reserved	These bits are read as 0.	R

- The settings for flush-right data with 10-bit accuracy



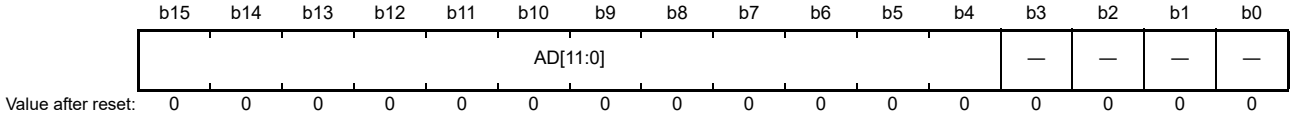
Bit	Symbol	Bit Name	Description	R/W
b9 to b0	AD[9:0]	Converted Value 9 to 0	10-bit A/D-converted value	R
b15 to b10	—	Reserved	These bits are read as 0.	R

- The settings for flush-right data with 8-bit accuracy



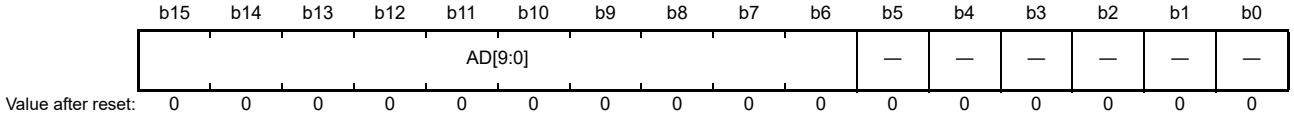
Bit	Symbol	Bit Name	Description	R/W
b7 to b0	AD[7:0]	Converted Value 7 to 0	8-bit A/D-converted value	R
b15 to b8	—	Reserved	These bits are read as 0.	R

- The settings for flush-left data with 12-bit accuracy



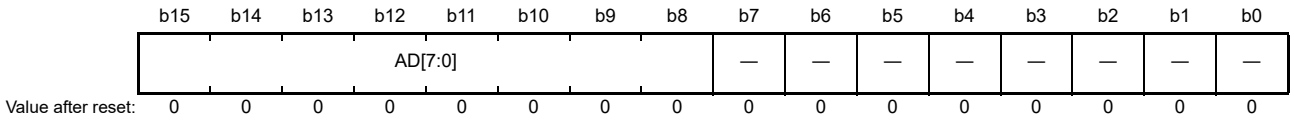
Bit	Symbol	Bit Name	Description	R/W
b3 to b0	—	Reserved	These bits are read as 0.	R
b15 to b4	AD[11:0]	Converted Value 11 to 0	12-bit A/D-converted value	R

- The settings for flush-left data with 10-bit accuracy



Bit	Symbol	Bit Name	Description	R/W
b5 to b0	—	Reserved	These bits are read as 0.	R
b15 to b6	AD[9:0]	Converted Value 9 to 0	10-bit A/D-converted value	R

- The settings for flush-left data with 8-bit accuracy



Bit	Symbol	Bit Name	Description	R/W
b7 to b0	—	Reserved	These bits are read as 0.	R
b15 to b8	AD[7:0]	Converted Value 7 to 0	8-bit A/D-converted value	R

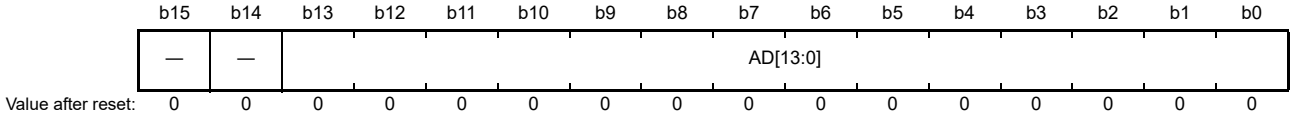
(2) When A/D-converted value average mode is selected

When A/D-converted value average mode is selected, the AD[11:0] bits indicate the mean of A/D-converted values on a specific channel. Even if A/D-converted value average mode is selected, the value is stored in the A/D data register according to the settings of the A/D data register format select bit in the same way as normal A/D conversion.

(3) When A/D-converted value addition mode is selected

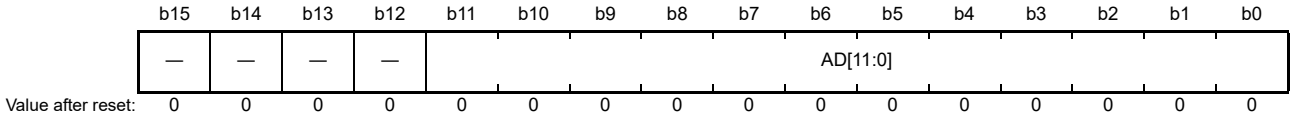
When A/D-converted value addition mode is selected, the AD[13:0] bits indicate the value that is obtained by adding up A/D-converted values on a specific channel. In A/D-converted value addition mode, the value obtained by adding up of A/D conversion results is retained in the A/D data register as a 2-bit-extended value of the conversion accuracy specified. Even if A/D-converted value addition mode is selected, the value is stored in the A/D data register according to the settings of the A/D data register format select bits.
The data formats for each given condition are shown below.

- The settings for flush-right data with 12-bit accuracy (when A/D-converted value addition mode is selected)



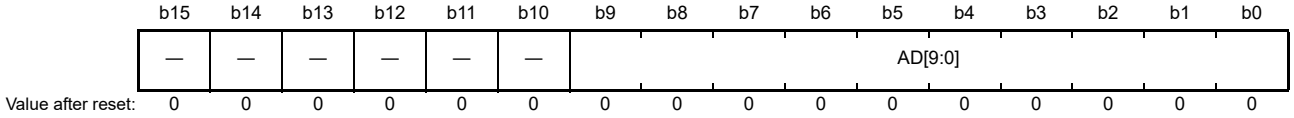
Bit	Symbol	Bit Name	Description	R/W
b13 to b0	AD[13:0]	Added Value 13 to 0	14-bit value obtained by adding up of A/D conversion results	R
b15, b14	—	Reserved	These bits are read as 0.	R

- The settings for flush-right data with 10-bit accuracy (when A/D-converted value addition mode is selected)



Bit	Symbol	Bit Name	Description	R/W
b11 to b0	AD[11:0]	Added Value 11 to 0	12-bit value obtained by adding up of A/D conversion results	R
b15 to b12	—	Reserved	These bits are read as 0.	R

- The settings for flush-right data with 8-bit accuracy (when A/D-converted value addition mode is selected)

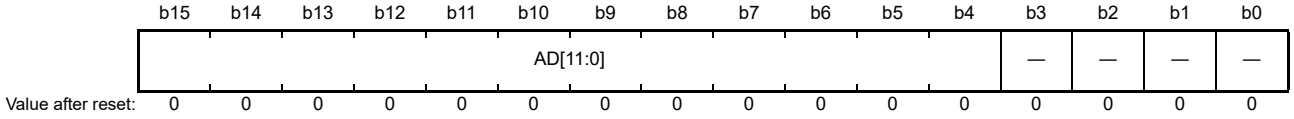


Bit	Symbol	Bit Name	Description	R/W
b9 to b0	AD[9:0]	Added Value 9 to 0	10-bit value obtained by adding up of A/D conversion results	R
b15 to b10	—	Reserved	These bits are read as 0.	R

- The settings for flush-left data with 12-bit accuracy (when A/D-converted value addition mode is selected)

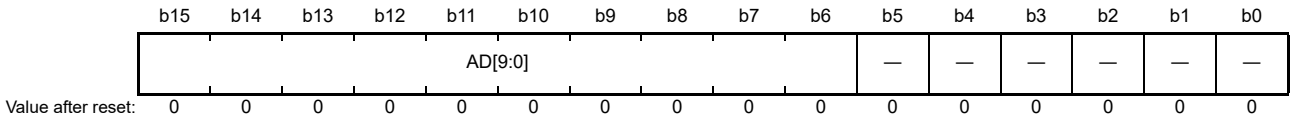
Bit	Symbol	Bit Name	Description	R/W
b1, b0	—	Reserved	These bits are read as 0.	R
b15 to b2	AD[13:0]	Added Value 13 to 0	14-bit value obtained by adding up of A/D conversion results	R

- The settings for flush-left data with 10-bit accuracy (when A/D-converted value addition mode is selected)



Bit	Symbol	Bit Name	Description	R/W
b3 to b0	—	Reserved	These bits are read as 0.	R
b15 to b4	AD[11:0]	Added Value 11 to 0	12-bit value obtained by adding up of A/D conversion results	R

- The settings for flush-left data with 8-bit accuracy (when A/D-converted value addition mode is selected)



Bit	Symbol	Bit Name	Description	R/W
b5 to b0	—	Reserved	These bits are read as 0.	R
b15 to b6	AD[9:0]	Added Value 9 to 0	10-bit value obtained by adding up of A/D conversion results	R

30.2.2 A/D Self-Diagnosis Data Register (ADRD)

ADRD is a 16-bit read-only register that holds the A/D conversion results based on the 12-bit A/D converter's self-diagnosis. In addition to the AD bit indicating A/D-converted value, the self-diagnosis status bit (DIAGST) is included in. In the ADRD register, the following different formats are used depending on the conditions below.

- The setting of the A/D data register format select bit (ADCER.ADRFT) (determining whether the data are flush-left or flush-right in the registers)
- The setting of the A/D conversion-accuracy selection bits (ADCER.ADPRC[1:0]) (8-, 10-, or 12-bit is selectable.)

The A/D-converted value addition mode and A/D-converted value average mode cannot be applied to the A/D self-diagnosis function. For details of self-diagnosis, see section 30.2.8 A/D Control Extended Register (ADCER). The data formats for each given condition are shown below.

- The settings for flush-right data with 12-bit accuracy

Address(es): S12ADC0.ADRD A008 C01Eh, S12ADC1.ADRD A008 C41Eh

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	DIAGST[1:0]		—	—	AD[11:0]											
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b11 to b0	AD[11:0]	Converted Value 11 to 0	12-bit A/D-converted value	R
b13, b12	—	Reserved	These bits are read as 0.	R
b15, b14	DIAGST[1:0]	Self-Diagnosis Status	b15 b14 0 0: Self-diagnosis has never been executed since power-on. 0 1: Self-diagnosis using the voltage of 0 V has been executed. 1 0: Self-diagnosis using the voltage of reference power supply*1 × 1/2 has been executed. 1 1: Self-diagnosis using the voltage of reference power supply*1 has been executed. For details of self-diagnosis, see section 30.2.8 A/D Control Extended Register (ADCER).	R

Note 1. "Reference voltage" refers to VREFH0 for unit 0 and to VREFH1 for unit 1.

- The settings for flush-right data with 10-bit accuracy

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	DIAGST[1:0]		—	—	—	—	AD[9:0]									
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b9 to b0	AD[9:0]	Converted Value 9 to 0	10-bit A/D-converted value	R
b13 to b10	—	Reserved	These bits are read as 0.	R
b15, b14	DIAGST[1:0]	Self-Diagnosis Status	b15 b14 0 0: Self-diagnosis has never been executed since power-on. 0 1: Self-diagnosis using the voltage of 0 V has been executed. 1 0: Self-diagnosis using the voltage of reference power supply*1 × 1/2 has been executed. 1 1: Self-diagnosis using the voltage of reference power supply*1 has been executed. For details of self-diagnosis, see section 30.2.8 A/D Control Extended Register (ADCER).	R

Note 1. “Reference voltage” refers to VREFH0 for unit 0 and to VREFH1 for unit 1.

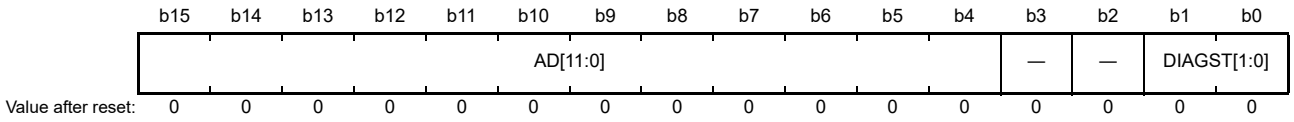
- The settings for flush-right data with 8-bit accuracy

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	DIAGST[1:0]		—	—	—	—	—	—	AD[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	AD[7:0]	Converted Value 7 to 0	8-bit A/D-converted value	R
b13 to b8	—	Reserved	These bits are read as 0.	R
b15, b14	DIAGST[1:0]	Self-Diagnosis Status	b15 b14 0 0: Self-diagnosis has never been executed since power-on. 0 1: Self-diagnosis using the voltage of 0 V has been executed. 1 0: Self-diagnosis using the voltage of reference power supply*1 × 1/2 has been executed. 1 1: Self-diagnosis using the voltage of reference power supply*1 has been executed. For details of self-diagnosis, see section 30.2.8 A/D Control Extended Register (ADCER).	R

Note 1. “Reference voltage” refers to VREFH0 for unit 0 and to VREFH1 for unit 1.

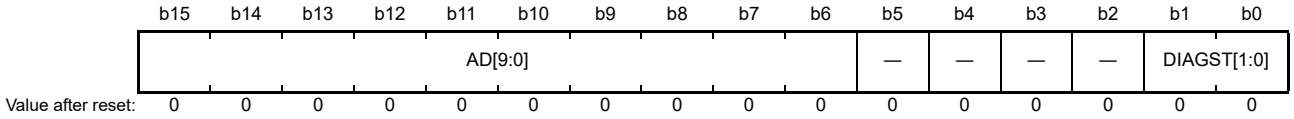
- The settings for flush-left data with 12-bit accuracy



Bit	Symbol	Bit Name	Description	R/W
b1, b0	DIAGST[1:0]	Self-Diagnosis Status	b1 b0 0 0: Self-diagnosis has never been executed since power-on. 0 1: Self-diagnosis using the voltage of 0 V has been executed. 1 0: Self-diagnosis using the voltage of reference power supply*1 × 1/2 has been executed. 1 1: Self-diagnosis using the voltage of reference power supply*1 has been executed. For details of self-diagnosis, see section 30.2.8 A/D Control Extended Register (ADCER).	R
b3, b2	—	Reserved	These bits are read as 0.	R
b15 to b4	AD[11:0]	Converted Value 11 to 0	12-bit A/D-converted value	R

Note 1. “Reference voltage” refers to VREFH0 for unit 0 and to VREFH1 for unit 1.

- The settings for flush-left data with 10-bit accuracy



Bit	Symbol	Bit Name	Description	R/W
b1, b0	DIAGST[1:0]	Self-Diagnosis Status	b1 b0 0 0: Self-diagnosis has never been executed since power-on. 0 1: Self-diagnosis using the voltage of 0 V has been executed. 1 0: Self-diagnosis using the voltage of reference power supply*1 × 1/2 has been executed. 1 1: Self-diagnosis using the voltage of reference power supply*1 has been executed. For details of self-diagnosis, see section 30.2.8 A/D Control Extended Register (ADCER).	R
b5 to b2	—	Reserved	These bits are read as 0.	R
b15 to b6	AD[9:0]	Converted Value 9 to 0	10-bit A/D-converted value	R

Note 1. “Reference voltage” refers to VREFH0 for unit 0 and to VREFH1 for unit 1.

- The settings for flush-left data with 8-bit accuracy

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	AD[7:0]								—	—	—	—	—	—	DIAGST[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	DIAGST[1:0]	Self-Diagnosis Status	b1 b0 0 0: Self-diagnosis has never been executed since power-on. 0 1: Self-diagnosis using the voltage of 0 V has been executed. 1 0: Self-diagnosis using the voltage of reference power supply*1 × 1/2 has been executed. 1 1: Self-diagnosis using the voltage of reference power supply*1 has been executed. For details of self-diagnosis, see section 30.2.8 A/D Control Extended Register (ADCER).	R
b7 to b2	—	Reserved	These bits are read as 0.	R
b15 to b8	AD[7:0]	Converted Value 7 to 0	8-bit A/D-converted value	R

Note 1. “Reference voltage” refers to VREFH0 for unit 0 and to VREFH1 for unit 1.

30.2.3 A/D Control Register (ADCSR)

ADCSR sets double trigger mode, A/D conversion start trigger; enables or disables scan end interrupt; selects the scan mode; and starts or stops A/D conversion.

Address(es): S12ADC0.ADCSR A008 C000h, S12ADC1.ADCSR A008 C400h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ADST	ADCS[1:0]		ADIE	—	—	TRGE	EXTRG	DBLE	GBADIE	—	DBLANS[4:0]				
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b4 to b0	DBLANS[4:0]	Double Trigger Channel Select	These bits select one analog input channel for double triggered operation. The setting is only effective while double trigger mode is selected.	R/W
b5	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b6	GBADIE	Group B Scan End Interrupt Enable	0: Disables S12GBADI interrupt generation upon group B scan completion. 1: Enables S12GBADI interrupt generation upon group B scan completion.	R/W
b7	DBLE	Double Trigger Mode Select	0: Deselects double trigger mode. 1: Selects double trigger mode.	R/W
b8	EXTRG	Trigger Select*1	0: A/D conversion is started by a synchronous trigger (TPUa, ELC). 1: A/D conversion is started by the asynchronous trigger (ADTRG0 in unit 0; ADTRG1 in unit 1).	R/W
b9	TRGE	Trigger Start Enable	0: Disables A/D conversion to be started by the synchronous or asynchronous trigger. 1: Enables A/D conversion to be started by the synchronous or asynchronous trigger.	R/W
b11, b10	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b12	ADIE	Scan End Interrupt Enable	0: Disables S12ADI interrupt generation upon scan completion. 1: Enables S12ADI interrupt generation upon scan completion.	R/W
b14, b13	ADCS[1:0]	Scan Mode Select	b14 b13 0 0: Single scan mode 0 1: Group scan mode 1 0: Continuous scan mode 1 1: Setting prohibited	R/W
b15	ADST	A/D Conversion Start	0: Stops A/D conversion process. 1: Starts A/D conversion process.	R/W

Note 1. Starting A/D conversion using an external pin (asynchronous trigger)
After a low-level signal is input to the external pin (ADTRG0 in unit 0; ADTRG1 in unit 1), write 1 to both the TRGE and EXTRG bits in ADCSR and change the signals of ADTRG0 in unit 0 and ADTRG1 in unit 1 to the high level. Thus the rising edge of ADTRG0 in unit 0 and ADTRG1 in unit 1 are detected and the scan conversion process is started. In this case, the pulse width of the high-level input must be at least 1.5 clock cycles of PCLK.

DBLANS[4:0] Bits (Double Trigger Channel Select)

The DBLANS[4:0] bits select one of the channels for A/D conversion data duplication in double trigger mode. The A/D conversion results of the analog input of the channel selected by the DBLANS[4:0] bits are stored into the A/D data register y when conversion is started by the first trigger in double trigger mode, and into the A/D data duplication register when started by the second trigger. Table 30.4 shows selection of the channel for double triggered operation.
When double trigger mode is selected, the channels selected by the ADANSA register are invalid, and the channel selected by the DBLANS[4:0] bits is subjected to A/D conversion instead.
When double trigger mode is used, do not select A/D conversion for the temperature sensor output. The DBLANS[4:0] bits should be set while the ADST bit is 0. (They should not be set simultaneously when 1 is written to the ADST bit.)
To use A/D-converted value addition/average mode while double trigger mode is set, the channel selected by the DBLANS[4:0] bits should be selected in the ADANSA register.

Table 30.4 Relationship between DBLANS Bit Settings and Double Trigger Enabled Channels

Unit 0		Unit 1	
DBLANS[4:0]	**Duplication Channel**	**DBLANS[4:0]**	**Duplication Channel**
00000	AN000	00000	AN100
00001	AN001	00001	AN101
00010	AN002	00010	AN102
00011	AN003	00011	AN103
00100	AN004	00100	AN104
00101	AN005	00101	AN105
00110	AN006	00110	AN106
00111	AN007	00111	AN107

Note: Settings other than above are prohibited.

GBADIE Bit (Group B Scan End Interrupt Enable)

The GBADIE bit sets whether to enable or disable group B scan end interrupt (S12GBADI) in group scan mode.

DBLE Bit (Double Trigger Mode Select)

Double trigger mode has a function to store the resulting data of A/D conversion started by the first and second synchronous triggers into separate registers.
When double trigger mode is selected, the channels specified in the ADANS register are invalid and the channel selected by the DBLANS[4:0] bits is effective instead. Double trigger mode is only operated by the synchronous trigger (TPUa, ELC) selected by the ADSTRGR.TRSA[5:0] bits. Neither asynchronous trigger nor software trigger can be operated in double trigger mode. The A/D conversion results started by the first trigger are stored into the A/D data register y and those started by the second trigger are stored into the A/D data duplication register. In this case, if the ADIE bit is set to 1, the interrupt is output not upon completion of the first scan but upon completion of the second scan.
In continuous scan mode, double trigger mode should not be selected. Also do not select double trigger mode for conversion of temperature sensor output.
The DBLE bit should be set after the ADST bit has been set to 0.

EXTRG Bit (Trigger Select)

The EXTRG bit selects the synchronous trigger or the asynchronous trigger as the trigger for starting A/D conversion.

TRGE Bit (Trigger Start Enable)

The TRGE bit sets whether to enable or disable starting of A/D conversion by the synchronous trigger and the asynchronous trigger.
This bit should be set to 1 in group scan mode.

ADIE Bit (Scan End Interrupt Enable)

The ADIE bit sets whether to enable or disable generation of the A/D scan end interrupt (S12ADI) in group scan mode (except for group B scan).
With double trigger mode deselected, the S12ADI interrupt is generated after the first scan is completed if the ADIE bit is set to 1.
With double trigger mode selected, the S12ADI interrupt is generated after the second scan is completed if the ADIE bit is set to 1 as long as the scan is started by the synchronous trigger (TPUa, ELC) selected by the ADSTRGR.TRSA[5:0] bits.

ADCS[1:0] Bits (Scan Mode Select)

The ADCS bit select the scan mode.
In single scan mode, A/D conversion is performed for the analog inputs of a maximum of eight channels in unit 0 and 8 channels in unit 1 selected with the ADANSA register in the ascending order of the channel number, and when one cycle of A/D conversion is completed for all the selected channels, the scan conversion is stopped.*1
In continuous scan mode, while the ADST bit in ADCSR is 1, A/D conversion is performed for the analog inputs of a maximum of eight channels in unit 0 and 8 channels in unit 1 selected with the ADANSA register in the ascending order of the channel number, and when one cycle of A/D conversion is completed for all the selected channels, A/D conversion is repeated from the first channel. If the ADST bit in ADCSR is set to 0 during continuous scan, A/D conversion is stopped even if scanning is in progress.*1
In group scan mode, A/D conversion is performed for the analog inputs (group A) of a maximum of eight channels in unit 0 and 8 channels in unit 1 selected with the ADANSA register in the ascending order of the channel number after scanning is started by the synchronous trigger (TPUa, ELC) selected by the TRSA[5:0] bits in ADSTRGR, and when one cycle of A/D conversion is completed for all the selected channels, A/D conversion is stopped. A/D conversion is also performed for the analog inputs (group B) of a maximum of eight channels in unit 0 and 8 channels in unit 1 selected with the ADANSB register in the ascending order of the channel number after scanning is started by the synchronous trigger (TPUa, ELC) selected by the TRSB[5:0] bits in ADSTRGR, and when one cycle of A/D conversion is completed for all the selected channels, A/D conversion is stopped. If the conversion processes in group A and B occur at the same time, ADC cannot control those conversion separately. In this case, set the group A priority control setting bit (ADGSPCR.PGS) in the A/D group scan priority control register (ADGSPCR) to 1 in order to assign a priority to conversion of group A.*1
In group scan mode, different channels and triggers should be selected for group A and group B.
The ADCS[1:0] bits should be set while the ADST bit is 0 (it should not be set simultaneously when 1 is written to the ADST bit.)

Note 1. When the temperature sensor output is selected, A/D conversion of the designated analog input channels is followed by A/D conversion of the temperature sensor output.

Table 30.5 shows the selectable targets for A/D conversion depending on the settings of scan mode and double trigger mode.

Table 30.5 Selectable Targets for A/D Conversion Depending on the Settings of Scan Mode and Double Trigger Mode

		Targets for A/D Conversion			
Scan Mode Setting	**Double Trigger Mode Setting**	**Self-Diagnosis**	**Analog Input (Including Group A)**	**Analog Input (Group B)**	**Temperature Sensor Output**
Single scan	DBLE = 0	√	√	×	√
	DBLE = 1	√	√ (1ch only)	×	×
Continuous scan	DBLE = 0	√	√	×	√
	DBLE = 1	×	×	×	×
Group scan	DBLE = 0	√	√	√	√
	DBLE = 1	×	√ (1ch only)	√	×

Note: √: Selectable; ×: Not selectable

ADST Bit (A/D Conversion Start)

The ADST bit starts or stops A/D conversion process.

Before the ADST bit is set to 1, set the A/D conversion clock, the conversion mode, and conversion target analog input.

[Setting conditions]

The ADST bit is set to 1 if one of the following conditions is satisfied:

- 1 is written by software.
- The synchronous trigger (TPUa, ELC) selected by the ADSTRGR.TRSA[5:0] bits is detected with ADCSR.EXTRG and ADCSR.TRGE bits being set to 0 and 1, respectively.
- The synchronous trigger (TPUa, ELC) selected by the ADSTRGR.TRSB[5:0] bits is detected with the ADCSR.TRGE bit being set to 1 in group scan mode.
- The asynchronous trigger is detected with the ADCSR.TRGE and ADCSR.EXTRG bits being set to 1 and the ADSTRGR.TRSA[5:0] bits being set to 000000b.
- With group-A priority control operation mode enabled (ADCSR.ADCS[1:0] bits = 01b and ADGSPCR.PGS bit = 1), a group B trigger is detected and A/D conversion of group B is started.
- With group-A priority control operation mode enabled (ADCSR.ADCS[1:0] bits = 01b and ADGSPCR.PGS bit = 1), the ADGSPCR.GBRSCN bit is set to 1 and A/D conversion of group B is restarted.
- With group-A priority control operation mode enabled (ADCSR.ADCS[1:0] bits = 01b and ADGSPCR.PGS bit = 1), the ADGSPCR.GBRP bit is set to 1 and A/D conversion of group B is started.

[Clearing conditions]

The ADST bit is cleared to 0 if one of the following conditions is satisfied:

- 0 is written by software.
- The A/D conversion of all the selected channels or the temperature sensor output is completed in single scan mode.
- Group A scan is completed in group scan mode.
- Group B scan is completed in group scan mode.
- With group-A priority control operation mode enabled (ADCSR.ADCS[1:0] bits = 01b and ADGSPCR.PGS bit = 1), a group A trigger is detected during group B A/D conversion and the scanning of group B is stopped.
- With group-A priority control operation mode enabled (ADCSR.ADCS[1:0] bits = 01b and ADGSPCR.PGS bit = 1), the ADGSPCR.GBRP bit is set to 1 and the scanning of group B started by a resumption trigger is completed.
- With group-A priority control operation mode enabled (ADCSR.ADCS[1:0] bits = 01b and ADGSPCR.PGS bit = 1), the ADGSPCR.GBRSCN bit is set to 1 and the scanning of group B by a trigger is completed.

Note: When group-A priority control operation mode has been enabled (ADCSR.ADCS[1:0] bits = 01b and ADGSPCR.PGS bit = 1), do not set the ADST bit to 1.

Note: When group-A priority control operation mode has been enabled (ADCSR.ADCS[1:0] bits = 01b and ADGSPCR.PGS bit = 1) and ADGSPCR.GBRP = 1, do not set the ADST bit to 0. When forcibly terminating A/D conversion, follow the procedure for clearing the ADST bit.

30.2.4 A/D Channel Select Register A (ADANSA)

ADANSA selects analog input channels for A/D conversion among AN000 to AN007 (unit 0) and AN100 to AN107 (unit 1). In group scan mode, this register selects group A channels.

Address(es): S12ADC0.ADANSA A008 C004h, S12ADC1.ADANSA A008 C404h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	ANSA[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	ANSA[7:0]	A/D Conversion Channel Select	0: AN000 to AN007 (unit 0) and AN100 to AN107 (unit 1) are not subjected to conversion. 1: AN000 to AN007 (unit 0) and AN100 to AN107 (unit 1) are subjected to conversion.	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

ANSA[7:0] Bits (A/D Conversion Channel Select)

The ANSA[7:0] bits select analog input channels for A/D conversion among AN000 to AN007 (unit 0) and AN100 to AN107 (unit 1). The channels to be selected and the number of channels can be arbitrarily set. In unit 0, the ANSA[0] bit corresponds to AN000 and the ANSA[7] bit corresponds to AN007. In unit 1, the ANSA[0] bit corresponds to AN100 and the ANSA[7] bit corresponds to AN107.

When double trigger mode is selected, the channel selected by the ANSA[7:0] bits is invalid, and the channel selected by the ADCSR.DBLANS[4:0] bits is selected in group A instead.

When group scan mode is selected, do not select the channels specified in the A/D channel select register B (ADANSB).

The ANSA[7:0] bits should be set while the ADCSR.ADST bit is 0.

30.2.5 A/D Channel Select Register B (ADANSB)

ADANSB selects analog input channels for A/D conversion among AN000 to AN007 (unit 0) and AN100 to AN107 (unit 1) in group B when group scan mode is selected. The ADANSB register is not used in any scan mode other than group scan mode.

Address(es): S12ADC0.ADANSB A008 C014h, S12ADC1.ADANSB A008 C414h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	ANSB[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	ANSB[7:0]	A/D Conversion Channel Select	0: AN000 to AN007 (unit 0) and AN100 to AN107 (unit 1) are not subjected to conversion. 1: AN000 to AN007 (unit 0) and AN100 to AN107 (unit 1) are subjected to conversion.	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

ANSB[7:0] Bits (A/D Conversion Channel Select)

The ANSB[7:0] bits select analog input channels for A/D conversion among AN000 to AN007 (unit 0) and AN100 to AN107 (unit 1) in group B when group scan mode is selected. The ADANSB register is used for group scan mode only; not used for any other modes. The channels specified in group A (the channels corresponding to group A, selected with the ADANSA register and the ADCSR.DBLANS[4:0] bits in double trigger mode) should be excluded as the channels to be selected and the number of channels to be set.

In unit 0, the ANSB[0] bit corresponds to AN000 and the ANSB[7] bit corresponds to AN007. In unit 1, the ANSB[0] bit corresponds to AN100 and the ANSB[7] bit corresponds to AN107.

The ANSB[7:0] bits should be set while the ADCSR.ADST bit is 0.

30.2.6 A/D-Converted Value Addition/Average Mode Select Register (ADADS)

ADADS selects the channels AN000 to AN007 and AN100 to AN107 on which A/D conversion is performed successively 2 to 4 times and then converted values are added (integrated) or averaged.

Address(es): S12ADC0.ADADS A008 C008h, S12ADC1.ADADS A008 C408h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	ADS[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	ADS[7:0]	A/D-Converted Value Addition/ Average Channel Select	0: A/D-converted value addition/average mode for AN000 to AN007 (unit 0) and AN100 to AN107 (unit 1) is not selected. 1: A/D-converted value addition/average mode for AN000 to AN007 (unit 0) and AN100 to AN107 (unit 1) is selected.	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

ADS[7:0] Bits (A/D-Converted Value Addition/Average Channel Select)

When the ADS[n] bit of the number that is the same as that of A/D-converted channel selected by the ANSA[n] bits (n = 0 to 7 in unit 0, or n = 0 to 7 in unit 1) in ADANSA or DBLANS[4:0] bits in ADCSR and ANSB[n] bits (n = 0 to 7 in unit 0, or n = 0 to 15 in unit 1) in ADANSB is set to 1, A/D conversion of analog input of the selected channels is performed successively 2 to 4 times that is set with the ADC[1:0] bits in ADADC. When the ADADC.AVEE bit is 0, the value obtained by addition (integration) is stored in the A/D data register. When the ADADC.AVEE bit is 1, the mean value of the results obtained by addition (integration) is stored in the A/D data register. As for the channel on which the A/D conversion is performed and addition/average mode is not selected, a normal one-time conversion is executed and the conversion result is stored to the A/D data register.

The ADS[7:0] bits should be set while the ADCSR.ADST bit is 0.

Figure 30.3 shows the scan sequence when ADS[2] and ADS[6] are set to 1.

It is assumed that the scan is operated in continuous scan mode (ADCS = 10b in ADCSR), addition mode is selected (ADADC.AVEE = 0), the number of additions is set to 3 (conversion is performed four times) (ADADC.ADC[1:0] = 11b), and AN000 to AN007 is selected (ADANSA.ANSA[7:0] = FFh). Conversion is started from AN000. Conversion for AN002 is performed four times, and the value obtained by addition (integration) is stored in the A/D data register 2. Then, conversion for AN003 is started. Conversion for AN006 is performed four times successively, and the value obtained by addition (integration) is stored in the A/D data register 6. After conversion of AN007 is completed, the same sequence repeats from AN000.

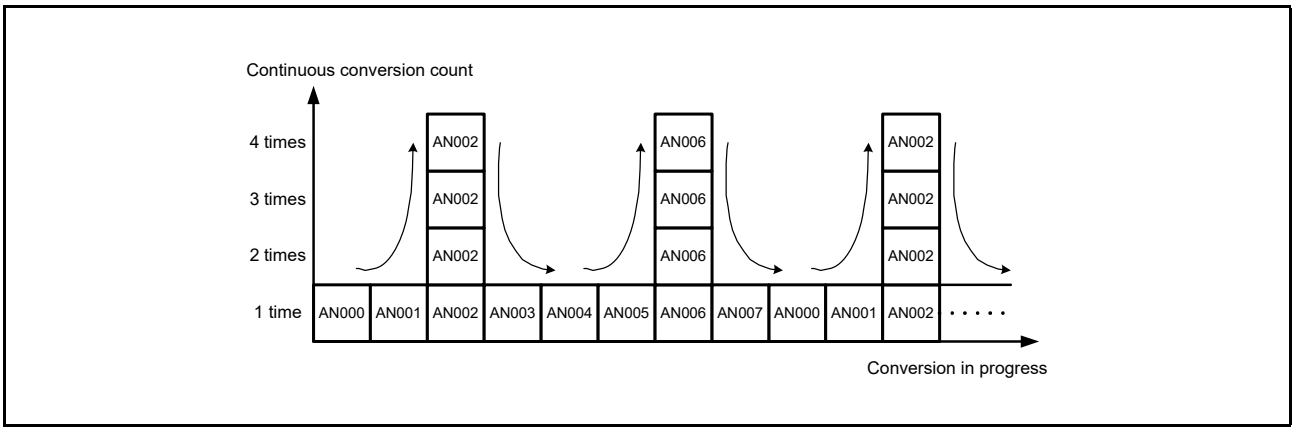


Figure 30.3 Scan Conversion Sequence with ADADC.ADC[1:0] = 11b, ADADC.AVEE = 0, ADS[2] = 1, and ADS[6] = 1

30.2.7 A/D-Converted Value Addition/Average Count Select Register (ADADC)

ADADC sets the addition count and average count for A/D conversion of the channel for which A/D-converted value addition/average mode is selected and for A/D conversion of temperature sensor output, and selects either addition or average mode.

Address(es): S12ADC0.ADADC A008 C00Ch, S12ADC1.ADADC A008 C40Ch

	b7	b6	b5	b4	b3	b2	b1	b0
	AVEE	—	—	—	—	—	ADC[1:0]	
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	ADC[1:0]	Addition Count Select	b1 b0 0 0: 1-time conversion (no addition; same as normal conversion) 0 1: 2-time conversion (addition once) 1 0: 3-time conversion (addition twice)*1 1 1: 4-time conversion (addition three times)	R/W
b6 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b7	AVEE	Average Mode Enable	0: Addition mode is selected. 1: Average mode is selected.	R/W

Note 1. When average mode is selected by setting the ADADC.AVEE bit to 1, do not set the addition count to three times (ADADC.ADC[1:0] = 10b)

ADC[1:0] Bits (Addition Count Select)

The ADC[1:0] bits set the addition count common to the channels for which A/D conversion and A/D-converted value addition/average mode is selected, including the channels selected in double trigger mode (by ADCSR.DBLANS[4:0] bits), and to A/D conversion of temperature sensor output.

When average mode is selected by setting the ADADC.AVEE bit to 1, do not set the addition count to three times (ADADC.ADC[1:0] = 10b).

The ADC[1:0] bits should be set while the ADCSR.ADST bit is 0. When self-diagnosis is executed (ADCER.DIAGM = 1), do not set the ADC[1:0] bits to any value other than 00b.

AVEE Bit (Average Mode Enable)

The AVEE bit selects addition or average mode for A/D conversion of the channel for which A/D conversion and A/D-converted value addition/average mode is selected, including the channels selected in double trigger mode (by ADCSR.DBLANS[4:0] bits) and temperature sensor output.

When average mode is selected by setting the ADADC.AVEE bit to 1, do not set the addition count to three times (ADADC.ADC[1:0] = 10b).

The AVEE bits should be set while the ADCSR.ADST bit is 0.

30.2.8 A/D Control Extended Register (ADCER)

ADCER sets self-diagnosis mode, format of the A/D data registers y (ADDRy), and automatic clearing of A/D data registers.

Address(es): S12ADC0.ADCER A008 C00Eh, S12ADC1.ADCER A008 C40Eh

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ADRFMT	—	—	—	DIAGM	DIAGLD	DIAGVAL[1:0]		—	—	ACE	—	—	ADPRC[1:0]		—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2, b1	ADPRC[1:0]	A/D Conversion Accuracy Specify	b2 b1 0 0: A/D conversion is performed with 12-bit accuracy. 0 1: A/D conversion is performed with 10-bit accuracy. 1 0: A/D conversion is performed with 8-bit accuracy. 1 1: Setting is prohibited.	R/W
b4, b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b5	ACE	A/D Data Register Automatic Clearing Enable	0: Disables automatic clearing. 1: Enables automatic clearing.	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b9, b8	DIAGVAL[1:0]	Self-Diagnosis Conversion Voltage Select	b9 b8 0 0: Setting prohibited when self-diagnosis is enabled 0 1: Uses the voltage of 0 V for self-diagnosis. 1 0: Uses the voltage of reference power supply*1 × 1/2 for self-diagnosis. 1 1: Uses the voltage of reference power supply*1 for self-diagnosis.	R/W
b10	DIAGLD	Self-Diagnosis Mode Select	0: Rotation mode for self-diagnosis voltage 1: Fixed mode for self-diagnosis voltage	R/W
b11	DIAGM	Self-Diagnosis Enable	0: Disables self-diagnosis of 12-bit A/D converter. 1: Enables self-diagnosis of 12-bit A/D converter.	R/W
b14 to b12	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15	ADRFMT	A/D Data Register Format Select	0: Right-alignment is selected for the A/D data register format. 1: Left-alignment is selected for the A/D data register format.	R/W

Note 1. “Reference voltage” refers to VREFH0 for unit 0 and to VREFH1 for unit 1.

ADPRC[1:0] Bits (A/D Conversion Accuracy Specify)

These bits select the A/D conversion accuracy among 8-, 10-, or 12-bit accuracy. When the A/D conversion accuracy is changed, the bit width of effective data stored in the result register and A/D conversion time are also changed.
See section 30.3.6 Analog Input Sampling and Scan Conversion Time for details.

ACE Bit (A/D Data Register Automatic Clearing Enable)

The ACE bit selects enabling or disabling of automatic clearing (all “0”) of the A/D data register (ADDRy, ADRD, ADDBLDR, or ADTSDR) when reading any of these registers by the CPU or DMACA. Automatic clearing of the A/D data register enables a failure which has not been updated in the A/D data register to be detected.

DIAGVAL[1:0] Bits (Self-Diagnosis Conversion Voltage Select)

These bits select the voltage value used in self-diagnosis voltage fixed mode. For details, refer to the descriptions of the ADCER.DIAGLD bit.

Self-diagnosis should not be executed by setting the ADCER.DIAGLD bit to 1 when the ADCER.DIAGVAL[1:0] bits are set to 00b.

DIAGLD Bit (Self-Diagnosis Mode Select)

The DIAGLD bit selects whether the three voltage values are rotated or the fixed voltage is used in self-diagnosis. Setting this bit (ADCER.DIAGLD) to 0 allows conversion of the voltages in rotation mode where 0, the reference power supply × 1/2, and the reference power supply are converted in this order. If the self-diagnosis voltage rotation mode is selected after a reset, rotation starts at 0 V. The fixed voltage specified by the ADCER.DIAGVAL[1:0] bits is converted when self-diagnosis voltage fixed mode is selected. In self-diagnosis voltage rotation mode, the self-diagnosis voltage value does not return to 0 when scan conversion is completed. When scan conversion is restarted, therefore, rotation starts at the voltage value following the previous value. If fixed mode is switched to rotation mode, rotation starts at the fixed voltage value.

The DIAGLD bit should be set while the ADCSR.ADST bit is 0.

DIAGM Bit (Self-Diagnosis Enable)

The DIAGM bit selects execution of self-diagnosis.

Self-diagnosis is used to detect a failure of the 12-bit A/D converter. Specifically, it is used to convert the voltage value selected from the internally generated voltage values 0, the reference power supply × 1/2, and the reference power supply. When conversion is completed, information on the converted voltage and the conversion result is stored into the self-diagnosis data register (ADRD). ADRD can then be read out by software to determine whether the conversion result falls within the normal range (normal) or not (abnormal). Self-diagnosis is executed once at the beginning of each scan, and one of the three voltages is converted. The execution time of self-diagnosis differs from the A/D conversion time of one channel.

When self-diagnosis is selected in double trigger mode, self-diagnosis is executed only for the first scan conversion by a synchronous trigger (TPUa, ELC), not executed in the second scan. When self-diagnosis is selected in group scan mode, self-diagnosis is separately executed in group A and B.

The DIAGM bit should be set while the ADCSR.ADST bit is 0.

ADRFMT Bit (A/D Data Register Format Select)

The ADRFMT bit specifies right-alignment or left-alignment for the data to be stored in ADDRy, ADDBLDR, ADTSDR, ADRD, or ADCMPDRy.

For details on the format of each data register, see section 30.2.1 A/D Data Registers y (ADDRy), A/D Data Duplication Register (ADDBLDR), A/D Temperature Sensor Data Register (ADTSDR), section 30.2.2 A/D Self-Diagnosis Data Register (ADRD), and section 30.2.20 A/D Compare Data Register y (ADCMPDRy) (y = 0, 1).

30.2.9 A/D Start Trigger Select Register (ADSTRGR)

ADSTRGR selects the A/D conversion start trigger.

Address(es): S12ADC0.ADSTRGR A008 C010h, S12ADC1.ADSTRGR A008 C410h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	TRSA[5:0]						—	—	TRSB[5:0]					
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b5 to b0	TRSB[5:0]	A/D Conversion Start Trigger Select for Group B	Select the A/D conversion start trigger for group B in group scan mode.	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b13 to b8	TRSA[5:0]	A/D Conversion Start Trigger Select	Select the A/D conversion start trigger in single scan mode and continuous mode. In group scan mode, the A/D conversion start trigger for group A is selected.	R/W
b15, b14	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

TRSB[5:0] Bits (A/D Conversion Start Trigger Select for Group B)

The TRSB[5:0] bits select the trigger to start scanning of the analog input selected in group B. The TRSB[5:0] bits require to be set only in group scan mode and are not used in any other scan mode. For the scan conversion start trigger for group B, setting a software trigger or an asynchronous trigger is prohibited. Therefore, the TRSB[5:0] bits should be set to the value other than 000000b and the ADCSR.TRGE bit should be set to 1 in group scan mode.

When group A is given priority in group scan mode, setting the ADGSPCR.GBRP bit to 1 allows group B to continuously operate in single scan mode. When setting the ADGSPCR.GBRP bit to 1, set the TRSB[5:0] bits to 3Fh.

Note that the issuance period of trigger for A/D conversion must be more than or equal to the actual scan conversion time (tSCAN). If the issuance period is less than tSCAN, A/D conversion by the trigger may have no effect.

Because an A/D conversion start trigger requires synchronization processing, a delay of the period for synchronization processing occurs. See section 30.3.6 Analog Input Sampling and Scan Conversion Time for details.

Table 30.6 lists the A/D conversion startup sources selected by the TRSB[5:0] bits.

TRSA[5:0] Bits (A/D Conversion Start Trigger Select)

The TRSA[5:0] bits select the trigger to start A/D conversion in single scan mode and continuous scan mode. In group scan mode, the trigger to start scanning of the analog input selected in group A is selected. When scanning is executed in group scan mode or double trigger mode, a software trigger and an asynchronous trigger cannot be used at the same time.

- When using the A/D conversion startup source of a synchronous trigger (TPUa, ELC), set the TRGE bit in ADCSR to 1 and set the EXTRG bit in ADCSR to 0.
- When using the asynchronous trigger (ADTRGn), set the TRGE bit in ADCSR to 1 and set the EXTRG bit in ADCSR to 1.
- Software trigger (ADCSR.ADST) is enabled regardless of the settings of the ADCSR.TRGE bit, the ADCSR.EXTRG bit, and the TRSA[5:0] bits.

Note that the issuance period of trigger for A/D conversion must be more than or equal to the actual scan conversion time (tSCAN). If the issuance period is less than tSCAN, A/D conversion by a trigger may have no effect. Because an A/D conversion start trigger requires synchronization processing, a delay of the period for synchronization processing occurs. See section 30.3.6 Analog Input Sampling and Scan Conversion Time for details.

Table 30.7 lists the selection of A/D activation sources selected by the TRSA[5:0] bits.

Table 30.6 Selection of A/D Activation Sources by the TRSB[5:0] Bits (for Group B only)

Module	Source	Remarks	TRSB[5]	TRSB[4]	TRSB[3]	TRSB[2]	TRSB[1]	TRSB[0]
Trigger source de-selection state			1	1	1	1	1	1
TPUa	TPTRGAN_0	Compare match with or input capture to TPUn.TGRA (n = 0 to 4)	0	1	1	1	1	1
	TPTRG0AN_0	Compare match with or input capture to TPU0.TGRA	1	0	0	0	0	0
ELC	ELCTRG0/ ELCTRG1	Event signals from the respective peripheral modules	1	1	0	0	0	0

Table 30.7 Selection of A/D Activation Sources by the TRSA[5:0] Bits

Module	Source	Remarks	TRSA[5]	TRSA[4]	TRSA[3]	TRSA[2]	TRSA[1]	TRSA[0]
Trigger source de-selection state			1	1	1	1	1	1
External	ADTRG0 ADTRG1	Input pin for the trigger	0	0	0	0	0	0
TPUa	TPTRGAN_0	Compare match with or input capture to TPUn.TGRA (n = 0 to 4)	0	1	1	1	1	1
	TPTRG0AN_0	Compare match with or input capture to TPU0.TGRA	1	0	0	0	0	0
ELC	ELCTRG0/ ELCTRG1	Event signals from the respective peripheral modules	1	1	0	0	0	0

30.2.10 A/D Conversion Extended Input Control Register (ADEXICR)

ADEXICR controls temperature sensor output.

Address(es): S12ADC0.ADEXICR A008 C012h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	TSSB	—	TSSA	—	—	—	—	—	—	—	TSSAD
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	TSSAD	Temperature Sensor Output A/D-Converted Value Addition/Average Mode Select	0: Temperature sensor output A/D-converted value addition/average mode is not selected. 1: Temperature sensor output A/D-converted value addition/average mode is selected.	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b8	TSSA	Temperature Sensor Output A/D Conversion Select	0: A/D conversion of temperature sensor output is disabled. 1: A/D conversion of temperature sensor output is enabled.	R/W
b9	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b10	TSSB	Temperature Sensor Output A/D Conversion Select	0: A/D conversion of temperature sensor output is disabled. 1: A/D conversion of temperature sensor output is enabled.	R/W
b15 to b11	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

TSSAD Bit (Temperature Sensor Output A/D-Converted Value Addition/Average Mode Select)

This bit selects A/D conversion of the temperature sensor output. When the TSSAD bit is set to 1, A/D conversion of the temperature sensor output is selected and performed successively 2 to 4 times that is set with the ADC[1:0] bits in ADADC. When the ADADC.AVEE bit is 0, the value obtained by addition (integration) is returned to the A/D temperature sensor data register (ADTSDR). When the ADADC.AVEE bit is 1, the mean value is returned to ADTSDR. The TSSAD bit should be set while the ADCSR.ADST bit is 0.

TSSA Bit (Temperature Sensor Output A/D Conversion Select)

This bit selects A/D conversion of the temperature sensor output for group A in single scan mode, continuous scan mode or group scan mode. When A/D conversion of the temperature sensor output is selected and performed, set the ADCSR.DBLE bit to 0.
The TSSA bit should be set while the ADCSR.ADST bit is 0.

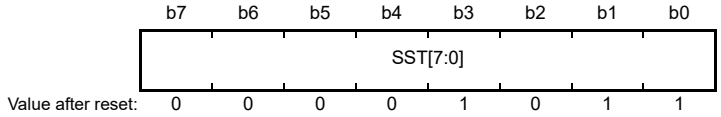
TSSB Bit (Temperature Sensor Output A/D Conversion Select)

This bit selects A/D conversion of the temperature sensor output for group B in group scan mode. When A/D conversion of the temperature sensor output is selected and performed, set the ADCSR.DBLE bit to 0.
The TSSB bit should be set while the ADCSR.ADST bit is 0. Do not set the TSSB bit to 1 while the TSSA bit is 1.

30.2.11 A/D Sampling State Register n (ADSSTRn) (n = 0 to 7, T)

The ADSSTRn register sets the sampling time for analog input.
If one state is one ADCLK (A/D conversion clock) cycle and the ADCLK clock is 60 MHz, one state is 16.7 ns. The initial value is 11 states. If the impedance of analog input signal source is too high to secure sufficient sampling time or if the ADCLK clock is slow, the sampling time can be adjusted by the value of SST[7:0] bits. The SST[7:0] bits should be set while the ADCSR.ADST bit is 0. The sampling time must be set to a value that is 5 states or more and is 255 or less.
Table 30.8 shows the relationship between the A/D sampling state register and the relevant channels.
For details, refer to section 30.3.6 Analog Input Sampling and Scan Conversion Time.

Address(es): S12ADC0.ADSSTR0 A008 C060h, S12ADC0.ADSSTRT A008 C070h, S12ADC0.ADSSTR1 A008 C073h, S12ADC0.ADSSTR2 A008 C074h, S12ADC0.ADSSTR3 A008 C075h, S12ADC0.ADSSTR4 A008 C076h, S12ADC0.ADSSTR5 A008 C077h, S12ADC0.ADSSTR6 A008 C078h, S12ADC0.ADSSTR7 A008 C079h, S12ADC1.ADSSTR0 A008 C460h, S12ADC1.ADSSTR1 A008 C473h, S12ADC1.ADSSTR2 A008 C474h, S12ADC1.ADSSTR3 A008 C475h, S12ADC1.ADSSTR4 A008 C476h, S12ADC1.ADSSTR5 A008 C477h, S12ADC1.ADSSTR6 A008 C478h, S12ADC1.ADSSTR7 A008 C479h



Bit	Symbol	Bit Name	Description	R/W
b7 to b0	SST[7:0]	Sampling Time Setting	These bits set the sampling time in the range from 5 to 255 states.	R/W

Table 30.8 Relationship between A/D Sampling State Register and Relevant Channels

	Corresponding Channels	
Bit Name	**Unit 0**	**Unit 1**
ADSSTR0.SST[7:0] bits*1	AN000	AN100
ADSSTR1.SST[7:0] bits	AN001	AN101
ADSSTR2.SST[7:0] bits	AN002	AN102
ADSSTR3.SST[7:0] bits	AN003	AN103
ADSSTR4.SST[7:0] bits	AN004	AN104
ADSSTR5.SST[7:0] bits	AN005	AN105
ADSSTR6.SST[7:0] bits	AN006	AN106
ADSSTR7.SST[7:0] bits	AN007	AN107
ADSSTRT.SST[7:0] bits	Temperature sensor output (in unit 0 only)	—

Note 1. When self-diagnosis function is selected, the sampling time set by the ADSSTR0.SST[7:0] bits is applied to it.

30.2.12 A/D Sample and Hold Circuit Control Register (ADSHCR)

ADSHCR sets the parameters related to channel-dedicated sample-and-hold circuits.

Address(es): S12ADC0.ADSHCR A008 C066h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	SHANS[3:0]				SSTSH[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	SSTSH[7:0]	Channel-Dedicated Sample-and-Hold Circuit Sampling Time Setting	Set the sampling time (4 to 255 states).	R/W
b11 to b8	SHANS[3:0]	Channel-Dedicated Sample-and-Hold Circuit Bypass Select	Select whether to use or not use (bypass) AN000 to AN003 channel-dedicated sample-and-hold circuits. 0: Bypass the channel-dedicated sample-and-hold circuits. 1: Use the channel-dedicated sample-and-hold circuits.	R/W
b15 to b12	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

SSTSH[7:0] Bits (Channel-Dedicated Sample-and-Hold Circuit Sampling Time Setting)

The SSTSH[7:0] bits set the sampling time for the channel-dedicated sample-and-hold circuits. If one state is one ADCLK (A/D conversion clock) cycle and the ADCLK clock is 60 MHz, one state is 16.7 ns. The initial value is 24 states. If the impedance of analog input signal source is too high to secure sufficient sampling time or if the ADCLK clock is slow, the sampling time can be adjusted. The SSTSH[7:0] bits should be set while the ADCSR.ADST bit is 0. The sampling time must be set to a value that is 4 states or more and is 255 or less. Also, the sampling time must be 0.4 µs or more.

SHANS[3:0] Bits (Channel-Dedicated Sample-and-Hold Circuit Bypass Select)

The SHANS[3:0] bits select whether to use or not use (bypass) AN000 to AN003 channel-dedicated sample-and-hold circuits. The SHANS[0] bit selects AN000, SHANS[1] bit selects AN001, SHANS[2] bit selects AN002, and SHANS[3] bit selects AN003. The SHANS[3:0] bits should be set while the ADST bit in ADCSR is 0.

If any channel from among AN000 to AN003 is selected for group B while operation is in group scan mode under group A priority control, make the setting to bypass the channel's dedicated sample-and-hold circuit.

The channels in unit 1 do not include channel-dedicated sample-and-hold circuits.

30.2.13 A/D Disconnection Detection Control Register (ADDISCR)

ADDISCR sets the disconnection detection assist function.

Address(es): S12ADC0.ADDISCR A008 C07Ah, S12ADC1.ADDISCR A008 C47Ah

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	ADNDIS[4:0]				
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b4 to b0	ADNDIS[4:0]	Disconnection Detection Assist Setting	Disconnection detection assist function is set.	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

ADNDIS[4:0] Bits (Disconnection Detection Assist Setting)

These bits selects either precharge or discharge and the period of precharge/discharge for the A/D disconnection detection assist function. Setting the ADNDIS[4] bit = 1 allows to select precharge and setting the ADNDIS[4] bit = 0 allows to select discharge. The period of precharge/discharge can be set with the ADNDIS[3:0] bits. When the ADNDIS[3:0] bits = 0000b, the disconnection detection assist function is not effective. Setting of the ADNDIS[3:0] bits to 0001b is prohibited. Except for the case of ADNDIS[3:0] = 0000b or 0001b, the specified value indicates the number of states for the period of precharge/discharge. When the temperature sensor output is converted or self-diagnosis is used, the disconnection detection assistance cannot be used. In that case, the ADNDIS[3:0] bits should be set to 0000b. When the ADNDIS[3:0] bits are set to any values other than 0000b or 0001b, and the disconnection detection assistance is enabled, the disconnection detection assistance for the channel-dedicated sample-and-hold circuit is also enabled. Disable the disconnection detection assistance function when pin-level self-diagnosis is executed.

30.2.14 A/D Group Scan Priority Control Register (ADGSPCR)

ADGSPCR is used to make settings for priority control of A/D conversion for group A in group scan mode.

Address(es): S12ADC0.ADGSPCR A008 C080h, S12ADC1.ADGSPCR A008 C480h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	GBRP	—	—	—	—	—	—	—	—	—	—	—	—	—	GBRSCN	PGS
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	PGS	Group-A Priority Control Setting*1	0: Operation is without group A priority control 1: Operation is with group A priority control	R/W
b1	GBRSCN	Group B Restart Setting	(Enabled only when PGS = 1. Set 0 when PGS = 0.) 0: Scanning for group B is not restarted when priority control is performed for group A. 1: Scanning for group B is restarted when priority control is performed for group A.	R/W
b14 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15	GBRP	Group B Single Scan Continuous Start*2	(Enabled only when PGS = 1. Set 0 when PGS = 0.) 0: Single scan for group B is not continuously activated. 1: Single scan for group B is continuously activated.	R/W

Note 1. When the PGS bit is to be set to 1, the ADCSR.ADCS[1:0] bits must be set to 01b (group scan mode). If the bits are set to any other values, proper operation cannot be guaranteed.
Note 2. When the GBRP bit has been set to 1, single scan is performed continuously for group B regardless of the setting of the GBRSCN bit.

PGS Bit (Group A Priority Control Setting)

This bit sets the priority of operation on group A. Set this bit to 1 when giving priority to operation on group A.
When the PGS bit is to be set to 1, the ADCSR.ADCS[1:0] bits must be set to 01b (group scan mode). If the bits are set to any other values, proper operation cannot be guaranteed.
When the PGS bit has been set to 0, A/D conversion must be stopped according to section 30.5.2 Notes on Stopping A/D Conversion. When the PGS bit has been set to 1, make register settings according to section 30.3.4.3 Operation under Group-A Priority Control.

GBRSCN Bit (Group B Restart Setting)

This bit controls the restarting of scan operation on group B when operation on group A is given priority.
If a scan operation on group B has been stopped by a group A trigger input with the GBRSCN bit set to 1, the scan operation is restarted on completion of the A/D conversion on group A. Also, if a group B trigger is input during A/D conversion on group A, the scan operation on group B is restarted on completion of the A/D conversion on group A. However, when this function is to be used, set the ratio between the frequency divisors for PCLKH and ADCLK to 1:1. This function cannot be used if the frequency division settings are not in accord with this.
If the GBRSCN bit has been set to 0, triggers that are input during A/D conversion are ignored. Also, the ADCSR.ADST bit must be 0 when the GBRSCN bit is to be set.
The setting of the GBRSCN bit has an effect when the PGS bit is set to 1.

GBRP Bit (Group B Single Scan Continuous Start)

This bit is set when a single scan operation is to be performed continuously on group B.
Setting the GBRP bit to 1 starts a single scan on group B. On completion of the scan, another single scan on group B is automatically started. If an A/D conversion on group B has been stopped due to an operation on group A that takes priority, single scan on group B is automatically restarted on completion of the A/D conversion on group A.
Disable group B trigger input before setting the GBRP bit to 1. Setting the GBRP bit to 1 invalidates the setting of the GBRSCN bit. The ADCSR.ADST bit must be 0 when the GBRP bit is to be set.
The setting of the GBRP bit is valid when the PGS bit is 1.

30.2.15 A/D Compare Control Register (ADCMPCR)

ADCMPCR is used to set compare function.

Address(es): S12ADC0.ADCMPCR A008 C090h, S12ADC1.ADCMPCR A008 C490h

	b7	b6	b5	b4	b3	b2	b1	b0
	CMPIE	WCMPE	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b5 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b6	WCMPE	Window Function	0: Window function disabled 1: Window function enabled	R/W
b7	CMPIE	Compare Interrupt Enable	0: Generation of an S12CMPI interrupt in response to matches with a condition for comparison is disabled. 1: Generation of an S12CMPI interrupt in response to matches with a condition for comparison is enabled.	R/W

WCMPE Bit (Window Function)

Set enable/disable of Window function.
For details about Window function, see section 30.2.18 A/D Compare Level Register (ADCMPLR).

CMPIE Bit (Compare Interrupt Enable)

This bit selects whether to enable or disable generation of a compare interrupt (S12CMPI) in response to a match.

30.2.16 A/D Compare Channel Select Register (ADCMPANSR)

ADCMPANSR is used to select analog input channels for comparison from among AN000 to AN007 (unit 0) and AN100 to AN107 (unit 1).

Address(es): S12ADC0.ADCMPANSR A008 C094h, S12ADC1.ADCMPANSR A008 C494h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	CMPS[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	CMPS[7:0]	Compare Channel Select	0: The corresponding channel from among AN000 to AN007 (unit 0) and AN100 to AN107 (unit 1) is not a target for comparison. 1: The corresponding channel from among AN000 to AN007 (unit 0) and AN100 to AN107 (unit 1) is a target for comparison.	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

CMPS[7:0] Bits (Compare Channel Select)

Setting the CMPS[n] bit which has the same number as the A/D channel selected by the ADANSA.ANSA[n] (n = 0 to 7 for unit 0, n = 0 to 7 for unit 1) or ADANSB.ANSB[n] (n = 0 to 7 for unit 0, n = 0 to 7 for unit 1) bit to 1 enables comparison with that channel.

Set the CMPS0[7:0] bits while ADCSR.ADST bit is 0.

30.2.17 A/D Compare Channel Select Extended Register (ADCMPANSER)

ADCMPANSER sets whether to use the temperature sensor output for comparison.

Address(es): S12ADC0.ADCMPANSER A008 C092h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	CMPSTS
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CMPSTS	Temperature Sensor Output Compare Select	0: Temperature sensor output is not a target for comparison. 1: Temperature sensor output is a target for comparison.	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

CMPSTS Bit (Temperature Sensor Output Compare Select)

Setting the CMPSTS bit to 1 while ADEXICR.TSSA or ADEXICR.TSSB = 1 enables comparison.

30.2.18 A/D Compare Level Register (ADCMPLR)

The ADCMPLR register sets the condition for use in comparing the values of the ADCMPDR0 and ADCMPDR1 registers with results of A/D conversion.
Set the ADCMPLR register while ADCSR.ADST is 0.

Address(es): S12ADC0.ADCMPLR A008 C098h, S12ADC1.ADCMPLR A008 C498h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	CMPL[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	CMPL[7:0]	Compare Level Select	Set the condition for comparison with the selected channels from among AN000 to AN007 (unit 0) and AN100 to AN107 (unit 1). When Window function is disabled (ADCMPCR.WCMPE bit = 0): 0: ADCMPDR0 register value > A/D-converted value 1: ADCMPDR0 register value < A/D-converted value When Window function is enabled (ADCMPCR.WCMPE bit = 1): 0: AD-converted value < ADCMPDR0 register value or A/D-converted value > ADCMPDR1 register value 1: ADCMPDR0 register value < A/D-converted value < ADCMPDR1 register value	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

CMPL[7:0] Bits (Compare Level Select)

The CMPL0[15:0] bits set the condition for use in comparison with the selected channel from among AN000 to AN007 (unit 0) and AN100 to AN107 (unit 1). A condition can be set for individual comparison of each analog input.
The CMPL[0] bit is used for AN000 (unit 0) and AN100 (unit 1), and the CMPL[7] bit is used for AN007 (unit 0) and AN107 (unit 1).
When the result of comparison matches the set condition, ADCMPSR.CMPFn is set to 1 and a compare interrupt (S12CMPI) is generated.

30.2.19 A/D Compare Level Extended Register (ADCMPLER)

The ADCMPLER register sets the condition for use in comparing the values of ADCMPDR0 and ADCMPDR1 registers with results of A/D conversion.

Set the ADCMPLER register while ADCSR.ADST is 0.

Address(es): S12ADC0.ADCMPLER A008 C093h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	CMPLTS
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CMPLTS	Temperature Sensor Output Compare Select	When Window function is disabled (ADCMPCR.WCMPE bit = 0): 0: ADCMPDR0 register value > A/D-converted value 1: ADCMPDR0 register value < A/D-converted value When Window function is enabled (ADCMPCR.WCMPE bit = 1): 0: AD-converted value < ADCMPDR0 register value or A/D-converted value > ADCMPDR1 register value 1: ADCMPDR0 register value < A/D-converted value < ADCMPDR1 register value	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

CMPLTS Bit (Temperature Sensor Output Compare Level Select)

This bit sets the condition for use in comparison with temperature sensor output.

When the result of comparison matches the set condition, ADCMPSER.CMPFTS is set to 1 and a compare interrupt (S12CMPI) is generated.

30.2.20 A/D Compare Data Register y (ADCMPDRy) (y = 0, 1)

Set the reference data for comparison with the selected channels.
This register is accessible even during A/D conversion which allows dynamic changing of the reference data. The ADCMPDR1 register is not used when the Window function is disabled.
The ADCMPDRy register use different formats depending on the following conditions.

- The value of A/D data register format select bit (flush-right or flush-left)
- The value of A/D-conversion accuracy specification bit (12 bits, 10 bits, 8 bits)
- The value of A/D-converted value addition/average mode select register (A/D-converted value addition mode selected and not selected)

(1) When A/D-Converted Value Addition Mode is Not Selected

- The settings for flush-right data with 12-bit accuracy

Address(es): S12ADC0.ADCMPDR0 A008 C09Ch, S12ADC0.ADCMPDR1 A008 C09Eh, S12ADC1.ADCMPDR0 A008 C49Ch, S12ADC1.ADCMPDR1 A008 C49Eh

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	CMPD[11:0]											
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b11 to b0	CMPD[11:0]	—	12-bit reference value	R/W
b15 to b12	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

- The settings for flush-right data with 10-bit accuracy

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	CMPD[9:0]									
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

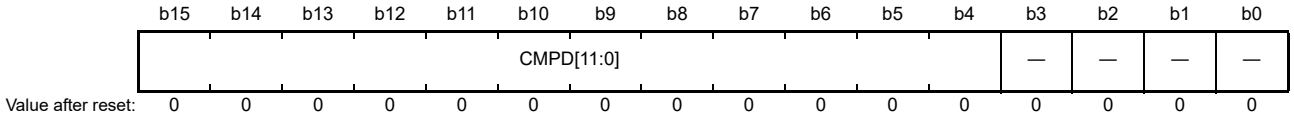
Bit	Symbol	Bit Name	Description	R/W
b9 to b0	CMPD[9:0]	—	10-bit reference value	R/W
b15 to b10	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

- The settings for flush-right data with 8-bit accuracy

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	CMPD[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

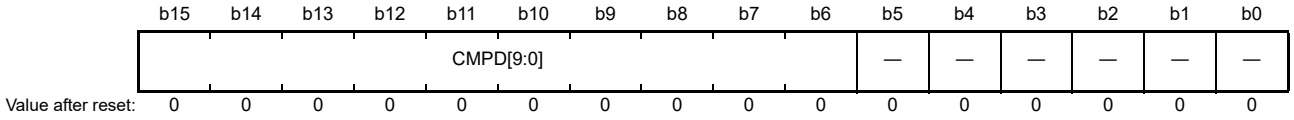
Bit	Symbol	Bit Name	Description	R/W
b7 to b0	CMPD[7:0]	—	8-bit reference value	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

- The settings for flush-left data with 12-bit accuracy



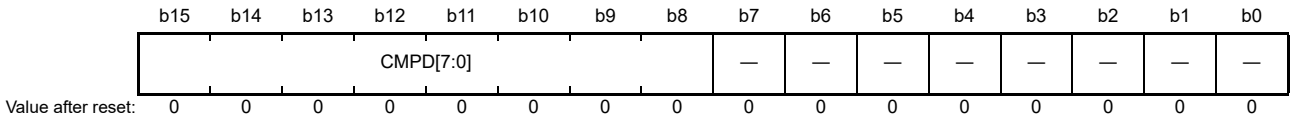
Bit	Symbol	Bit Name	Description	R/W
b3 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15 to b4	CMPD[11:0]	—	12-bit reference value	R/W

- The settings for flush-left data with 10-bit accuracy



Bit	Symbol	Bit Name	Description	R/W
b5 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15 to b6	CMPD[9:0]	—	10-bit reference value	R/W

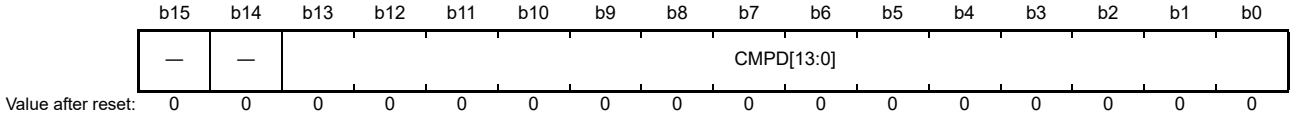
- The settings for flush-left data with 8-bit accuracy



Bit	Symbol	Bit Name	Description	R/W
b7 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15 to b8	CMPD[7:0]	—	8-bit reference value	R/W

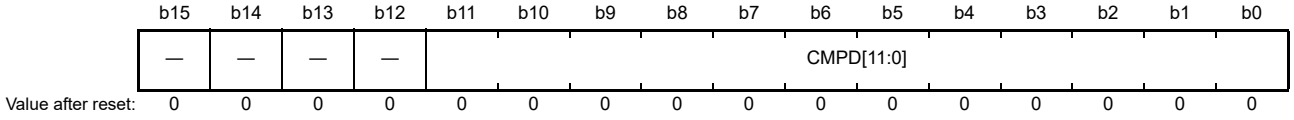
(2) When A/D-Converted Value Addition Mode is Selected

- The settings for flush-right data with 12-bit accuracy (when A/D-converted value addition mode is selected)



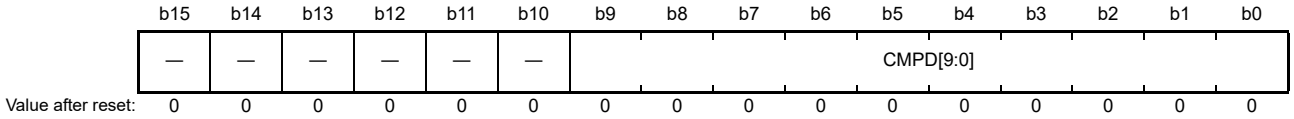
Bit	Symbol	Bit Name	Description	R/W
b13 to b0	CMPD[13:0]	—	14-bit reference value	R/W
b15, b14	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

- The settings for flush-right data with 10-bit accuracy (when A/D-converted value addition mode is selected)



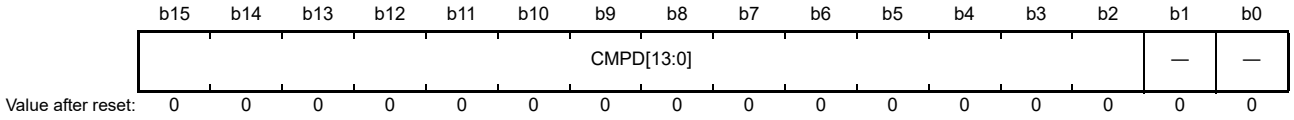
Bit	Symbol	Bit Name	Description	R/W
b11 to b0	CMPD[11:0]	—	12-bit reference value	R/W
b15 to b12	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

- The settings for flush-right data with 8-bit accuracy (when A/D-converted value addition mode is selected)



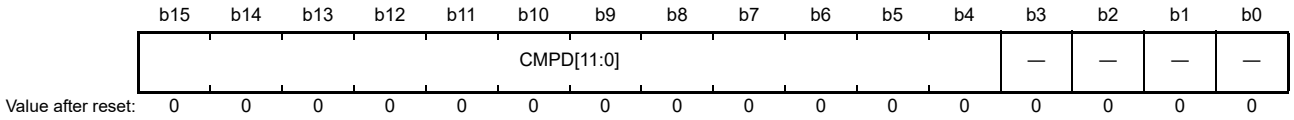
Bit	Symbol	Bit Name	Description	R/W
b9 to b0	CMPD[9:0]	—	10-bit reference value	R/W
b15 to b10	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

- The settings for flush-left data with 12-bit accuracy (when A/D-converted value addition mode is selected)



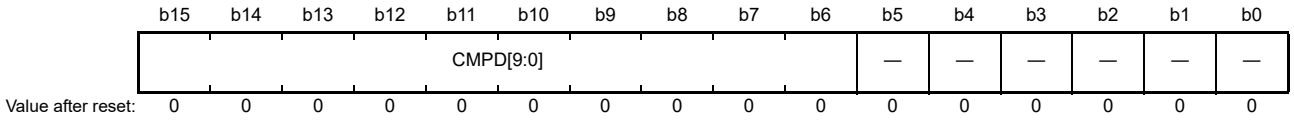
Bit	Symbol	Bit Name	Description	R/W
b1, b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15 to b2	CMPD[13:0]	—	14-bit reference value	R/W

- The settings for flush-left data with 10-bit accuracy (when A/D-converted value addition mode is selected)



Bit	Symbol	Bit Name	Description	R/W
b3 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15 to b4	CMPD[11:0]	—	12-bit reference value	R/W

- The settings for flush-left data with 8-bit accuracy (when A/D-converted value addition mode is selected)



Bit	Symbol	Bit Name	Description	R/W
b5 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15 to b6	CMPD[9:0]	—	10-bit reference value	R/W

30.2.21 A/D Compare Status Register (ADCMPSR)

The ADCMPSR register stores the compare results of compare function.

Address(es): S12ADC0.ADCMPSR A008 C0A0h, S12ADC1.ADCMPSR A008 C4A0h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	CMPF[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	CMPF[7:0]	Compare Flag	Indicates compare results of AN000 to AN007 (unit 0) and AN100 to AN107 (unit 1). 0: Condition for comparison was not met. 1: Condition for comparison was met.	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

CMPF[7:0] Bits (Compare Flag)

These bits are status flags to indicate the results of comparison with the selected analog inputs from among AN000 to AN007 (unit 0) and AN100 to AN107 (unit 1). When the result of comparison on completion of A/D conversion matches the condition set in ADCMPLR.CMPLn, the corresponding flags are set to 1.

When the ADCMPCR.CMPIE bit is 1, a compare interrupt (S12CMPI) request is generated when the setting of the flag becomes 1.

The CMPF[0] bit is used for AN000 (unit 0)/AN100 (unit 1), the CMPF[7] bit is used for AN007 (unit 0)/AN107 (unit 1).

The value 1 cannot be written to the CMPFn bit.

[Setting condition]

- The condition set in ADCMPLR.CMPLn is met.

[Clearing condition]

- 0 is written after reading 1.

30.2.22 A/D Compare Status Extended Register (ADCMPSER)

The ADCMPSER register is a status register that indicates the compare results of temperature sensor output.

Address(es): S12ADC0.ADCMPSER A008 C0A4h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	CMPFTS
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CMPFTS	Temperature Sensor Output Compare Flag	0: Condition for comparison was not met. 1: Condition for comparison was met.	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

CMPFTS Bit (Temperature Sensor Output Compare Flag)

This bit is a status flag to indicate the result of comparison with temperature sensor output. When the result of comparison on completion of A/D conversion matches the condition set in ADCMPLER.CMPLTS, this flag is set to 1. When the ADCMPCR.CMPIE bit is 1, a compare interrupt (S12CMPI) request is generated when the setting of the flag becomes 1.

The value 1 cannot be written to the CMPFTS bit.

[Setting condition]

- The condition set in ADCMPLER.CMPLTS is met.

[Clearing condition]

- 0 is written after reading 1.

30.2.23 A/D Pin-Level Self-Diagnosis Control Register (ADTDCR)

ADTDCR controls pin-level self-diagnosis function. For details about this function, see section 30.3.12 Pin-Level Self-Diagnosis Function.

Address(es): S12ADC0.ADTDCR A008 C0C8h, S12ADC1.ADTDCR A008 C4C8h

	b7	b6	b5	b4	b3	b2	b1	b0
	TDE	—	—	—	—	—	TDLV[1:0]	
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	TDLV[1:0]	Pin-level Self-diagnosis Level Select	b1 b0 0 0: Input channels with even numbers are discharged to AVSS, and input channels with odd numbers are charged to AVCC. 0 1: Input channels with even numbers are charged to AVCC, and input channels with odd numbers are discharged to AVSS. 1 0: Input channels with even numbers are discharged to AVSS, and input channels with odd numbers are charged to AVCC × 1/2. 1 1: Input channels with even numbers are charged to AVCC × 1/2, and input channels with odd numbers are discharged to AVSS.	R/W
b6 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b7	TDE	Pin-level Self-diagnosis Enable	0: Disables pin-level self-diagnosis. 1: Enables pin-level self-diagnosis.	R/W

TDLV[1:0] Bits (Pin-level Self-diagnosis Level Select)

These bits select the pin level for pin-level self-diagnosis.
Set the TDLV[1:0] bits while ADCSR.ADST bit is 0.

TDE Bit (Pin-level Self-diagnosis Enable)

This bit selects whether to perform pin-level self-diagnosis.
Set the TDE bit while ADCSR.ADST bit is 0.

30.2.24 A/D Error Control Register (ADERCR)

ADERCR controls error detection function. For details about errors, see section 30.3.13 Error Detection Function.

Address(es): S12ADC0.ADERCR A008 C0CAh, S12ADC1.ADERCR A008 C4CAh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	OWEIE	—	—
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b2	OWEIE	Overwrite Error Interrupt Enable	0: Disables interrupt generation when an overwrite error is detected. 1: Enables interrupt generation when an overwrite error is detected.	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

OWEIE Bit (Overwrite Error Interrupt Enable)

This bit sets whether to enable or disable generation of an error interrupt request (S12ADE) when an overwrite error is detected.

Set the OWEIE bit while ADCSR.ADST bit is 0.

30.2.25 A/D Error Clear Register (ADERCLR)

The A/D error clear register is a write-only register for clearing errors.

Address(es): S12ADC0.ADERCLR A008 C0CBh, S12ADC1.ADERCLR A008 C4CBh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	OWEC	—	—
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	—	Reserved	The write value should be 0.	W
b2	OWEC	Overwrite Error Clear	Writing 0: Disables clearing overwrite errors. Writing 1: Enables clearing overwrite errors.	W
b7 to b3	—	Reserved	The write value should be 0.	W

30.2.26 A/D Overwrite Error Register (ADOWER)

The ADOWER register is a status register that indicates that the results of A/D conversion in the ADDRy register were not read and an overwrite occurred.

Address(es): S12ADC0.ADOWER A008 C0D2h, S12ADC1.ADOWER A008 C4D2h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	OWE[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	OWE[7:0]	Overwrite Error	0: No overwrite error occurred in S12ADC0.ADDR0 to S12ADC0.ADDR7 (unit 0) and S12ADC1.ADDR0 to S12ADC1.ADDR7 (unit 1). 1: An overwrite error occurred in S12ADC0.ADDR0 to S12ADC0.ADDR7 (unit 0) and S12ADC1.ADDR0 to S12ADC1.ADDR7 (unit 1).	R
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R

OWE[7:0] Bits (Overwrite Error 0)

The OWE[7:0] bits are set to 1 when the results of the next A/D conversion are stored in the ADDRy register after the A/D conversion finishes without the results of A/D conversion in the ADDRy register being read.

The OWE[7:0] bits are cleared to 0 when 1 is written to the ADERCLR.OWEC register. The OWE[7:0] bits cannot be written to 1.

The OWE[0] bit corresponds to S12ADC0.ADDR0 and the OWE[7] bit corresponds to S12ADC0.ADDR7 for unit 0.

The OWE[0] bit corresponds to S12ADC1.ADDR0 and the OWE[7] bit corresponds to S12ADC1.ADDR7 for unit 1.

30.2.27 A/D Overwrite Error Extended Register (ADOWEER)

The ADOWEER register is a status register that indicates that the results of A/D conversion in the ADDBLDR, ADRD, ADTSDR registers were not read and an overwrite occurred.

Address(es): S12ADC0.ADOWEER A008 C0D6h, S12ADC1.ADOWEER A008 C4D6h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	TSOWE	DIAGOWE	DOWE	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	—	Reserved	These bits are read as 0.	R
b2	DOWE	A/D Data Duplication Register Overwrite Error	0: No overwrite error occurred in ADDBLDR. 1: An overwrite error occurred in ADDBLDR.	R
b3	DIAGOWE	A/D Self-diagnosis Data Register Overwrite Error	0: No overwrite error occurred in ADRD. 1: An overwrite error occurred in ADRD.	R
b4	TSOWE	A/D Temperature Sensor Data Register Overwrite Error	0: No overwrite error occurred in ADTSDR. 1: An overwrite error occurred in ADTSDR.	R
b15 to b5	—	Reserved	These bits are read as 0.	R

DOWE Bit (A/D Data Duplication Register Overwrite Error)

DOWE bit is set to 1 when the results of the next A/D conversion are stored in the ADDBLDR register after the A/D conversion finishes without the results of A/D conversion in the ADDBLDR register being read.
DOWE bit is cleared to 0 when 1 is written to the ADERCLR.OWEC register. DOWE bit cannot be written to 1.

DIAGOWE Bit (A/D Self-diagnosis Data Register Overwrite Error)

DIAGOWE bit is set to 1 when the results of the next A/D conversion are stored in the ADRD register after the A/D conversion finishes without the results of A/D conversion in the ADRD register being read.
DIAGOWE bit is cleared to 0 when 1 is written to the ADERCLR.OWEC register. DIAGOWE bit cannot be written to 1.

TSOWE Bit (A/D Temperature Sensor Data Register Overwrite Error)

TSOWE bit is set to 1 when the results of the next A/D conversion are stored in the ADTSDR register after the A/D conversion finishes without the results of A/D conversion in the ADTSDR register being read.
TSOWE bit is cleared to 0 when 1 is written to the ADERCLR.OWEC register. TSOWE bit cannot be written to 1.

30.3 Operation

30.3.1 Scanning Operation

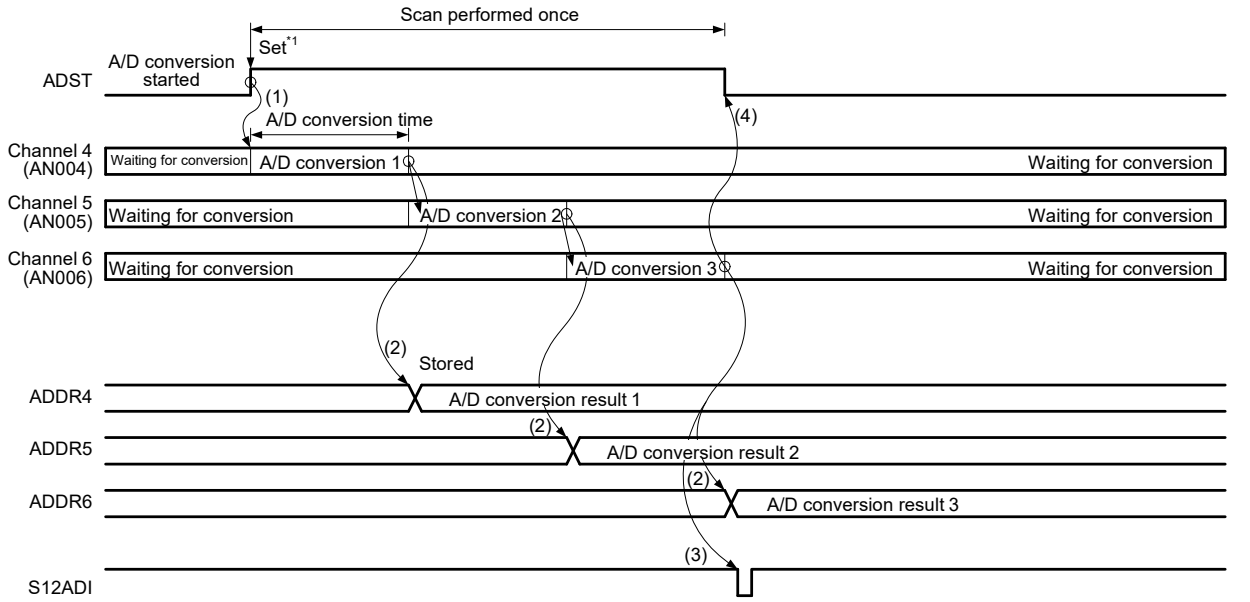
In scanning, A/D conversion is performed sequentially on the analog inputs of the specified channels.
There are three operating modes: single scan mode, continuous scan mode, and group scan mode. In single scan mode, one or more specified channels are scanned once. In continuous scan mode, one or more specified channels are scanned repeatedly until the ADST bit in ADCSR is cleared to 0 from 1. In group scan mode, the selected channels of group A and the selected channels of group B are scanned once after starting to be scanned according to the respective synchronous triggers (TPUa, ELC).
In single scan mode and continuous scan mode, A/D conversion is performed for ANn pins selected by the ADANSA register, starting from the pin with the smallest number n. In group scan mode, A/D conversion is performed for ANn pins selected by the ADANSA register for group A, and performed for ANn pins selected by the ADANSB register for group B, respectively, starting from the pin with the smallest number n.
When self-diagnosis is selected, it is executed once at the beginning of each scan and one of the three voltages internally generated in the 12-bit A/D converter is converted.
The temperature sensor output can be scanned at the same time as the analog input of channels, and A/D conversion of the analog input of channels and the temperature sensor output is performed in that order.
Double trigger mode is to be used with single scan mode or group scan mode. With double trigger mode being enabled, A/D conversion data of a channel selected by the DBLANS[4:0] bits in ADCSR is duplicated only if the conversion is started by any of the synchronous triggers (TPUa, ELC) selected by the TRSA[5:0] bits in ADSTRGR.
When any of AN000 to AN003 channels is set for a channel-dedicated sample-and-hold circuit by the SHANS[3:0] bits in ADSHCR, the target analog input specified is sampled and held before the first A/D conversion of each scan.

30.3.2 Single Scan Mode

30.3.2.1 Basic Operation (without Channel-Dedicated Sample-and-Hold Circuits)

In basic operation of single scan mode, A/D conversion is performed once on the analog input of the specified channels as below.

(1) When the ADST bit in ADCSR is set to 1 (A/D conversion start) by a software trigger, a synchronous trigger input (TPUa, ELC), or an asynchronous trigger input, A/D conversion is performed for ANn pins selected by the ADANSA register, starting from the pin with the smallest number n.
(2) Each time A/D conversion of a single channel is completed, the A/D conversion result is stored into the corresponding A/D data register (ADDRy).
(3) When A/D conversion of all the selected channels is completed, an S12ADI interrupt is generated if the ADIE bit in ADCSR is 1 (S12ADI interrupt upon scanning completion is enabled).
(4) The ADST bit remains 1 (A/D conversion start) during A/D conversion, and is automatically cleared to 0 when A/D conversion of all the selected channels is completed. Then the 12-bit A/D converter enters a waiting state.



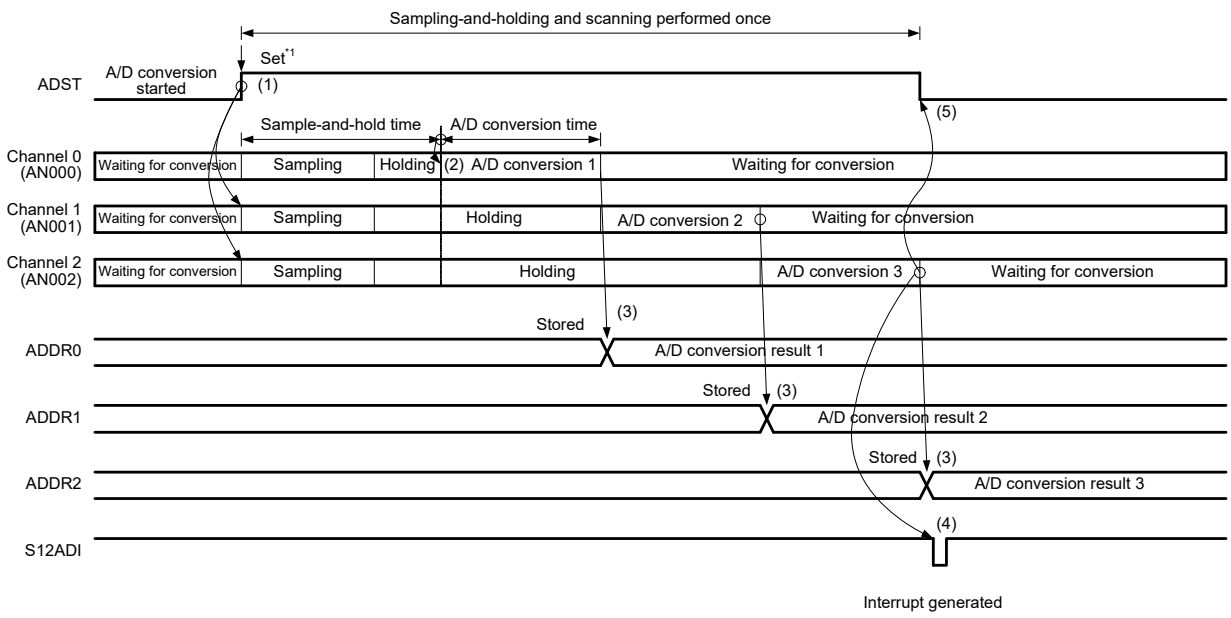
Note 1. The arrows in this figure indicate instruction execution by software.

Figure 30.4 Example of Operation in Single Scan Mode (Basic Operation: AN004 to AN006 Selected)

30.3.2.2 Basic Operation (with Channel-Dedicated Sample-and-Hold Circuits)

When the channel-dedicated sample-and-hold circuit is used, sample-and-hold operation is first performed, and then A/D conversion is performed once on the analog input of all the selected channels as below. The channels whose channel-dedicated sample-and-hold circuit is to be used can be selected by the SHANS[3:0] bits in ADSHCR.

(1) Analog input sampling of all the channels whose channel-dedicated sample-and-hold circuit is to be used is started when the ADST bit in ADCSR is set to 1 (A/D conversion start) by a software trigger, a synchronous trigger input (TPUa, ELC), or an asynchronous trigger input.
(2) After sample-and-hold operation, A/D conversion is performed for ANn pins selected by the ADANSA register, starting from the channel with the smallest number n.
(3) Each time A/D conversion of a single channel is completed, the A/D conversion result is stored into the corresponding A/D data register (ADDRy).
(4) When A/D conversion of all the selected channels is completed, an S12ADI interrupt is generated if the ADIE bit in ADCSR is 1 (S12ADI interrupt upon scanning completion is enabled).
(5) The ADST bit in ADCSR remains 1 (A/D conversion start) during A/D conversion, and is automatically cleared to 0 when A/D conversion of all the selected channels is completed. Then the 12-bit A/D converter enters a waiting state.



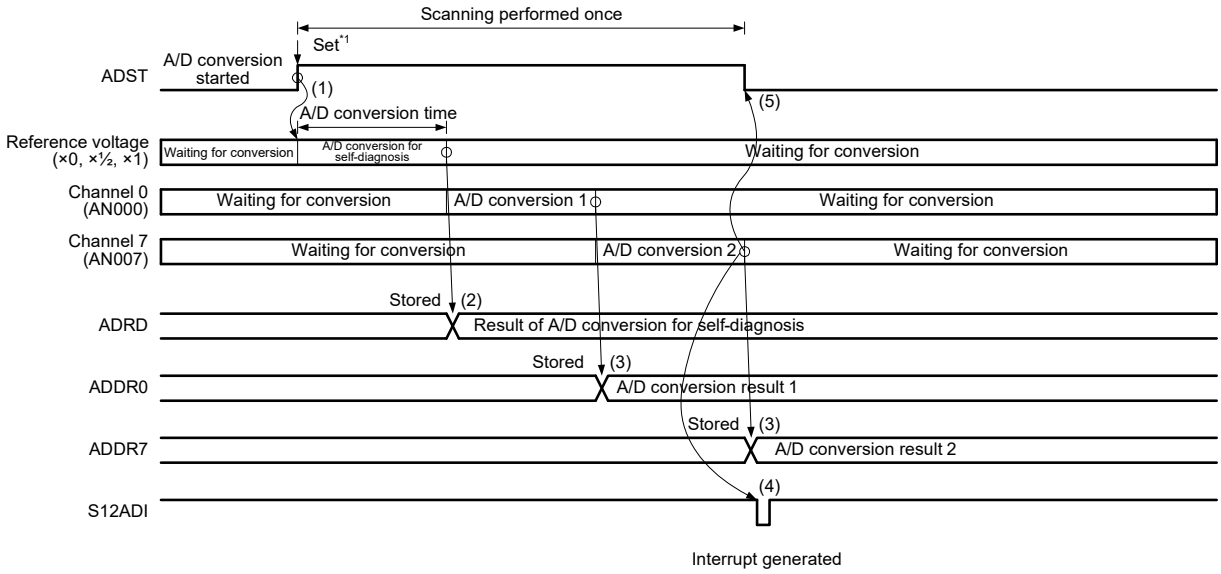
Note 1. The arrows in this figure indicate instruction execution by software.

Figure 30.5 Example of Operation in Single Scan Mode (Channel-Dedicated Sample-and-Hold Circuits Used: AN000 to AN002 Selected)

30.3.2.3 Channel Selection and Self-Diagnosis (without Channel-Dedicated Sample-and-Hold Circuits)

When channels and self-diagnosis are selected, A/D conversion is first performed for the reference voltage VREFH0 (unit 0) or VREFH1 (unit 1) (reference voltage ×0, ×1/2, or ×1) supplied to the A/D converter, and then A/D conversion is performed once on the analog input of the selected channels as below.

(1) A/D conversion for self-diagnosis is first started when the ADST bit in ADCSR is set to 1 (A/D conversion start) by a software trigger, a synchronous trigger input (TPUa, ELC), or an asynchronous trigger input.
(2) When A/D conversion for self-diagnosis is completed, the A/D conversion result is stored into the A/D self-diagnosis data register (ADRD). A/D conversion is then performed for ANn pins selected by the ADANSA register, starting from the channel with the smallest number n.
(3) Each time A/D conversion of a single channel is completed, the A/D conversion result is stored into the corresponding A/D data register (ADDRy).
(4) When A/D conversion of all the selected channels is completed, an S12ADI interrupt is generated if the ADIE bit in ADCSR is 1 (S12ADI interrupt upon scanning completion enabled).
(5) The ADST bit remains 1 (A/D conversion start) during A/D conversion, and is automatically cleared to 0 when A/D conversion of all the selected channels is completed. Then the 12-bit A/D converter enters a waiting state.



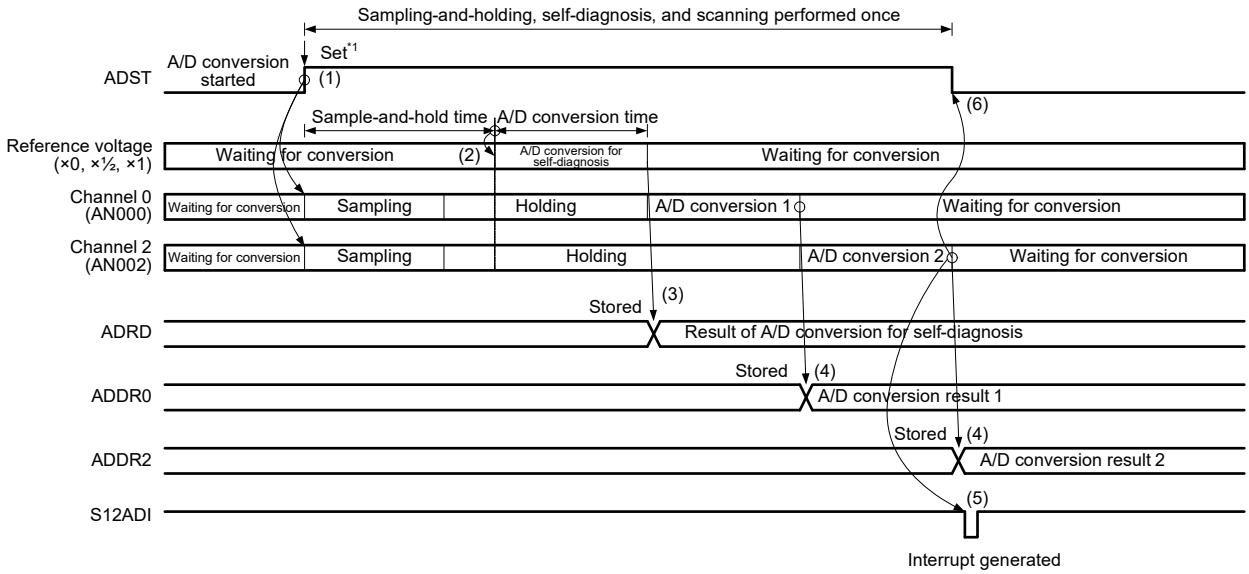
Note 1. The arrows in this figure indicate instruction execution by software.

Figure 30.6 Example of Operation in Single Scan Mode (Basic Operation: AN000 and AN007 Selected + Self-Diagnosis)

30.3.2.4 Channel Selection and Self-Diagnosis (with Channel-Dedicated Sample-and-Hold Circuits)

When the channel-dedicated sample-and-hold circuit is used and channels and self-diagnosis are selected, sample-and-hold operation is first performed, and then A/D conversion is performed once for the reference voltage VREFH0 (unit 0) or VREFH1 (unit 1) (reference voltage ×0, ×1/2, or ×1) supplied to the 12-bit A/D converter as below. After that, A/D conversion is performed only once on the analog input of the selected channels. The ADSHCR.SHANS[3:0] bits are used to select the channels for which the channel-dedicated sample-and-hold circuit is used.

(1) Analog input sampling of all the channels whose channel-dedicated sample-and-hold circuit is to be used is started when the ADST bit in ADCSR is set to 1 (A/D conversion start) by a software trigger, a synchronous trigger input (TPUa, ELC), or an asynchronous trigger input.
(2) After sample-and-hold operation, A/D conversion is started by self-diagnosis.
(3) When A/D conversion due to self-diagnosis is completed, the A/D conversion result is stored into the A/D self-diagnosis data register (ADRD). A/D conversion is then performed for ANn pins selected by the ADANSA register, starting from the channel with the smallest number n.
(4) Each time A/D conversion of a single channel is completed, the A/D conversion result is stored into the corresponding A/D data register (ADDRy).
(5) When A/D conversion of all the selected channels is completed, an S12ADI interrupt is generated if the ADIE bit in ADCSR is 1 (S12ADI interrupt upon scanning completion enabled).
(6) The ADST bit remains 1 (A/D conversion start) during A/D conversion, and is automatically cleared to 0 when A/D conversion of all the selected channels is completed. Then the 12-bit A/D converter enters a waiting state.



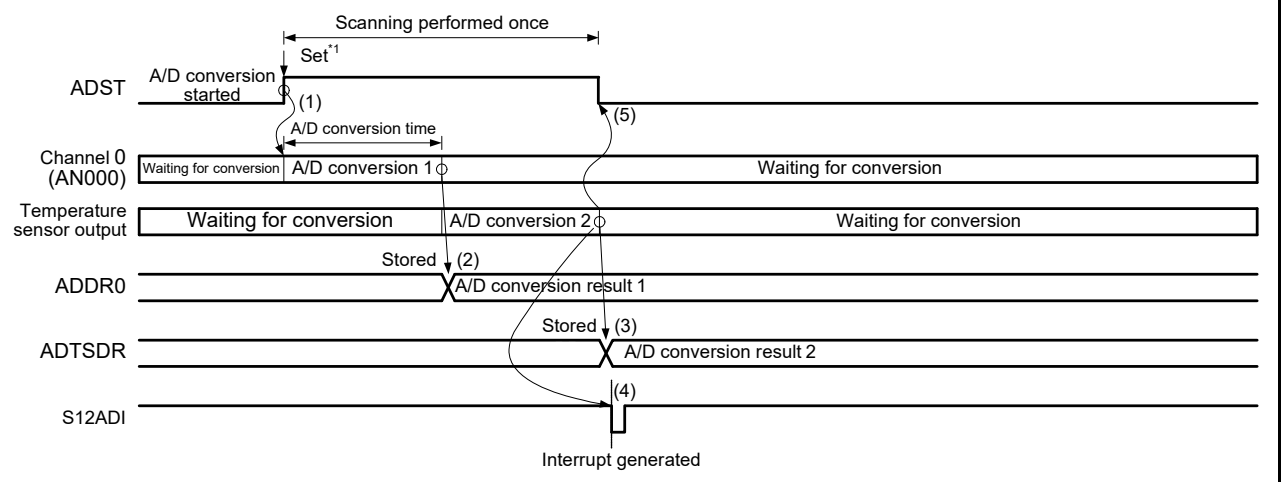
Note 1. The arrows in this figure indicate instruction execution by software.

Figure 30.7 Example of Operation in Single Scan Mode (Channel-Dedicated Sample-and-Hold Circuits Used; AN000 and AN002 Selected + Self-Diagnosis)

30.3.2.5 A/D Conversion of Temperature Sensor Output

When the channels and temperature sensor output are selected at the same time, A/D conversion is first performed on the analog input of the selected channels, and then A/D conversion is performed once on the temperature sensor output as below.
With the channels deselected, selecting only the temperature sensor output is also possible.

(1) When software, synchronous trigger (TPUa, ELC), or asynchronous trigger sets the ADCSR.ADST bit to 1 (starting A/D conversion), conversion for the ANn pins selected in the ADANSA register starts in order from the channel with the lowest number n.
(2) On completion of A/D conversion on the channel, the result is stored in the corresponding A/D data register (ADDRy), and then A/D conversion of temperature sensor output starts.
(3) On completion of A/D conversion of temperature sensor output, the result is stored in the corresponding A/D temperature sensor data register (ADTRDR).
(4) If the ADCSR.ADIE bit is set to 1 (enabling S12ADI0 interrupt generation upon scan conversion completion), an S12ADI interrupt is generated.
(5) The ADCSR.ADST bit remains 1 (starting A/D conversion) during A/D conversion, and is automatically cleared to 0 upon completion of A/D conversion. Then the 12-bit A/D converter enters a waiting state.



Note 1. The arrows in this figure indicate instruction execution by software.

Figure 30.8 Example of Operation in Single Scan Mode (Basic Operation: AN000 and Temperature Sensor Output Selected)

30.3.2.6 A/D Conversion in Double Trigger Mode

When double trigger mode is selected in single scan mode, A/D conversion is performed for two rounds of single scan operation started by a synchronous trigger (TPUa, ELC) as a sequence as shown below.
The temperature sensor output A/D conversion select bits (ADEXICR.TSSA and ADEXICR.TSSB) should be set to 0.
Duplication of A/D conversion data is enabled by setting the channel numbers to be duplicated to the DBLANS[4:0] bits in ADCSR and setting the DBLE bit in ADCSR to 1. When the DBLE bit in ADCSR is set to 1, channel selection using the ADANSA register is invalid. In double trigger mode, a synchronous trigger (TPUa, ELC) should be selected using the TRSA[5:0] bits in ADSTRGR; the EXTRG bit and TRGE bit in ADCSR should be set to 0 and 1, respectively. Software trigger should not be used.

(1) When the ADST bit in ADCSR is set to 1 (A/D conversion start) by a synchronous trigger input (TPUa, ELC), A/D conversion is started on the single channel selected by the DBLANS[4:0] bits in ADCSR.
(2) Each time A/D conversion of a single channel is completed, the A/D conversion result is stored into the corresponding A/D data register (ADDRy).
(3) The ADST bit in ADCSR is automatically cleared to 0 and the 12-bit A/D converter enters a waiting state. Here, an S12ADI interrupt request is not generated irrespective of the ADIE (S12ADI interrupt upon scanning completion enabled) bit setting in ADCSR.
(4) When the ADST bit in ADCSR is set to 1 (A/D conversion start) by the second synchronous trigger input, A/D conversion is started on the single channel selected by the DBLANS[4:0] bits in ADCSR.
(5) When A/D conversion is completed, the A/D conversion result is stored into the A/D data duplication register (ADDBLDR), which is exclusively used in double trigger mode.
(6) If the ADIE bit in ADCSR is 1 (S12ADI interrupt upon scanning completion enabled), an S12ADI interrupt is generated.
(7) The ADST bit remains 1 (A/D conversion start) during A/D conversion, and is automatically cleared to 0 when A/D conversion is completed. Then the 12-bit A/D converter enters a waiting state.

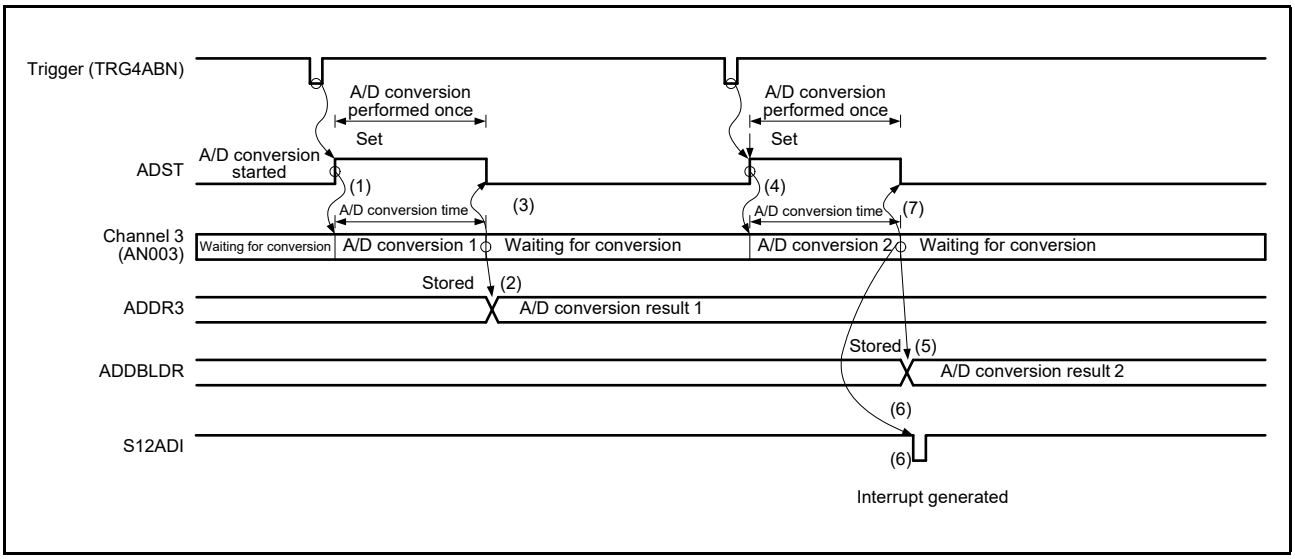


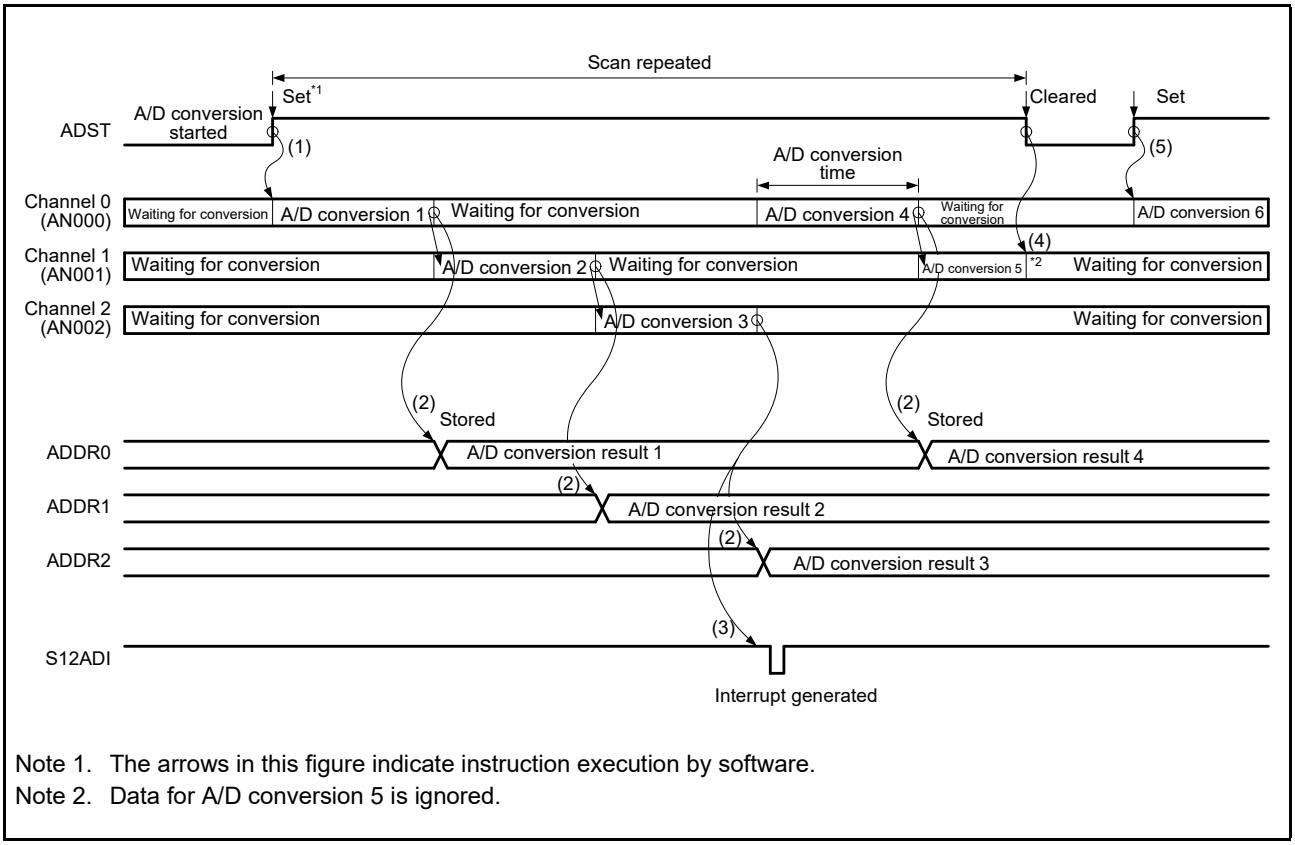
Figure 30.9 Example of Operation in Single Scan Mode (Double Trigger Mode Selected: AN003 Duplicated; TRG4ABN Selected as Trigger; Self-Diagnosis Deselected)

30.3.3 Continuous Scan Mode

30.3.3.1 Basic Operation (without Channel-Dedicated Sample-and-Hold Circuits)

In basic operation of continuous scan mode, A/D conversion is performed repeatedly on the analog input of the channels selected by the ADANSA register as below.

(1) When the ADST bit in ADCSR is set to 1 (A/D conversion start) by a software trigger, a synchronous trigger input (TPUa, ELC), or an asynchronous trigger input, A/D conversion is performed for ANn pins selected by the ADANSA register, starting from the channel with the smallest number n.
(2) Each time A/D conversion of a single channel is completed, the A/D conversion result is stored into the corresponding A/D data register (ADDRy).
(3) When A/D conversion of all the selected channels is completed, an S12ADI interrupt is generated if the ADCSR.ADIE bit is 1 (S12ADI interrupt upon scanning completion enabled).
The 12-bit A/D converter sequentially starts A/D conversion for ANn pins selected by the ADANSA register, starting from the channel with the smallest number n.
(4) The ADST bit in ADCSR is not automatically cleared and steps 2 and 3 are repeated as long as the bit remains 1 (A/ D conversion start). When the ADCSR.ADST bit is set to 0 (A/D conversion stop), A/D conversion stops and the 12-bit A/D converter enters a waiting state.
(5) When the ADST bit is later set to 1 (A/D conversion start), A/D conversion is started again for ANn pins selected by the ADANSA register, starting from the channel with the smallest number n.



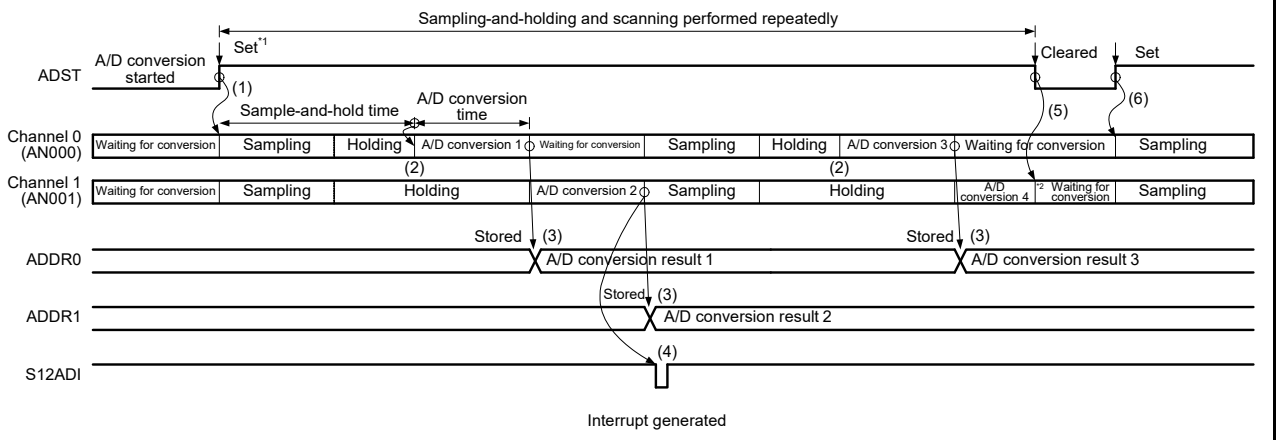
Note 1. The arrows in this figure indicate instruction execution by software.
Note 2. Data for A/D conversion 5 is ignored.

Figure 30.10 Example of Operation in Continuous Scan Mode (Basic Operation: AN000 to AN002 Selected)

30.3.3.2 Basic Operation (with Channel-Dedicated Sample-and-Hold Circuits)

When the channel-dedicated sample-and-hold circuit is used, sample-and-hold operation is first performed, and then A/D conversion is repeated on the analog input of all the selected channels as below. The channels whose channel-dedicated sample-and-hold circuit is to be used can be selected by the SHANS[3:0] bits in ADSHCR.

(1) Analog input sampling of all the channels whose channel-dedicated sample-and-hold circuit is to be used is started when the ADST bit in ADCSR is set to 1 (A/D conversion start) by a software trigger, a synchronous trigger input (TPUa, ELC), or an asynchronous trigger input.
(2) After sample-and-hold operation, A/D conversion is performed for ANn pins selected by the ADANSA register, starting from the channel with the smallest number n.
(3) Each time A/D conversion of a single channel is completed, the A/D conversion result is stored into the corresponding A/D data register (ADDRy).
(4) When A/D conversion of all the selected channels is completed, an S12ADI interrupt is generated if the ADCSR.ADIE bit is 1 (S12ADI interrupt upon scanning completion is enabled). At the same time, analog input sampling is started for all the channels whose channel-dedicated sample-and-hold circuit is to be used.
(5) The ADST bit is not automatically cleared and steps 2 to 4 are repeated as long as the bit remains 1. When the ADST bit is set to 0 (A/D conversion stop), A/D conversion stops and the 12-bit A/D converter enters a waiting state.
(6) When the ADST bit is then set to 1 (A/D conversion start), analog input sampling is started again for all the channels whose channel-dedicated sample-and-hold circuit is to be used.



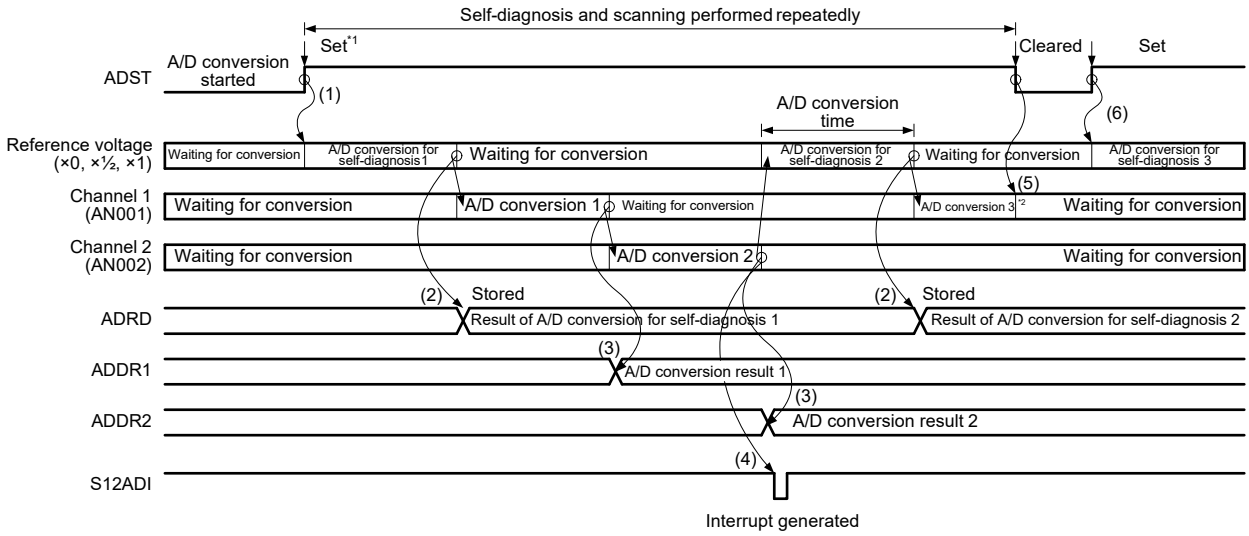
Note 1. The arrows in this figure indicate instruction execution by software.
Note 2. Data for A/D conversion 4 is ignored.

Figure 30.11 Example of Operation in Continuous Scan Mode (Channel-Dedicated Sample-and-Hold Circuits Used; AN000 and AN001 Selected)

30.3.3.3 Channel Selection and Self-Diagnosis (without Channel-Dedicated Sample-and-Hold Circuits)

When channels and self-diagnosis are selected at the same time, A/D conversion is first performed for the reference voltage VREFH0 (unit 0) or VREFH1 (unit 1) (reference voltage ×0, ×1/2, or ×1) supplied to the 12-bit A/D converter, and then A/D conversion is performed on the analog input of the selected channels, which sequence is repeated as below.

(1) A/D conversion due to self-diagnosis is first started when the ADST bit in ADCSR is set to 1 (A/D conversion start) by a software trigger, a synchronous trigger input (TPUa, ELC), or an asynchronous trigger input.
(2) When A/D conversion due to self-diagnosis is completed, the A/D conversion result is stored into the A/D self-diagnosis data register (ADRD). A/D conversion is then performed for ANn pins selected by the ADANSA register, starting from the channel with the smallest number n.
(3) Each time A/D conversion of a single channel is completed, the A/D conversion result is stored into the corresponding A/D data register (ADDRy).
(4) When A/D conversion of all the selected channels is completed, an S12ADI interrupt is generated if the ADCSR.ADIE bit is 1 (S12ADI interrupt upon scanning completion is enabled). At the same time, the 12-bit A/D converter starts A/D conversion due to self-diagnosis and then starts A/D conversion on ANn pins selected by the ADANSA register, starting from the channel with the smallest number n.
(5) The ADST bit in ADCSR is not automatically cleared and steps 2 to 4 are repeated as long as the bit remains 1. When the ADST bit in ADCSR is set to 0 (A/D conversion stop), A/D conversion stops and the 12-bit A/D converter enters a waiting state.
(6) When the ADST bit is later set to 1 (A/D conversion start), conversion is started again from the A/D conversion due to self-diagnosis.



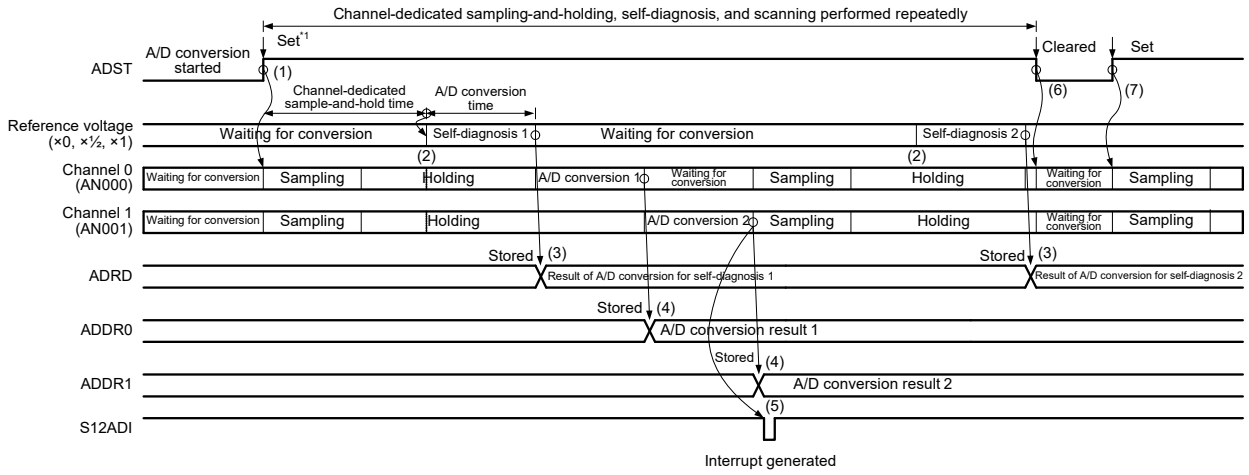
Note 1. The arrows in this figure indicate instruction execution by software.
Note 2. Data for A/D conversion 3 is ignored.

Figure 30.12 Example of Operation in Continuous Scan Mode (Basic Operation; AN001 and AN002 Selected + Self-Diagnosis)

30.3.3.4 Channel Selection and Self-Diagnosis (with Channel-Dedicated Sample-and-Hold Circuits)

When the channel-dedicated sample-and-hold circuit is used and channels and self-diagnosis are selected, sample-and-hold operation is first performed, and then A/D conversion is performed for the reference voltage VREFH0 (unit 0) or VREFH1 (unit 1) (reference voltage ×0, ×1/2, or ×1) supplied to the 12-bit A/D converter, and A/D conversion is performed on the analog input of the selected channels, which sequence is repeated as below. The ADSHCR.SHANS[3:0] bits are used to select the channels for which the channel-dedicated sample-and-hold circuit is used.

(1) Analog input sampling of all the channels whose channel-dedicated sample-and-hold circuit is to be used is started when the ADST bit in ADCSR is set to 1 (A/D conversion start) by a software trigger, a synchronous trigger input (TPUa, ELC), or an asynchronous trigger input.
(2) After sample-and-hold operation, A/D conversion is started by self-diagnosis.
(3) When A/D conversion due to self-diagnosis is completed, the A/D conversion result is stored into the A/D self-diagnosis data register (ADRD). A/D conversion is then performed for ANn pins selected by the ADANSA register, starting from the channel with the smallest number n.
(4) Each time A/D conversion of a single channel is completed, the A/D conversion result is stored into the corresponding A/D data register (ADDRy).
(5) When A/D conversion of all the selected channels is completed, an S12ADI interrupt is generated if the ADCSR.ADIE bit is 1 (S12ADI interrupt upon scanning completion enabled). At the same time, analog input sampling is started for all the channels whose channel-dedicated sample-and-hold circuit is to be used.
(6) The ADST bit is not automatically cleared and steps 2 to 5 are repeated as long as the bit remains 1. When the ADST bit is set to 0 (A/D conversion stop), A/D conversion stops and the 12-bit A/D converter enters a waiting state.
(7) When the ADST bit is then set to 1 (A/D conversion start), analog input sampling is started again for all the channels whose channel-dedicated sample-and-hold circuit is to be used.



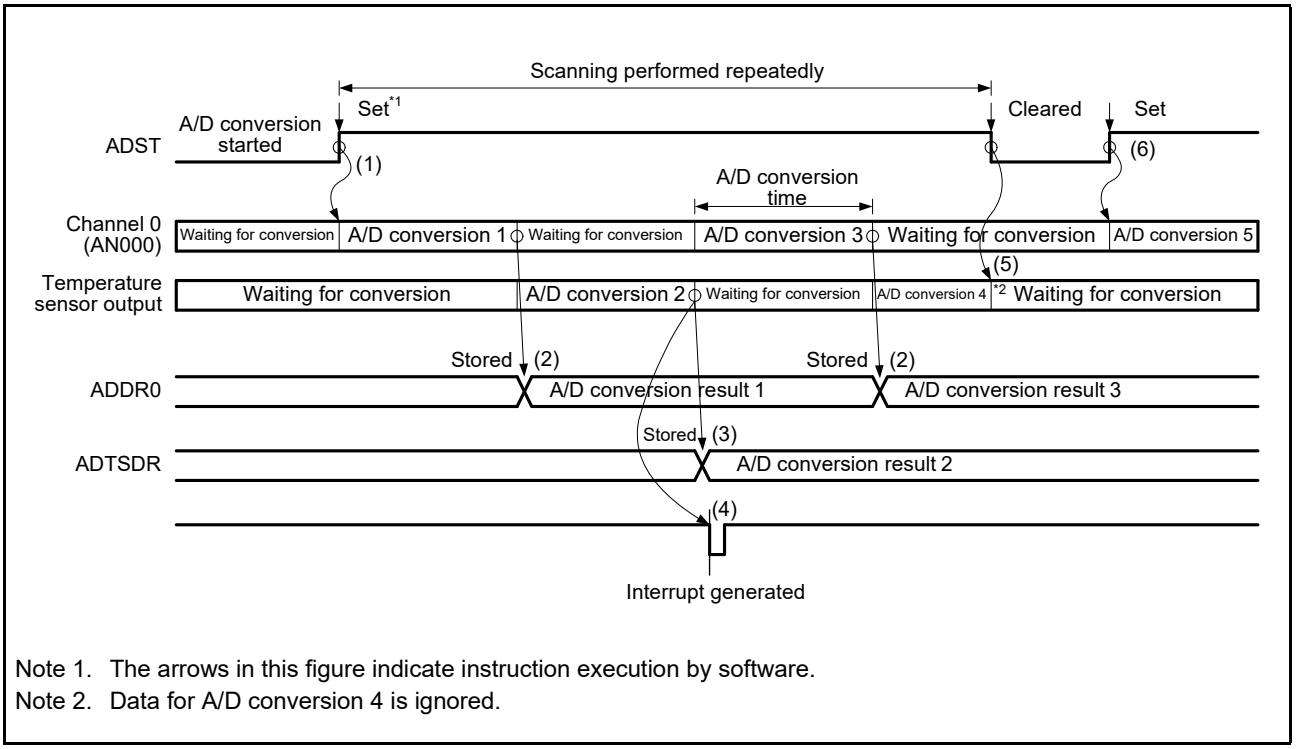
Note 1. The arrows in this figure indicate instruction execution by software.

Figure 30.13 Example of Operation in Continuous Scan Mode (Channel-Dedicated Sample-and-Hold Circuits Used; AN000 and AN001 Selected + Self-Diagnosis)

30.3.3.5 A/D Conversion of Temperature Sensor Output

When the channels and temperature sensor output are selected at the same time, A/D conversion is first performed on the analog input of the selected channels, and then A/D conversion of the temperature sensor output is repeated as below. With the channels to be subjected to A/D conversion deselected, selecting only the temperature sensor output is also possible.

(1) When software, synchronous trigger (TPUa, ELC), or asynchronous trigger sets the ADCSR.ADST bit to 1 (starting A/D conversion), conversion for the ANn pins selected in the ADANSA register starts in order from the channel with the lowest number n.
(2) On completion of A/D conversion on the selected channel, the result is stored in the corresponding A/D data register (ADDRy), and then A/D conversion of temperature sensor output starts.
(3) On completion of A/D conversion of temperature sensor output, the result is stored in the corresponding A/D temperature sensor data register (ADTRDR).
(4) If the ADCSR.ADIE bit is set to 1 (enabling S12ADI interrupt generation upon scan conversion completion), an S12ADI interrupt is generated. Furthermore, the 12-bit A/D converter continuously starts A/D conversion for the ANn pins selected in the ADANSA register in order from the channel with the lowest number n.
(5) The ADCSR.ADST bit is not cleared automatically, and steps 2 to 4 are repeated as long as this bit remains set to 1. When the ADCSR.ADST bit is set to 0 (stopping A/D conversion), A/D conversion stops and the 12-bit A/D converter enters a waiting state.
(6) If the ADCSR.ADST bit is later set to 1 (starting A/D conversion), A/D conversion starts again for ANn pins selected by the ADANSA register in order from the channel with the lowest number n.



Note 1. The arrows in this figure indicate instruction execution by software.
Note 2. Data for A/D conversion 4 is ignored.

Figure 30.14 Example of Operation in Continuous Scan Mode (Basic Operation: AN000 and Temperature Sensor Output or Internal Reference Voltage Selected)

30.3.4 Group Scan Mode

30.3.4.1 Basic Operation

In basic operation of group scan mode, A/D conversion is performed once on the analog inputs of all the specified channels in group A and group B after scanning is started by a synchronous trigger (TPUa, ELC) as below. Scan operation of each group is similar to the scan operation in single scan mode.

The synchronous triggers of group A and B can be selected using the TRSA[5:0] and TRSB[5:0] bits in ADSTRGR, respectively. The different triggers should be set for group A and group B to prevent simultaneous A/D conversion of group A and group B. Software trigger should not be used.

The ADANSA register and the TSSA bit in the ADEXICR register are used to select the channels subject to A/D conversion for group A. The ADANSB register and the TSSB bit in the ADEXICR register are used to select the channels subject to A/D conversion for group B. Group A and group B cannot use the same channels.

When self-diagnosis is selected in group scan mode, self-diagnosis is separately executed for group A and group B.

30.3.4.2 A/D Conversion in Double Trigger Mode

When double trigger mode is selected in group scan mode, A/D conversion is performed for two rounds of single scan operation started by a synchronous trigger (TPUa, ELC) as a sequence for group A. For group B, single scan operation started by a synchronous trigger (TPUa, ELC) is performed once.
In group scan mode, the synchronous triggers of group A and B can be selected using the TRSA[5:0] and TRSB[5:0] bits in ADSTRGR, respectively. The different triggers should be selected for group A and group B to prevent simultaneous A/D conversion of group A and group B. Software trigger, or asynchronous trigger (ADTRGn) should not be used.
The DBLANS[4:0] bits in the ADCSR register are used to select the channels subject to A/D conversion for group A. The ADANSB register is used to select the channels subject to A/D conversion for group B. The same channels cannot be selected for both groups.
When double trigger mode is selected in group scan mode, the temperature sensor output A/D conversion select bits (ADEXICR.TSSA and ADEXICR.TSSB) should be set to 0 (deselected).
When double trigger mode is selected in group scan mode, self-diagnosis cannot be selected.
Duplication of A/D conversion data is enabled by setting the channel numbers to be duplicated to the DBLANS[4:0] bits in ADCSR and setting the DBLE bit in ADCSR to 1.

30.3.4.3 Operation under Group-A Priority Control

Setting the ADGSPCR.PGS bit to 1 in group scan mode makes operation proceed under group-A priority control. When setting the PGS bit in the ADPGSCR register to 1, follow the procedure in Figure 30.15 to set the relevant registers. If the procedure is not followed, A/D conversion operation and stored data are not guaranteed.
In operation in basic group scan mode, input of the trigger for the other group during A/D conversion in group A or group B is ignored. Under group-A priority control, if a group-A trigger is input during A/D conversion for group B, A/D conversion for group B is discontinued and A/D conversion for group A proceeds. If the setting of the ADGSPCR.GBRSCN bit is 0, the converter enters the waiting state on completion of the A/D conversion for group A, without restarting the A/D conversion for group B (that was discontinued by the group-A trigger). If the setting of the ADGSPCR.GBRSCN bit is 1, the converter automatically restarts scanning for group B from the head of the group after completion of the A/D conversion for group A. Table 30.9 summarizes operations in response to the input of a trigger during A/D conversion with the settings of the ADGSPCR.GBRSCN bit.

Scan operations in group A or group B are the same in single scan mode. Furthermore, single scanning for group B continues to proceed if the ADGSPCR.GBRP bit is set to 1.

In group scan mode, select a synchronous trigger for group A using the ADSTRGR.TRSA[5:0] bits and select a synchronous trigger different from that of group A for group B using the ADSTRGR.TRSB[5:0] bits. Set the ADSTRGR.TRSB[5:0] bits to 3Fh when setting the ADGSPCR.GBRP bit to 1. Furthermore, as targets for A/D conversion, select channels for group A using the ADANSA register and the ADEXICR.TSSA bit, and for group B, select channels different from those for group A using the ADANSB register and the ADEXICR.TSSB bit.

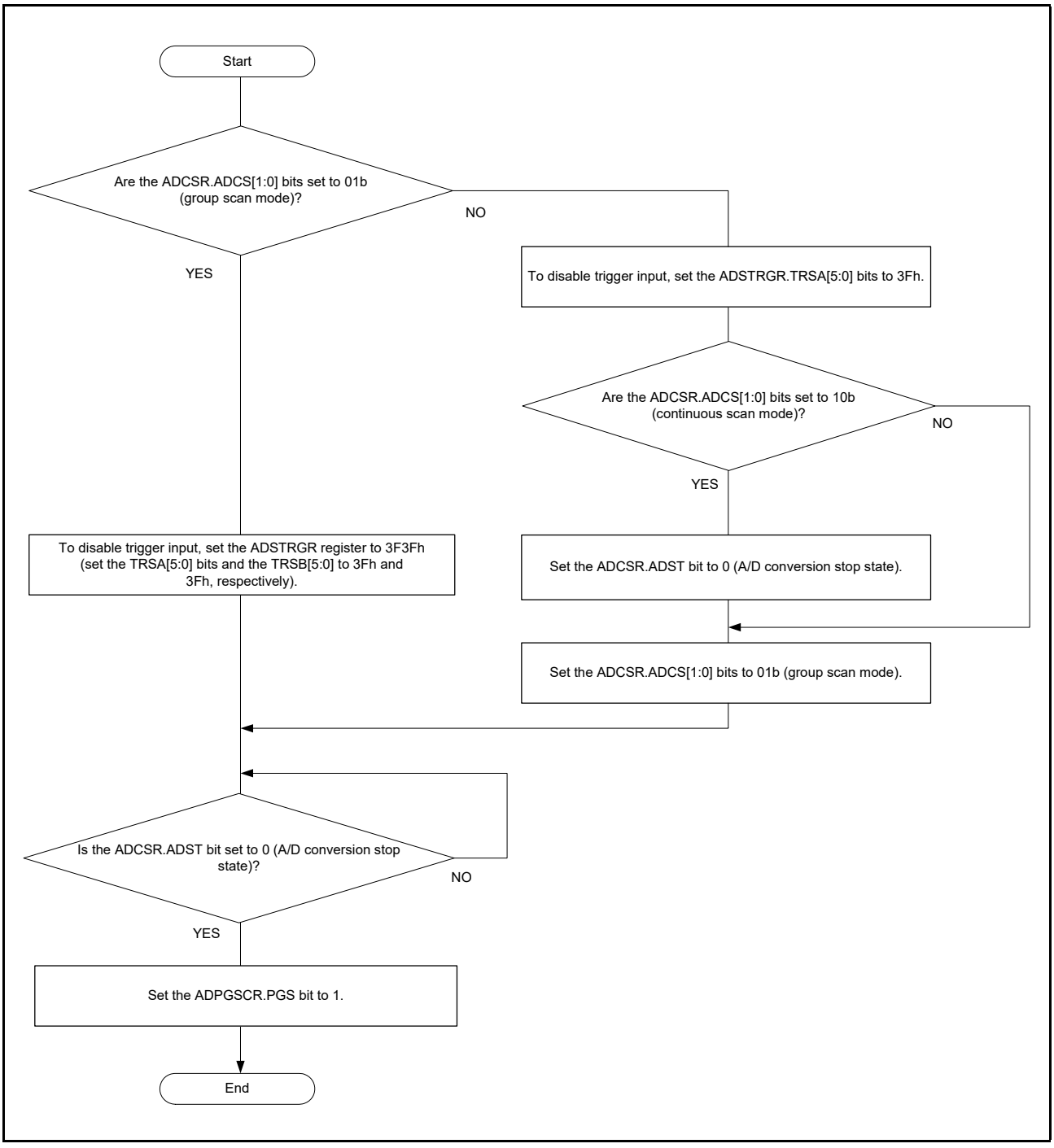


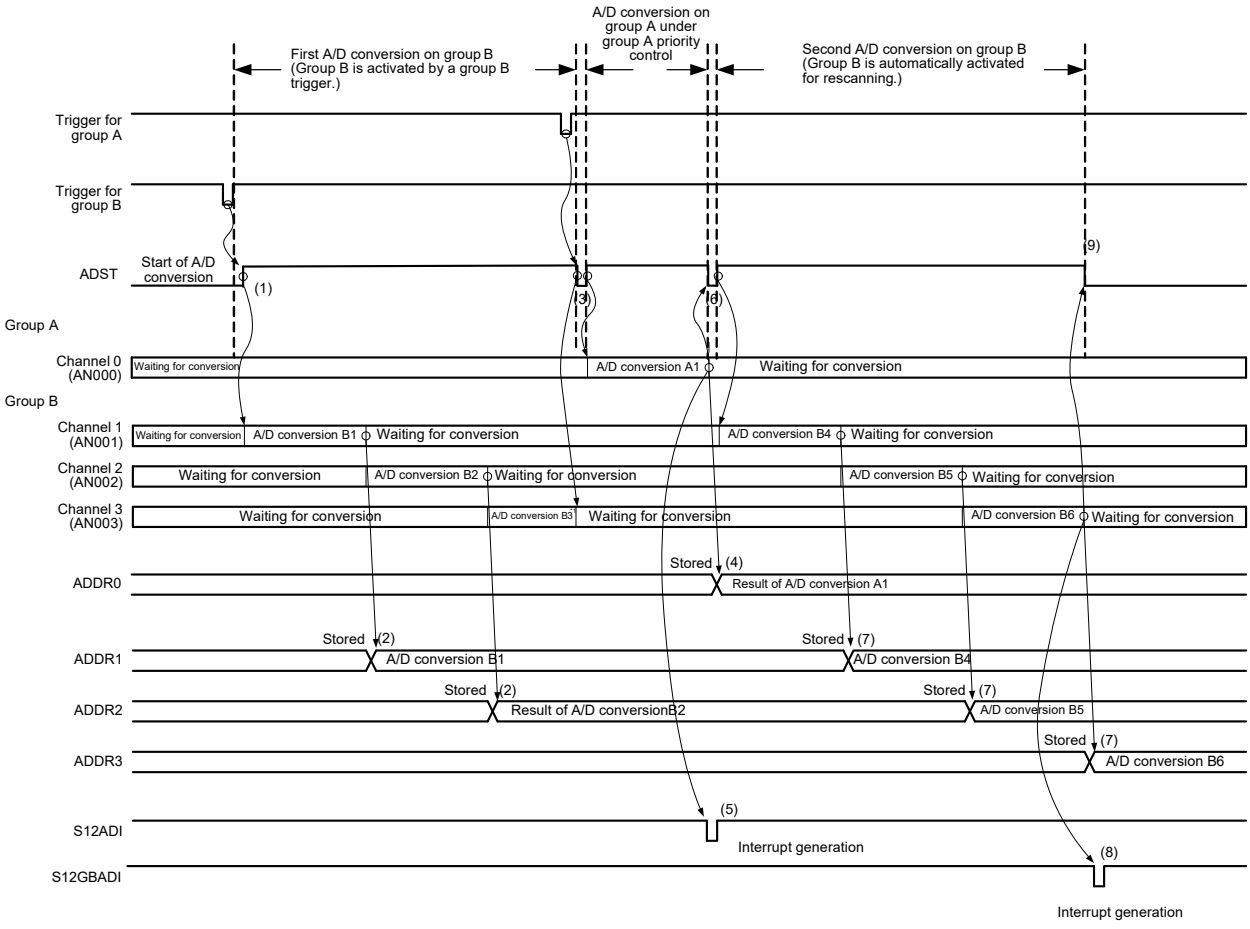
Figure 30.15 Flow of Setting the ADPGSCR.PGS Bit

Table 30.9 Control of A/D Conversion Operations According to the Settings of the ADGSPCR.GBRSCN Bit

A/D Conversion Operation	Trigger Input	ADGSPCR.GBRSCN = 0	ADGSPCR.GBRSCN = 1
When A/D conversion for group A is in progress	Input of trigger for group A	Trigger input is ineffective.	Trigger input is ineffective.
	Input of trigger for group B	Trigger input is ineffective.	A/D conversion is performed on group B after A/D conversion on group A is completed.
When A/D conversion for group B is in progress	Input of trigger for group A	Conversion for group B that is in progress is discontinued and conversion for group A starts.	• Conversion in progress for group B is discontinued and conversion for group A starts. • Conversion for group B starts after conversion for group A is completed.
	Input of trigger for group B	Trigger input is ineffective.	Trigger input is ineffective.

The following describes the operations in group scan mode under group-A priority control (i.e. ADGSPCR.GBRSCN = 1 and ADGSPCR.GBRP = 0) when channel 0 is selected for group A and channels 1 to 3 are selected for group B.

(1) When input of a trigger for group B sets the ADCSR.ADST bit to 1 (starting A/D conversion), conversion for the ANn channel pins selected in the ADANSB register starts in order from the channel with the lowest number.
(2) On completion of A/D conversion for a channel, the result is stored in the corresponding A/D data register (ADDRy).
(3) The ADCSR.ADST bit is cleared on the input of a trigger for group A while A/D conversion in group B is in progress, and the latter is discontinued. After that, the ADCSR.ADST bit is set to 1 (starting A/D conversion), and conversion for the ANn channel pins selected in the ADANSA register starts in order from the channel with the lowest number.
(4) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy).
(5) An S12ADI interrupt is generated if the setting of the ADCSR.ADIE bit is 1 (S12ADI interrupt upon scanning completion is enabled).
(6) After the ADST bit is automatically cleared, again, the bit is automatically set to 1 (starting A/D conversion) and conversion for the ANn channel pins of group B selected in the ADANSB register starts in order from the channel with the lowest number.
(7) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy).
(8) An S12GBADI interrupt is generated if the setting of the ADCSR.GBADIE bit is 1 (S12GBADI interrupt upon group B scanning completion enabled).
(9) The ADST bit retains the value 1 (starting A/D conversion) during A/D conversion and is automatically cleared on completion of conversion, after which the A/D converter enters the waiting state.

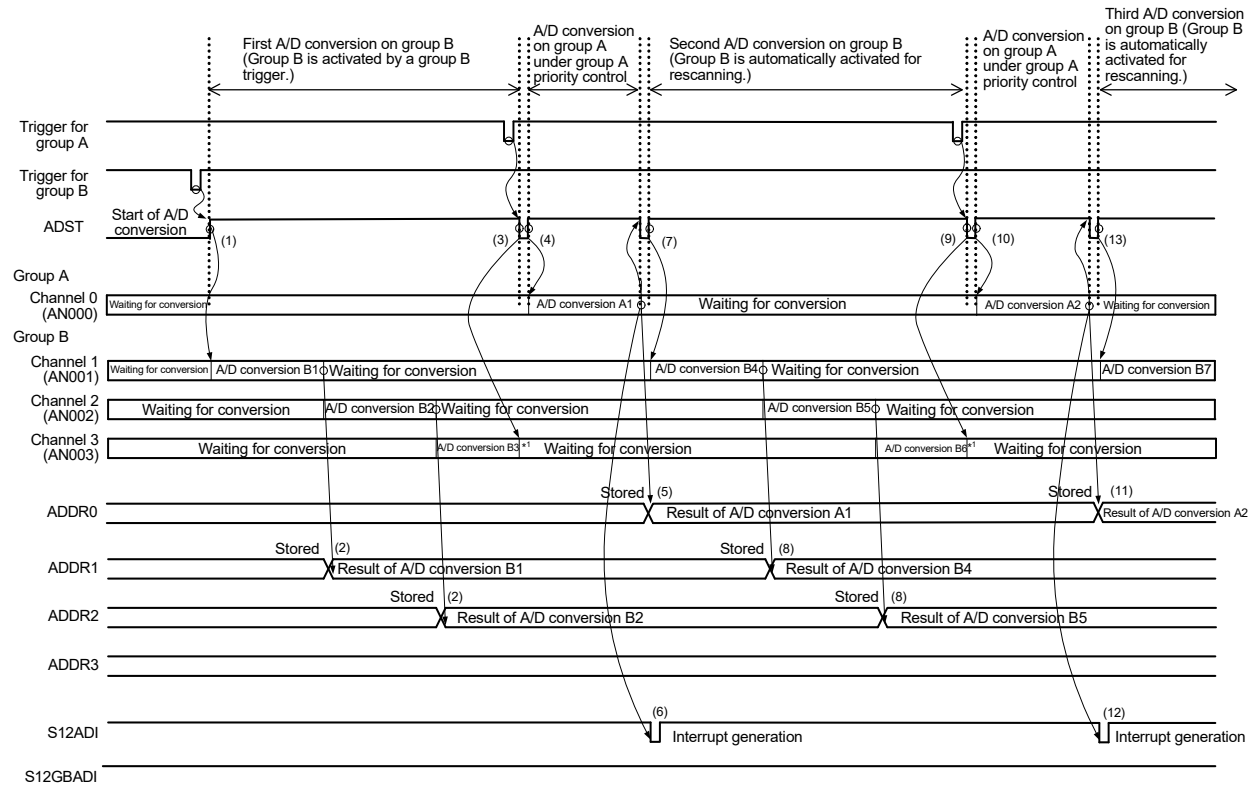


Note 1. Converted data from A/D conversion B3 are ignored.

Figure 30.16 Example of Operations under Group-A Priority Control (1) (when ADGSPCR.GBRSCN = 1 and ADGSPCR.GBRP = 0)

The following is an example when a group A trigger is input again during rescanning operation on group B. In this example, channel 0 is selected for group A and channels 1 to 3 are selected for group B when operation on group A is given priority (ADGSPCR.GBRSCN = 1, ADGSPCR.GBRP = 0).

(1) When a group B trigger input sets the ADCSR.ADST bit to 1 (starting A/D conversion), conversion for the ANn channel pins of group B selected in the ADANSB register starts in order from the channel with the lowest number n.
(2) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy).
(3) The ADCSR.ADST bit is cleared to 0 (stopping A/D conversion) on the input of a trigger for group A while A/D conversion in group B is in progress, and the latter is discontinued.
(4) After that, the ADCSR.ADST bit is set to 1 automatically and A/D conversion for the ANn group A channel pins selected in the ADANSA register starts in order from the channel with the lowest number n.
(5) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy).
(6) An S12ADI interrupt is generated if the setting of the ADCSR.ADIE bit is 1 (S12ADI interrupt upon scanning completion is enabled).
(7) On completion of A/D conversion on the group A, rescanning operation on group B sets the ADCSR.ADST bit to 1 automatically if the setting of the ADGSPCR.GBRSCN bit is 1 (enabling rescanning operation). After that, A/D conversion for the ANn group B channel pins selected in the ADANSB register starts again in order from the channel with the lowest number n.
(8) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy).
(9) If a group A trigger is input during A/D conversion on group B for rescanning, the ADCSR.ADST bit is cleared to 0 (stopping A/D conversion) and the ongoing A/D conversion on group B is stopped.
(10) After that, the ADCSR.ADST bit is set to 1 automatically and A/D conversion for the ANn group A channel pins selected in the ADANSA register starts in order from the channel with the lowest number n.
(11) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy).
(12) An S12ADI interrupt is generated if the setting of the ADCSR.ADIE bit is 1 (S12ADI interrupt upon scanning completion is enabled).
(13) On completion of A/D conversion on group A, rescanning operation on group B sets the ADCSR.ADST bit to 1 automatically if the setting of the ADGSPCR.GBRSCN bit is 1 (enabling rescanning operation). After that, A/D conversion for the ANn group B channel pins selected in the ADANSB register starts again in order from the channel with the lowest number n.
(14) If a group A trigger is input during A/D conversion on group B for rescanning, steps 9 to 13 are repeated. If a group A trigger is not input, the ADCSR.ADST bit is cleared automatically on completion of A/D conversion on group B and the 12-bit A/D converter enters a waiting state.



Note 1. Conversion data from A/D conversion B3 and B6 are ignored.

Figure 30.17 Example of Operations under Group-A Priority Control (2) (when ADGSPCR.GBRSCN = 1 and ADGSPCR.GBRP = 0)

The following is an example of a rescanning operation in which a group B trigger is input during A/D conversion on group A. In this example, channels 1 to 3 are selected for group A and channel 0 is selected for group B when operation on group A is given priority (ADGSPCR.GBRSCN = 1, ADGSPCR.GBRP = 0).

(1) When input of a trigger for group A sets the ADCSR.ADST bit to 1 (starting A/D conversion), conversion for the ANn channel pins of group A selected in the ADANSA register starts in order from the channel with the lowest number.
(2) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy).
(3) If a group B trigger is input during A/D conversion on group A, A/D conversion on group B can be performed after the A/D conversion on group A is completed. (However, if group A triggers are input continuously, the scan operation on group B is cancelled by conversion on group A and is not performed.)
(4) On completion of the A/D conversion on the group A, an S12ADI interrupt is generated if the setting of the ADCSR.ADIE bit is 1 (S12ADI interrupt upon scanning completion is enabled).
(5) On completion of the A/D conversion on the group A, activation of group B for rescanning sets the ADCSR.ADST bit to 1 automatically.
After that, conversion for the ANn channel pins of group B selected in the ADANSB register starts again in order from the channel with the lowest number.
(6) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy).
(7) On completion of the rescanning operation on the group B, an S12GBADI interrupt is generated if the setting of the ADCSR.GBADIE bit is 1 (S12GBADI interrupt upon scanning completion is enabled).
(8) The ADST bit retains the value 1 (starting A/D conversion) during A/D conversion and is automatically cleared on completion of conversion, after which the A/D converter enters a waiting state.

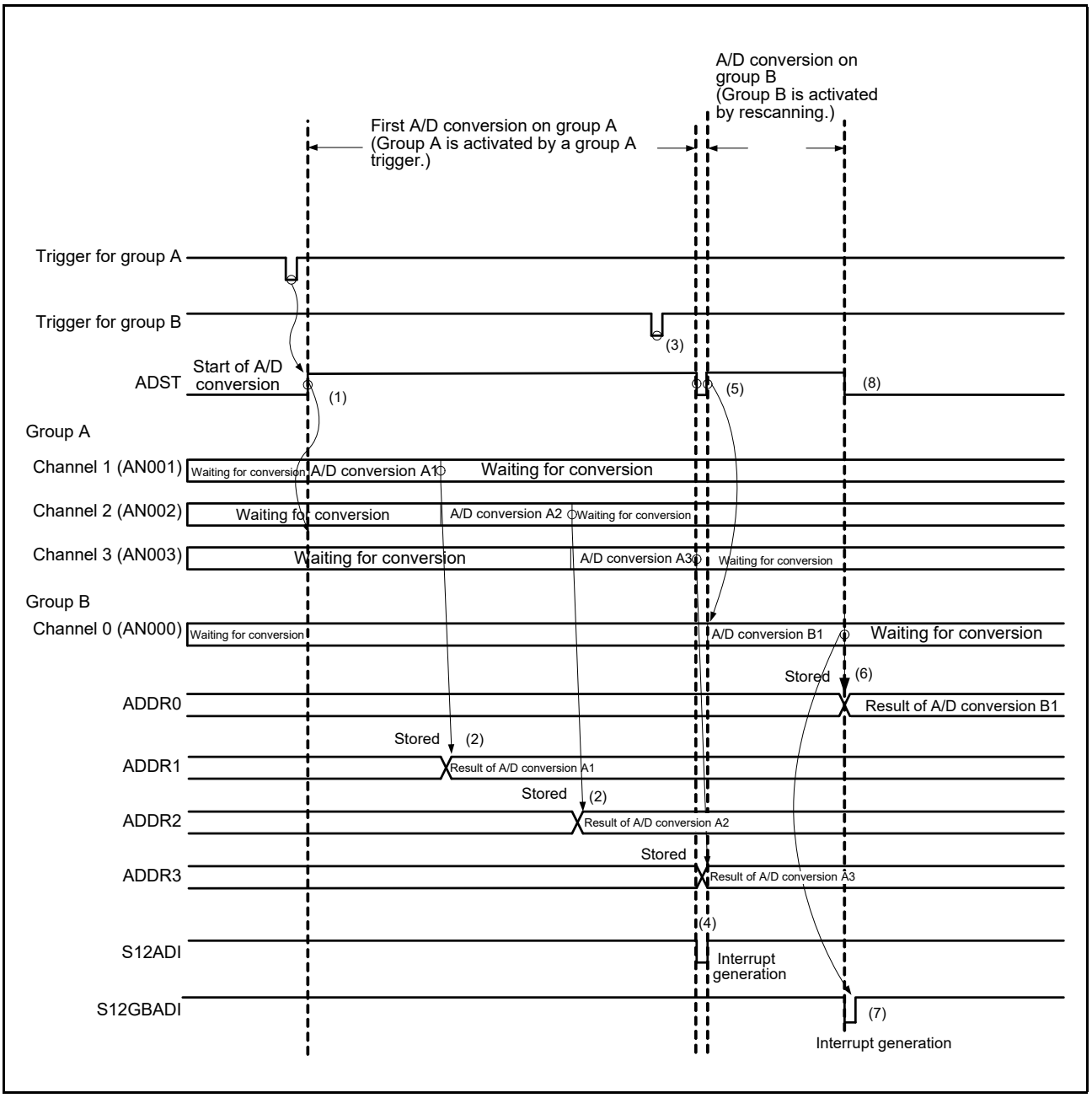
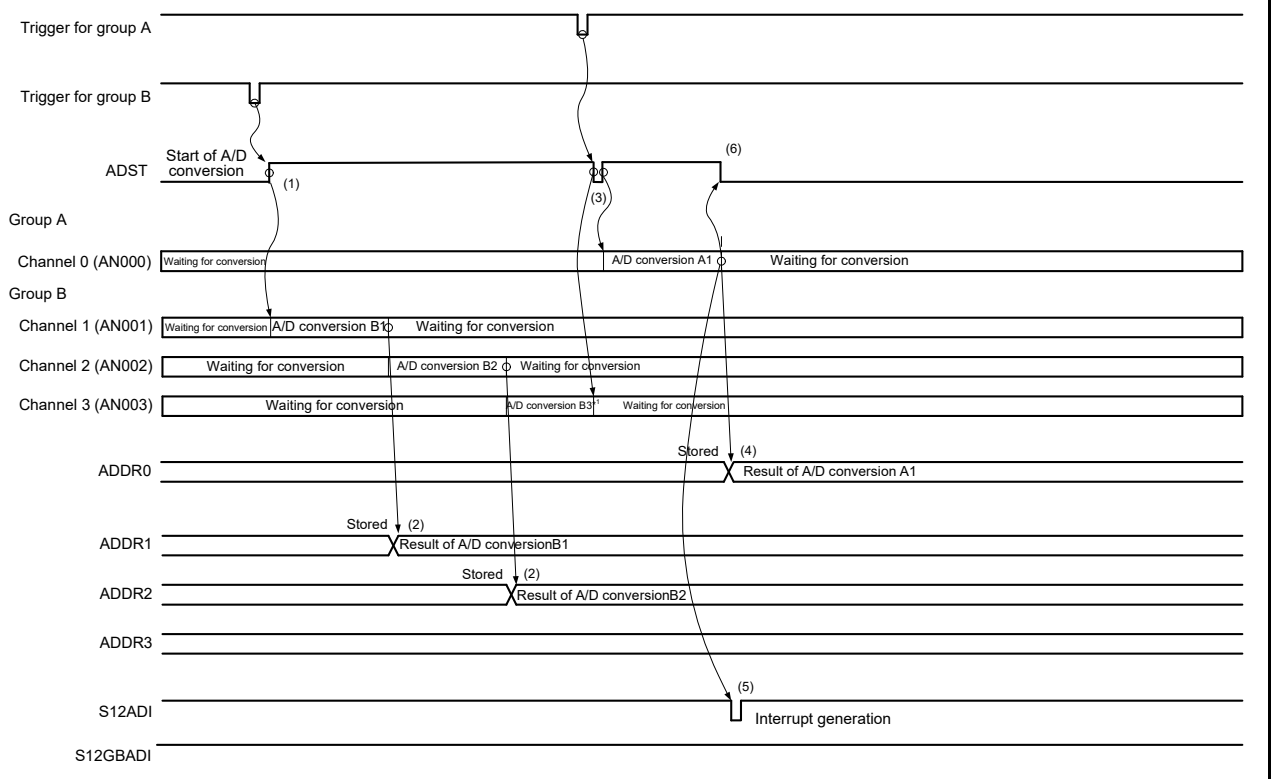


Figure 30.18 Example of Operations under Group-A Priority Control (3) (when ADGSPCR.GBRSCN = 1 and ADGSPCR.GBRP = 0)

The following is an example of operation under group-A priority control in which channel 0 is selected for group A and channels 1 to 3 are selected for group B (ADGSCR.GBRSCN = 0, ADGSCR.GBRP = 0).

(1) When input of a trigger for group B sets the ADCSR.ADST bit to 1 (starting A/D conversion), conversion for the ANn channel pins selected in the ADANSB register starts in order from the channel with the lowest number.
(2) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy).
(3) If a group A trigger is input during A/D conversion on group B, the ADCSR.ADST bit is cleared to 0 and the ongoing A/D conversion on group B is stopped. After that, the ADCSR.ADST bit is set to 1 (starting A/D conversion) and conversion for the ANn channel pins selected in the ADANSA register starts in order from the channel with the lowest number.
(4) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy).
(5) An S12ADI interrupt is generated if the setting of the ADCSR.ADIE bit is 1 (S12ADI interrupt upon scanning completion is enabled).
(6) The ADCSR.ADST bit retains the value 1 (starting A/D conversion) during A/D conversion and is cleared on completion of conversion, after which the A/D converter enters the waiting state.

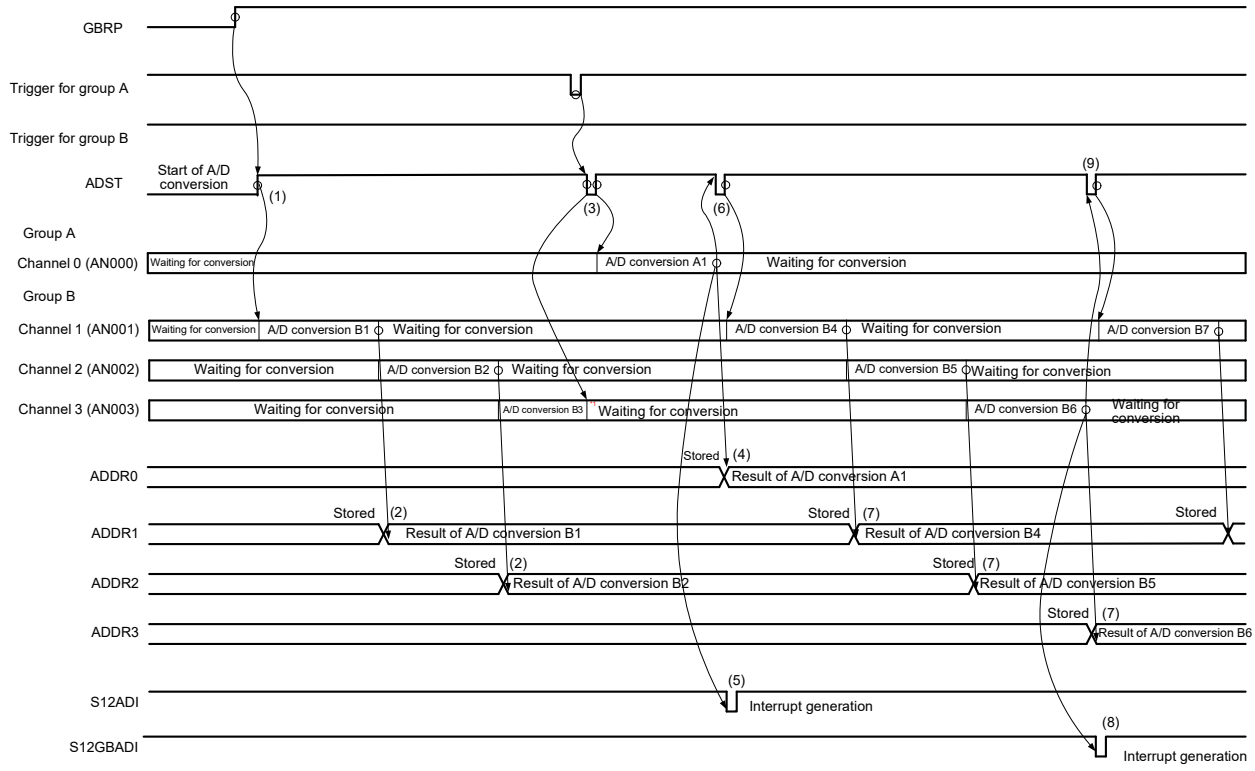


Note 1. The converted data of A/D conversion B3 is ignored.

Figure 30.19 Example of Operation under Group-A Priority Control (4) (when ADGSPCR.GBRSCN = 0 and ADGSPCR.GBRP = 0)

The following is an example of operation under group A priority control in which channel 0 is selected for group A and channels 1 to 3 are selected for group B (ADGSCR.GBRP = 1).

(1) The ADCSR.ADST bit is set to 1 (starting A/D conversion) when ADGSPCR.GBRP is set to 1, and conversion for the ANn channel pins selected in the ADANSB register starts in order from the channel with the lowest number.
(2) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy).
(3) If a group A trigger is input during A/D conversion on group B, the ADCSR.ADST bit is cleared to 0 and the ongoing A/D conversion on group B is stopped. After that, the ADCSR.ADST bit is set to 1 (starting A/D conversion) and conversion for the ANn channel pins selected in the ADANSA register starts in order from the channel with the lowest number.
(4) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy).
(5) An S12ADI interrupt is generated if the setting of the ADCSR.ADIE bit is 1 (S12ADI interrupt upon scanning completion is enabled).
(6) After the ADST bit is automatically cleared and then is automatically set to 1 (starting A/D conversion) again, conversion for the ANn channel pins selected in the ADANSB register starts again in order from the channel with the lowest number.
(7) On completion of A/D conversion on a single channel, the result is stored in the corresponding A/D data register (ADDRy).
(8) An S12GBADI interrupt is generated if the setting of the ADCSR. GBADIE bit is 1 (S12GBADI interrupt upon completion of scanning for group B is enabled).
(9) After the ADST bit is automatically cleared, again, the bit is automatically set to 1 (starting A/D conversion), and then conversion for the ANn channel pins selected in the ADANSB register starts again in order from the channel with the lowest number. Steps 6 to 9 are repeated as long as the ADGSPCR.GBRP bit remains 1.
Clearing of the ADCSR.ADST bit to 0 is prohibited while the ADGSPCR.GBRP bit is set to 1.
Follow the procedure for clear operation by software through the ADCSR.ADST bit, shown in Figure 30.31, if you wish to forcibly stop A/D conversion while ADGSPCR.GBRP = 1.



Note 1. Converted data from A/D conversion B3 are ignored.

Figure 30.20 Example of Operation under Group-A Priority Control (5) (when ADGSPCR.GBRP = 1)

30.3.5 Comparison

Comparison is of a reference value set in a register with the result of A/D conversion on selected channels. Self-diagnosis function and double-triggered mode are not available while comparison is in use.

Operation using comparison in combination with continuous scan mode is described below.

(1) A/D conversion is started in the order of the selected channels and the temperature sensor output when ADCSR.ADST is set to 1 (to start A/D conversion) by software, or in response to a synchronous trigger (TPUa, ELC) or asynchronous trigger.

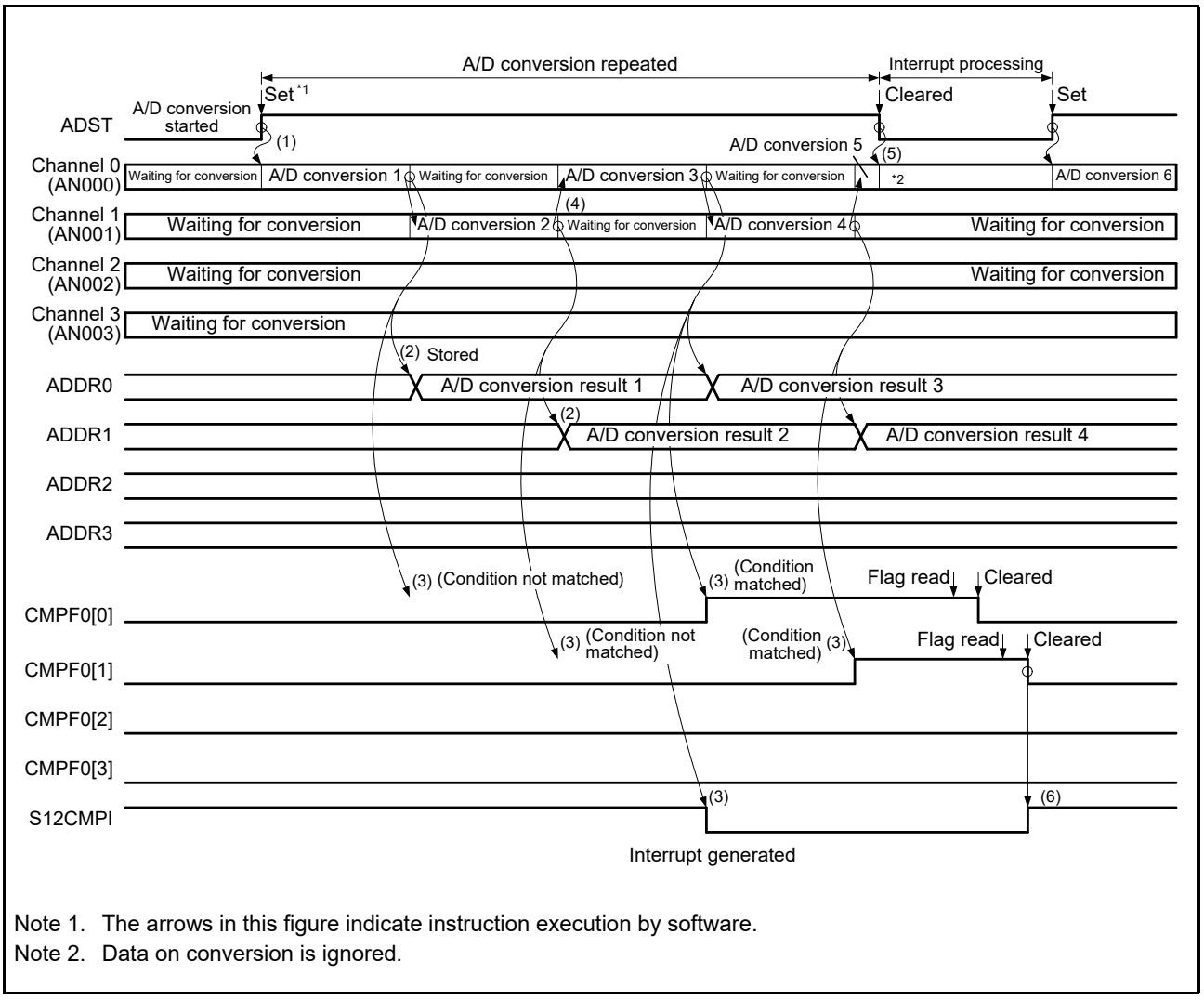
(2) When A/D conversion is completed, the result is stored in the A/D self-diagnosis data register (ADDRy, ADTSDR). If the register is selected for comparison by the settings of the ADCMPANSR and ADCMPANSER registers, its value is then compared with that of the ADCMPDR0/1 registers.

(3) If the result of comparison meets the condition set in the ADCMPCR.WCMPE bit, the ADCMPLR register, and the ADCMPLER register, the bits ADCMPSR.CMPFn and ADCMPSER.CMPFTS are set to 1. If the setting of the ADCMPCR.CMPIE bit is 1 at this time, an S12CMPI interrupt is also generated.

(4) When A/D conversion is completed for all selected channels, A/D conversion is started again.

(5) If the setting of the ADCSR.ADST bit is 0 (A/D conversion stop) after the S12CMPI interrupt is accepted, the interrupt processing proceeds for channels that have the compare flag.

(6) The S12CMPI interrupt signal is deasserted when all of the compare flags have been cleared. To start further comparison, start A/D conversion again.



Note 1. The arrows in this figure indicate instruction execution by software.
Note 2. Data on conversion is ignored.

Figure 30.21 Example of Operation Using Compare Function (AN000, AN001, AN002, and AN003 Selected for Comparison)

30.3.6 Analog Input Sampling and Scan Conversion Time

Scan conversion can be activated either by a software trigger; a synchronous trigger (TPUa, ELC); or an asynchronous trigger (ADTRGn). After the start-of-scanning-delay time (t_D) has elapsed, processing by the channel-dedicated sample-and-hold circuits, processing for disconnection detection assistance, and processing of conversion for self-diagnosis proceed, and this is followed by A/D conversion.

Figure 30.22 shows the scan conversion timing in single scan mode, in which scan conversion is activated by a software trigger or a synchronous trigger (TPUa, ELC). Figure 30.23 shows the scan conversion timing in single scan mode, in which scan conversion is activated by an asynchronous trigger ADTRGn. The scan conversion time (t_{SCAN}) includes the start-of-scanning-delay time (t_D), channel-dedicated sample-and-hold circuit processing time (t_{SPLSH})*1, disconnection detection assistance processing time (t_{DIS})*2, self-diagnosis A/D conversion processing time (t_{DIAG})*3, A/D conversion processing time (t_{CONV}), channel-dedicated sample-and-hold circuit end time (t_{SHED})*4, and end-of-scanning-delay time (t_{ED}).

The A/D conversion processing time (t_{CONV}) consists of input sampling time (t_{SPL}) and time for conversion by successive approximation (t_{SAM}). The sampling time (t_{SPL}) is used to charge sample-and-hold circuits in the A/D converter. If there is not sufficient sampling time due to the high impedance of an analog input signal source, or if the A/D conversion clock (ADCLK) is slow, sampling time can be adjusted using the ADSSTR register.

The time for conversion by successive approximation (t_{SAM}) is at 13 ADCLK states with 12-bit accuracy selected, 11 ADCLK states with 10-bit accuracy selected, and 9 ADCLK states with 8-bit accuracy selected. Table 30.11 shows the scan conversion time.

The scan conversion time (t_{SCAN}) in single scan mode for which the number of selected channels is n can be determined as follows:

$$t_{SCAN} = t_D + t_{SPLSH} + (t_{DIS} \times n) + t_{DIAG} + (t_{CONV} \times n) + t_{ED}$$

The scan conversion time for the first cycle in continuous scan mode is t_{SCAN} for single scan minus t_{ED} plus t_{SHED}. The scan conversion time for the second and subsequent cycles in continuous scan mode is fixed to $t_{SPLSH} + (t_{DIS} \times n) + t_{DIAG} + (t_{CONV} \times n) + t_{SHED}$.

Note 1. When no channel-dedicated sample-and-hold circuits are used, $t_{SH} = 0$.
Note 2. When disconnection detection assistance is not selected, $t_{DIS} = 0$.
Note 3. When the self-diagnosis function is not used, $t_{DIAG} = 0$.
Note 4. When no channel-dedicated sample-and-hold circuits are used, $t_{SHED} = 0$. Here, continuous scan mode is assumed. In single scan mode and group scan mode, t_{SHED} is included in the end-of-scanning-delay time (t_{ED}).

Table 30.10 Example of Setting the ADSSTR Register

Use	Setting Range	Sampling Time*1
Standard (initial value)	0Bh	0.18 µs (For ADCLK = 60 MHz)
Use this range if there is not sufficient sampling time due to the high impedance of an analog input signal source.	0Ch to FFh	Example: FFh 4.3 µs (For ADCLK = 60 MHz)
Use this range if ADCLK is less than 60 MHz and the sampling time needs to be less than the initial value.	05h to 0Ah	Example: 0Ah 0.67 µs (For ADCLK = 15 MHz)

Note 1. The sampling time is determined by the following formula.

$$\text{Sampling time (µs)} = \frac{\text{ADSSTR register setting}}{\text{ADCLK (MHz)}}$$

Table 30.11 Times for Conversion during Scanning (in Numbers of Cycles of the ADCLK and PCLKH)

Item			Symbol		Type/Conditions: Synchronous Trigger TPUa, ELC	Asynchronous Trigger	Software Trigger	Unit
Scan start processing time*1, *2	A/D conversion on group A under group A priority control.	Group B is to be stopped. (Group A is activated after group B is stopped due to an A/D conversion source of group A.)	t_D		3 PCLKH + 6 ADCLK	—	—	Cycle
		Group B is not to be stopped. (Activation by an A/D conversion source of group A.)			2 PCLKH + 4 ADCLK	—	—	
	A/D conversion when self-diagnosis is enabled	A/D conversion for self-diagnosis is to be started.			2 PCLKH + 6 ADCLK	4 PCLKH + 6 ADCLK	6 ADCLK	
		Normal A/D conversion is to be started after completion of self-diagnosis conversion.			2 ADCLK	2 ADCLK	2 ADCLK	
		A/D conversion for self-diagnosis is to be started after completion of conversion for continuous scan on the last channel specified.			2 ADCLK	2 ADCLK	2 ADCLK	
	Other than above				2 PCLKH + 4 ADCLK	4 PCLKH + 4 ADCLK	4 ADCLK	
Channel-dedicated sample-and-hold processing time*1	Sampling time		t_{SPLSH}*3	t_{SH}	The setting of ADSHCR.SSTSH[7:0] (initial value = 18h) × ADCLK			
	Wait time between sampling and A/D Conversion			t_W	12 ADCLK			
Disconnection detection assistance processing time			t_{DIS}		The setting of ADNDIS[3:0] (initial value = 00h) × ADCLK			
Self-diagnosis conversion processing time*1	Sampling time		t_{DIAG}	t_{SPL}	The setting of ADSSTR0 (initial value = 0Bh) × ADCLK			
	Time for conversion by successive approximation	12-bit conversion accuracy		t_{SAM}	15 ADCLK			
		10-bit conversion accuracy			13 ADCLK			
		8-bit conversion accuracy			11 ADCLK			
A/D conversion processing time*1	Sampling time		t_{CONV}*3	t_{SPL}	The setting of ADSSTRn (n = 0 to 7, L, T) (initial value = 0Bh) × ADCLK			
	Time for conversion by successive approximation	12-bit conversion accuracy		t_{SAM}	13 ADCLK			
		10-bit conversion accuracy			11 ADCLK			
		8-bit conversion accuracy			9 ADCLK			
Channel-dedicated sample-and-hold end processing time			t_{SHED}		2 ADCLK			
Scan end processing time*1			t_{ED}		1 PCLKH + 3 ADCLK			

Note 1. Refer to Figure 30.22 and Figure 30.23 for illustration of times t_D, t_{SPLSH}, t_{DIAG}, t_{CONV}, and t_{ED}.
Note 2. This is the maximum time required from software writing or trigger input to A/D conversion start.
Note 3. Set "t_{SPLSH} + t_{CONV}" so that it satisfies the A/D conversion characteristics shown in section 34.5 A/D Conversion Characteristics

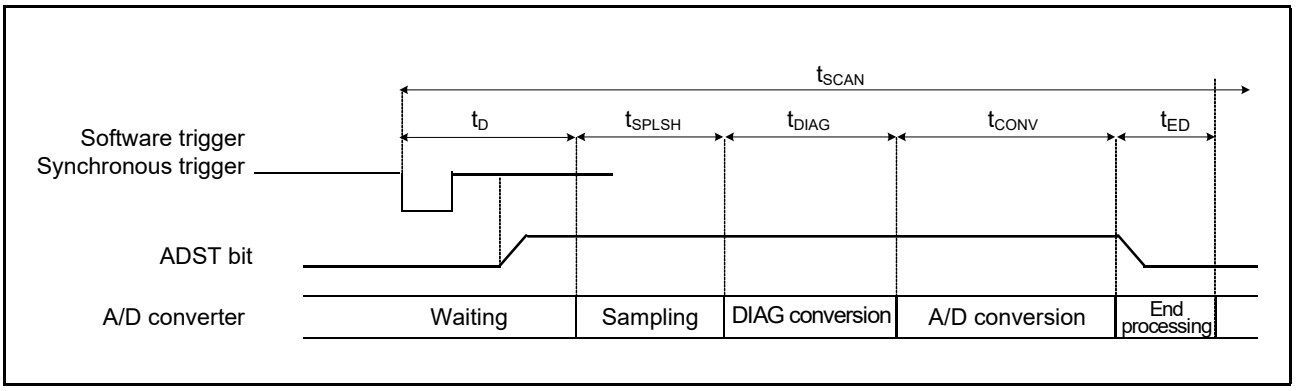


Figure 30.22 Scan Conversion Timing (Activated by Software or Synchronous Trigger Input (TPUa, ELC))

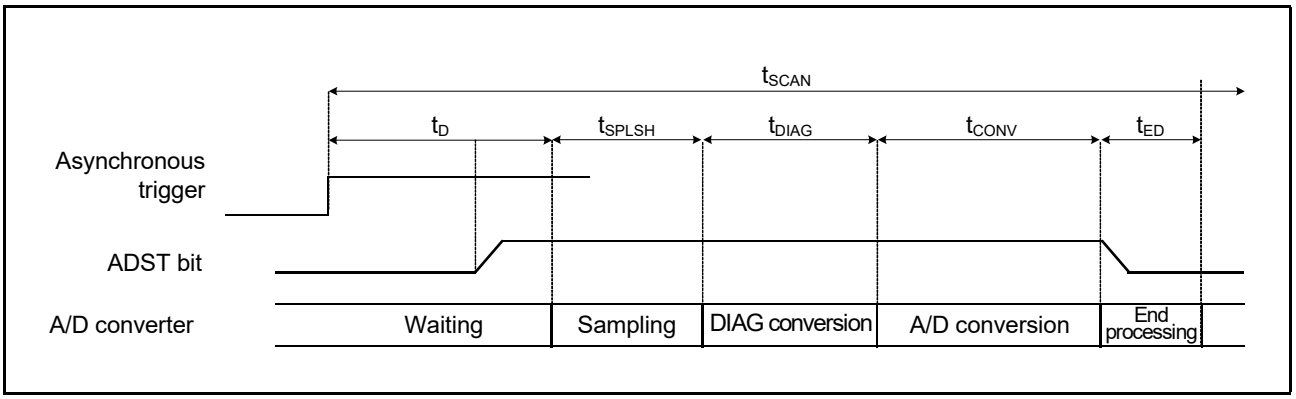


Figure 30.23 Scan Conversion Timing (Activated by Asynchronous Trigger Input (ADTRGn))

30.3.7 Usage Example of A/D Data Register Automatic Clearing Function

In A/D-converted value addition/average mode, the A/D data register automatic clearing function can be used when A/D conversion of the analog input of the selected channels or A/D conversion of the temperature sensor output is selected. Setting the ACE bit in ADCER to 1 automatically clears the A/D data registers (ADDRy, ADRD, ADDBLDR, ADTSDR) to 0000h when the A/D data registers are read by the CPU or DMAC.

This function enables detection of update failures of the A/D data registers (ADDRy, ADRD, ADDBLDR, ADTSDR). The following describes the examples in which the function to automatically clear the ADDRy register is enabled and disabled.

In a case where the ACE bit in ADCER is 0 (automatic clearing is disabled), if the A/D conversion result (0222h) is not written to the ADDRy register for some reason, the old data (0111h) will be the ADDRy value. Furthermore, if this ADDRy value is read into a general register using an A/D scan end interrupt, the old data (0111h) can be saved in the general register. When checking whether there is an update failure, it is necessary to frequently save the old data in the RAM or a general register.

In a case where the ACE bit in ADCER is 1 (automatic clearing is enabled), when ADDRy = 0111h is read by the CPU or DMAC, ADDRy is automatically cleared to 0000h. After that, if the A/D conversion result 0222h cannot be transferred to ADDRy for some reason, the cleared data (0000h) remains as the ADDRy value. If this ADDRy value is read into a general register using an A/D scan end interrupt at this point, 0000h will be saved in the general register. Occurrence of an ADDRy update failure can be determined by simply checking that the read data value is 0000h.

30.3.8 A/D-Converted Value Addition/Average Mode

In A/D-converted value addition mode, the same channel is A/D-converted two to four consecutive times and the sum of the converted values is stored in the data register. In A/D-converted value average mode, the same channel is A/D-converted two to four consecutive times and the mean of the converted values is stored in the data register. The use of the average of these results can improve the accuracy of A/D conversion, depending on the types of noise components that affect A/D conversion. This function, however, cannot always guarantee an improvement in A/D conversion accuracy. The A/D-converted value addition/average mode can be specified when A/D conversion of the channel select analog input or temperature sensor output is selected.

30.3.9 Disconnection Detection Assist Function

This converter incorporates the function to fix the charge for sampling capacitance to the specified state (VREFH0 or VREFL0 for unit 0; VREFH1 or VREFL1 for unit 1) before start of A/D conversion. This function enables disconnection detection in wiring of analog inputs.

Figure 30.24 illustrates the A/D conversion operation when the disconnection detection assist function is used. Figure 30.25 shows an example of disconnection detection when precharge is selected. Figure 30.26 shows an example of disconnection detection when discharge is selected.

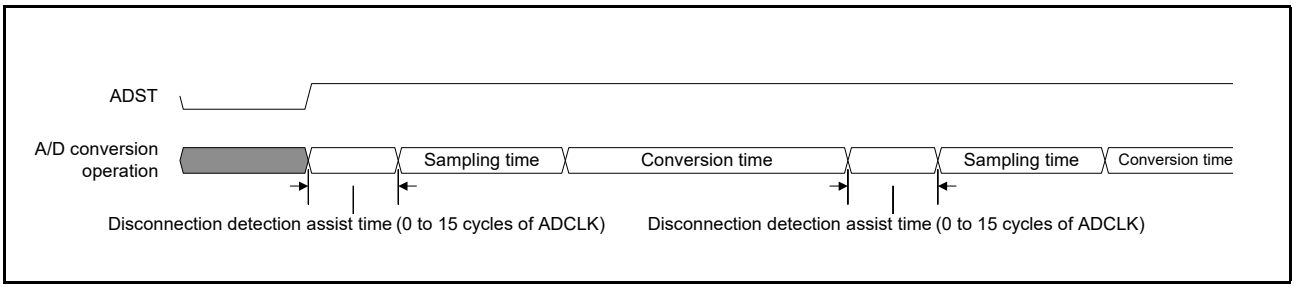
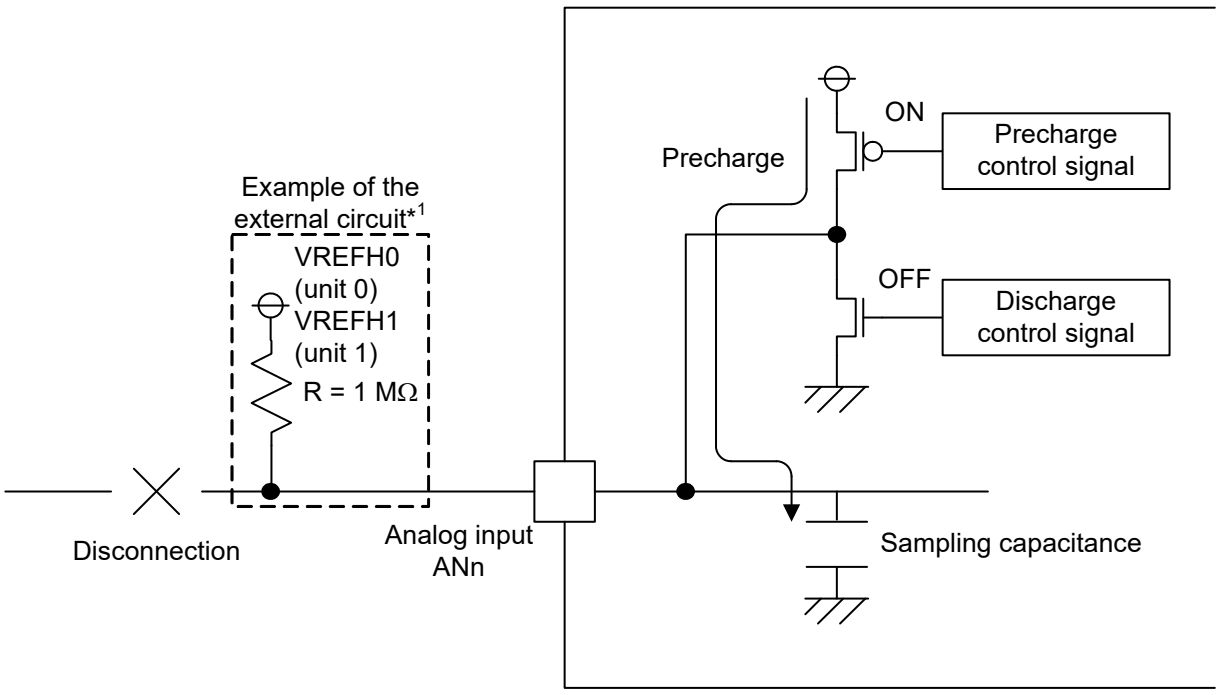
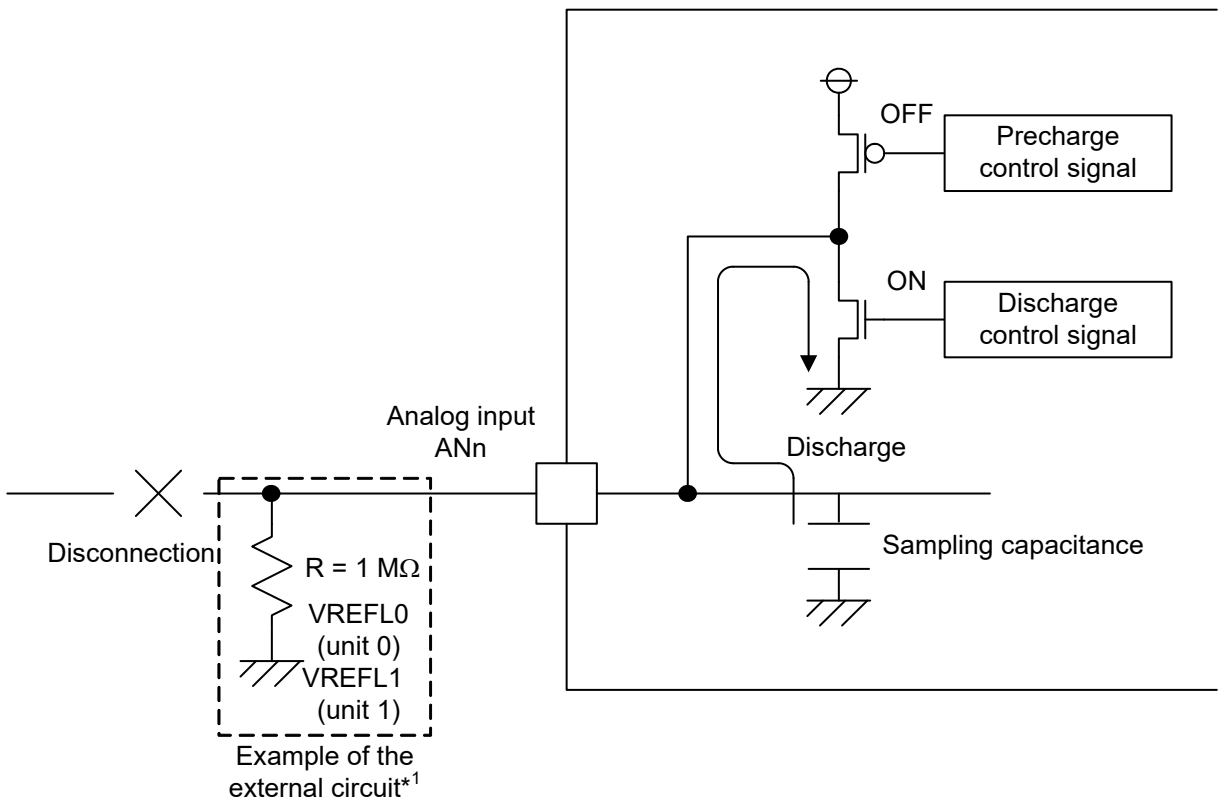


Figure 30.24 Operation of A/D Conversion when the Disconnection Detection Assist Function is Used



Note 1. The converted result should be used after fully evaluated because the result data when disconnection occurs varies depending on the external circuit.

Figure 30.25 Example of Disconnection Detection when Precharge is Selected



Note 1. The converted result should be used after fully evaluated because the result data when disconnection occurs varies depending on the external circuit.

Figure 30.26 Example of Disconnection Detection when Discharge is Selected

30.3.10 Starting A/D Conversion with Asynchronous Trigger

The A/D conversion can be started by the input of an asynchronous trigger. To start up the A/D converter by an asynchronous trigger, the A/D conversion start trigger select bits (ADSTRGR.TRSA[5:0]) should be set to 000000b and a low-level signal should be input to the asynchronous trigger (ADTRGn pin). Both the ADCSR.TRGE and ADCSR.EXTRG bits then should be set to 1. Figure 30.27 shows a timing of the asynchronous trigger input.
For the time between setting the ADST bit to 1 and starting A/D conversion, refer to section 30.5.3 A/D Conversion Restarting Timing and Termination Timing.

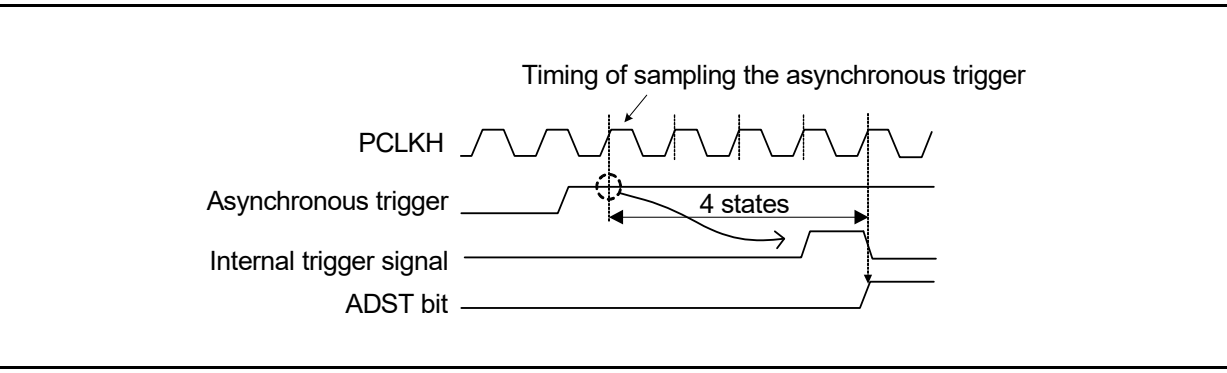


Figure 30.27 Asynchronous Trigger Input Timing

30.3.11 Starting A/D Conversion with Synchronous Trigger from Peripheral Module

The A/D conversion can be started by a synchronous trigger (TPUa, ELC). To start the A/D conversion by a synchronous trigger, the ADCSR.TRGE bit should be set to 1, the ADCSR.EXTRG bit should be cleared to 0, and the relevant sources should be selected by the ADSTRGR.TRSA[5:0] and ADSTRGR.TRSB[5:0] bits.

30.3.12 Pin-Level Self-Diagnosis Function

The pin-level self-diagnosis function performs A/D conversion using different voltage levels for input channels with odd numbers and for input channels with even numbers, respectively, to diagnose abnormality of routes from the ANn pins. Different voltage levels can be set in the ADTDCR register, with a combination of AVSS, AVCC, and 1/2 × AVCC. Users can select any physical channels to be tested in the pin-level self-diagnosis function. Furthermore, pin-level self-diagnosis can be performed for all scan operations.

30.3.12.1 Pin-Level Self-Diagnosis in Single Scan Mode (without Channel-Dedicated Sample-and-Hold Circuits)

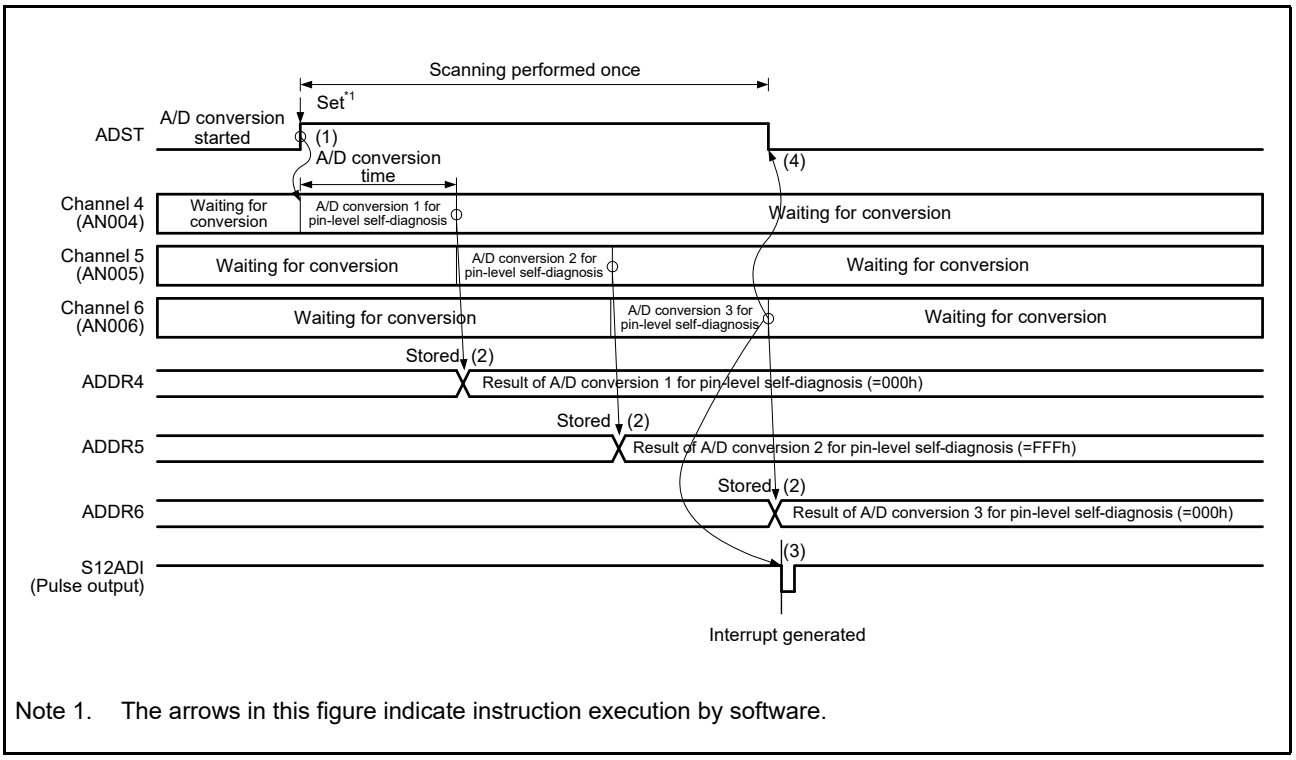
During pin-level self-diagnosis in single scan mode, A/D conversion of the analog input on the specified channel is performed for one cycle only as follows:

(1) When the ADST bit in ADCSR is set to 1 (A/D conversion start) by a software trigger, a synchronous trigger input (TPUa, ELC), or an asynchronous trigger input, A/D conversion is performed for ANn pins selected by the ADANSA register, starting from the pin with the smallest number n.

(2) Each time A/D conversion of a single channel is completed, the A/D conversion result is stored into the corresponding A/D data register (ADDRy).

(3) When A/D conversion of all the selected channels is completed, an S12ADI interrupt is generated if the ADIE bit in ADCSR is 1 (S12ADI interrupt upon completion of scanning is enabled).

(4) The ADST bit in ADCSR remains 1 (A/D conversion start) during A/D conversion, and is automatically cleared to 0 when A/D conversion of all the selected channels is completed. Then the 12-bit A/D converter enters a waiting state.



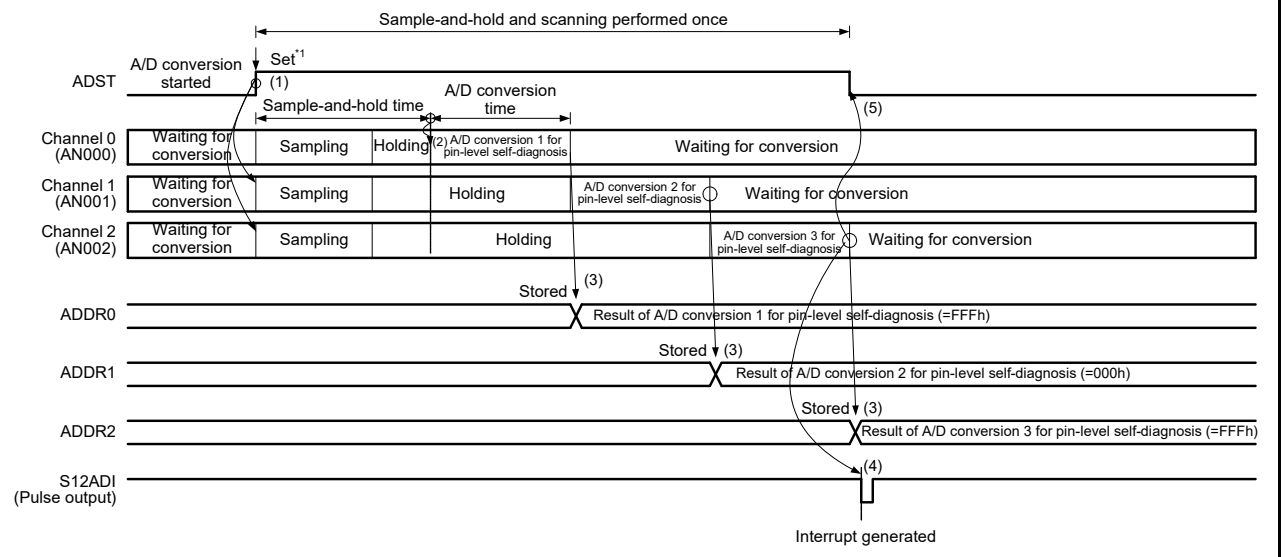
Note 1. The arrows in this figure indicate instruction execution by software.

Figure 30.28 Example Operation of Pin-Level Self-Diagnosis in Single Scan Mode (Basic Operation: AN004 to AN006 Selected/ADTDCR.TDLV[1:0] = 00b)

30.3.12.2 Pin-Level Self-Diagnosis in Single Scan Mode (with Channel-Dedicated Sample-and-Hold Circuits)

When pin-level self-diagnosis is performed with the channel-dedicated sample-and-hold circuit used, sample-and-hold operation is first performed, and then A/D conversion is performed once on the analog input of all the selected channels as below. The channels whose channel-dedicated sample-and-hold circuit is to be used can be selected by the SHANS[3:0] bits in ADSHCR.

(1) When the ADST bit in ADCSR is set to 1 (A/D conversion start) by a software trigger, a synchronous trigger input (TPUa, ELC), or an asynchronous trigger input, sampling is started for analog input of all the channels that use channel-dedicated sample-and-hold circuit.
(2) After sample-and-hold operation, A/D conversion is performed for ANn pins selected by the ADANSA register, starting from the channel with the smallest number n.
(3) Each time A/D conversion of a single channel is completed, the A/D conversion result is stored into the corresponding A/D data register (ADDRy).
(4) When A/D conversion of all the selected channels is completed, an S12ADI interrupt is generated if the ADIE bit in ADCSR is 1 (S12ADI interrupt upon completion of scanning is enabled).
(5) The ADST bit in ADCSR remains 1 (A/D conversion start) during A/D conversion, and is automatically cleared to 0 when A/D conversion of all the selected channels is completed. Then the 12-bit A/D converter enters a waiting state.



Note 1. The arrows in this figure indicate instruction execution by software.

Figure 30.29 Example Operation of Pin-Level Self-Diagnosis in Single Scan Mode (with Channel-Dedicated Sample-and-Hold Circuits: AN000 to AN002 Selected/ADTDCR.TDLV[1:0] = 01b)

30.3.13 Error Detection Function

The overwrite error detection function detects update of the A/D conversion results held in various A/D data registers, without a need of reading those results. The 12-bit A/D converter can generate S12ADOWEI interrupt, which is an overwrite error interrupt request to the error control module (ECM).

Setting the OWEIE bit in the ADERCR register to 1 enables generation of S12ADOWEI interrupt, and clearing the bit to 0 disables generation of S12ADOWEI interrupt. Reading the ADOWER 0/1 and ADOWEER registers can determine the A/D data register in which the overwrite error occurred. Writing 1 to the OWEC bit in the ADERCLR register clears all the overwrite error flags held in the ADOWER 0/1 and ADOWEER registers.

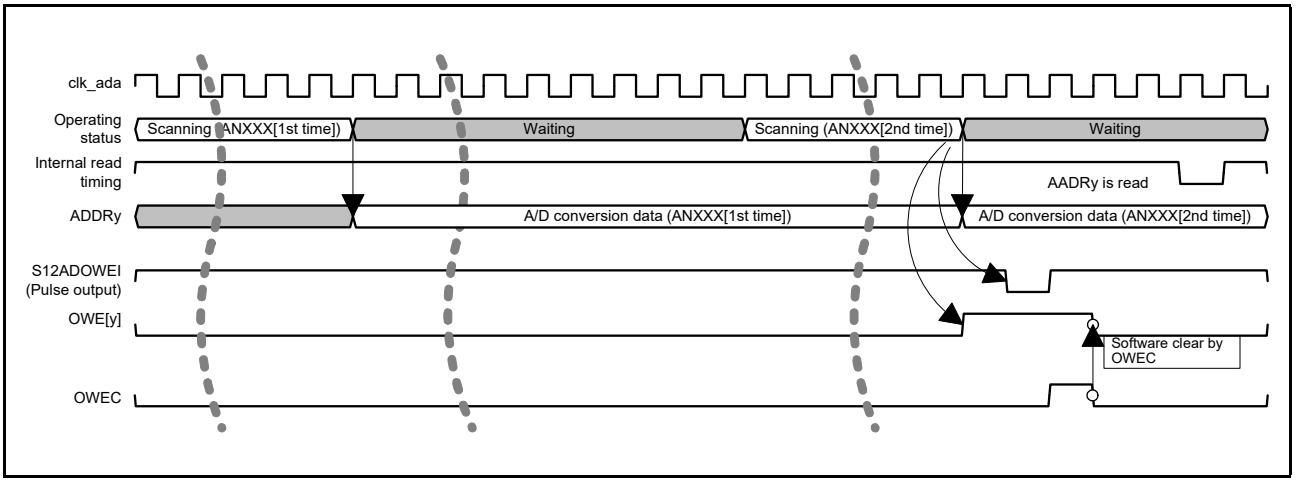


Figure 30.30 Example of Generation of AD Overwrite Error Interrupt

30.4 Interrupt Sources and DMAC Transfer Requests

30.4.1 Interrupt Requests

The 12-bit A/D converter can generate scan end interrupt requests S12ADI and S12GBADI. The module also generates the S12CMPI interrupt in response to matches with a condition for comparison.
Setting the ADCSR.ADIE bit to 1 and 0 enables and disables generation of an S12ADI interrupt, respectively; similarly, setting the ADCSR.GBADIE bit to 1 and 0 enables and disables generation of an S12GBADI interrupt, respectively.
Setting the ADCMPCR.CMPIE bit to 1 and 0 enables and disables generation of an S12CMPI interrupt, respectively.
In addition, the DMAC can be started up when an S12ADI or an S12GBADI interrupt is generated. Using an S12ADI or an S12GBADI interrupt to allow the DMAC to read the converted data enables continuous A/D conversion without burden on software.
For details on DMAC settings, see section 14, DMA Controller (DMACAa).
The S12ADI and S12GBADI interrupts are output according to the settings of scan mode and double trigger mode as shown in Table 30.12.

Table 30.12 Relationship between Mode Setting and S12ADI Interrupt Output

Scan Mode	Double Trigger Mode (DBLE)	Trigger	S12ADI Interrupt (ADIE = 1)	S12GBADI Interrupt (GBADIE = 1)
Single scan mode	DBLE = 0	Software trigger	Output on completion of each scan	Not output (group B scan is disabled)
		Synchronous trigger	Output on completion of each scan	Not output (group B scan is disabled)
		Asynchronous trigger	Output on completion of each scan	Not output (group B scan is disabled)
	DBLE = 1	Software trigger (setting prohibited)*1	—	—
		Synchronous trigger	Output on completion of each even-order scan	Not output (group B scan is disabled)
		Asynchronous trigger (setting prohibited)*1	—	—
Continuous scan mode	Setting prohibited	Software trigger	Output on completion of each scan	Not output (group B scan is disabled)
		Synchronous trigger	Output on completion of each scan	Not output (group B scan is disabled)
		Asynchronous trigger	Output on completion of each scan	Not output (group B scan is disabled)
Group scan mode	DBLE = 0	Software trigger (setting prohibited)*1	—	—
		Synchronous trigger	Output on completion of each scan for group A	Output on completion of each scan for group B
		Asynchronous trigger (setting prohibited)*1	—	—
	DBLE = 1	Software trigger (setting prohibited)*1	—	—
		Synchronous trigger	Output on completion of each even-order scan for group A	Output on completion of each scan for group B
		Asynchronous trigger (setting prohibited)*1	—	—

Note 1. Setting a software trigger and asynchronous trigger in double trigger mode is prohibited.
Setting a software trigger and asynchronous trigger in group scan mode is also prohibited.

30.4.2 Scan End Event Output to ELC

The event link controller (ELC) enables a link operation with the modules specified in advance using S12ADI interrupt request signal as an event signal. The S12GBADI and S12CMPI interrupt request signals cannot be used as an event signal. An event signal can be output regardless of the settings of the corresponding interrupt request enable bits. The 12-bit A/D converter outputs an A/D conversion end event.

30.5 Usage Notes

30.5.1 Notes on Reading Data Registers

The A/D data registers, A/D data duplication registers, A/D data duplication register A, A/D data duplication register B, A/D temperature sensor data register, and A/D self-diagnosis data register must be read in word units. If a register is read twice in byte units, that is, the upper byte and lower byte are separately read, the A/D-converted value having been read first may disagree with the A/D-converted value having been read for the second time.

30.5.2 Notes on Stopping A/D Conversion

To select an asynchronous trigger or a synchronous trigger as the condition for starting A/D conversion and stop A/D conversion, follow the procedure in Figure 30.31.

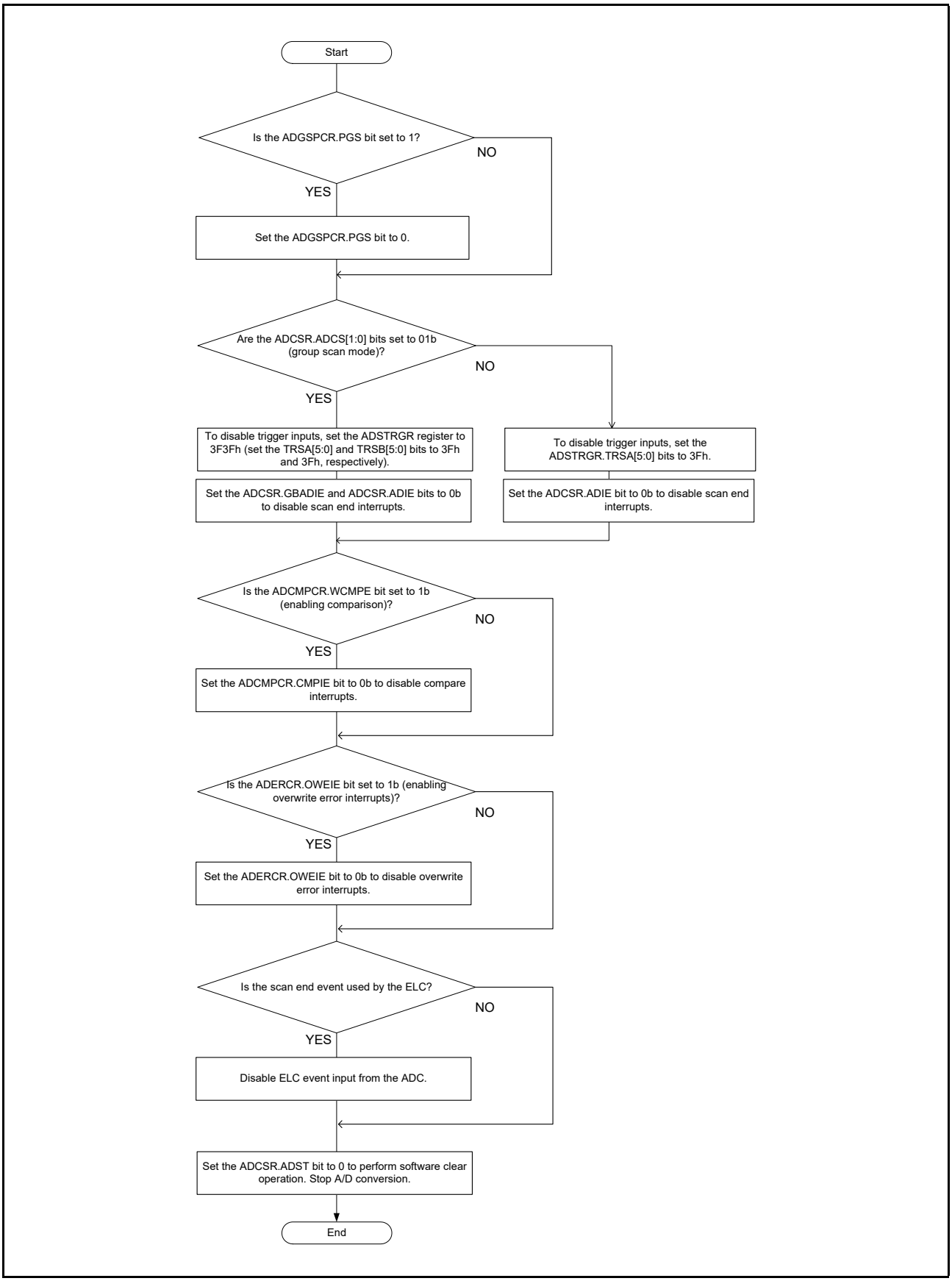


Figure 30.31 Procedures for Clear Operation by Software through the ADCSR.ADST Bit

30.5.3 A/D Conversion Restarting Timing and Termination Timing

It takes a maximum of six ADCLK cycles until the idle analog unit of the 12-bit A/D converter is restarted after the ADST bit in ADCSR is set to 1. It takes a maximum of two ADCLK cycles until the operating analog unit of the 12-bit A/D converter is terminated if the ADST bit in ADCSR is set to 0.

30.5.4 Notes on Scan End Interrupt Handling

When scanning the same analog input twice using any trigger, if the CPU does not complete reading out the A/D-converted data by the time the A/D conversion of the first analog input for the second scan ends after the first scan end interrupt is generated, the first A/D-converted data is overwritten with the second A/D-converted data and an overwrite error occurs.

30.5.5 Module Stop Function Setting

Operation of the 12-bit A/D converter can be disabled or enabled using the module stop control register C (MSTPCRC). The initial setting is for operation of the 12-bit A/D converter to be halted. Releasing the module-stop state enables access to the registers. After release from the module-stop state, wait for at least 1µs before starting A/D conversion. For details, see section 9, Low-Power Consumption Function.

30.5.6 Notes on Entering Low-Power Consumption States

Before the transition to the module-stop state or software standby mode, make sure to stop A/D conversion. Here, set the ADST bit in ADCSR to 0, and secure certain period of time until the analog unit of the 12-bit A/D converter is stopped. Follow the procedure given below to secure this time.
Follow the procedure for clear operation by software through the ADCSR.ADST bit, shown in Figure 30.31. Then, after two clock cycles of ADCLK pass, place the 12-bit A/D converter in the module-stop state or software standby mode.
To place the 12-bit A/D converter in standby mode, set the MSTPCRC.MSTPCRC4 bit (unit 1) or the MSTPCRC.MSTPCRC5 bit (unit 0) to 1.

30.5.7 Error in Absolute Accuracy When Disconnection Detection Assistance is in Use

Using disconnection detection assistance leads to an error in absolute accuracy of the A/D converter. This is because an error voltage is input to the analog input pins due to the resistive voltage division between the pull-up or pull-down resistor (Rp) and the resistance of the signal source (Rs). This error in absolute accuracy is calculated from the following formula. Only use disconnection detection assistance after thorough evaluation.

Maximum error in absolute accuracy (LSB) = 4095 × Rs / Rp

30.5.8 Caution When Using Disconnection Detection Assistance

Disable the disconnection detection assistance function when pin-level self-diagnosis is executed.

30.5.9 Caution When Using Self-Diagnosis

When the channel-dedicated sample-and-hold circuit is to be used for simultaneous sampling of four channels (AN000 to AN003) and self-diagnosis is to be enabled, set the sampling time tSPL to 0.4 µs by using ADSSTR0.SST[7:0] to satisfy the holding characteristics of sample-and-hold circuits, 3.2 µs (max.).

30.5.10 Setting to Restart Conversion by Group B in Group Scan Mode (when Group A is Given Priority)

To enable setting for restarting group B (i.e., PGS = 1 and GBRSCN = 1) in group scan mode (when group A is given priority), set the ratio between the frequency divisors for PCLKH and ADCLK to 1:1. Conversion by group B cannot be restarted in group scan mode (when group A is given priority) unless the frequency division settings are in accord with this.

30.5.11 Allowable Signal Source Impedance

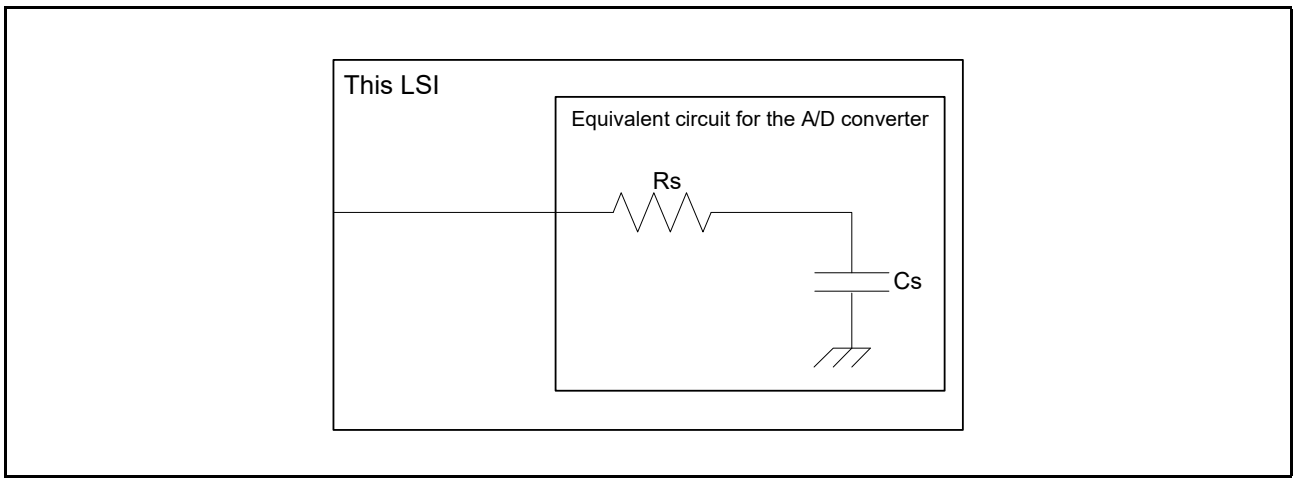


Figure 30.32 Internal Equivalent Circuit for Analog Input Pin

Table 30.13 Analog Pin Specifications

Item		Min.	Typ.	Max.	Unit
Equivalent circuit in ADC unit 0	Rs	—	1.5	—	kΩ
	Cs	—	8	—	pF
Equivalent circuit in ADC unit 1	Rs	—	3.0	—	kΩ
	Cs	—	16	—	pF

Note: The capacity of an actual product pin is calculated by adding the pin capacity shown in section 34, Electrical Characteristics, to the capacity of the internal equivalent circuit.

31. Temperature Sensor

31.1 Overview

This LSI includes a temperature sensor. The temperature sensor outputs a voltage which varies with the temperature. The 12-bit A/D converter (unit 0) can convert the voltage from the sensor into a digital value. The user can then obtain the temperature around this LSI by converting the value into the temperature.
Table 31.1 lists the specifications of the temperature sensor, and Figure 31.1 shows a block diagram of the temperature sensor.

Table 31.1 Specifications of Temperature Sensor

Item	Description
Temperature sensor voltage output	Temperature sensor outputs a voltage to the 12-bit A/D converter (unit 0).
Low-power consumption function	The module-stop state is selectable.

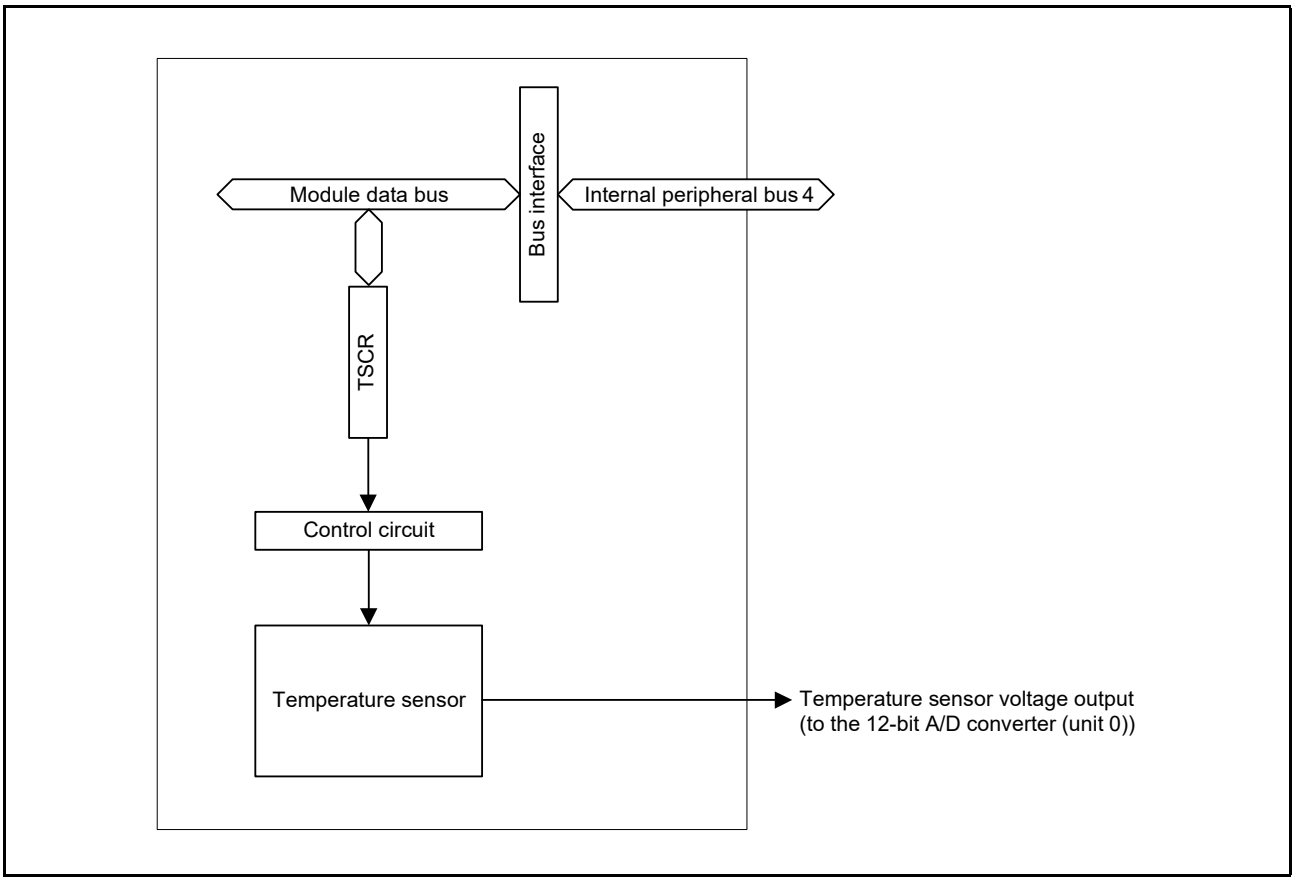


Figure 31.1 Block Diagram of Temperature Sensor

31.2 Register Descriptions

31.2.1 Temperature Sensor Control Register (TSCR)

Address(es): A008 0A00h

	b7	b6	b5	b4	b3	b2	b1	b0
	TSEN	—	—	TSOE	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b3 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b4	TSOE	Temperature Sensor Output Enable	0: Disables output from the temperature sensor to the 12-bit A/D converter (unit 0). 1: Enables output from the temperature sensor to the 12-bit A/D converter (unit 0).	R/W
b6, b5	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b7	TSEN	Temperature Sensor Enable	0: Stops the temperature sensor. 1: Starts the temperature sensor.	R/W

The settings of TSCR register have the timing restrictions shown in Figure 31.3.

31.3 Using the Temperature Sensor

The temperature sensor outputs a voltage which varies with the temperature.
This voltage is converted to a digital value by the 12-bit A/D converter (unit 0). The user can then obtain the temperature around this LSI by converting the value into the temperature.

31.3.1 Preparation for Using the Temperature Sensor

The voltage output by the temperature sensor is proportional to temperature, which can be calculated according to the following formula.

Formula for the temperature characteristic:
T = (Vs – V1)/Slope + T1

T: Measured temperature (°C)
Vs: Voltage output by the temperature sensor at the time of temperature measurement (V)
T1: Temperature experimentally measured at one point (°C)
V1: Voltage output by the temperature sensor at the time of measurement of T1 (V)
T2: Temperature at the experimental measurement of another point (°C)
V2: Voltage output by the temperature sensor at the time of measurement of T2 (V)
Slope: Temperature gradient by the temperature sensor (V/°C); slope = (V2 – V1)/(T2 – T1)

Characteristics vary from sensor to sensor. Therefore, the following experimental measurement at two different temperatures is recommended.
Use the 12-bit A/D converter (unit 0) to measure the voltage V1 output by the temperature sensor at temperature T1. Again, using the 12-bit A/D converter (unit 0), measure the voltage V2 output by the temperature sensor at a different temperature T2. Obtain the temperature gradient (slope = (V2 – V1)/(T2 – T1)) from these results.
Subsequently, obtain temperatures by substituting the slope into the formula for the temperature characteristic (T = (Vs – V1)/slope + T1).

If you are using the temperature gradient given in section 34.6, Temperature Sensor Characteristics, use the 12-bit A/D converter (unit 0) to measure the voltage V1 output by the temperature sensor at temperature T1, and then calculate the temperature characteristic by using the formula below.
However, this method gives less accurate temperatures than measurement at two points.

T = (Vs – V1)/Slope + T1

T: Measured temperature (°C)
Vs: Voltage output by the temperature sensor at the time of temperature measurement (V)
T1: Temperature experimentally measured at one point (°C)
V1: Voltage output by the temperature sensor at the time of measurement of T1 (V)
Slope: Temperature gradient (V/°C) given in section 34.6, Temperature Sensor Characteristics.

- Calculation of the voltage output by the temperature sensor (when AD-converted value addition mode is not selected)
  The voltage output by the temperature sensor (V) can be calculated from the following formula.

$$\text{Output voltage (V)} = \text{AVREFH0 voltage (V)} \times \frac{\text{ADTSDR register value}}{2^n}$$

n: The precision of the AD conversion that is specified by the ADCER.ADPRC[1:0] bits (n = 8, 10, 12).

Remark: When the AD-converted value addition mode is selected, a value corresponding to the number of additions specified by the ADADC register is stored in the ADTSDR register. This requires another calculation of the value calculated from the above formula, i.e., dividing it the specified number of additions.

31.3.2 Setting of 12-Bit A/D Converter (Unit 0)

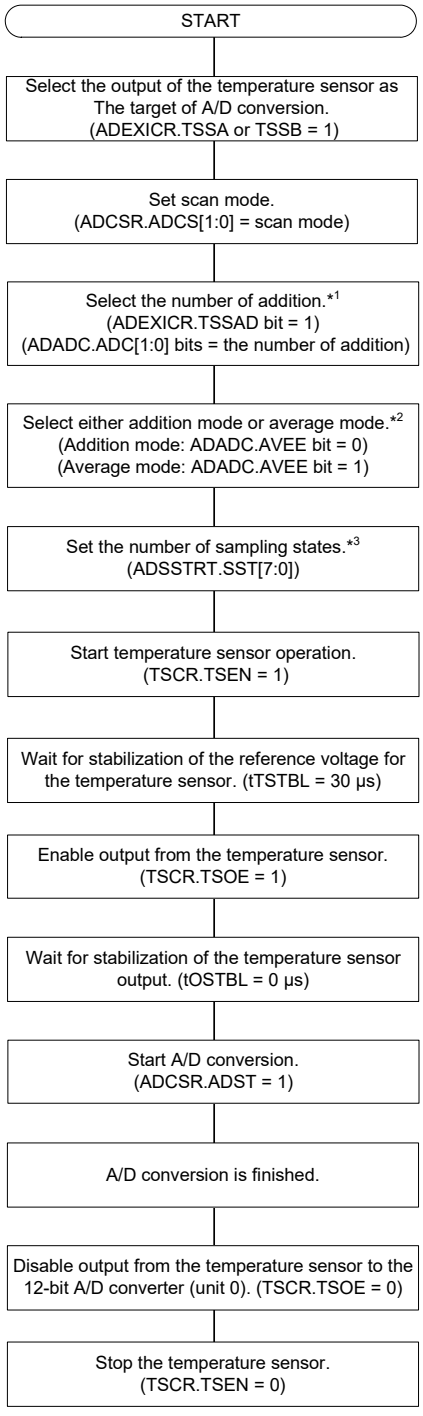
For A/D conversion of temperature sensor output voltages, 12-bit A/D converter (unit 0) registers should be set as follows.

- Setting the Temperature Sensor Voltage as an A/D Conversion Target
  Select A/D conversion of the voltage from the temperature sensor by setting the temperature sensor output A/D conversion select bit in the A/D conversion extended input control register (ADEXICR.TSSA or TSSB) to 1.
- Setting Scan Mode
  Select scan mode by setting the scan mode select bits in the A/D control register (ADCSR.ADCS[1:0]).
- Setting Addition/Average Mode
  For A/D conversion of the temperature sensor output, additional or average mode is selectable. To use either additional or average mode, set the temperature sensor output A/D converted value addition mode select bit in the A/D conversion extended input control register (ADEXICR.TSSAD) to 1, and the addition count select bits in the A/D converted value addition count select register (ADADC.ADC[1:0]) to the desired number of addition. Furthermore, clear the AVEE bit in ADADC to 0 to select addition mode; set the AVEE bit in ADADC to 1 to select average mode. In average mode, however, the ADC[1:0] bits in ADADC should not be set to 10b.
- Setting the Number of Sampling States of the 12-bit A/D converter (unit 0)
  The number of states for sampling of the output of the temperature sensor for A/D conversion is selectable. The initial setting is 11 states. To change the number of states for sampling from 11 states, set the sampling time setting bits in A/D sampling state register T (ADSSTRT.SST[7:0]), when the ADST bit in ADCSR is 0.

Setting the A/D conversion start bit in the A/D control register (ADCSR.ADST) to 1 starts A/D conversion, and the result is stored in the A/D temperature sensor data register (ADTSDR). If you will be using A/D conversion of the output from the temperature sensor, do so in accord with section 31.3.3, Procedure for Using the Temperature Sensor.

31.3.3 Procedure for Using the Temperature Sensor

Figure 31.2 shows the procedure for using the temperature sensor.



Note 1. Unless addition/average mode is set, this setting is not required. To select average mode, do not set the ADC[1:0] bits to 10b.
Note 2. Unless addition/average mode is set, this setting is not required.
Note 3. The initial value for the number of states over which sampling proceeds is 11. If the number of states is not to be changed, this setting is not required.

Figure 31.2 Procedure for Using the Temperature Sensor

31.3.4 Timing of A/D Conversion of Temperature Sensor Output

Figure 31.3 shows the timing from the start of temperature-sensor operation until the completion of A/D conversion when only the output from the temperature sensor is to be A/D converted and conversion is in single-scan mode. The times shown in the figure are described in Table 31.2.

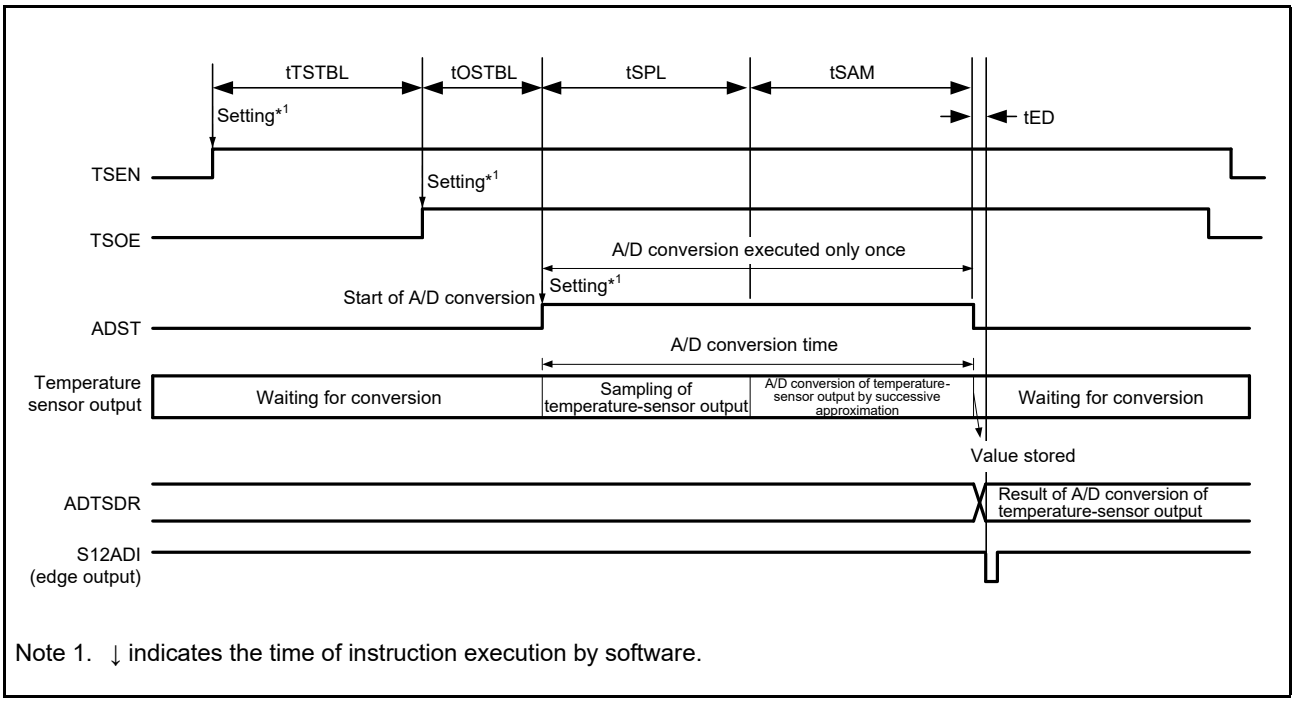


Figure 31.3 Timing from the Start of Temperature-Sensor Operation until Completion of A/D Conversion

Table 31.2 Time until Completion of A/D Conversion after the Start of Temperature-Sensor Operation

Item	Symbol	Time
Waiting time for temperature-sensor reference-voltage stabilization	tTSTBL	30 µs (min)
Waiting time for temperature-sensor output stabilization	tOSTBL	0 µs (min)
12-bit A/D converter (unit 0) input sampling time	tSPL	4.25 µs min. ADSSTRT setting × ADCLK cycles
Time for A/D conversion by successive approximation	tSAM	12-bit conversion accuracy: 13 ADCLK 10-bit conversion accuracy: 11 ADCLK 8-bit conversion accuracy: 9 ADCLK
Scan conversion end delay time	tED	1 PCLKH + 3 ADCLK

31.4 Usage Note

31.4.1 Module-Stop Function Setting

The corresponding bit in module stop control register C (MSTPCRC) can be used to enable and disable the temperature sensor. The initial setting is for the temperature sensor to be stopped. The register becomes accessible on release from the module-stop state. For details, see section 9, Low-Power Consumption Function.

32. Data Operation Circuit (DOC)

32.1 Overview

The data operation circuit (DOC) is used to compare, add, and subtract 16-bit data.
Table 32.1 lists the data operation circuit specifications and Figure 32.1 shows a block diagram of the data operation circuit.

- 16-bit data is compared and an interrupt can be generated when a selected condition applies.
- 16-bit data can be added.
- 16-bit data can be subtracted.

Table 32.1 DOC Specifications

Item	Description
Data operation function	16-bit data comparison, addition, and subtraction
Lower power consumption function	Module-stop state can be set.
Interrupts	An interrupt occurs at the following timings: • The compared values either match or mismatch • The result of data addition is greater than FFFFh • The result of data subtraction is less than 0000h
Event link function (event signal output)	An interrupt occurs at the following timings: • The compared values either match or mismatch • The result of data addition is greater than FFFFh • The result of data subtraction is less than 0000h

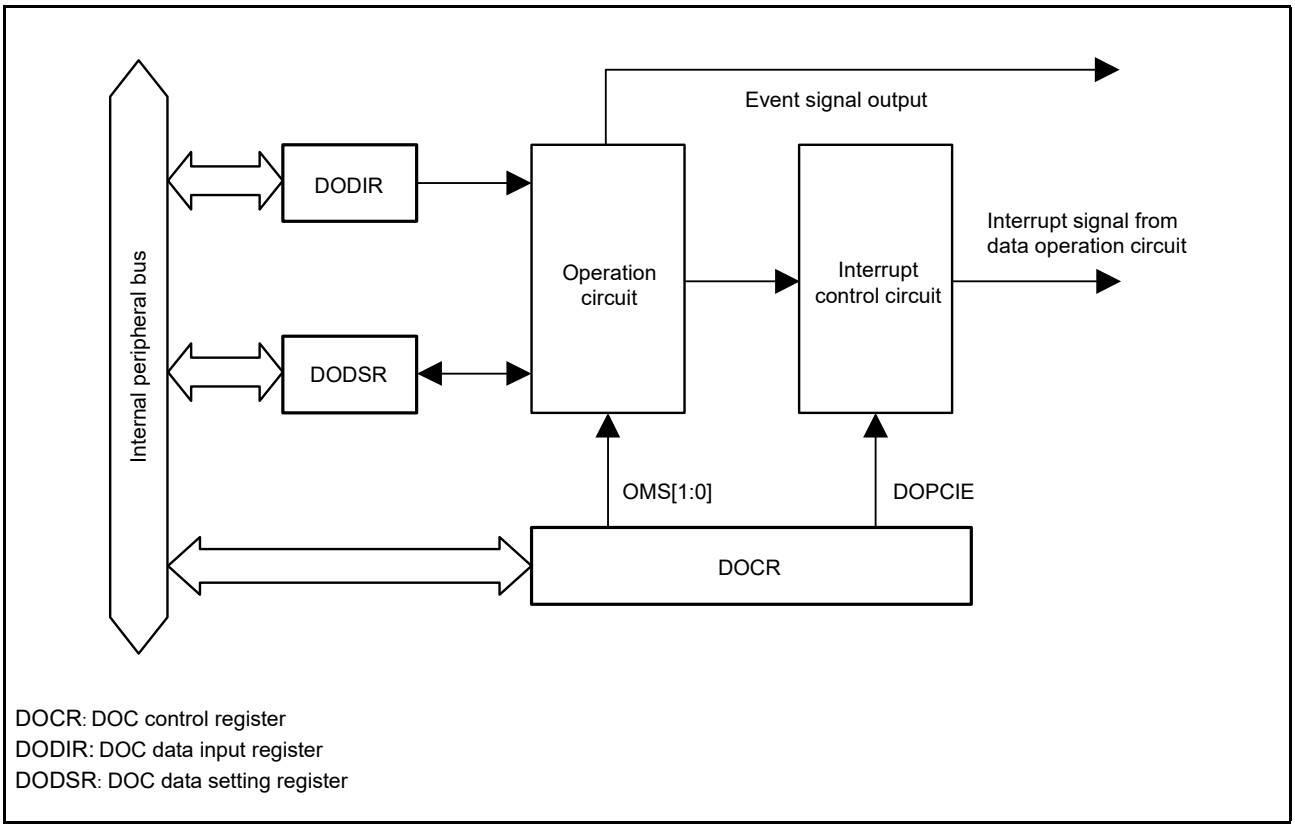


Figure 32.1 Block Diagram of Data Operation Circuit (DOC)

32.2 Register Descriptions

32.2.1 DOC Control Register (DOCR)

The DOCR register controls the DOC.

Address(es): A008 1200h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	DOPCFCL	DOPCF	DOPCIE	—	DCSEL	OMS[1:0]	
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	OMS[1:0]	Operating Mode Select	Selects an operating mode of the data operation circuit. b1 b0 0 0: Data comparison mode 0 1: Data addition mode 1 0: Data subtraction mode 1 1: Setting prohibited	R/W
b2	DCSEL	Detection Condition Select	Selects a condition for detecting data comparison result. 0: Data mismatch is detected. 1: Data match is detected. Note: Valid only in the data comparison mode.	R/W
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b4	DOPCIE	Data Operation Circuit Interrupt Enable	Specifies whether to enable or disable data operation circuit interrupt requests. 0: Disables interrupts from the data operation circuit. 1: Enables interrupts from the data operation circuit.	R/W
b5	DOPCF	Data Operation Circuit Flag	[Setting conditions] When any of the following is met: • The condition selected by the DCSEL bit is met. • A result of data addition is greater than FFFFh. • A result of data subtraction is less than 0000h. [Clearing condition] Writing 1 to the DOPCFCL bit	R
b6	DOPCFCL	DOPCF Clear	Setting this bit to 1 clears the DOPCF flag of this register. This bit is always read as 0. Writing 0 to this bit has no effect. Only 1 can be written to it.	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

32.2.2 DOC Data Input Register (DODIR)

The DODIR register is a 16-bit readable/writable register in which 16-bit data for use in operations are stored.

Address(es): A008 1202h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

32.2.3 DOC Data Setting Register (DODSR)

The DODSR register is a 16-bit readable/writable register. This register stores 16-bit data for use as a reference in data comparison mode. This register also stores the results of operations in data addition and data subtraction modes.

Address(es): A008 1204h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

32.3 Operation

32.3.1 Data Comparison Mode

Figure 32.2 shows an example of the steps involved in data comparison mode operation by the data operation circuit. The following is an example of operation when DCSEL is set to 0 (data mismatch is detected as a result of data comparison).

(1) Writing 00b to the DOCR.OMS[1:0] bits selects data comparison mode.
(2) The 16-bit reference data is set in DODSR.
(3) 16-bit data for comparison is written to DODIR.
(4) Writing of 16-bit data continues until all data for comparison have been written to DODIR.
(5) If a value written to DODIR does not match that in DODSR, the DOCR.DOPCF flag is set to 1. When the DOCR.DOPCIE bit is 1, a data operation circuit interrupt is also generated.

Note: When the DODIR register has the value after reset (0000h), the DOCR.DOPCF flag does not change to 1 if a value other than 0000h is set in the DODSR register. To perform data comparison, set the DODSR register and write a value to the DODIR register. If the DODSR register is set again after data comparison, the value remaining in the DODIR register will not be compared again with the value set in the DODSR register. That is, comparison proceeds in response to writing to the DODIR register but not in response to writing to the DODSR register.

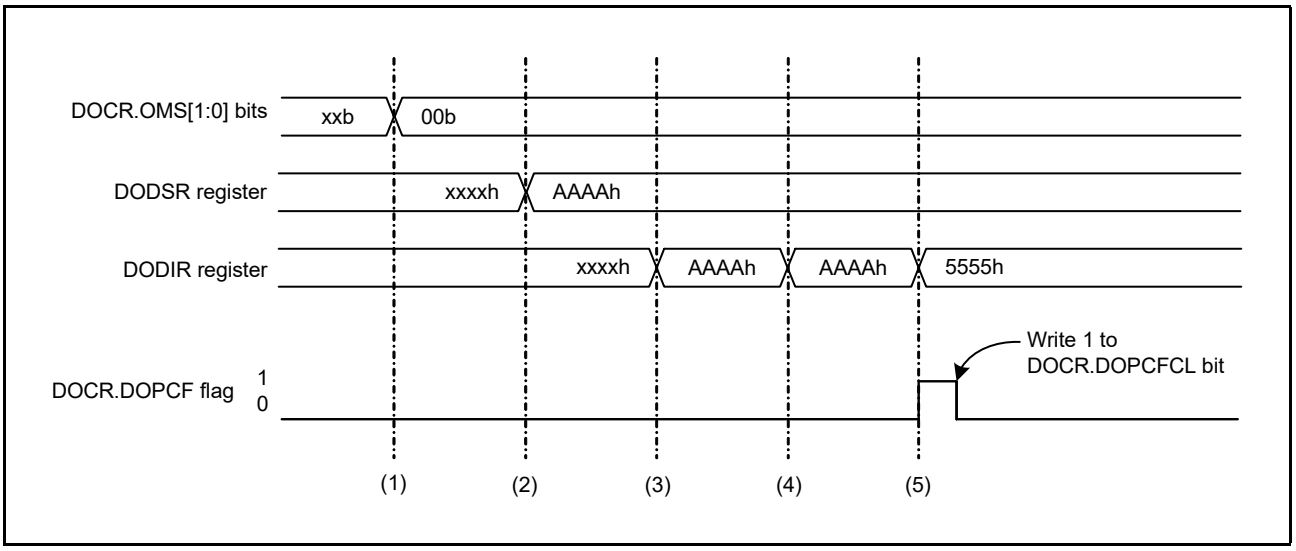


Figure 32.2 Example of Operation in Data Comparison Mode

32.3.2 Data Addition Mode

Figure 32.3 shows an example of the steps involved in data addition mode operation by the data operation circuit.

(1) Writing 01b to the DOCR.OMS[1:0] bits selects data addition mode.
(2) 16-bit data is set in the DODSR register as the value after reset.
(3) 16-bit data to be added is written to DODIR. The result of the operation is stored in DODSR.
(4) Writing of 16-bit data continues until all data for addition have been written to DODIR.
(5) If the result of an operation is greater than FFFFh, the DOCR.DOPCF flag is set to 1. When the DOCR.DOPCIE bit is 1, a data operation circuit interrupt is also generated.

Note: When the DODIR register has the value after reset (0000h), data addition is not performed if a value other than 0000h is set in the DODSR register. To perform data addition, set the DODSR register and write a value to be added into the DODIR register. If the DODSR register is set again after data addition, the value remaining in the DODIR register will not be added again to the value set in the DODSR register. That is, addition proceeds in response to writing to the DODIR register but not in response to writing to the DODSR register.

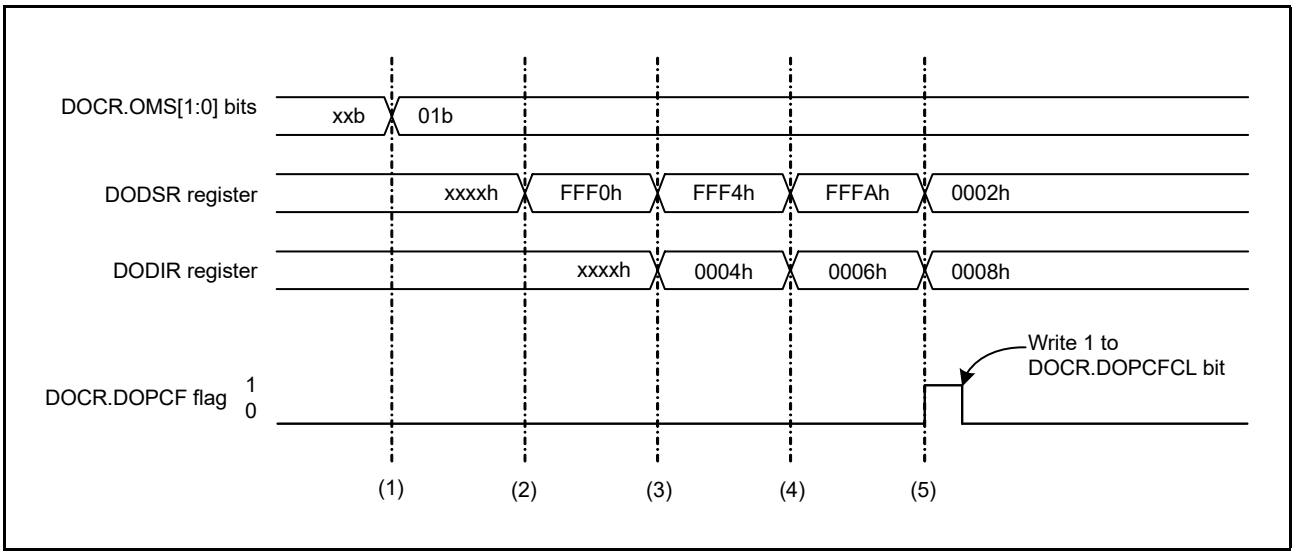


Figure 32.3 Example of Operation in Data Addition Mode

32.3.3 Data Subtraction Mode

Figure 32.4 shows an example of the steps involved in data subtraction mode operation by the data operation circuit.

(1) Writing 10b to the DOCR.OMS[1:0] bits selects data subtraction mode.
(2) 16-bit data is set in the DODSR register as the value after reset.
(3) 16-bit data to be subtracted is written to DODIR. The result of the operation is stored in DODSR.
(4) Writing of 16-bit data continues until all data for subtraction have been written to DODIR.
(5) If the result of an operation is less than 0000h, the DOCR.DOPCF flag is set to 1. When the DOCR.DOPCIE bit is 1, a data operation circuit interrupt is also generated.

Note: When the DODIR register has the value after reset (0000h), data subtraction is not performed if a value other than 0000h is set in the DODSR register. To perform data subtraction, set the DODSR register and write a value to be subtracted into the DODIR register. If the DODSR register is set again after data subtraction, the value remaining in the DODIR register will not be subtracted again from the value set in the DODSR register. That is, subtraction proceeds in response to writing to the DODIR register but not in response to writing to the DODSR register.

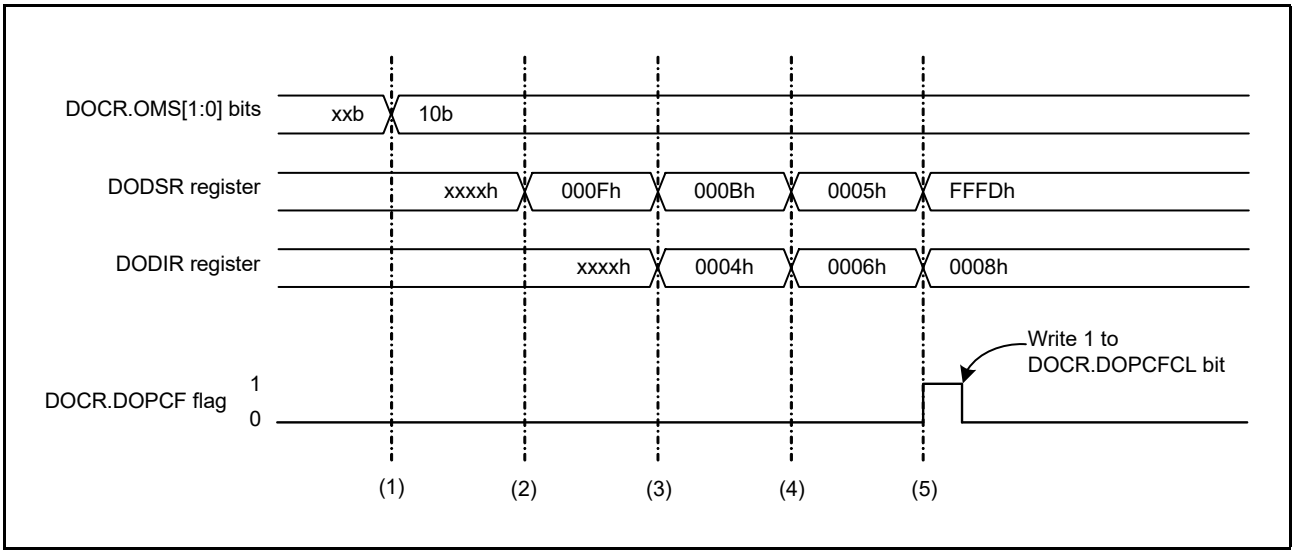


Figure 32.4 Example of Operation in Data Subtraction Mode

32.4 Interrupt Requests

The data operation circuit generates the data operation circuit interrupt as an interrupt request. When an interrupt source is generated, the data operation circuit flag (DOCR.DOPCF) corresponding to the interrupt is set to 1. Table 32.2 describes the interrupt request.

Table 32.2 Interrupt Request from Data Operation Circuit

Interrupt Request	Data Operation Circuit Flag	Interrupt Generation Timing
Data operation circuit interrupt	DOPCF	• The result of data addition is greater than FFFFh • The result of data subtraction is less than 0000h • The compared values either match or mismatch

Note: The data operation circuit interrupt is not conveyed to the interrupt controller, but it is conveyed to the error control module (ECM) as a DOC operation error.

32.5 Event Link Output

The DOC outputs event signals for the event link controller (ELC) under the following conditions, and these can be used to initiate operations by other modules selected in advance.

- The compared values either match or mismatch
- The result of data addition is greater than FFFFh
- The result of data subtraction is less than 0000h

32.5.1 Interrupt Handling and Event Linking

The data operation circuit (DOC) has a bit (DOCR.DOPCIE) to enable or disable interrupts. An interrupt request signal is output for the CPU when an interrupt source is generated while the corresponding enable bit is enabled.
In contrast, an event link output signal is sent to other modules as an event signal via the ELC when an interrupt source is generated, regardless of the setting of the corresponding interrupt enable bit.

32.6 Usage Note

32.6.1 Module Stop Function Setting

Operation of the data operation circuit can be disabled or enabled using module stop control register C (MSTPCRC). The value after reset indicates that the data operation circuit is in the stop state. Register access is enabled by canceling the module-stop state. For details, see section 9, Low-Power Consumption Function.

33. RAM (Product Option)

This LSI has on-chip high-speed RAM (with ECC error correction). The RAM, whose capacity is 1 Mbyte (512 Kbytes × 2), is installed.

33.1 Overview

Table 33.1 lists the specifications of the RAM.

Table 33.1 Specifications of RAM

Item	Description
RAM capacity	1 Mbyte
RAM addresses (for access from the Cortex-R4)	On-chip extended SRAM (area 1) 0400 0000h to 0407 FFFFh 2400 0000h to 2407 FFFFh (mirror) On-chip extended SRAM (area 2) 2000 0000h to 2007 FFFFh 2200 0000h to 2207 FFFFh (mirror)
Low-power consumption function	Operate only during access
Error checking	1-bit error correction, 2-bit error detection: The ECC decoder must be enabled to use these functions.
Initializing function	All areas of RAM are initialized to zero by a reset.

Note: The RAM is reset by the sources from the RES# pin input, the ECM, and software.

33.2 Register Descriptions

33.2.1 Protect Command Register (RAMPCMD)

The RAMPCMD register is used to provide write protection for registers that might seriously affect the system in order to prevent the application system from inadvertently stopping due to, for example, runaway of a program. Writing to the protected registers is disabled unless the PROTREL bit is set to 1.

Address(es): A00F 3000h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PROTREL
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	PROTREL	Enable Write Access to Protected Registers	1: Write access is enabled. 0: Write access is disabled (protected status).	R/W

To set the PROTREL bit of the RAMPCMD register to 1, the sequence shown below must be used. No special sequence is required to clear this bit to 0 or to read the register.

1. Write 0000 00A5h to the RAMPCMD register as a specific value.
2. Write 0000 0001h to the RAMPCMD register.
3. Write 0000 FFFEh to the RAMPCMD register.
4. Write 0000 0001h to the RAMPCMD register.

Note 1. In steps 1, 2, and 3, nothing is written to the register.
Note 2. Be sure to clear the PROTREL bit to 0 after writing to target registers is completed.

Table 33.2 Write-Protection Target Registers

Register Name	Symbol	R/W
Protect command register	RAMPCMD	R/W
ECC decoder configuration register	RAMEDC	R/W
ECC encoder configuration register	RAMEEC	R/W

33.2.2 ECC Decoder Configuration Register (RAMEDC)

The RAMEDC register controls the ECC decoders for the on-chip extended SRAM (areas 1 and 2).

Address(es): A00F 3100h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ECC_ENABLE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECC_ENABLE	ECC Decoder Enable	0: ECC decoder is disabled. 1: ECC decoder is enabled.	R/W
b31 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

When the RAMEDC.ECC_ENABLE bit is set to 1, ECC decoder logic for the on-chip extended SRAM (areas 1 and 2) is enabled, and the following functions are enabled, and this is conveyed to the error control module (ECM).

- 1-bit ECC error: Corrects the read data and conveys the error to the error control module (ECM).
- 2-bit ECC error: Conveys the error to the error control module (ECM).

When the RAMEDC.ECC_ENABLE bit is set to 0 (the “disabled” setting), even if an ECC error occurs, read data is not corrected or the error signal is not conveyed to the ECM. Therefore, retention of the status in the RAMDBEST register, capturing an error address in the RAMDBEAD register, and error count in the RAMDBECNT register are disabled.

Note 1. Switch this register while no masters are accessing the RAM.

Note 2. Writing to this register is disabled unless the write-protection is canceled by the RAMPCMD register.

33.2.3 ECC Encoder Configuration Register (RAMEEC)

The RAMEEC register controls the self-test for the ECC circuit of the on-chip extended SRAM (areas 1 and 2).
If the RAMEEC.DBE_DISTn (n = 0 to 15) bit is set to 1, the Syndrome value (ECC redundancy bit data) is latched when the RAM corresponding to each bit is accessed. Then, when the RAM is accessed the next time, the latched Syndrome value is written to the RAM to inject an ECC error.
If the RAMEEC.DBE_DISTn (n = 0 to 15) bit is set to 0, the normal Syndrome value is always written to the RAM corresponding to each bit.

Address(es): A00F 3104h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	DBE_DIST15	DBE_DIST14	DBE_DIST13	DBE_DIST12	DBE_DIST11	DBE_DIST10	DBE_DIST9	DBE_DIST8	DBE_DIST7	DBE_DIST6	DBE_DIST5	DBE_DIST4	DBE_DIST3	DBE_DIST2	DBE_DIST1	DBE_DIST0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	DBE_DIST0	On-Chip Extended SRAM (area 1) BANK0 Way0 Syndrome Error Injection Enable	0: Normal Syndrome value is always written to the RAM. 1: The Syndrome value for the previous access is written to the RAM.	R/W
b1	DBE_DIST1	On-Chip Extended SRAM (area 1) BANK0 Way1 Syndrome Error Injection Enable		R/W
b2	DBE_DIST2	On-Chip Extended SRAM (area 1) BANK0 Way2 Syndrome Error Injection Enable		R/W
b3	DBE_DIST3	On-Chip Extended SRAM (area 1) BANK0 Way3 Syndrome Error Injection Enable		R/W
b4	DBE_DIST4	On-Chip Extended SRAM (area 1) BANK1 Way0 Syndrome Error Injection Enable		R/W
b5	DBE_DIST5	On-Chip Extended SRAM (area 1) BANK1 Way1 Syndrome Error Injection Enable		R/W
b6	DBE_DIST6	On-Chip Extended SRAM (area 1) BANK1 Way2 Syndrome Error Injection Enable		R/W
b7	DBE_DIST7	On-Chip Extended SRAM (area 1) BANK1 Way3 Syndrome Error Injection Enable		R/W
b8	DBE_DIST8	On-Chip Extended SRAM (area 2) BANK0 Way0 Syndrome Error Injection Enable		R/W
b9	DBE_DIST9	On-Chip Extended SRAM (area 2) BANK0 Way1 Syndrome Error Injection Enable		R/W
b10	DBE_DIST10	On-Chip Extended SRAM (area 2) BANK0 Way2 Syndrome Error Injection Enable		R/W
b11	DBE_DIST11	On-Chip Extended SRAM (area 2) BANK0 Way3 Syndrome Error Injection Enable		R/W
b12	DBE_DIST12	On-Chip Extended SRAM (area 2) BANK1 Way0 Syndrome Error Injection Enable		R/W
b13	DBE_DIST13	On-Chip Extended SRAM (area 2) BANK1 Way1 Syndrome Error Injection Enable		R/W
b14	DBE_DIST14	On-Chip Extended SRAM (area 2) BANK1 Way2 Syndrome Error Injection Enable		R/W
b15	DBE_DIST15	On-Chip Extended SRAM (area 2) BANK1 Way3 Syndrome Error Injection Enable		R/W
b31 to b16	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Switch this register while no masters are accessing the RAM.
Note 2. Writing to this register is disabled unless the write-protection is canceled by the RAMPCMD register.

33.2.4 2-Bit ECC Error Status Register (RAMDBEST)

The RAMDBEST register indicates the 2-bit ECC error status for the on-chip extended SRAM (areas 1 and 2). After the error signal is conveyed to the error control module (ECM), read this register to identify the BANK and WAY in which a 2-bit ECC error occurred.

Address(es): A00F 3108h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	DBE_RAM15	DBE_RAM14	DBE_RAM13	DBE_RAM12	DBE_RAM11	DBE_RAM10	DBE_RAM9	DBE_RAM8	DBE_RAM7	DBE_RAM6	DBE_RAM5	DBE_RAM4	DBE_RAM3	DBE_RAM2	DBE_RAM1	DBE_RAM0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	DBE_RAM0	On-Chip Extended SRAM (area 1) Bank0 Way0 2-Bit ECC Error Detection	0: No error 1: Error occurred.	R
b1	DBE_RAM1	On-Chip Extended SRAM (area 1) Bank0 Way1 2-Bit ECC Error Detection		R
b2	DBE_RAM2	On-Chip Extended SRAM (area 1) Bank0 Way2 2-Bit ECC Error Detection		R
b3	DBE_RAM3	On-Chip Extended SRAM (area 1) Bank0 Way3 2-Bit ECC Error Detection		R
b4	DBE_RAM4	On-Chip Extended SRAM (area 1) Bank1 Way0 2-Bit ECC Error Detection		R
b5	DBE_RAM5	On-Chip Extended SRAM (area 1) Bank1 Way1 2-Bit ECC Error Detection		R
b6	DBE_RAM6	On-Chip Extended SRAM (area 1) Bank1 Way2 2-Bit ECC Error Detection		R
b7	DBE_RAM7	On-Chip Extended SRAM (area 1) Bank1 Way3 2-Bit ECC Error Detection		R
b8	DBE_RAM8	On-Chip Extended SRAM (area 2) Bank0 Way0 2-Bit ECC Error Detection		R
b9	DBE_RAM9	On-Chip Extended SRAM (area 2) Bank0 Way1 2-Bit ECC Error Detection		R
b10	DBE_RAM10	On-Chip Extended SRAM (area 2) Bank0 Way2 2-Bit ECC Error Detection		R
b11	DBE_RAM11	On-Chip Extended SRAM (area 2) Bank0 Way3 2-Bit ECC Error Detection		R
b12	DBE_RAM12	On-Chip Extended SRAM (area 2) Bank1 Way0 2-Bit ECC Error Detection		R
b13	DBE_RAM13	On-Chip Extended SRAM (area 2) Bank1 Way1 2-Bit ECC Error Detection		R
b14	DBE_RAM14	On-Chip Extended SRAM (area 2) Bank1 Way2 2-Bit ECC Error Detection		R
b15	DBE_RAM15	On-Chip Extended SRAM (area 2) Bank1 Way3 2-Bit ECC Error Detection		R
b31 to b16	—	Reserved	These bits are read as 0.	R

Note 1. Reading this register clears the ECC error source.

33.2.5 2-Bit ECC Error Address Register (RAMDBEAD)

The RAMDBEAD register is a read-only register that holds the address where a 2-bit ECC error was found.
When a 2-bit ECC error is detected, the ECC error generation address is captured with the detection signal as a trigger, and then is stored in the ADDRESS[15:0] bits.
The register in which an ECC error generation address has been captured cannot retain the next ECC error generation address unless the LOCK bit of the register is enabled and the register is read. Therefore, if you want to capture a new ECC error generation address, you must first read this register.

Address(es): A00F 310Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	BANK[1:0]		ADDRESS[15:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ADDRESS[15:0]														—	LOCK
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	LOCK	Lock Enable	0: Register unlocked (2-bit ECC error generation address can be captured.) 1: Register locked (2-bit ECC error generation address cannot be captured.) Read this register to unlock the registers.	R
b1	—	Reserved	This bit is read as 0.	R
b17 to b2	ADDRESS [15:0]	2-Bit ECC Error Generation Address	The 2-bit ECC error generation address is retained.	R
b19, b18	BANK[1:0]	2-Bit ECC Error Generation BANK	These bits indicate the number of the bank where a 2-bit ECC error was encountered. 0: On-chip extended SRAM (area 1) BANK0 1: On-chip extended SRAM (area 1) BANK1 2: On-chip extended SRAM (area 2) BANK0 3: On-chip extended SRAM (area 2) BANK1	R
b31 to b20	—	Reserved	These bits are read as 0.	R

Note: If 2-bit ECC errors occur at the same time in different WAYs, the priority of captured addresses is as follows:
On-chip extended SRAM (area 1) BANK0 WAY0 > On-chip extended SRAM (area 1) BANK0 WAY1 >
On-chip extended SRAM (area 1) BANK0 WAY2 > On-chip extended SRAM (area 1) BANK0 WAY3 >
On-chip extended SRAM (area 1) BANK1 WAY0 > On-chip extended SRAM (area 1) BANK1 WAY1 >
On-chip extended SRAM (area 1) BANK1 WAY2 > On-chip extended SRAM (area 1) BANK1 WAY3 >
On-chip extended SRAM (area 2) BANK0 WAY0 > On-chip extended SRAM (area 2) BANK0 WAY1 >
On-chip extended SRAM (area 2) BANK0 WAY2 > On-chip extended SRAM (area 2) BANK0 WAY3 >
On-chip extended SRAM (area 2) BANK1 WAY0 > On-chip extended SRAM (area 2) BANK1 WAY1 >
On-chip extended SRAM (area 2) BANK1 WAY2 > On-chip extended SRAM (area 2) BANK1 WAY3

33.2.6 2-Bit ECC Error Counter Register (RAMDBECNT)

The RAMDBECNT register is a read-only register that retains the 2-bit ECC error count.
If a 2-bit ECC error is detected, the error counter is incremented with the detection signal as a trigger.
If the counter value exceeds the maximum (Fh), it is cleared to 0h.

Address(es): A00F 3110h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	ERRCOUNT[3:0]			
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b3 to b0	ERRCOUNT [3:0]	2-Bit ECC Error Counter	The 2-bit ECC error count is retained.	R
b31 to b4	—	Reserved	These bits are read as 0.	R

Note: Even if 2-bit ECC errors occur at the same time in different WAYs, the count-up value is 1.

33.3 Description of Operation

33.3.1 Configuration of Memory Map

This LSI has two 512-Kbyte RAMs as the extended internal SRAM, each having a two-bank four-way configuration.

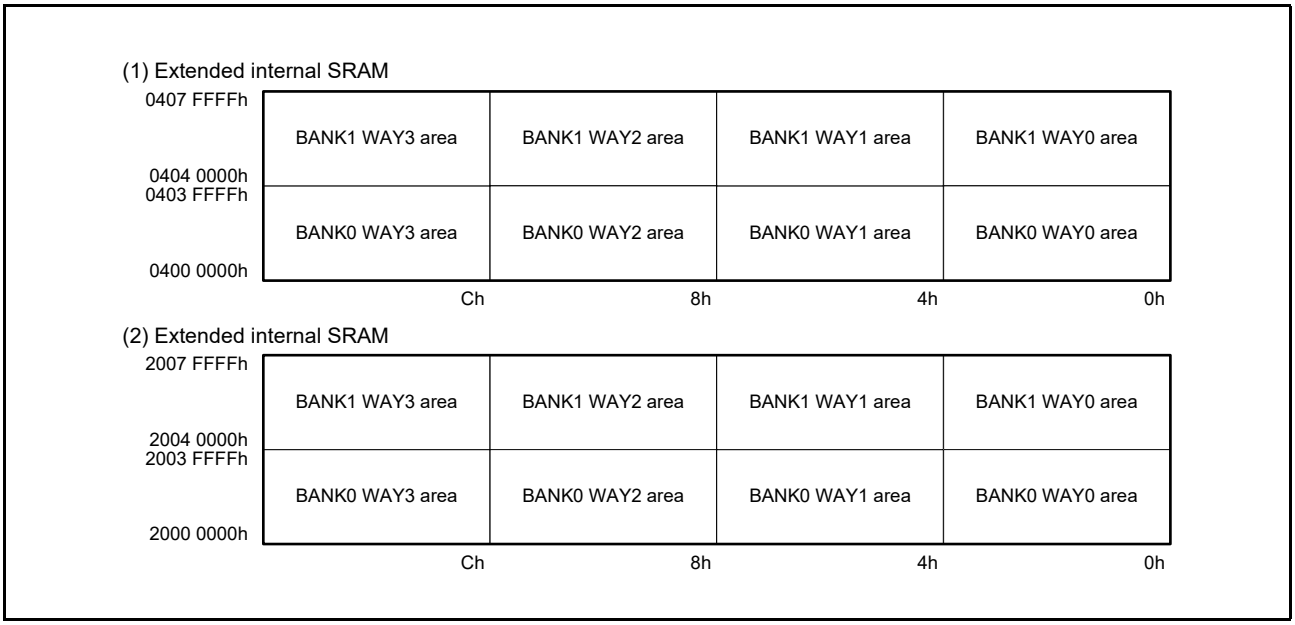


Figure 33.1 Configuration of Memory

33.3.2 ECC Error Correction Function

The RAMEDC register enables or disables ECC error correction in the 1-Mbyte space. ECCs can be used to correct 1-bit errors and detect 2-bit errors. The error control module (ECM) detects the sources of 2-bit errors, and the RAMDBEST register is used to check the way in which an error was found. Moreover, the RAMDBEAD register can be used to identify the address where a 2-bit error was found, and the RAMDBECNT indicates the number of 2-bit errors that have been encountered.

33.3.3 Self-Testing of the ECC Circuit

Self-testing of the ECC circuit proceeds in way units. The RAMEEC register sets the target area for each way. The following shows an example of the procedure for self-testing of the ECC circuit.

(1) Example of ECC error-injection setting procedure

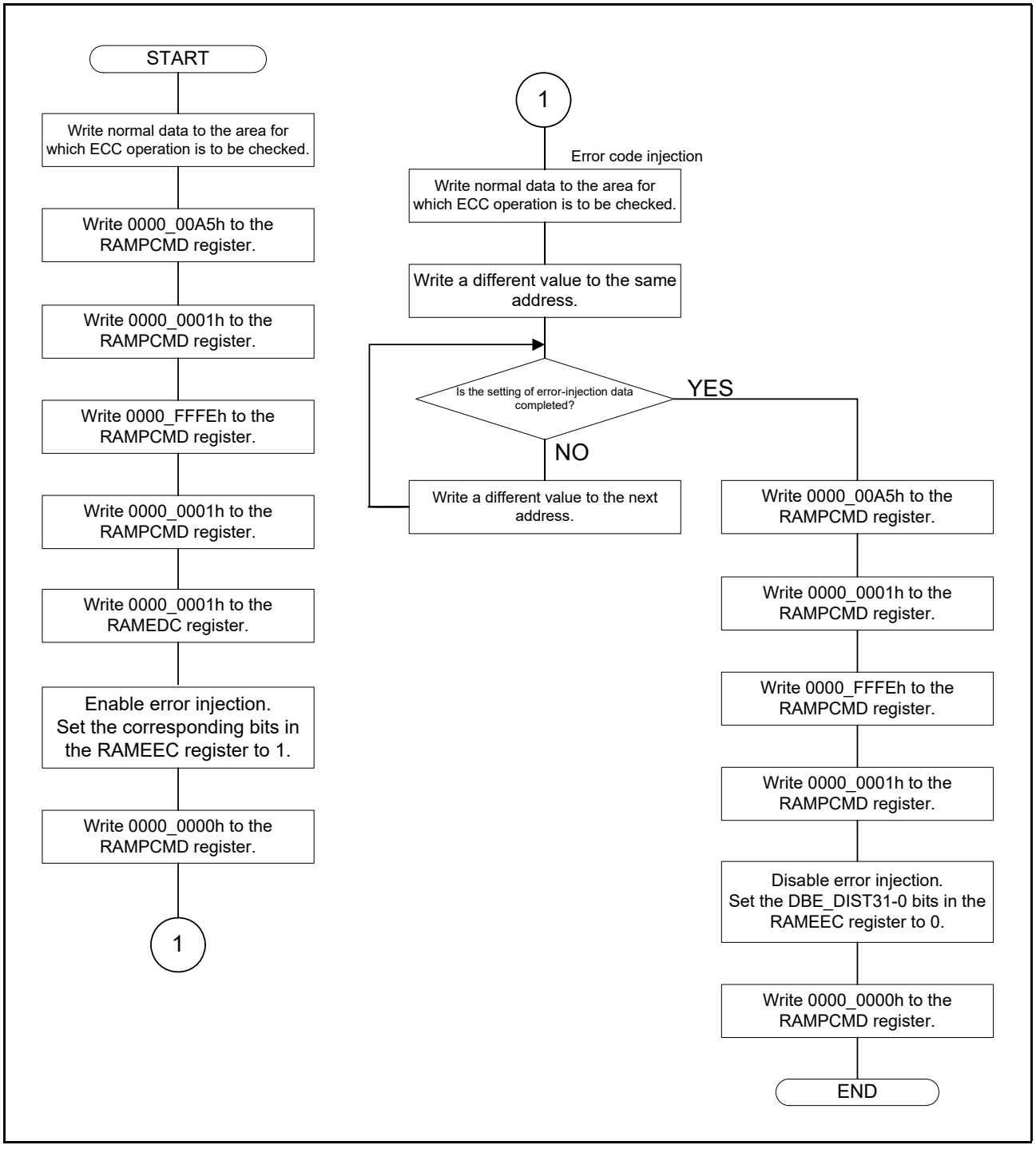
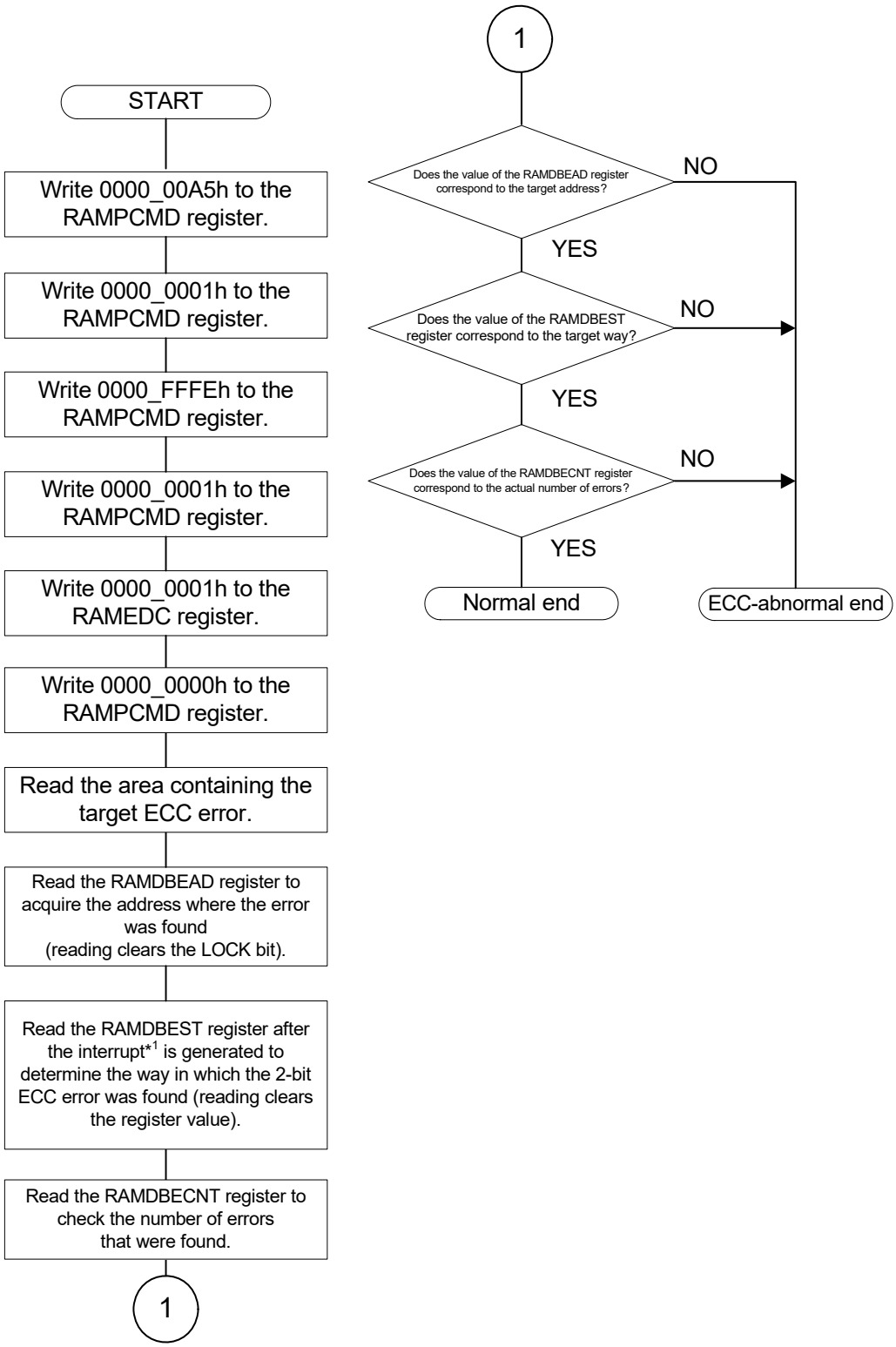


Figure 33.2 Example of ECC Error-Injection Setting Procedure

(2) Procedure for Checking ECC Operation



Note 1. An ERRD interrupt via the ECM is used to connect the interrupt signal to the VIC.

Figure 33.3 Procedure of Checking ECC Operation

34. Electrical Characteristics

34.1 Absolute Maximum Ratings

Table 34.1 Absolute Maximum Rating

Conditions: VSS = PLLVSS0 = PLLVSS1 = AVSS0 = AVSS1 = VREFL0 = VREFL1 = 0 V

Item	Symbol	Value	Unit
Power supply voltage (I/O)	VCCQ33	-0.3 to +4.2	V
Power supply voltage (1.2 V I/O)	VCCQ12	–0.3 to +1.6	V
Power supply voltage (internal)	VDD	-0.3 to +1.6	V
PLL power supply voltage	PLLVDD0, PLLVDD1	-0.3 to +1.6	V
Input voltage (except 1.2-V I/O ports and ports for 5-V tolerant*1)	V_{in1}	-0.3 to VCCQ33 + 0.3*4	V
Input voltage (ports for 5-V tolerant*1)	V_{in2}	-0.3 to +5.5*3	V
Input voltage (1.2 V I/O port)	V_{in3}	–0.3 to VCCQ12 + 0.3*5	V
Analog power supply voltage	AVCC0, AVCC1*2	-0.3 to +4.2	V
Reference power supply voltage	VREFH0, VREFH1	-0.3 to (AVCC0, AVCC1) + 0.3*4	V
Analog input voltage	V_{AN}	-0.3 to (AVCC0, AVCC1) + 0.3*4	V
Operating temperature (junction temperature)	Tj	-40 to +110	°C
Storage temperature	T_{stg}	-55 to +125	°C

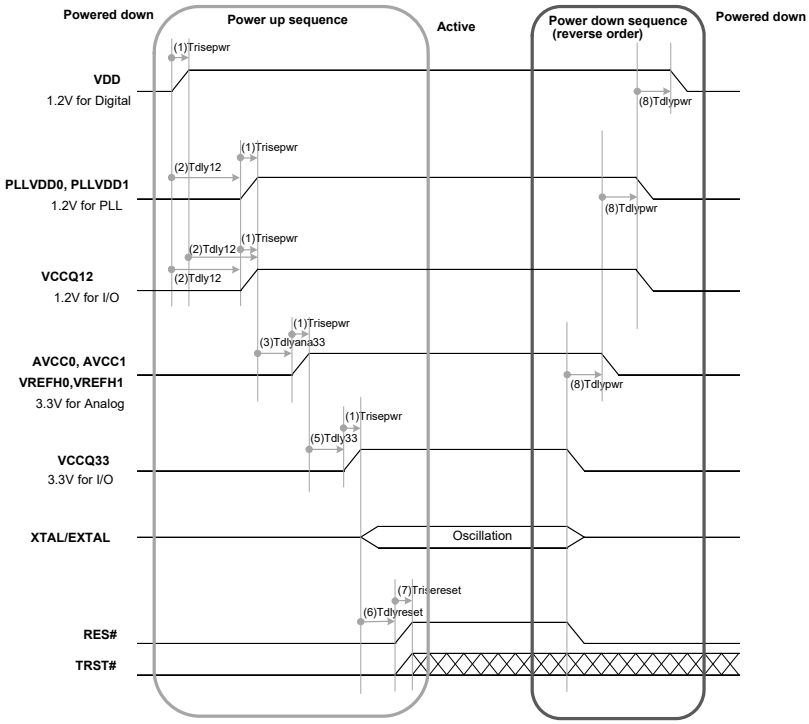
[Usage Notes]

1. Do not directly connect output pins (I/O pins in output state) of IC products to other output pins (including I/O pins in output state) , power pins, or GND pins. However, output pins are directly connectable in an external circuit where timing design is provided to avoid conflict of outputs of high-impedance pins such as I/O pins.
2. If even a single item exceeds the absolute maximum rating for even a moment, it may degrade the product's quality. In other words, the absolute maximum rating is a rated value that potentially causes physical damage to products. Use products with a margin of the absolute maximum rating.
   Specified values and conditions shown in DC characteristics and AC characteristics are the range of normal operation and quality assurance of products.

Note 1. Ports PC2, PC3, PC6, and PC7 are 5-V tolerant.
Note 2. When the A/D converter unit 0 is not to be used, connect the AVCC0 and VREFH0 pins to VCCQ33 and the AVSS0 and VREFL0 pins to VSS, respectively. Do not leave these pins open. In the same way, when the A/D converter unit 1 is not to be used, connect the AVCC1 and VREFH1 pins to VCCQ33 and the AVSS1 and VREFL1 pins to VSS, respectively. Do not leave these pins open.
Note 3. When VCCQ33 is less than 3.0 V, the rated value of ports for 5-V tolerant is 3.6 V.
Note 4. Do not exceed the absolute maximum rating, 4.2 V.
Note 5. Do not exceed the absolute maximum rating, 1.6 V.

34.2 Power On/Off Sequence

Turn on and off each power supply voltage according to the procedure shown in the figure below.
When turning on the power, be sure to fix TRST# pins and RES# pins to the low level. Otherwise, initialization is not performed successfully.



Timing

		Value		
No.	Item	min	typ	max
(1)	Trisepwr	100 µs	—	50 ms
(2)	Tdly12	0 ms	—	100 ms
(3)	Tdlyana33	0 ms	—	100 ms
(4)	Tdly33	0 ms	—	100 ms
(5)	Tdlyreset	10 ms	—	—
(6)	Trisereset	—	—	150 µs
(7)	Tdlypwr	0 ms	—	—

Note 1. Turn on all power supply voltages and reset signals with a monotonic increase and turn off with a monotonic decrease.
Note 2. Do not apply a negative voltage to power supply voltages.
Note 3. Before turning on the power, be sure to make reset pins (TRST# and RES#) active (low). Otherwise, inputs and outputs on the pins may be unstable. If this may also be a problem when turning off the power, make reset pins (TRST# and RES#) active (low.)

Note: If the power on/off sequence is not met (out of operation guarantee range), the pin input state and output state may be undefined.

Figure 34.1 Power On/Off Sequence

34.3 DC Characteristics

- Conditions: VDD = VCCQ12 = PLLVDD0 = PLLVDD1 = 1.14 to 1.26 V,
  VCCQ33 = AVCC0 = AVCC1 =3.0 to 3.6 V
  VREFH0 = 2.5 to 3.6 V (when AVCC0 ≥ VREFH0),
  VREFH1 = 2.5 to 3.6 V (when AVCC1 ≥ VREFH1),
  VSS = PLLVSS0 = PLLVSS1 = AVSS0 = AVSS1 = VREFL0 = VREFL1 = 0 V,
  Tj = -40 to 110°C

Table 34.2 DC Characteristics (1)

Item	Symbol	min	typ	max	Unit	Test Conditions
Power supply voltage (I/O)	VCCQ33	3.0	3.3	3.6	V	
Power supply voltage (1.2V I/O)	VCCQ12	1.14	1.2	1.26	V	
Power supply voltage (internal)	VDD	1.14	1.2	1.26	V	
PLL power supply voltage	PLLVDD0, PLLVDD1	1.14	1.2	1.26	V	
Analog power supply voltage	AVCC0, AVCC1	3.0	3.3	3.6	V	

Table 34.3 DC Characteristics (2) [Power Supply]

Item	Type	Symbol	typ	max	Unit	Test Conditions
Normal operation	VDD	VIcc	160	440	mA	Tj = -40 to 110ºC R7S910020 R7S910021 R7S910120 R7S910121
			220	511	mA	Tj = -40 to 110ºC R7S910022 R7S910023 R7S910122 R7S910123
	PLLVDD0 + PLLVDD1	PLLIcc	3.2	5	mA	
	VCCQ12	V12Icc	1*1, *2	—	mA	
	VCCQ33	V33Icc	19*1, *2	—	mA	
	AVCC0	AV0Icc	2	5	mA	A/D conversion (unit 0)
	AVCC1	AV1Icc	0.7	1.5	mA	A/D conversion (unit 1)
	VREFH0	VRF0Icc	0.07	0.2	mA	A/D conversion (unit 0)
	VREFH1	VRF1Icc	0.07	0.2	mA	A/D conversion (unit 1)
Standby mode with all modules inactive (reference value)	VDD	VIcc	41	—	mA	
	PLLVDD0 + PLLVDD1	PLLIcc	3.2	—	mA	
	VCCQ12	V12Icc	0.1*1, *2	—	mA	
	VCCQ33	V33Icc	0.35*1, *2	—	mA	
	AVCC0	AV0Icc	0.64	—	µA	
	AVCC1	AV1Icc	0.32	—	µA	
	VREFH0	VRF0Icc	0.24	—	µA	
	VREFH1	VRF1Icc	0.24	—	µA	

Note 1. These values are reference values. The actual operating current greatly depends on the system (such as unsharpened waveforms due to I/O load and toggle frequency). Be sure to measure these current values in the system.
Note 2. V33Icc + V12Icc must be 80 mA or less. (ΣI_{OH} in Table 34.7)

Table 34.4 DC Characteristics (3)*1

Item		Symbol	min	typ	max	Unit	Test Conditions
Schmitt trigger Input voltage	Other than 5-V tolerant pins	V_{IH1}	2.4	—	VCCQ33 + 0.3	V	
		V_{IL1}	–0.3	—	0.8	V	
		ΔV_{T1}	VCCQ33 × 0.05	—	—	V	
	5-V tolerant pins*2	V_{IH2}	VCCQ33 × 0.7	—	5.3*3	V	
		V_{IL2}	–0.3	—	VCCQ33 × 0.3	V	
		ΔV_{T2}	VCCQ33 × 0.05	—	—	V	
Input high level voltage (except for schmitt trigger input pins)		V_{IH3}	2.4	—	VCCQ33 + 0.3	V	
Input low level voltage (except for schmitt trigger input pins)		V_{IL3}	–0.3	—	0.8	V	
Output high level voltage	Other than 5-V tolerant pins	V_{OH}	VCCQ33 – 0.5	—	—	V	I_{OH} = -2 mA
Output low level voltage	Other than 5-V tolerant pins	V_{OL1}	—	—	0.4	V	I_{OL1} = 2 mA
	5-V tolerant pins*2	V_{OL2}	—	—	0.4	V	I_{OL2} = 3 mA
			—	—	0.6	V	I_{OL2} = 6 mA
Input leakage current		$\lvert I_{in} \rvert$	—	—	1.0	µA	$V_{in1} = V_{in2}$ = 0 V $V_{in1} = V_{in2}$ = VCCQ33
Three-state leakage current (off state)	Input/output and output pins excluding 5-V tolerant pins	$\lvert I_{TSI} \rvert$	—	—	1.0	µA	V_{in1} = 0 V V_{in1} = VCCQ33
	5-V tolerant pins*2		—	—	5.0	µA	V_{in2} = 0 V V_{in2} = VCCQ33
Input pull-up MOS current and resistance	Ports , P90 to P97	I_{pu1}	–300	—	–30	µA	VCCQ33 = 3.0 to 3.6 V $V_{in1} = V_{in2}$ = 0 V
		R_{pu1}	10	—	120	kΩ	
	Pins other than the above*4	I_{pu2}	–120	—	–7	µA	VCCQ33 = 3.0 to 3.6 V $V_{in1} = V_{in2}$ = 0 V
		R_{pu2}	25	—	515	kΩ	
Input pull-down MOS current and resistance	Ports , P90 to P97	I_{pd1}	30	—	300	µA	VCCQ33 = 3.0 to 3.6 V $V_{in1} = V_{in2}$ = VCCQ33
		R_{pd1}	10	—	120	kΩ	
	Pins other than the above*4	I_{pd2}	7	—	120	µA	VCCQ33 = 3.0 to 3.6 V $V_{in1} = V_{in2}$ = VCCQ33
		R_{pd2}	25	—	515	kΩ	
Pin capacity	All input/output and input pins	C_{in}	—	—	10	pF	

Note 1. Ports P50 to P56 are not included.
Note 2. Ports PC2, PC3, PC6, and PC7 are 5-V tolerant.
Note 3. When VCCQ33 is less than 3.00 V, do not apply voltage of 3.6 V or higher to 5-V tolerant pins.
Note 4. 5-V tolerant pins are not included.

Table 34.5 DC Characteristics for 1.2-V Pin*1

Item	Symbol	min	typ	max	Unit	Test Conditions
Input high level voltage	V_{IH12}	0.84	—	V_{CCQ12} + 0.3	V	
Input low level voltage	V_{IL12}	-0.3	—	0.36	V	
Input leakage current	I_{IN12}	-100	—	100	µA	
Output high level voltage*2	V_{OH12}	1.0	—	—	V	I_{OH} = -100 µA
Output low level voltage*2	V_{OL12}	—	—	0.2	V	I_{OL} = 100 µA
Output high level current*2	I_{OH12}	—	—	–2	mA	V_{OH} = 1.0 V
Output low level current*2	I_{OL12}	4	—	—	mA	V_{OL} = 0.2 V
Input pull-up MOS current and resistance	I_{pu12}	—	-9	—	µA	VCCQ12 = 1.2V V_{in1} = V_{in2} = 0V
	R_{pu12}	—	133	—	kΩ	
Input pull-down MOS current and resistance	I_{pd12}	—	7.5	—	µA	VCCQ12 = 1.2V V_{in1} = V_{in2} = VCCQ12
	R_{pd12}	—	160	—	kΩ	
Pin capacity	C_{in12}	—	—	10	pF	

Note 1. For P50, P51, P52, P53, P54, P55, and P56 pins.
Note 2. When the DSCR register of P50, P51, P52, P53, P54, P55, and P56 pins is set to 11 (1.2-V driving output).

Table 34.6 DC Characteristics for 12-Bit A/D Converter

Item		Symbol	min	typ	max	Unit	Test Conditions
Analog input voltage	Analog input pin AN00n (n = 0 to 7)	V_{AN00}	VREFL0	—	VREFH0	V	
	Analog input pin AN10n (n = 0 to 7)	V_{AN10}	VREFL1	—	VREFH1	V	

Table 34.7 Permissible Output Currents

Item		Symbol	min	typ	max	Unit
Permissible output low current (average value per pin)	Other than 5-V tolerant pins	I_{OL1}	—	—	2.0	mA
	5-V tolerant pins	I_{OL2}	—	—	3.0	mA
Permissible output low current (maximum value per pin)	Other than 5-V tolerant pins	I_{OL1}	—	—	4.0	mA
	5-V tolerant pins	I_{OL2}	—	—	6.0	mA
Permissible output low current (total)	Total of all output pins	ΣI_{OL}	—	—	80	mA
Permissible output high current (average value per pin)	All output pins	I_{OH}	—	—	-2.0	mA
Permissible output high current (maximum value per pin)	All output pins	I_{OH}	—	—	-4.0	mA
Permissible output high current (total)	Total of all output pins	ΣI_{OH}	—	—	-80	mA

[Usage Note] All output current values shall be within the values in Table 34.7 to ensure the reliability of this LSI.

34.4 AC Characteristics

- Conditions: VDD = VCCQ12 = PLLVDD0 = PLLVDD1 = 1.14 to 1.26 V,
  VCCQ33 = AVCC0 = AVCC1 = 3.0 to 3.6 V
  VREFH0 = 2.5 to 3.6 V (when AVCC0 ≥ VREFH0),
  VREFH1 = 2.5 to 3.6 V (when AVCC1 ≥ VREFH1),
  VSS = PLLVSS0 = PLLVSS1 = AVSS0 = AVSS1 = VREFL0 = VREFL1 = 0 V,
  Tj = -40 to 110°C

Table 34.8 Operating Frequency

Item			Symbol	min	max	Unit
Operating frequency	CPU clock (CPUCLK)	112-pin TFBGA	f	150	450	MHz
	System clock (ICLK)			150		
	Peripheral module clock (PCLKA)			150		
	Peripheral module clock (PCLKB)			75		
	Peripheral module clock (PCLKD)			75		
	Peripheral module clock (PCLKE)			18.75	75	
	Peripheral module clock (PCLKF)			7.5	60	
	Peripheral module clock (PCLKG)			7.5	60	
	Peripheral module clock (PCLKH)			60		
	High-speed serial clock (SERICLK)			120	150	

- 1.2-V I/O clock cycle
  Output load conditions: C = 200 pF

Item	Symbol	min	max	Unit	Test Conditions
Output clock cycle*1	t_{prd}	250	—	ns	Figure 34.2

Note 1. When the DSCR register of the P50, P51, P52, P53, P54, P55, and P56 pins is set to 11 (1.2-V driving output)

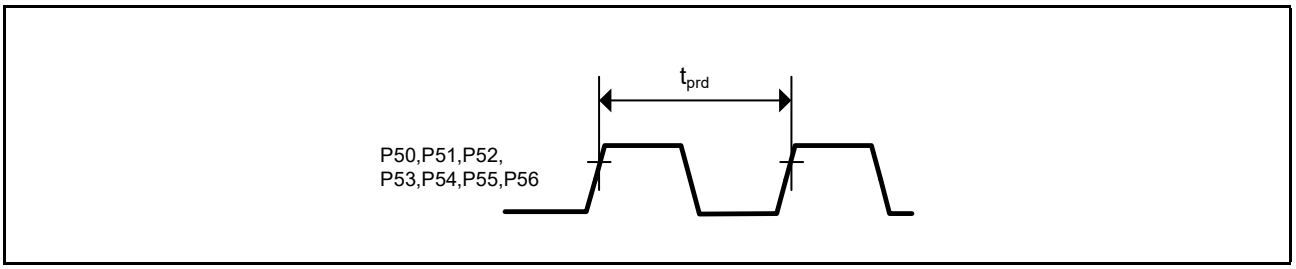


Figure 34.2 1.2-V I/O Clock Cycle

34.4.1 Clock Timing

Table 34.9 XTAL Clock Timing

Item	Symbol	min	typ	max	Unit
XTAL clock oscillator output cycle*1	$t_{XTALcyc}$		40.00 ± 50 ppm		ns

Note 1. When using the XTAL clock, ask the oscillator manufacturer to evaluate oscillation of the oscillator. For the oscillation stabilization time, see the evaluation result provided by the oscillator manufacturer.

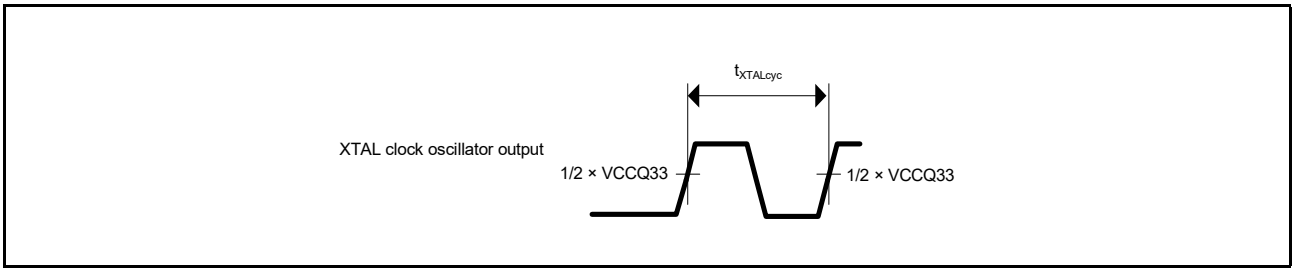


Figure 34.3 XTAL Clock Oscillator Output Timing

Table 34.10 LOCO Clock Timing

Item	Symbol	min	typ	max	Unit	Test Conditions
LOCO clock cycle time	t_{Lcyc}	4.62	4.17	3.79	µs	
LOCO clock oscillation frequency	f_{LOCO}	216	240	264	kHz	
LOCO clock oscillation stabilization wait time	t_{LOCOWT}	—	—	40	µs	Figure 34.4

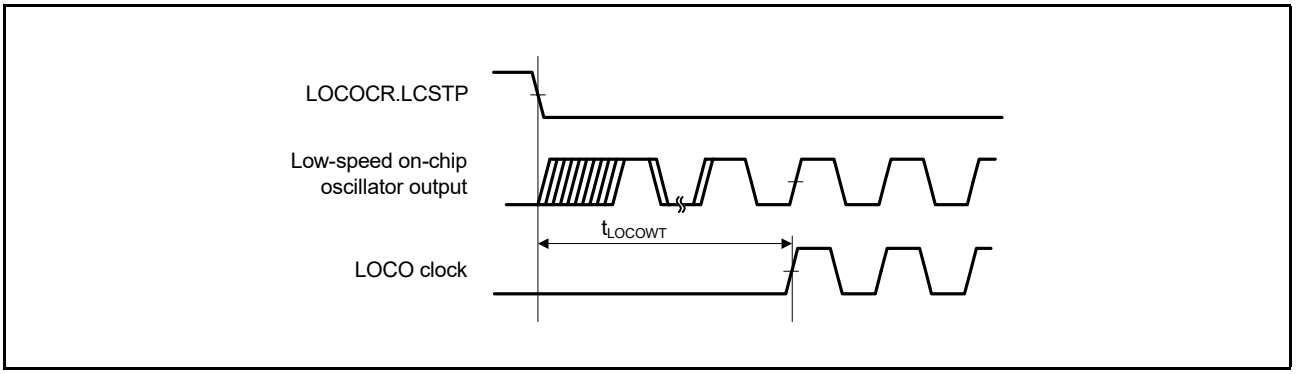


Figure 34.4 LOCO Clock Oscillation Start Timing

34.4.2 Reset Timing and Interrupt Timing

Table 34.11 Reset Timing and Interrupt Timing

Item		Symbol	Min*1	typ	max	Unit	Test Conditions
RES# pulse width	At power on	$T_{dlyreset}$	10	—	—	ms	Figure 34.5
	Other than above	$T_{dlyreset2}$	1	—	—	ms	
RES# rising time		$T_{risereset}$	—	—	150	µs	
TRST# pulse width	At power on	$T_{dlyreset}$	10	—	—	ms	
	Other than above	$T_{dlyreset2}$	1	—	—	ms	
TRST# rising time		$T_{risereset}$	—	—	150	µs	
NMI pulse width		t_{NMIW}	t_{Icyc} × 2	—	—	ns	Figure 34.6
IRQ pulse width		t_{IRQW}	t_{Icyc} × 2	—	—	ns	Figure 34.7

Note 1. t_{Icyc}: ICLK cycle

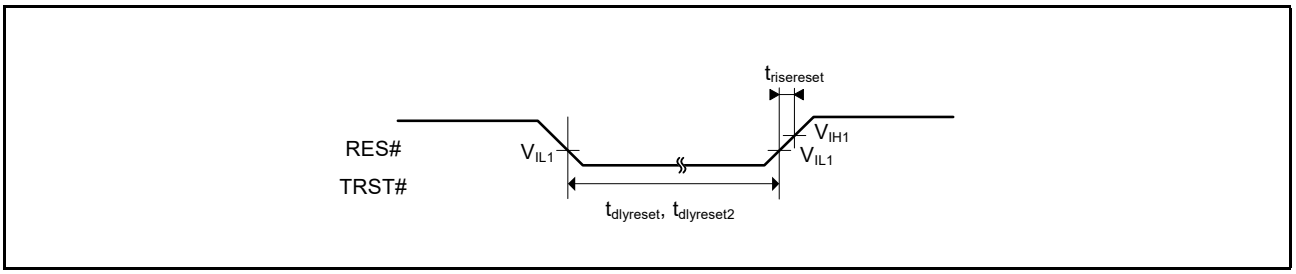


Figure 34.5 Reset Input Timing

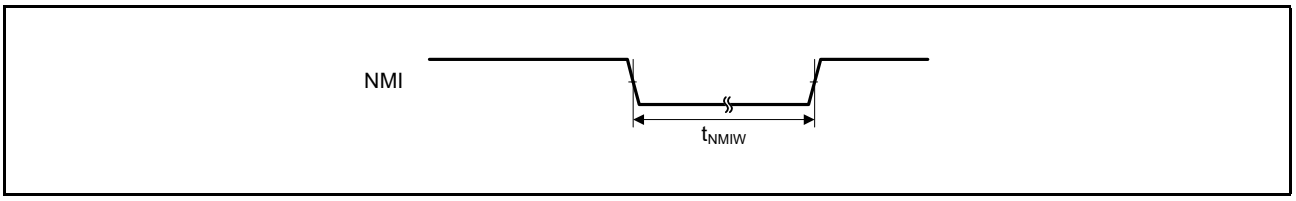


Figure 34.6 NMI Interrupt Input Timing

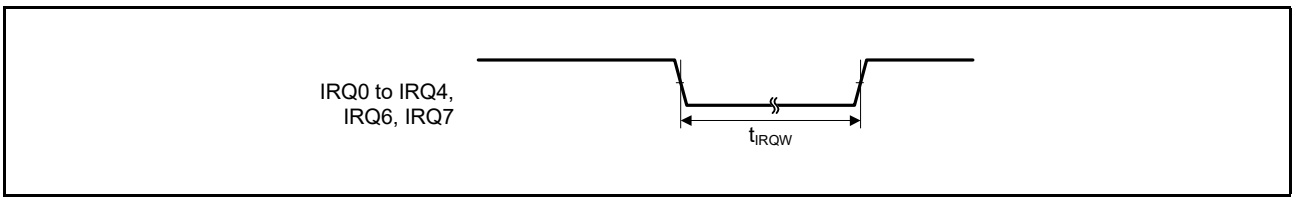


Figure 34.7 IRQ Interrupt Input Timing

34.4.3 On-Chip Peripheral Module Timing

34.4.3.1 I/O Port Timing

Table 34.12 I/O Port Timing

Item		Symbol	min	max	Unit*1	Test Conditions
I/O port	Input data pulse width	t_{PRW}	1.5	—	t_{PBcyc}	Figure 34.8

Note 1. t_{PBcyc}: PCLKB cycle

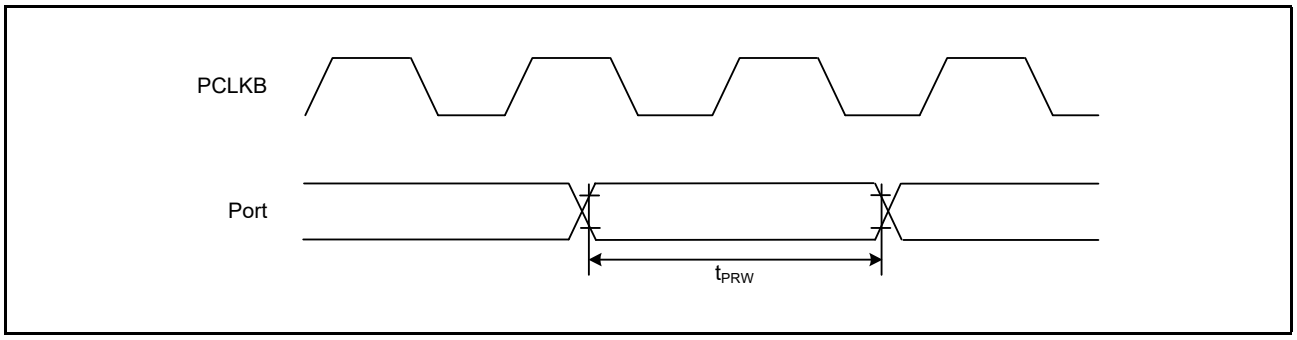


Figure 34.8 I/O Port Input Timing

34.4.3.2 TPUa Timing

Table 34.13 TPUa Timing

Item			Symbol	min	max	Unit*1	Test Conditions
TPUa	Input capture input pulse width	Single-edge setting	t_{TICW}	1.5	—	t_{PDcyc}	Figure 34.9
		Both-edge setting		2.5	—		
	Timer clock pulse width	Single-edge	t_{TCKWH}, t_{TCKWL}	1.5	—	t_{PDcyc}	Figure 34.10
		Both-edge setting		2.5	—		
		Phase counting mode		2.5	—		

Note 1. t_{PDcyc}: PCLKD cycle

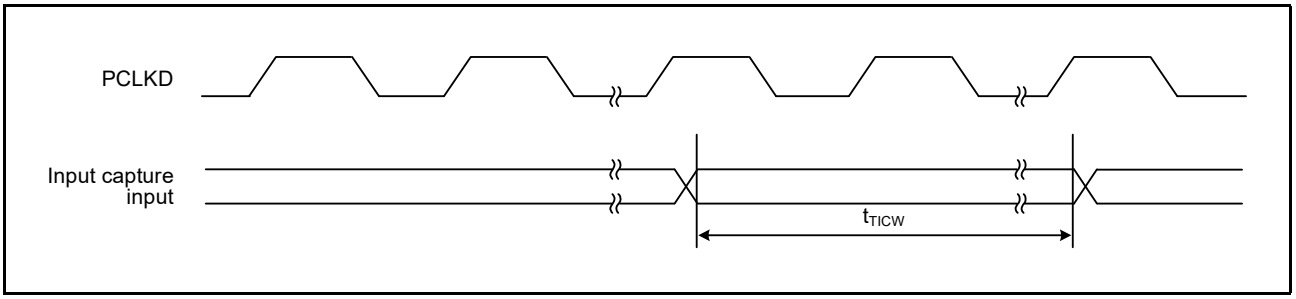


Figure 34.9 TPUa Input Capture Input Timing

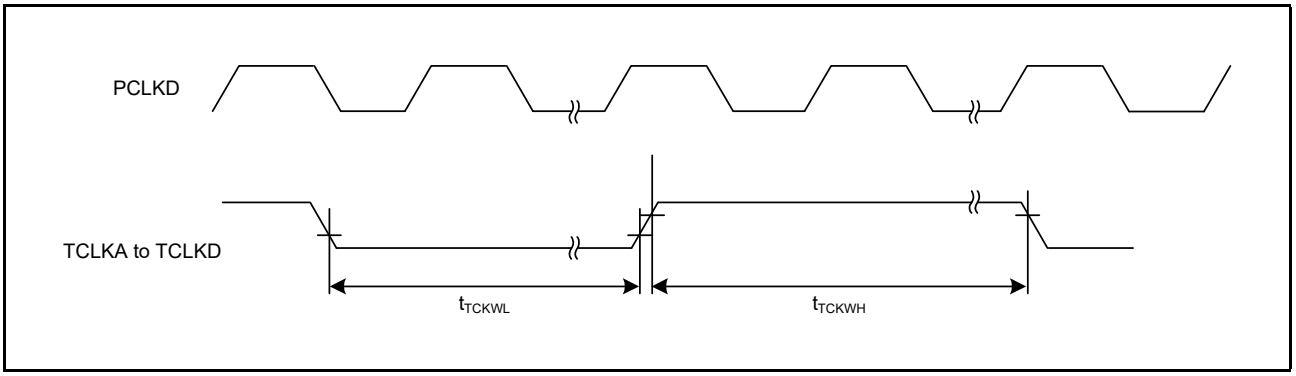


Figure 34.10 TPUa Clock Input Timing

34.4.3.3 CMTW Timing

Table 34.14 CMTW Timing

Item			Symbol	min	max	Unit*1	Test Conditions
CMTW	Input capture input pulse width	Single-edge setting	$t_{CMTWICW}$	1.5	—	t_{PDcyc}	Figure 34.11
		Both-edge setting		2.5	—		

Note 1. t_{PDcyc}: PCLKD cycle

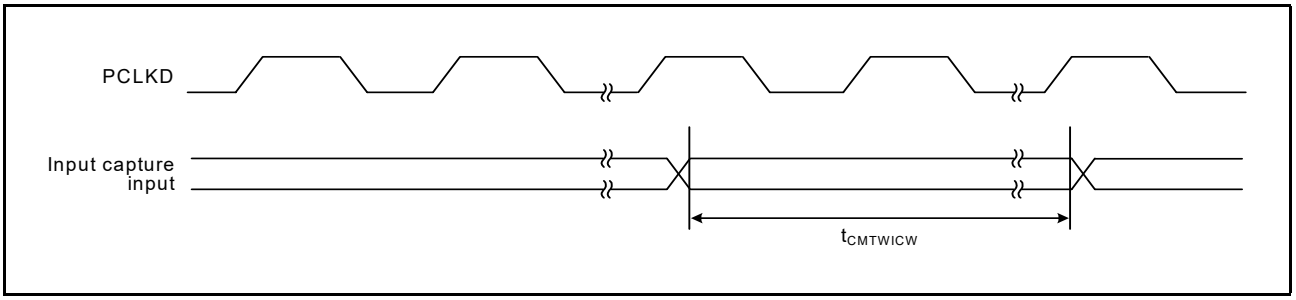


Figure 34.11 CMTW Input Capture Input Timing

34.4.3.4 A/D Converter Trigger Timing

Table 34.15 A/D Converter Trigger Timing

Item			Symbol	min	max	Unit*1	Test Conditions
A/D converter	A/D converter trigger input pulse width	ADTRG0	t_{TRGW}	1.5	—	t_{PFcyc}	Figure 34.12
		ADTRG1		1.5		t_{PGcyc}	Figure 34.13

Note 1. t_{PFcyc}: PCLKF cycle, t_{PGcyc}: PCLKG cycle

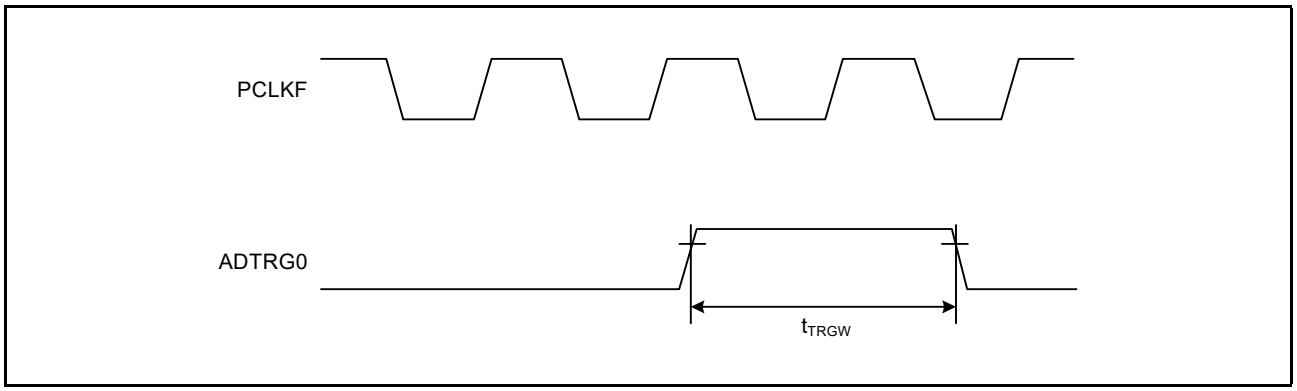


Figure 34.12 A/D Converter Trigger Input Timing (ADTRG0)

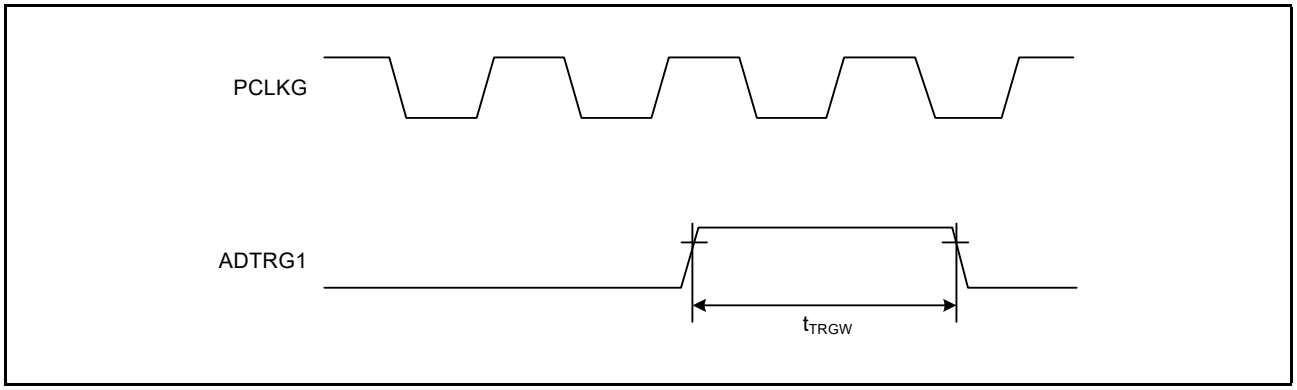


Figure 34.13 A/D Converter Trigger Input Timing (ADTRG1)

34.4.3.5 SCIFA Timing

Table 34.16 SCIFA Timing

Output load conditions: V_{OH} = VCCQ33 × 0.5, V_{OL1} = VCCQ33 × 0.5, C = 30 pF

Item			Symbol	min*1	max*1	Unit*1	Test Conditions
SCIFA	Input clock cycle	Asynchronous	t_{Scyc}	4	—	t_{SEcyc}	Figure 34.14
		Clock synchronous		12	—		
	Input clock pulse width		t_{SCKW}	0.4	0.6	t_{Scyc}	
	Input clock rising time		t_{SCKr}	—	5	ns	
	Input clock falling time		t_{SCKf}	—	5	ns	
	Output clock cycle	Asynchronous*2	t_{Scyc}	8	—	t_{SEcyc}	
		Clock synchronous		4	—		
	Output clock pulse width		t_{SCKW}	0.4	0.6	t_{Scyc}	
	Output clock rising time		t_{SCKr}	—	9	ns	
	Output clock falling time		t_{SCKf}	—	9	ns	
	Transmit data delay time	Internal clock	t_{TXD}	-10	10	ns	Figure 34.15
		External clock		3 × t_{SEcyc}	4 × t_{SEcyc} + 20		
	Receive data setup time	Internal clock	t_{RXS}	3 × t_{SEcyc} + 20	—	ns	
		External clock		t_{SEcyc} + 10	—		
	Receive data hold time	Internal clock	t_{RXH}	-3 × t_{SEcyc}	—	ns	
		External clock		2 × t_{SEcyc} + 10	—		

Note 1. t_{SEcyc}: SERICLK cycle
Note 2. When the SEMR.ABCS0 bit = 1 and the SEMR.BGDM bit = 1

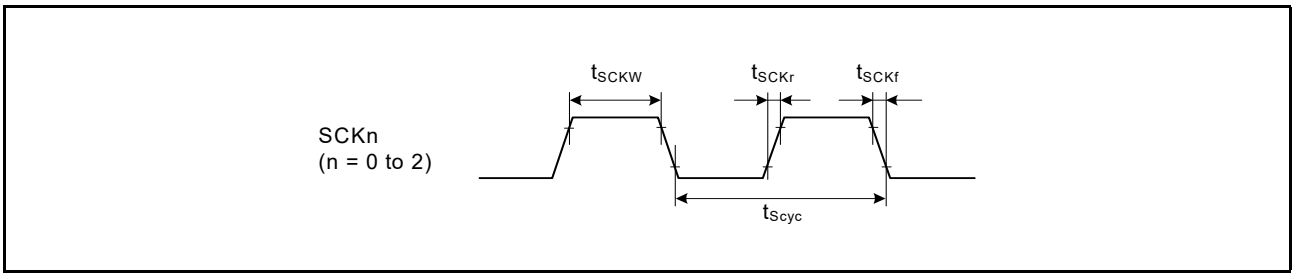


Figure 34.14 SCK Clock Input Timing

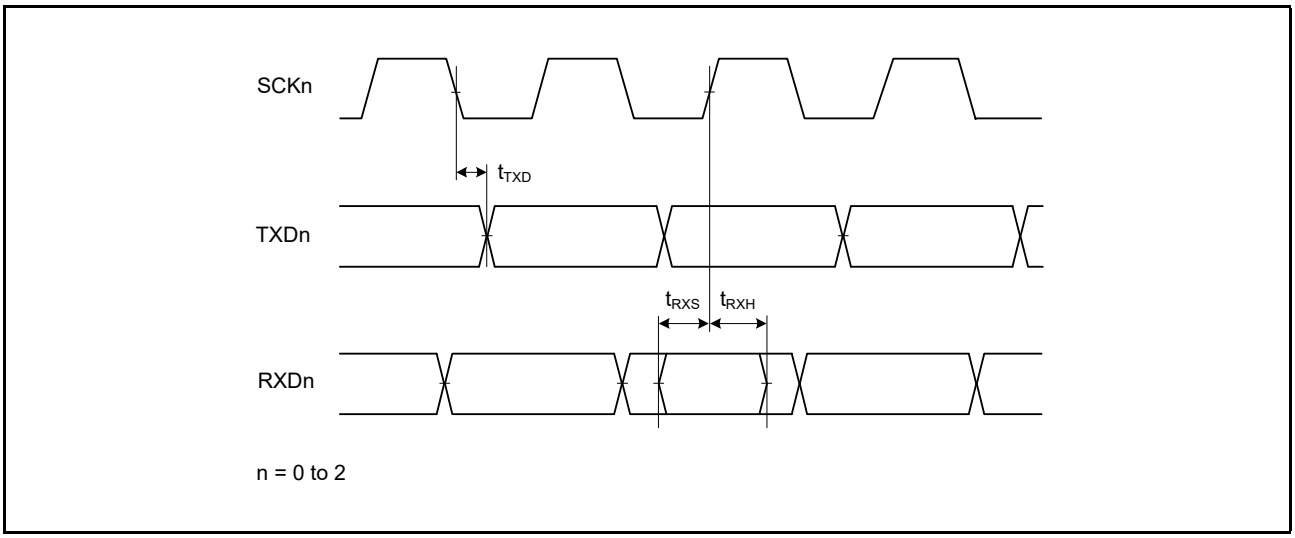


Figure 34.15 SCIFA Input/Output Timing/Clock Synchronous Mode

34.4.3.6 RSPIa Timing

Table 34.17 RSPIa Timing

Output load conditions: V_{OH} = VCCQ33 × 0.5, V_{OL1} = VCCQ33 × 0.5, C = 30 pF

Item			Symbol*1	Min*1	Max*1	Unit*1	Test Conditions
RSPIa	RSPCK clock cycle	Master	t_{SPcyc}	4	4096	t_{SEcyc}	Figure 34.16
		Slave*4		8	4096		
	RSPCK clock high level pulse width	Master	t_{SPCKWH}	$(t_{SPcyc} - t_{SPCKR} - t_{SPCKF}) / 2 - 3$	—	ns	
		Slave		0.4	—	t_{SPcyc}	
	RSPCK clock low level pulse width	Master	t_{SPCKWL}	$(t_{SPcyc} - t_{SPCKR} - t_{SPCKF}) / 2 - 3$	—	ns	
		Slave		0.4	—	t_{SPcyc}	
	RSPCK clock rising/ falling time	Output	t_{SPCKr}, t_{SPCKf}	—	9	ns	
		Input		—	10	ns	
	Data input setup time	Master	t_{SU}	6	—	ns	Figure 34.17 to Figure 34.20
		Slave		$8 - t_{SEcyc}$	—		
	Data input hold time	Master	t_H	t_{SEcyc}	—	ns	
		Slave		$8 + 2 \times t_{SEcyc}$	—		
	SSL setup time	Master	t_{LEAD}	$N \times t_{Spcyc} - 3$*2	$N \times t_{Spcyc} + 3$*2	ns	
		Slave		4	—	t_{SEcyc}	
	SSL hold time	Master	t_{LAG}	$N \times t_{Spcyc} - 3$*3	$N \times t_{Spcyc} + 3$*3	ns	
		Slave		4	—	t_{SEcyc}	
	Data output delay time	Master	t_{OD}	—	6	ns	
		Slave		—	$3 \times t_{SEcyc} + 20$ *4		
	Data output hold time	Master	t_{OH}	0	—	ns	
		Slave		0	—		
	Continuous transmission delay	Master	t_{TD}	$t_{SPcyc} + 2 \times t_{SEcyc}$	$8 \times t_{SPcyc} + 2 \times t_{SEcyc}$	ns	
		Slave		$4 \times t_{SEcyc}$	—		
	MOSI, MISO rising/ falling time	Output	t_{Dr}, t_{Df}	—	9	ns	
		Input		—	10	ns	
	SSL rising/falling time	Output	t_{SSLr}, t_{SSLf}	—	9	ns	
		Input		—	10	ns	
	Slave access time		t_{SA}	—	4	t_{SEcyc}	Figure 34.19, Figure 34.20
	Slave output release time		t_{REL}	—	3	t_{SEcyc}	

Note 1. t_{SEcyc}: SERICLK cycle
Note 2. N = SPCKD set value + 1 (1 to 8)
Note 3. N = SSLND set value + 1 (1 to 8)
Note 4. The data output delay time may become longer than half a cycle of the RSPCK clock depending on the bit rate setting. Be sure to satisfy the conditions required for the electrical characteristics of the master device.

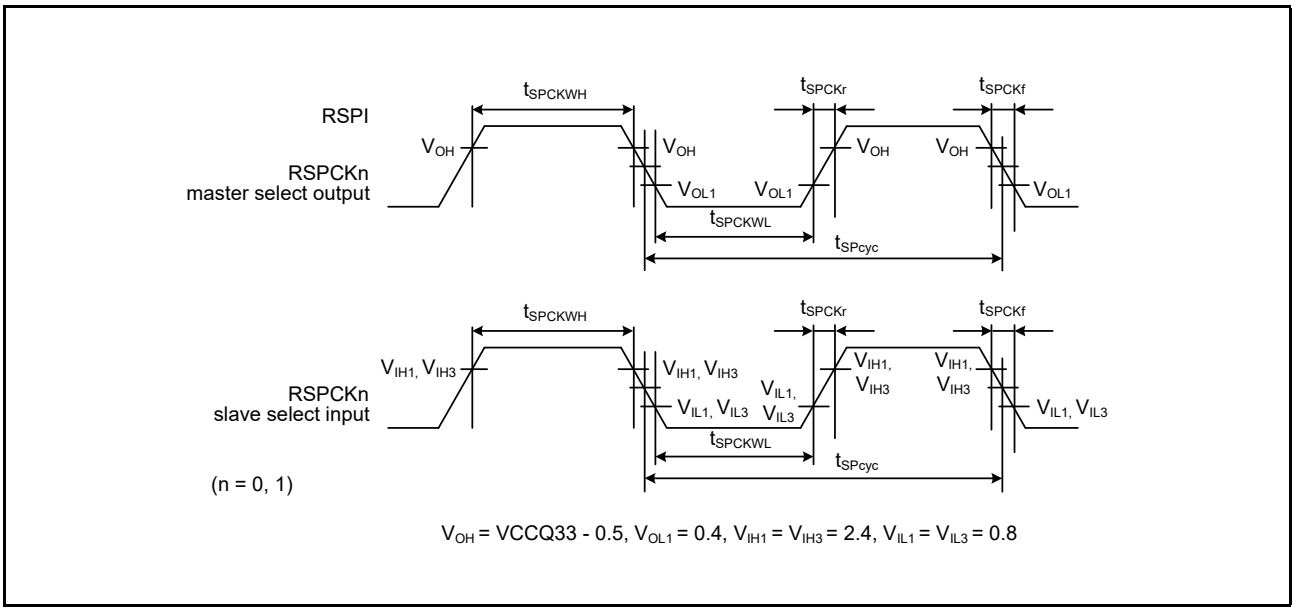


Figure 34.16 RSPIa Clock Timing

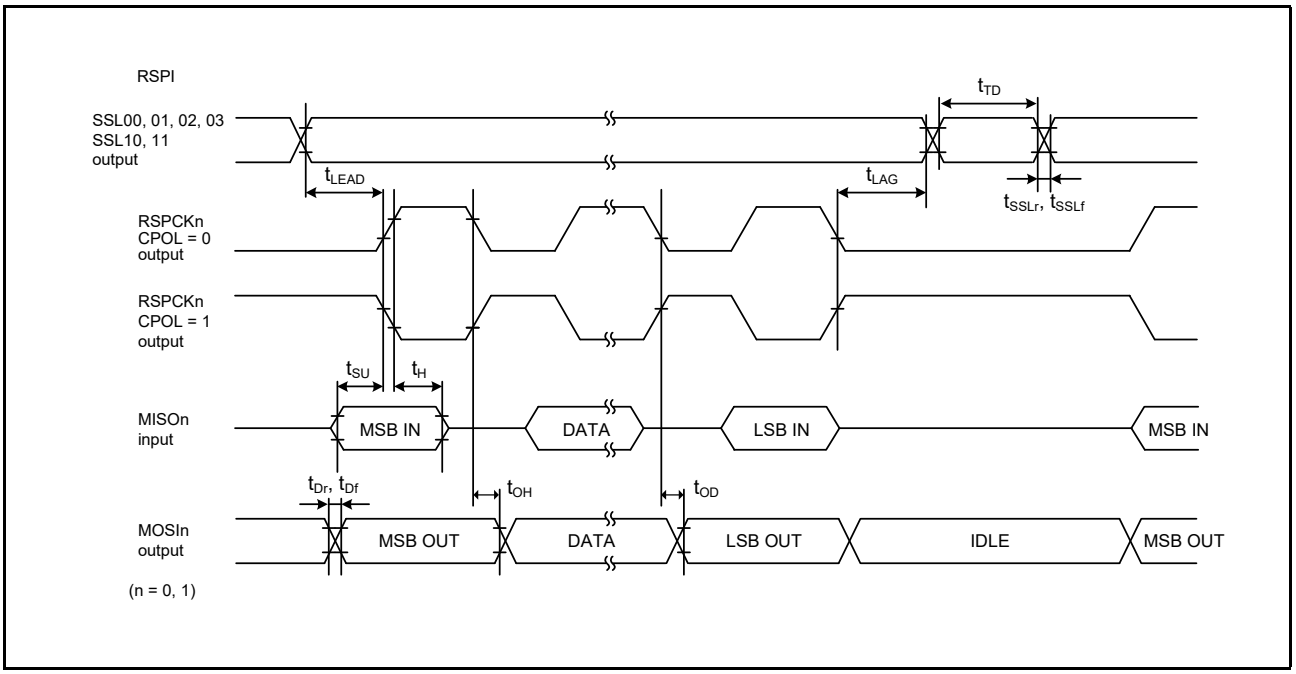


Figure 34.17 RSPIa Timing (Master, CPHA = 0)

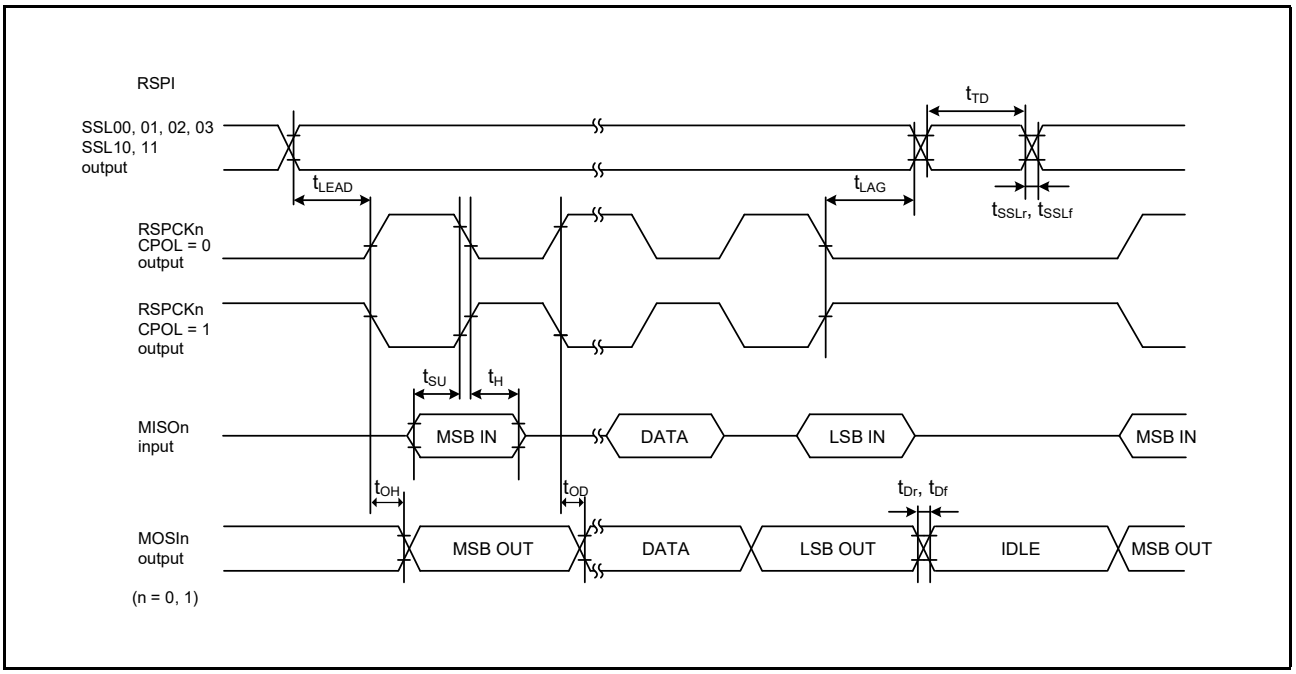


Figure 34.18 RSPIa Timing (Master, CPHA = 1)

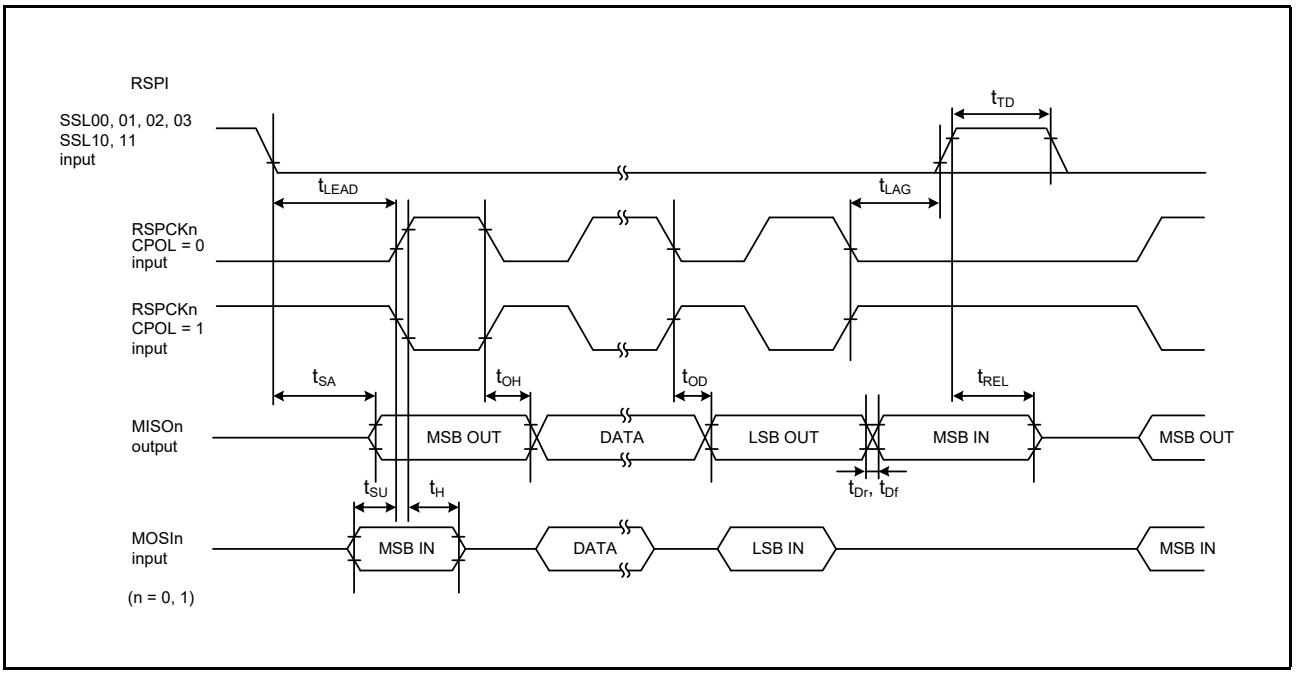


Figure 34.19 RSPI Timing (Slave, CPHA = 0)

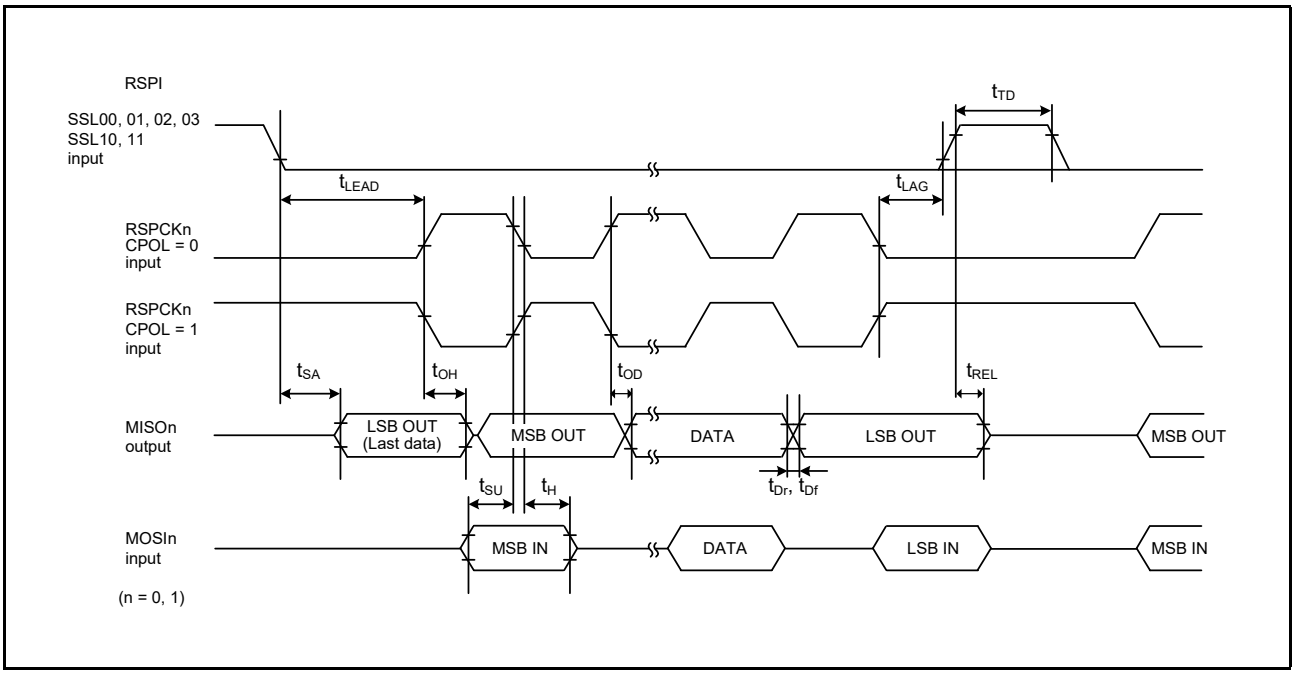


Figure 34.20 RSPI Timing (Slave, CPHA = 1)

34.4.3.7 SPIBSC Timing

Table 34.18 SPIBSC Timing

Output load conditions: V_{OH} = VCCQ33 × 0.5, V_{OL1} = VCCQ33 × 0.5, C = 30 pF

Item		Symbol	min	max	Unit*1	Test Conditions
SPIBSC	SPBCLK clock cycle	t_{SPBcyc}	2	4080	t_{PAcyc}	Figure 34.21
	SPBCLK high level pulse width	t_{SPBWH}	0.45	0.55	t_{SPBcyc}	
	SPBCLK low level pulse width	t_{SPBWL}	0.45	0.55	t_{SPBcyc}	
	Data input setup time	t_{SU}	3.5	—	ns	Figure 34.22, Figure 34.23, Figure 34.24
	Data input hold time	t_{H}	0.5	—	ns	
	SSL setup time	t_{LEAD}	1 × t_{SPBcyc} – 3	8 × t_{SPBcyc}	ns	
	SSL hold time	t_{LAG}	1.5 × t_{SPBcyc}	8.5 × t_{SPBcyc} + 3	ns	
	Continuous transfer delay time	t_{TD}	1	8	t_{SPBcyc}	
	Data output delay time	t_{OD}	—	3.6	ns	
	Data output hold time	t_{OH}	-1	—	ns	
	Data output buffer on time	t_{BON}	—	3.6	ns	Figure 34.25, Figure 34.26, Figure 34.27
	Data output buffer off time	t_{BOFF}	-7	0	ns	

Note 1. t_{PAcyc}: PCLKA cycle

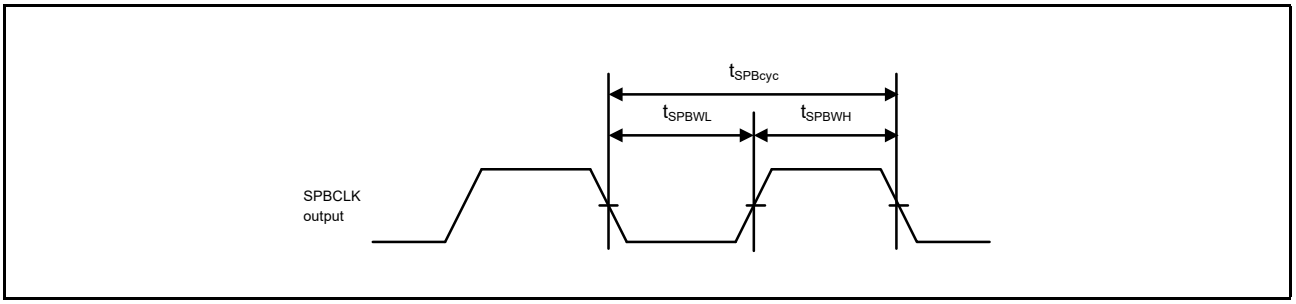


Figure 34.21 SPIBSC Clock Timing

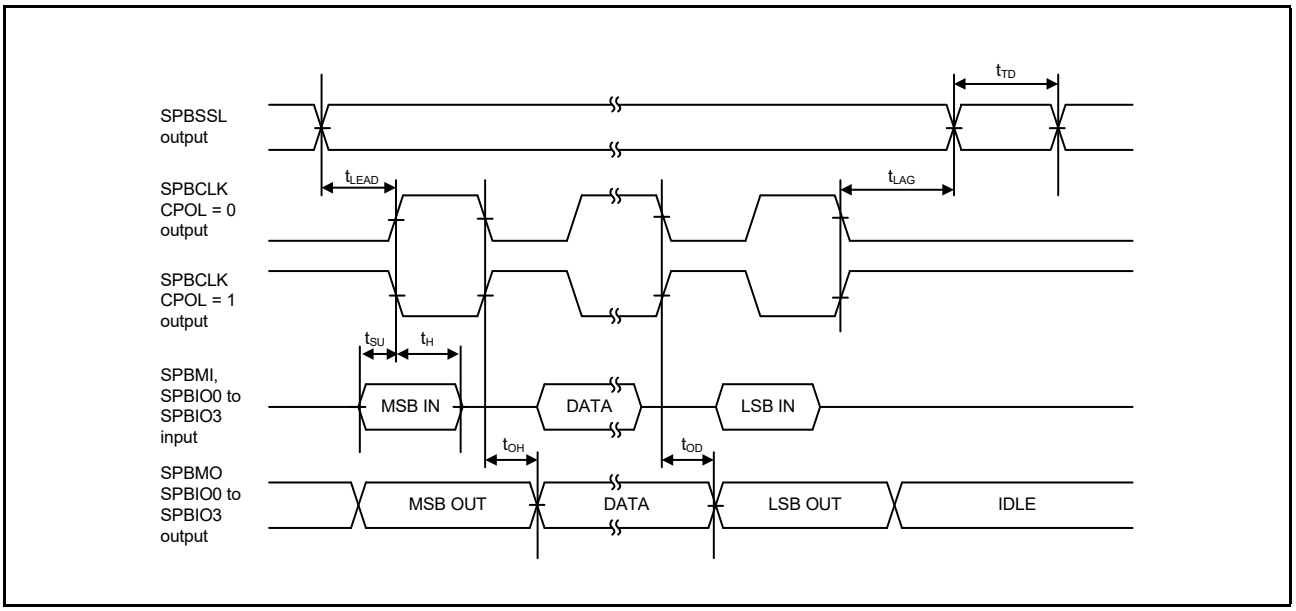


Figure 34.22 SPIBSC Transmit/Receive Timing (CPHAT = 0, CPHAR = 0)

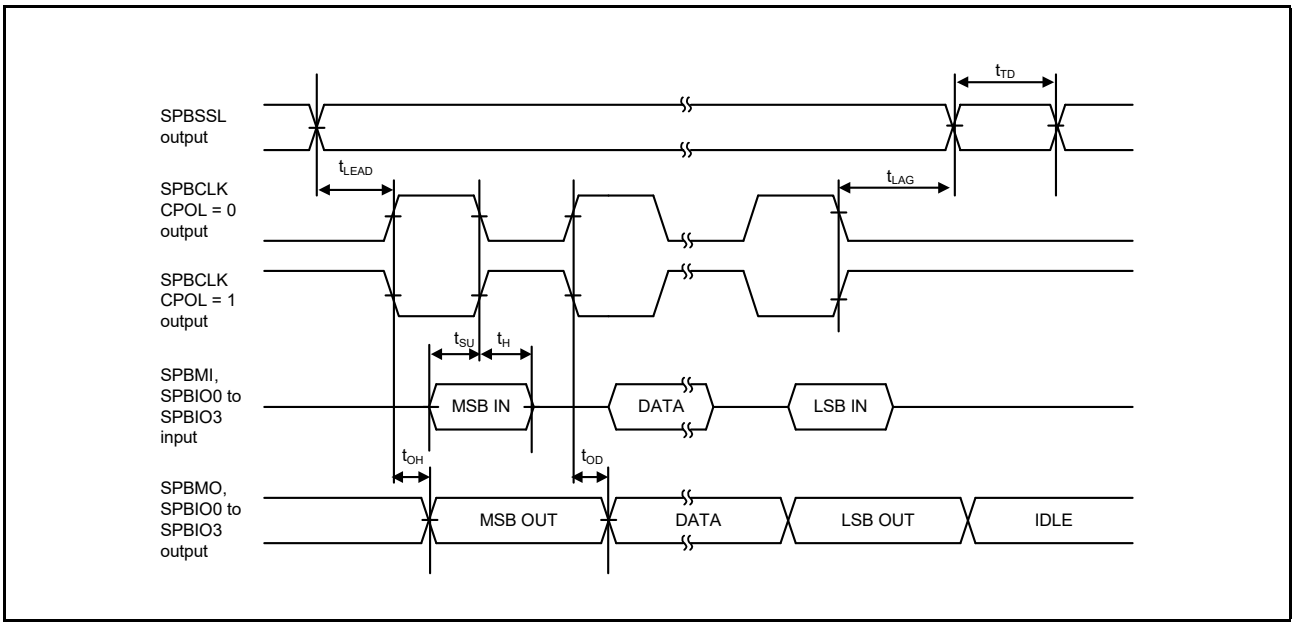


Figure 34.23 SPIBSC Transmit/Receive Timing (CPHAT = 1, CPHAR = 1)

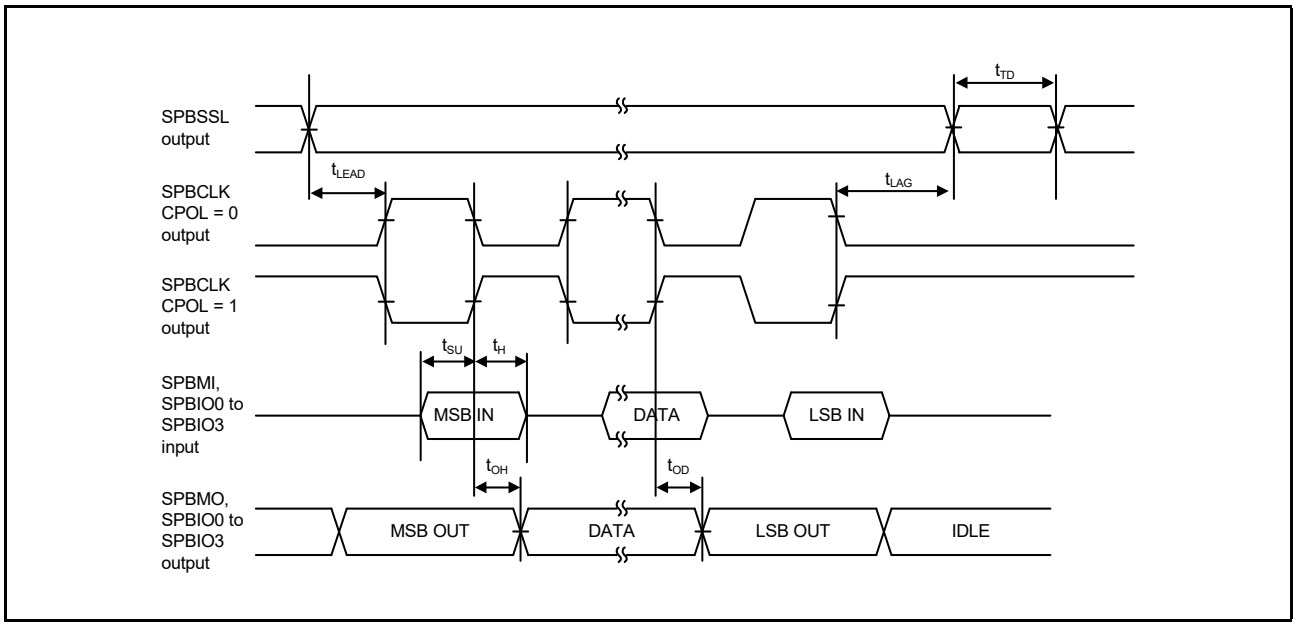


Figure 34.24 SPIBSC Transmit/Receive Timing (CPHAT = 0, CPHAR = 1)

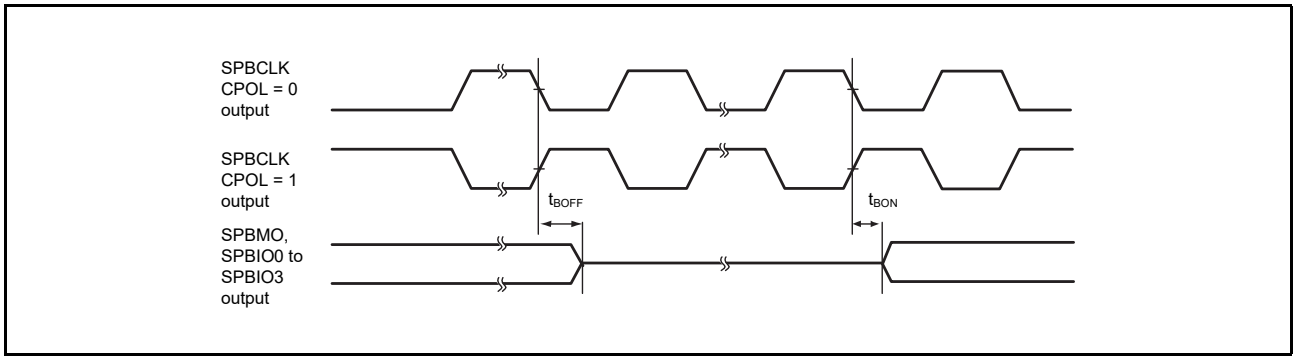


Figure 34.25 SPIBSC Buffer On/Off Timing (CPHAT = 0, CPHAR = 0)

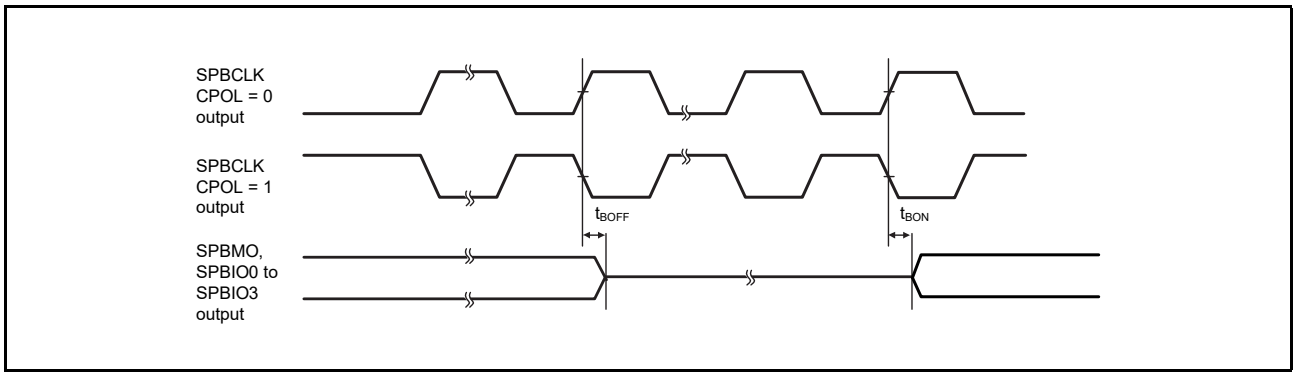


Figure 34.26 SPIBSC Buffer On/Off Timing (CPHAT = 1, CPHAR = 1)

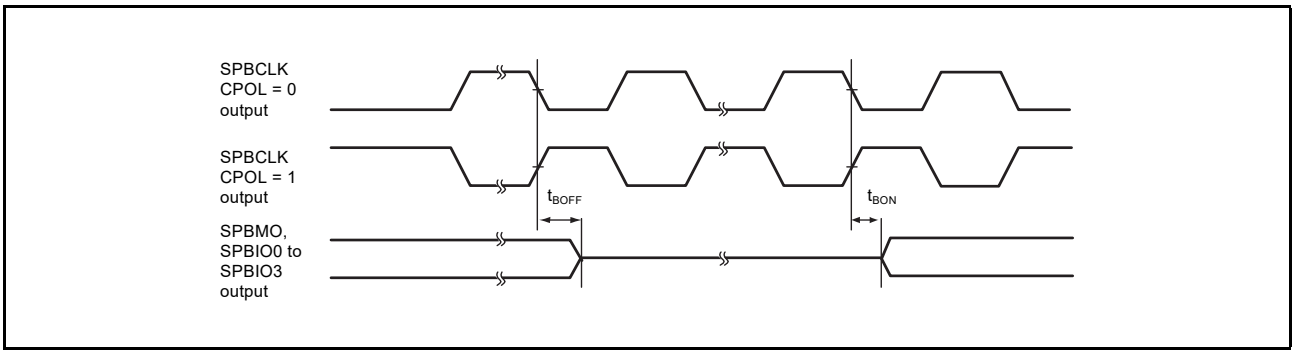


Figure 34.27 SPIBSC Buffer On/Off Timing (CPHAT = 0, CPHAR = 1)

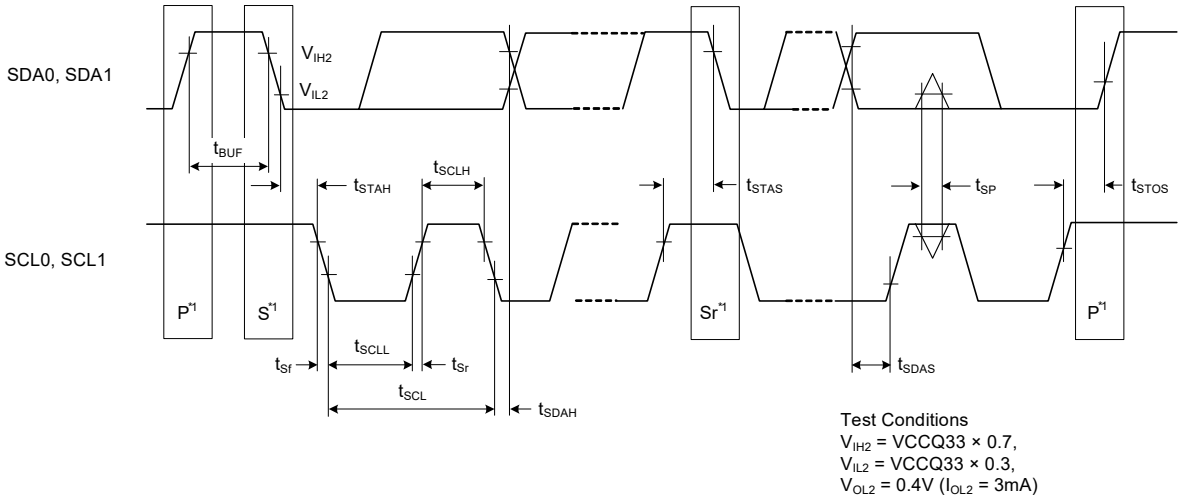
34.4.3.8 RIICa Timing

Table 34.19 RIICa Timing

Output load conditions: V_{OL2} = 0.4 V, I_{OL2} = 3 mA

Item		symbol	min*2	max*2	Unit*1	Test Conditions
RIICa (Standard-mode)	SCL input cycle time	t_{SCL}	6(12) × t_{IICcyc} + 1300	—	ns	Figure 34.28
	SCL input high pulse width	t_{SCLH}	3(6) × t_{IICcyc} + 300	—	ns	
	SCL input low pulse width	t_{SCLL}	3(6) × t_{IICcyc} + 300	—	ns	
	SCL, SDA input rising time	t_{sr}	—	1000	ns	
	SCL, SDA input falling time	t_{sf}	—	300	ns	
	SCL, SDA input spike pulse removal time	t_{SP}	0	1(4) × t_{IICcyc}	ns	
	SDA input bus free time	t_{BUF}	3(6) × t_{IICcyc} + 300	—	ns	
	Start condition input hold time	t_{STAH}	t_{IICcyc} + 300	—	ns	
	Restart condition input setup time	t_{STAS}	1000	—	ns	
	Stop condition input setup time	t_{STOS}	1000	—	ns	
	Data input setup time	t_{SDAS}	t_{IICcyc} + 50	—	ns	
	Data input hold time	t_{SDAH}	0	—	ns	
	SCL, SDA capacitive load	C_b	—	400	pF	
RIICa (Fast-mode)	SCL input cycle time	t_{SCL}	6(12) × t_{IICcyc} + 600	—	ns	
	SCL input high pulse width	t_{SCLH}	3(6) × t_{IICcyc} + 300	—	ns	
	SCL input low pulse width	t_{SCLL}	3(6) × t_{IICcyc} + 300	—	ns	
	SCL, SDA input rising time	t_{sr}	—*4	300	ns	
	SCL, SDA input falling time	t_{sf}	—*4	300	ns	
	SCL, SDA input spike pulse removal time	t_{SP}	0	1(4) × t_{IICcyc}	ns	
	SDA input bus free time	t_{BUF}	3(6) × t_{IICcyc} + 300	—	ns	
	Start condition input hold time	t_{STAH}	t_{IICcyc} + 300	—	ns	
	Restart condition input setup time	t_{STAS}	300	—	ns	
	Stop condition input setup time	t_{STOS}	300	—	ns	
	Data input setup time	t_{SDAS}	t_{IICcyc} + 50	—	ns	
	Data input hold time	t_{SDAH}	0	—	ns	
	SCL, SDA capacitive load*3	C_b	—	400	pF	

Note 1. t_{IICcyc}: RIIC internal reference clock (IICϕ) cycle
Note 2. The value out of parentheses is applicable when the value of the ICMR3.NF[1:0] bits is 00b while the digital filter is enabled by setting ICFER.NFE = 1. The value within parentheses is applicable when the value of the ICMR3.NF[1:0] bits is 11b while the digital filter is enabled by setting ICFER.NFE = 1.
Note 3. C_b is the total capacitance of the bus lines.
Note 4. The minimum values are not specified for tsr and tsf in Fast-mode.



Note 1. S, P, and Sr indicate the following conditions.
S: Start condition
P: Stop condition
Sr: Restart condition

Figure 34.28 RIICa Bus Interface Input/Output Timing

34.4.3.9 Serial Management Interface Timing

(1) Serial Management Interface (Slave) Timing

Table 34.20 Serial Management Interface (Slave) Timing*1

Output load conditions: V_{OH12} = 1.0V, V_{OL12} = 0.2V, C = 30pF

Item	Symbol	min	max	Unit	Test Conditions
MDC input cycle	t_{MDC}	250	—	ns	Figure 34.29
MDIO setup time	t_{SMDIO}	10	—	ns	
MDIO hold time	t_{MHDIO}	10	—	ns	
MDIO output delay time	t_{DMDIO}	—	175	ns	

Note 1. When the DSCR register for the P50, P51, P52, P53, P54, P55, and P56 pins is set to 11 (1.2-V driving output)

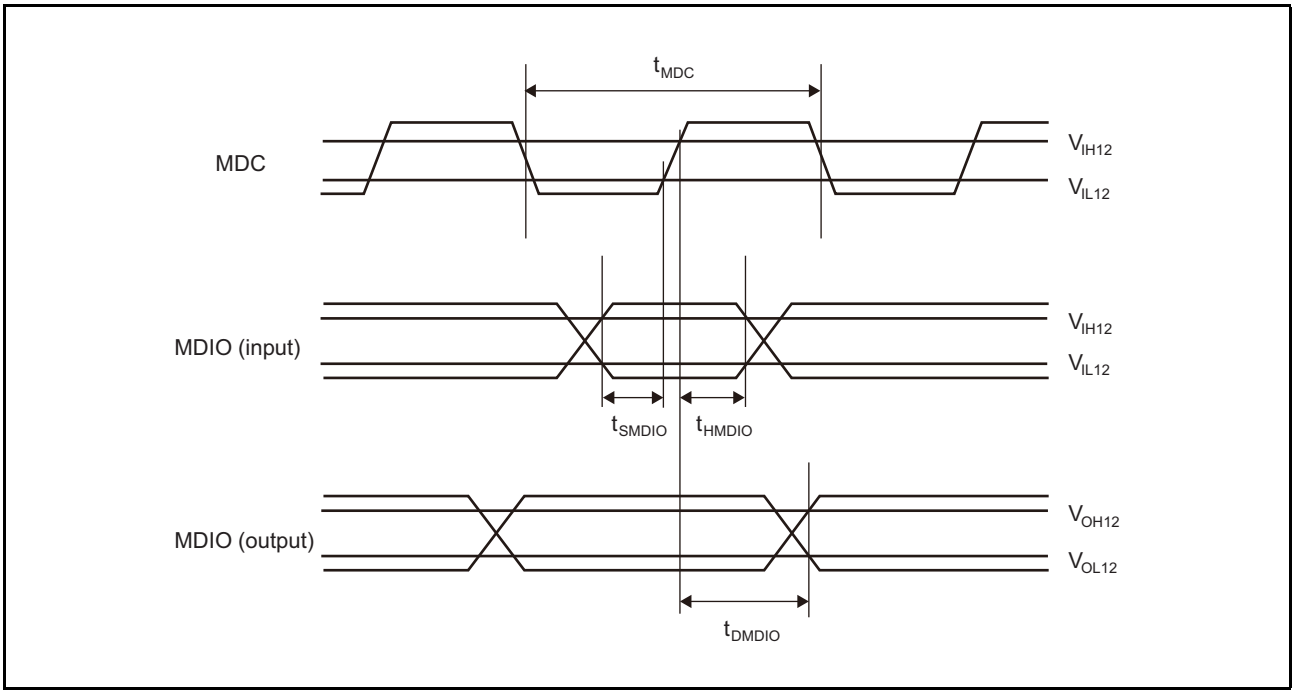


Figure 34.29 Serial Management Interface (Slave) Timing

(2) Serial Management Interface (Master, Channel 1) Timing

Table 34.21 Serial Management Interface (Master, Channel 1) Timing

Output load conditions: V_{OH1} = VCCQ33 – 0.5V, V_{OL1} = 0.4V, C = 30pF

Item	Symbol	min	max	Unit	Test Conditions
MDC output cycle	t_{MMDC1}	100	—	ns	Figure 34.30
MDIO setup time	$t_{SMMDIO1}$	40	—	ns	
MDIO hold time	$t_{HMMDIO1}$	0	—	ns	
MDIO output delay time	$t_{DMMDIO1}$	-20	20	ns	

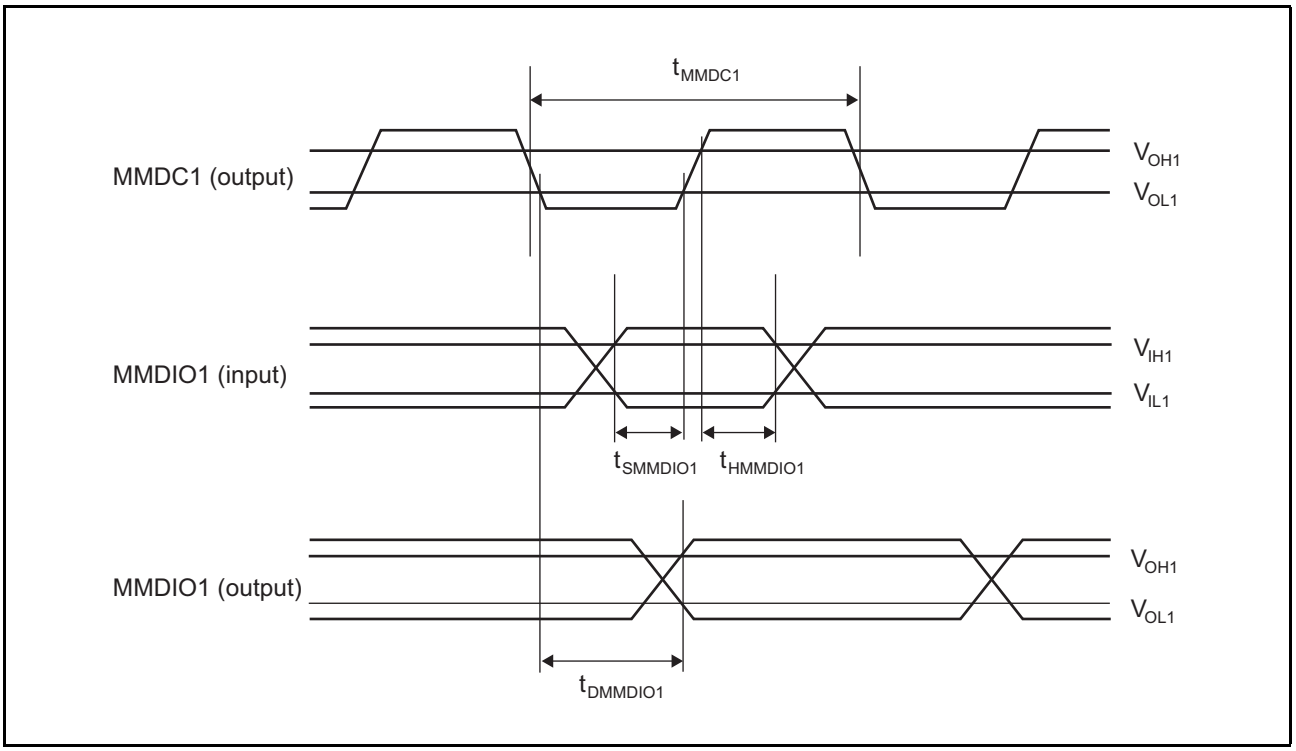


Figure 34.30 Serial Management Interface (Master, Channel 1) Timing

34.5 A/D Conversion Characteristics

- Conditions: VDD = VCCQ12 = PLLVDD0 = PLLVDD1 = 1.14 to 1.26 V,
  VCCQ33 = AVCC0 = AVCC1 = 3.0 to 3.6 V
  VREFH0 = 3.0 to 3.6 V (AVCC0 ≥ VREFH0),
  VREFH1 = 3.0 to 3.6 V (AVCC1 ≥ VREFH1),
  VSS = PLLVSS0 = PLLVSS1 = AVSS0 = AVSS1 = VREFL0 = VREFL1 = 0 V,
  Tj = -40 to 110°C

Table 34.22 12-Bit A/D (Unit 0) Conversion Characteristics 1 (1 / 2)

Item		min	typ	max	Unit	Test Conditions
Resolution		8	—	12	Bit	
Analog input capacitance		—	—	30	pF	
Channel-dedicated sample-and-hold circuits in use (AN000 to AN003) When disconnection detection assistance is in use	Conversion time*1 (Operation at PCLKF = 60 MHz) Permissible signal source impedance Max. = 1.0 kΩ	1.2 (0.4 + 0.4) *2	—	3.6	µs	• Sampling of channel-dedicated sample-and-hold circuits in 24 states • Sampling in 24 states
	Offset error	—	—	±7.5	LSB	
	Full-scale error	—	—	±7.5	LSB	
	Quantization error	—	±0.5	—	LSB	
	Absolute accuracy	—	—	±7.5	LSB	
	DNL differential nonlinearity error	—	—	±3.0	LSB	
	INL integral nonlinearity error	—	—	±4.0	LSB	
	Holding characteristics of sample-and-hold circuits	—	—	3.2	µs	Self-diagnosis + 4-channel simultaneous sampling
	Dynamic range	0.25	—	VREFH0 – 0.25	V	
Channel-dedicated sample-and-hold circuits in use (AN000 to AN003) When disconnection detection assistance is not in use	Conversion time*1 (Operation at PCLKF = 60 MHz) Permissible signal source impedance Max. = 1.0 kΩ	1.2 (0.4 + 0.4) *2	—	3.6	µs	• Sampling of channel-dedicated sample-and-hold circuits in 24 states • Sampling in 24 states
	Offset error	—	—	±6.5	LSB	
	Full-scale error	—	—	±6.5	LSB	
	Quantization error	—	±0.5	—	LSB	
	Absolute accuracy	—	—	±6.5	LSB	
	DNL differential nonlinearity error	—	—	±3.0	LSB	
	INL integral nonlinearity error	—	—	±4.0	LSB	
	Holding characteristics of sample-and-hold circuits	—	—	3.2	µs	Self-diagnosis + 4-channel simultaneous sampling
	Dynamic range	0.25	—	VREFH0 – 0.25	V	

Table 34.22 12-Bit A/D (Unit 0) Conversion Characteristics 1 (2 / 2)

Item		min	typ	max	Unit	Test Conditions
Channel-dedicated sample-and-hold circuits not in use (AN000 to AN007)	Conversion time*1 (Operation at PCLKF = 60 MHz) Permissible signal source impedance Max. = 1.0 kΩ	0.483 (0.267)*2	—	—	µs	Sampling in 16 states
	Offset error	—	—	±5.0	LSB	
	Full-scale error	—	—	±5.0	LSB	
	Quantization error	—	±0.5	—	LSB	
	Absolute accuracy	—	—	±6.0	LSB	
	DNL differential nonlinearity error	—	—	±2.5	LSB	
	INL integral nonlinearity error	—	—	±3.0	LSB	

Note 1. The conversion time is the total of the sampling time and the comparison time (t_{SPLSH} + t_{CONV} in Figure 30.22 and Figure 30.23 in section 30, 12-Bit A/D Converter (S12ADCa)). The number of sampling states is indicated for each item in Test Conditions.
Note 2. The value in parentheses indicates the sampling time.

Table 34.23 12-Bit A/D (Unit 1) Conversion Characteristics 1

Item		min	typ	max	Unit	Test Conditions
Resolution		8	—	12	Bit	
Conversion time*1 (Operation at PCLKF = 60 MHz)	Permissible signal source impedance Max. = 1.0 kΩ	0.883 (0.667)*2	—	—	µs	Sampling in 40 states
Analog input capacitance		—	—	30	pF	
Offset error		—	—	±6.0	LSB	
Full-scale error		—	—	±6.0	LSB	
Quantization error		—	±0.5	—	LSB	
Absolute accuracy		—	—	±6.0	LSB	
DNL differential nonlinearity error		—	—	±3.0	LSB	
INL integral nonlinearity error		—	—	±4.0	LSB	

Note 1. The conversion time is the total of the sampling time and the comparison time (t_{SPLSH} + t_{CONV} in Figure 30.22 and Figure 30.23 in section 30, 12-Bit A/D Converter (S12ADCa)). The number of sampling states is indicated for each item in Test Conditions.
Note 2. The value in parentheses indicates the sampling time.

- Conditions: VDD = VCCQ12 = PLLVDD0 = PLLVDD1 = 1.14 to 1.26 V,
  VCCQ33 = AVCC0 = AVCC1 = 3.0 to 3.6 V,
  VREFH0 = 2.5 to 3.0 V (AVCC0 ≥ VREFH0),
  VREFH1 = 2.5 to 3.0 V (AVCC1 ≥ VREFH1),
  VSS = PLLVSS0 = PLLVSS1 = AVSS0 = AVSS1 = VREFL0 = VREFL1 = 0 V
  Tj = -40 to 110ºC

Table 34.24 12-Bit A/D (Unit 0) Conversion Characteristics 2

Item		min	typ	max	Unit	Test Conditions
Resolution		8	—	12	Bit	
Analog input capacitance		—	—	30	pF	
Channel-dedicated sample-and-hold circuits in use (AN000 to AN007)	Conversion time*1 (Operation at PCLKF = 60 MHz) Permissible signal source impedance Max. = 1.0kΩ	0.883 (0.667) *2	—	—	µs	Sampling in 40 states
When disconnection detection assistance is not in use	Offset error	—	—	±8.7	LSB	
	Full-scale error	—	—	±8.7	LSB	
	Quantization error	—	±0.5	—	LSB	
	Absolute accuracy	—	—	±8.7	LSB	
	DNL differential nonlinearity error	—	—	±5.0	LSB	
	INL integral nonlinearity error	—	—	±6.0	LSB	

Note 1. The conversion time is the total of the sampling time and the comparison time (t_{SPLSH} + t_{CONV} in Figure 30.22 and Figure 30.23 in section 30, 12-Bit A/D Converter (S12ADCa)). The number of sampling states is indicated for each item in Test Conditions.
Note 2. The value in parentheses indicates the sampling time.

Table 34.25 12-Bit A/D (Unit 1) Conversion Characteristics 2

Item		min	typ	max	Unit	Test Conditions
Resolution		8	—	12	Bit	
Conversion time*1 (Operation at PCLKF = 60 MHz)	Permissible signal source impedance Max. = 1.0kΩ	0.883 (0.667)*2	—	—	µs	Sampling in 40 states
Analog input capacitance		—	—	30	pF	
Offset error		—	—	±8.7	LSB	
Full-scale error		—	—	±8.7	LSB	
Quantization error		—	±0.5	—	LSB	
Absolute accuracy		—	—	±8.7	LSB	
DNL differential nonlinearity error		—	—	±5.0	LSB	
INL integral nonlinearity error		—	—	±6.0	LSB	

Note 1. The conversion time is the total of the sampling time and the comparison time (t_{SPLSH} + t_{CONV} in Figure 30.22 and Figure 30.23 in section 30, 12-Bit A/D Converter (S12ADCa)). The number of sampling states is indicated for each item in Test Conditions.
Note 2. The value in parentheses indicates the sampling time.

34.6 Temperature Sensor Characteristics

- Conditions: VDD = PLLVDD0 = PLLVDD1 = 1.14 to 1.26 V,
  VCCQ33 = AVCC0 = AVCC1 = VREFH0 = VREFH1 = 3.0 to 3.6 V
  VSS = PLLVSS0 = PLLVSS1 = AVSS0 = AVSS1 = VREFL0 = VREFL1 = 0 V,
  Tj = -40 to 110°C

Table 34.26 Temperature Sensor Characteristics

Item	min	typ	max	Unit	Test Conditions
Relative accuracy	—	±1	—	°C	
Temperature slope	—	4.1	—	mV/°C	
Output voltage (at 25°C)	—	1.21	—	V	
Temperature sensor start time	—	—	30	µs	
Sampling time	4.25	—	—	µs	ADSSTRT.SST[7:0] = 255 states (when PCLKF [ADC (unit0) sampling CLK] = 60 MHz)

34.7 Oscillation Stop Detection Timing

Table 34.27 Oscillation Stop Detection Circuit Characteristics

Item	Symbol	min	typ	max	Unit	Test Conditions
Clock switching time	t_{dr}	—	—	1	ms	Figure 34.31

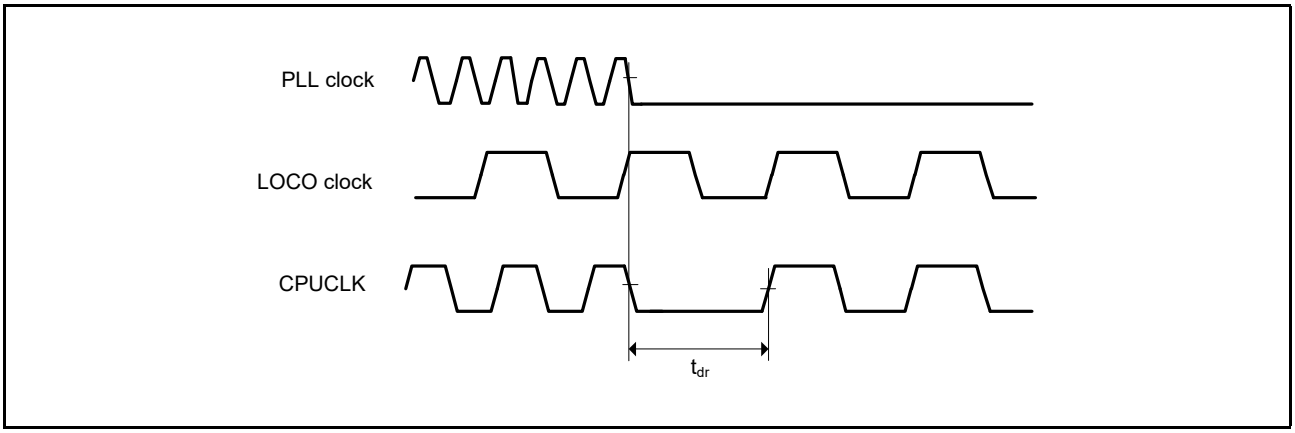


Figure 34.31 Oscillation Stop Detection Timing

34.8 Debug Interface Timing

Table 34.28 Debug Interface Timing

Output load conditions: V_{OH} = VCCQ33 – 0.5 V, V_{OL1} = 0.4 V

Item	Symbol	Min.	Max.	Unit	Reference Figure
TCK cycle time	t_{TCKcyc}	30	—	ns	Figure 34.32
TCK high pulse width	t_{TCKH}	0.4	0.6	t_{TCKcyc}	
TCK low pulse width	t_{TCKL}	0.4	0.6	t_{TCKcyc}	
TDI setup time	t_{TDIS}	5	—	ns	Figure 34.33 Output load: 30 pF
TDI hold time	t_{TDIH}	5	—	ns	
TMS/SWDIO setup time	t_{TMSS}	5	—	ns	
TMS/SWDIO hold time	t_{TMSH}	5	—	ns	
SWDIO delay time	t_{SWDO}	—	15	ns	
TDO delay time	t_{TDOD}	—	15	ns	
Trace clock cycle	t_{TCYC}	26.6	—	ns	Figure 34.34 Output load: 15 pF
Trace data delay time	t_{TDT}	0.25 × t_{TCYC} – 2	0.25 × t_{TCYC} + 2	ns	

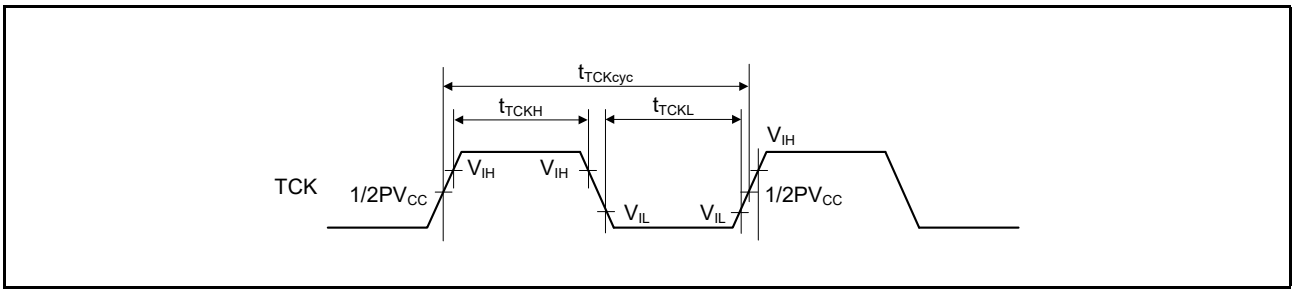


Figure 34.32 TCK Input Timing

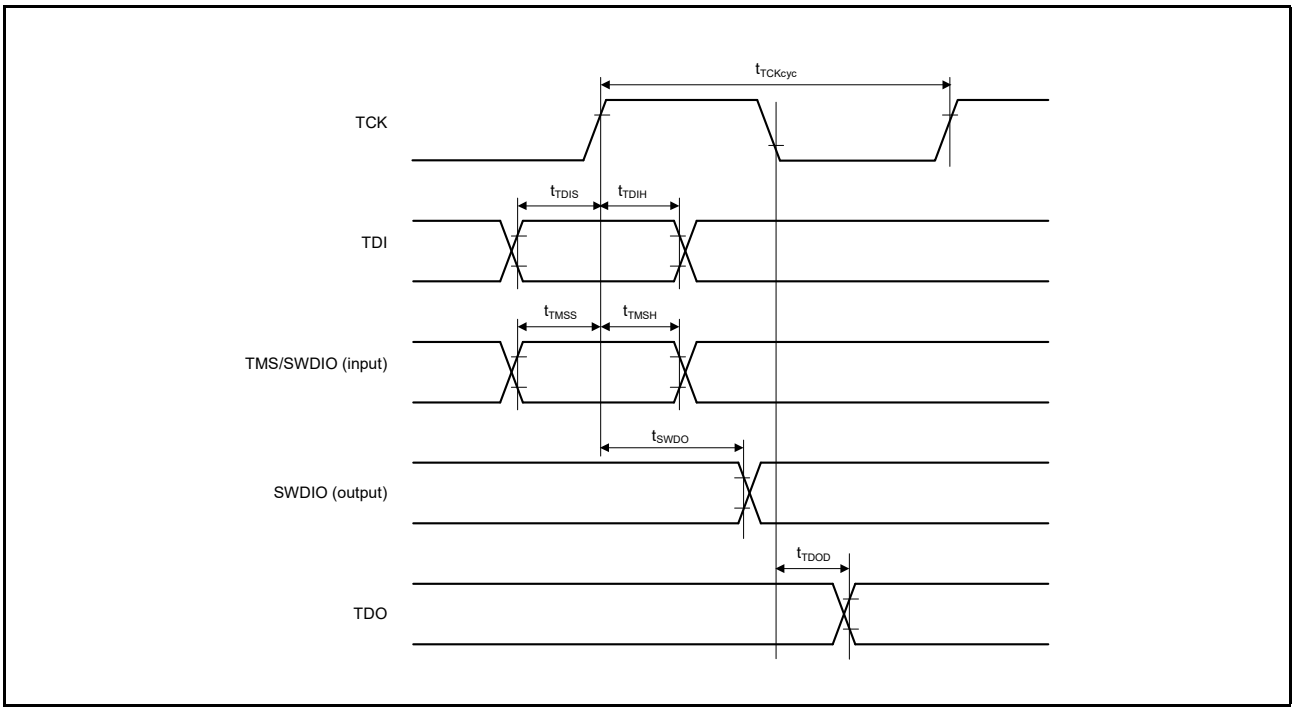


Figure 34.33 Data Transfer Timing

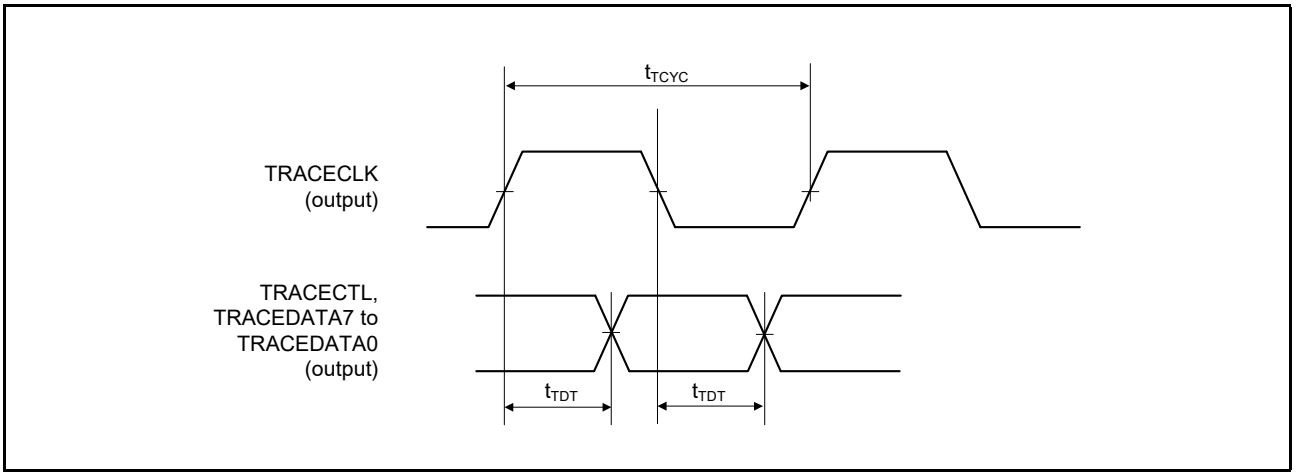
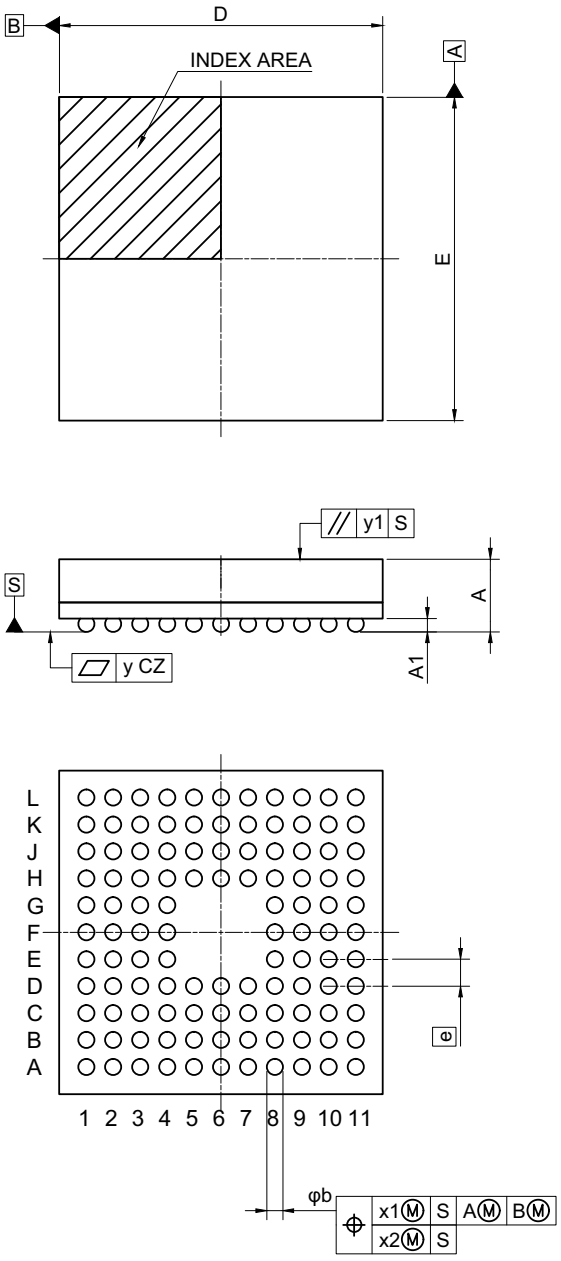


Figure 34.34 Trace Interface Timing

Appendix 1. Outer Dimensions Diagram

JEITA Package code	RENESAS code	Previous code	MASS(TYP.)[g]
P-LFBGA112-6x6-0.50	PLBG0112KA-A	-	0.09



Reference Symbol	Dimension in Millimeters		
	Min.	Nom.	Max.
D	5.80	6.00	6.20
E	5.80	6.00	6.20
A	—	—	1.70
A1	0.20	0.25	0.30
e	—	0.50	—
b	0.25	0.30	0.35
x1	—	—	0.20
x2	—	—	0.05
y	—	—	0.08
y1	—	—	0.20

Figure A 112-Pin FBGA (PLBG0112KA-A)

REVISION HISTORY — RZ/T1-M Group User's Manual: Hardware

Rev.	Date	Description	
		Page	Summary
0.60	Nov. 14, 2014	—	First edition, issued
0.70	Dec. 25, 2014	Features	
		28	■Operating temperature range: Heading title and description corrected
		Section 4. Address Space	
		60	4.1 Address Space: Description corrected
		61	Figure 4.1 Memory Map: Note 1, corrected
		Section 5. I/O Registers	
		103	Table 5.1 List of I/O Registers (Address Order) (39 / 42): The addresses and module symbols of ADCMPANSER and ADCMPLER, corrected. The ADCMPANSER and ADCMPLER lines, moved next to the ADCMPCR line.
		Section 7. Clock Generation Circuit	
		113	Table 7.1 Specifications of Clock Generation Circuit: Description of Frequency multiplication ratio at PLL0 circuit, corrected
		117	7.2.1 System Clock Control Register (SCKCR): In the bit chart and the table of bits, bit 17 corrected to reserved
		Section 10. Debugging Interface	
		148	Figure 10.1 CoreSight Configuration Diagram, corrected
		Section 12. Interrupt Controller (ICUA)	
		168	12.2.4 Non-maskable Interrupt Status Register (NMISR): In the bit chart, the name of bit 0 corrected
		Section 14. DMA Controller (DMACAa)	
		252	Table 14.1 Specifications of DMAC: Description of Interrupt request, corrected
		283	14.2.22 Descriptor Interval Register n (DSCITVL_X (X = A or B)): Functional description of the DITVL bit, corrected
		306	14.3.3.2 Round-Robin Mode: Description corrected
		307	Table 14.19 Method of Detecting DMA Transfer Request Signals: Description of Edge detection corrected
		325	Table 14.27 DMA Transfer Setting Example 4 (Descriptor 1): Table header corrected (Item → Description)
		326	Table 14.28 DMA Transfer Setting Example 4 (Descriptor 2): Data size, corrected (64 bits → 256 bits)
		327	Table 14.30 Descriptor Settings: Descriptor 1, Descriptor 2, and Descriptor 3 at CFG (Configuration) corrected
		Section 16. I/O Ports	
		361	16.3.1 Port Direction Register (PDR): The erroneous description on the address, corrected
		Section 18. 16-Bit Timer Pulse Unit (TPUa)	
		431	18.3.4 Cascaded Operation: Note: corrected
		433	Table 18.23 PWM Output Registers and Output Pins Register: Description in the Register column, corrected
		Section 24. Serial Communications Interface with FIFO (SCIFA)	
		576	24.3.10 FIFO Control Register (FCR): Register contents table, Functional description of RFRST and TFRST, corrected
		Section 25. I^2C Bus Interface (RIIC)	
		639	25.2.14 I^2C Bus Bit Rate High-Level Register (ICBRH): Note 1. corrected
		Section 29. Error Control Module (ECM)	

Rev.	Date	Description	
		Page	Summary
0.70	Dec. 25, 2014	822	Table 29.2 ECM Error Input (1 / 2): Module and Function for Error Source Numbers 1 and 3, corrected
		826	29.2.3 ECM Master/Checker Error Source Status Register 0 (ECMmESSTR0 (m = M or C)): Functional description of ECMmSSE000 and ECMmSSE002 corrected
		856	29.2.23 ECM Protection Command Register (ECMPCMD1): The table of bits, corrected (Symbol, Bit Name, and Description of bits 7 to 0 and bits 31 to 8, exchanged)
		878	29.3.4.1 Protection Unlock Sequence: The description of procedure 3, corrected
		Section 30. 12-Bit A/D Converter (S12ADCa)	
		916	30.2.18 A/D Compare Channel Select Extended Register (ADCMPANSER): Address corrected
		918	30.2.20 A/D Compare Level Extended Register (ADCMPLER): Address corrected
		Section 33. RAM (Product Option)	
		982	Table 33.1 Specifications of RAM: The description of the “Item” column, corrected
		991 to 993	33.3 Description of Operation, added
		Section 34. Electrical Characteristics (Target)	
		994	Table 34.1 Absolute Maximum Rating Conditions: VSS = PLLVSS0 = PLLVSS1 = AVSS0 = AVSS1 = VREFL0 = VREFL1 = VSS_USB = 0 V, corrected (Operating temperature(Junction temperature))
0.80	Apr. 21, 2015	Section 1 Overview	
		40 to 42	Table 1.5 Pin Assignments (112-Pin FBGA), pin names changed (MOSIn_BLUE → MOSIn, MOSIn_RED → MOSIn)
		43 to 45	Table 1.6 List of Pin and Pin Functions (112-Pin FBGA), pin names changed (MOSIn_BLUE → MOSIn, MOSIn_RED → MOSIn)
		Section 4 Address Space	
		61	Figure 4.1 Memory Map: Note 6. added
		Section 5 I/O Registers	
		64	(1) I/O register addresses (address order): Description on the number of access cycles, deleted
		65 to 106	Table 5.1 List of I/O Registers (Address Order) Address: A007 D000h deleted
		65 to 106	Table 5.1 List of I/O Registers (Address Order) Address: A007 D004h deleted
		65 to 106	Table 5.1 List of I/O Registers (Address Order) Address: A007 D040h deleted
		65 to 106	Table 5.1 List of I/O Registers (Address Order) Address: A007 D044h deleted
		65 to 106	Table 5.1 List of I/O Registers (Address Order) Address: A007 D080h deleted
		65 to 106	Table 5.1 List of I/O Registers (Address Order) Address: A007 D0F8h deleted
		Section 6 Reset	
		107	Table 6.2 Targets to be Initialized for Each Reset Type (√: To be initialized, —: No change), ERROROUT# pin output deleted
		107	Table 6.2 Targets to be Initialized for Each Reset Type (√: To be initialized, —: No change), Note 4 deleted
		110	6.2.2 Software Reset Register (SWRR1), bit chart: symbol SWRR1 corrected
		Section 7 Clock Generation Circuit	
		118	7.2.2 System Clock Control Register 2 (SCKCR2): Value after reset of b4 in the bit chart, changed
		Section 8 Clock Monitor Circuit (CLMA)	
		130	Table 8.1 Specifications of CLMAn (n = 2 to 0): Table title changed
		131	Figure 8.1 Block Diagram of CLMAn (n = 2 to 0): Description was added to the top of the figure.

Rev.	Date	Description	
		Page	Summary
0.80	Apr. 21, 2015	131	Figure 8.1 Block Diagram of CLMAn (n = 2 to 0): Figure title changed
		132	8.2.1 CLMAn Control Registers 0 (CLMAnCTL0) (n = 2 to 0): Note 1. changed
		Section 10 Debugging Interface	
		147	10.1 Overview: Description changed
		147	Table 10.1 CoreSight Specifications: Table title changed
		148	Figure 10.1 Block Diagram of CoreSight: Figure title changed
		149	Table 10.3 CTI Trigger Input and Output (Cortex-R4F): Note 1. deleted
		151	Table 10.6 Configuration of Pins for the Debugging Interface: Description on the top was changed.
		151	Table 10.6 Configuration of Pins for the Debugging Interface: Table title changed
		152	10.2.1 Debugging Interface Control Register (DBGIFCNT) Symbol: SWVSEL was changed in the bit chart.
		Section 12 Interrupt Controller (ICUA)	
		162	12.1 Overview: Description changed
		162	Table 12.1 Specifications of Interrupt Controller: Table title changed
		163	Figure 12.1 Block Diagram of Interrupt Controller: Figure title changed
		214	12.4.2.13 Interrupt Address Register (HVA0) Symbol: HVA in the bit chart, changed
		230 to 237	Table 12.3 Cortex-R4F/DMAC Interrupt Vector Table: Note 1. added
		246	Figure 12.9 Concept of Multiple Interrupts (2/2): Note 1. changed to Note 2. in interrupt request u (level 2)
		Section 14 DMA Controller (DMACAa)	
		253	14.2.1 • For N0SA_n_N and N1SA_n_N (normal mode) Symbol: SA in the bit chart, changed
		254	14.2.1 • For N0SA_n_W and N1SA_n_W (write-only mode) Symbol: WD in the bit chart, changed
		255	14.2.2 Next Destination Address Register n (N0DA_n and N1DA_n) Symbol: DA in the bit chart, changed
		256	14.2.3 Next Transaction Byte Register n (N0TB_n and N1TB_n) Symbol: TB in the bit chart, changed
		257	14.2.4 Current Source Address Register (CRSA_n) Symbol: CRSA in the bit chart, changed
		258	14.2.5 Current Destination Address Register (CRDA_n) Symbol: CRDA in the bit chart, changed
		259	14.2.6 Current Transaction Byte Register (CRTB_n) Symbol: CRTB in the bit chart, changed
		267	14.2.11 Common Control Register (CMNCR): Value after reset of b3 in the bit chart, changed
		267	14.2.11 Common Control Register (CMNCR): Functional description of b3 in the bit chart, changed
		270	14.2.13 Channel Configuration Register n (CHCFG_n): Description changed
		274	14.2.15 Next Link Address Register n (NXLA_n) Symbol: NXLA in the bit chart, changed
		275	14.2.16 Current Link Address Register n (CRLA_n) Symbol: CRLA in the bit chart, changed
		276	14.2.17 Source Continuous Register n (SCNT_n) Symbol: SCNT in the bit chart, changed
		277	14.2.18 Source Skip Register n (SSKP_n) Symbol: SSKP in the bit chart, changed
		279	14.2.19 Destination Continuous Register n (DCNT_n) Symbol: DCNT in the bit chart, changed
		280	14.2.20 Destination Skip Register n (DSKP_n) Symbol: DSKP in the bit chart, changed

Rev.	Date	Description	
		Page	Summary
0.80	Apr. 21, 2015	301	14.3.1.2(3), (6) Descriptor write-back: Description at the bottom, changed
		305	14.3.2.1 Single Transfer Mode: Description changed
		305	14.3.2.2 Block Transfer Mode: Description changed
		307	14.3.4 DMA Transfer Request: Description changed
		308 to 309	Table 14.20 DMA Transfer Request Detection Operation Setting Table: Two notes added
		317	14.4.2DMA Transfer Completion Interrupts: Description changed
		317	14.4.2 DMA Transfer Completion Interrupts: Caution added
		321	Table 14.24 DMA Transfer Setting Example 2: Setting Example 2 at the bottom, changed
		322	Figure 14.20 Setting Example 2: Flow changed
		327	Table 14.30 Descriptor Settings: Descriptor 1, Descriptor 2, and Descriptor 3 values of CFG (Configuration), changed
		Section 17 Multi-Function Pin Controller (MPC)	
		372	17.2.2 P0n Pin Function Control Register (P0nPFS) (n = 0), description corrected
		373	17.2.3 P1n Pin Function Control Register (P1nPFS) (n = 0), description corrected
		374	17.2.4 P2n Pin Function Control Register (P2nPFS) (n = 1, 2, 7), description corrected
		375	17.2.5 P3n Pin Function Control Register (P3nPFS) (n = 3 to 5), description corrected
		376	17.2.6 P4n Pin Function Control Register (P4nPFS) (n = 0, 2, 4), description corrected
		377	17.2.7 P6n Pin Function Control Register (P6nPFS) (n = 0 to 5), description corrected
		378	17.2.8 P7n Pin Function Control Register (P7nPFS) (n = 1 to 3), description corrected
		379	17.2.9 P9n Pin Function Control Register (P9nPFS) (n = 0 to 7), description corrected
		380	17.2.10 PAn Pin Function Control Register (PAnPFS) (n = 3 to 5), description corrected
		381	17.2.11 PCn Pin Function Control Register (PCnPFS) (n = 2, 3, 6, 7), description corrected
		382	17.2.12 PEn Pin Function Control Register (PEnPFS) (n = 0 to 7), description corrected
		383	17.2.13 PGn Pin Function Control Register (PGnPFS) (n = 2 to 6), description corrected
		385	Table 17.14 Register Settings, Note 1 corrected
		387	Table 17.15 List of Modules and Associated Pin Functions for Which Input is Always Enabled (2 / 2), entries corrected
		Section 21 Watchdog Timer (WDTA)	
		—	Heading title: 21.1.1 Block Diagram, deleted
		518	Figure 21.1 WDT Block Diagram: Description on the top was changed.
		518	Figure 21.1 WDT Block Diagram, changed
		—	Heading title: 21.5 Usage Notes, deleted
		—	21.5.1 Clock Division Ratio Setting, deleted
		Section 22 Independent Watchdog Timer (IWDTa)	
		—	Heading title: 22.1.1 Block Diagram, deleted
		534	Figure 22.1 IWDT Block Diagram, changed
		Section 24 Serial Communications Interface with FIFO (SCIFA)	
		561	24. Serial Communications Interface with FIFO (SCIFA): Description changed
		561	Table 24.1 Specifications of SCIFA: Description on Channel, changed
		—	Heading title: 24.2 Input/Output Pins, deleted
		562	Table 24.2 Pin Configuration: Description on the top was changed.
		562	Table 24.2 Pin Configuration: Table title changed
		571	24.2.8 Bit Rate Register (BRR): Note added
		574	Table 24.9 Maximum Bit Rates with External Clock Input (in Clock Synchronous Mode), totally changed
		580	24.3.12 Serial Port Register (SPTR): Description changed

Rev.	Date	Description	
		Page	Summary
0.80	Apr. 21, 2015	580	24.3.12 Serial Port Register (SPTR) Values after reset of symbols: RTS2DT, CTS2DT, SCKDT, and SPB2DT in the bit chart, changed
		Section 25 I2C Bus Interface (RIIC)	
		615	Table 25.2 Pin Configuration: Description on the top was changed.
		625	25.2.5 I2C Bus Mode Register 3 (ICMR3) Symbol of b7 was changed in the bit chart.
		642	Table 25.5 Examples of ICBRH/ICBRL Settings for Transfer Rate: Note changed
		685	Table 25.6 Interrupt Sources: Table heading, Priority, deleted
		Section 26 Serial Peripheral Interface (RSPIa)	
		689	26.1 Overview, description corrected
		689 to 690	Table 26.1 RSPI Specifications, description of the settings for channels corrected
		Section 27 SPI Multi I/O Bus Controller (SPIBSC)	
		771	Table 27.1 SPIBSC Specifications: Description of Figure 37.1, Block Diagram of SPIBSC, at the bottom, added
		—	Heading title: 27.2 Block Diagram, deleted
		—	Figure 27.1 Block Diagram: Description on the top, deleted
		772	Figure 27.1 Block Diagram: Figure title changed
		—	Heading title: 27.3 Input/Output Pins, deleted
		772	Table 27.2 Pin Configuration of the SPIBSC: Description on the top, changed
		772	Table 27.2 Pin Configuration of the SPIBSC: Table title changed
		773	27.4.1 Common Control Register (CMNCR) Functional description of b6 in the bit chart, changed
		788	27.4.11 SPI Mode Address Setting Register (SMADR) Symbol: ADR was changed in the bit chart.
		791	27.4.14 SPI Mode Read Data Register 0 (SMRDR0) Symbol: RDATA0 was changed in the bit chart.
		791	27.4.14 SPI Mode Read Data Register 0 (SMRDR0) Value after reset was changed in the bit chart.
		791	27.4.15 SPI Mode Write Data Register 0 (SMWDR0) Symbol: WDATA0 was changed in the bit chart.
		Section 28 CRC Operation Units (CRC)	
		816	28.2.1 CRC Data Input Register (CRCDIR), bit chart: symbol CRCDIR corrected
		817	28.2.2 CRC Data Output Register (CRCDOR), bit chart: CRCDOR corrected
		Section 29 Error Control Module (ECM)	
		821	29.1 Overview: Description changed
		821	Table 29.1 Specifications of ECM: Table title changed
		821	Table 29.1 Specifications of ECM: Description on safety processing, changed
		822	Figure 29.1 Block Diagram of ECM, changed
		823 to 824	Table 29.2 ECM Error Input Description on Error Source Number: 95, changed
		—	29.2.1 ECM Master/Checker Error Set Trigger Register (ECMmESET (m = M or C)), deleted
		—	29.2.2 ECM Master/Checker Error Clear Trigger Register (ECMmECLR (m = M or C)), deleted
		828	29.2.3 ECM Master/Checker Error Source Status Register 2 (ECMmESSTR2 (m = M or C)) Symbol of b30 was changed in the bit chart.
		828	29.2.3 ECM Master/Checker Error Source Status Register 2 (ECMmESSTR2 (m = M or C)) Symbol, bit name, description of b30 was changed in the table of bits.
		—	29.2.7 ECM Error Pulse Configuration Register (ECMEPCFG), deleted
		830	29.2.5 ECM Maskable Interrupt Configuration Register 0 (ECMMICFG0) Functional description of b2 and b0 in the table of bits, changed

Rev.	Date	Description	
		Page	Summary
0.80	Apr. 21, 2015	835	29.2.8 ECM Non-maskable Interrupt Configuration Register 0 (ECMNMICFG0) Functional description of b2 and b0 in the table of bits, changed
		840	29.2.11 ECM Internal Reset Configuration Register 0 (ECMIRCFG0) Functional description of b2 and b0 in the table of bits, changed
		845	29.2.14 ECM Error Mask Register 0 (ECMEMK0) Functional description of b2 and b0 in the table of bits, changed
		850	29.2.17 ECM Error Source Status Clear Trigger Register 0 (ECMESSTC0) Functional description of b2 and b0 in the table of bits, changed
		854	29.2.19 ECM Error Source Status Clear Trigger Register 2 (ECMESSTC2) Symbol of b30 in the bit chart, changed
		854	29.2.19 ECM Error Source Status Clear Trigger Register 2 (ECMESSTC2) Symbol, bit name, description of b30 was changed in the table of bits.
		856	29.2.22 ECM Pseudo Error Trigger Register 0 (ECMPE0) Functional description of b2 and b0 in the table of bits, changed
		860	29.2.26 ECM Delay Timer Register (ECMDTMR): Description changed
		862	29.2.28 ECM Delay Timer Configuration Register 0 (ECMDTMCFG0) Functional description of b2 and b0 in the table of bits, changed
		868	29.2.31 ECM Delay Timer Configuration Register 3 (ECMDTMCFG3) Functional description of b2 and b0 in the table of bits, changed
		—	29.2.37 ECM Error Output Clear Disable Configuration Register (ECMEOCCFG), deleted
		—	29.3.1 Operations for Error Output, deleted
		—	29.3.2 Loop-Back Function, deleted
		876	29.3.3 Timeout Function for Interrupt Processing by Using the Delay Timer: Description changed
		—	29.3.6 Setting of Disabling Error Output Clear, deleted
		877	29.4.1 Notes Regarding ECMCLK: Description changed
		Section 30 12-Bit A/D Converter (S12ADCa)	
		878	30.1 Overview, description corrected
		879 to 880	Table 30.1 Specifications of 12-Bit A/D Converter, Input channels and A/D conversion clock: Note 1 corrected to Note 2
		909	30.2.13 A/D Sample and Hold Circuit Control Register (ADSHCR), address corrected
		964	Figure 30.31 Procedures for Clear Operation by Software through the ADCSR.ADST Bit, “Is data transfer to EMU2 used?” deleted from the flow
		964	Figure 30.31 Procedures for Clear Operation by Software through the ADCSR.ADST Bit, “Disable data transfer from the ADC to EMU2.” deleted from the flow
		Section 31 Temperature Sensor	
		972	31.4.1 Module-Stop Function Setting, description corrected
		Section 32 Data Operation Circuit (DOC)	
		973	Figure 32.1 Block Diagram of Data Operation Circuit (DOC): Figure title changed
		Section 33 RAM (Product Option)	
		980	Table 33.1 Specifications of RAM: Caution added
		987	33.2.5 2-Bit ECC Error Address Register (RAMDBEAD): Description changed
		987	33.2.5 2-Bit ECC Error Address Register (RAMDBEAD) Symbol: ADDRESS of b31 to b20 in the bit chart was changed to “—”
		987	33.2.5 2-Bit ECC Error Address Register (RAMDBEAD) Symbol: ADDRESS of b19 and b18 in the bit chart was changed to “BANK”
		987	33.2.5 2 2-Bit ECC Error Address Register (RAMDBEAD) Symbol: ADDRESS of b31 to b20 in the table of bits was changed to “Reserved”
		987	33.2.5 2-Bit ECC Error Address Register (RAMDBEAD) Symbol: ADDRESS of b19 and b18 in the table of bits was changed to “BANK”
		990	33.3.3 (1) Example of ECC error-injection setting procedure Flow procedure “Enable error injection”, changed

Rev.	Date	Description	
		Page	Summary
0.80	Apr. 21, 2015	991	33.3.3 (2) Procedure for Checking ECC Operation After the flow procedure “Write 0000_0001h to the RAMEDC register”, add the procedure “Write 0000_0000h to the RAMPCMD register”.
		991	33.3.3 (2) Procedure for Checking ECC Operation The flow procedure “Read the RAMDBEAD register to acquire the address where the error was found”, changed
		991	33.3.3 (2) Procedure for Checking ECC Operation The flow procedure “Read the RAMDBECNT register to check the number of errors that were found”, changed
		991	33.3.3 (2) Procedure for Checking ECC Operation The flow procedure “Write 0000_0000h to the RAMPCMD register”, deleted
		Section 34 Electrical Characteristics (Target)	
		993	Figure 34.1 Power On/Off Sequence: Period of (2) Tdly12, corrected
		1002	Heading title: 34.4.4.1 I/O Port Timing, added
		1003	Heading title: 34.4.4.2 TPUa Timing, added
		1004	Heading title: 34.4.4.3 CMTW Timing, added
		1005	Heading title: 34.4.4.4 A/D Converter Trigger Timing, added
		1006	Heading title: 34.4.4.5 SCIFA Timing, added
		1008	Heading title: 34.4.4.6 RSPIa Timing, added
		1012	Heading title: 34.4.4.7 SPIBSC Timing, added
		1014	Figure 34.25 SPIBSC Transmit/Receive Timing (CPHAT = 0, SPHAR = 1), added
		1015	Figure 34.28 SPIBSC Buffer On/Off Timing (CPHAT = 0, SPHAR = 1), added
		1016	Heading title: 34.4.4.8 RIICa Timing, added
		1018	Table 34.21 12-Bit A/D (Unit 0) Conversion Characteristics Description on “Channel-dedicated sample-and-hold circuits in use (AN000 to AN003), When disconnection detection assistance is in use”, changed
		1018	Table 34.21 12-Bit A/D (Unit 0) Conversion Characteristics Description on “Channel-dedicated sample-and-hold circuits in use (AN000 to AN003), When disconnection detection assistance is not in use”, added
		1018	Table 34.21 12-Bit A/D (Unit 0) Conversion Characteristics Offset error (max.) of Channel-dedicated sample-and-hold circuits in use (AN000 to AN003), changed
		1018	Table 34.21 12-Bit A/D (Unit 0) Conversion Characteristics Full-scale error (max.) of Channel-dedicated sample-and-hold circuits in use (AN000 to AN003), changed
		1018	Table 34.21 12-Bit A/D (Unit 0) Conversion Characteristics Absolute accuracy (max.) of Channel-dedicated sample-and-hold circuits in use (AN000 to AN003), changed
		1018	Table 34.21 12-Bit A/D (Unit 0) Conversion Characteristics Conversion time (min.) and test condition of Channel-dedicated sample-and-hold circuits not in use (AN000 to AN007), changed
		1018	Table 34.21 12-Bit A/D (Unit 0) Conversion Characteristics INL integral nonlinearity error (max.) of Channel-dedicated sample-and-hold circuits not in use (AN000 to AN007), changed
		1019	Table 34.22 12-Bit A/D (Unit 1) Conversion Characteristics Conversion time (min.) and test condition, changed
		1020	Table 34.23 Temperature Sensor Characteristics: Typ. of output voltage, changed
0.90	Sep. 07, 2015	1. Overview	
		32	Table 1.1 Outline of Specifications, “TBD” in the category Power supply voltage deleted, “TBD” in the description changed to PLBG0112KA-A
		3. Operating Modes	
		52	Figure 3.2 Memory Assignment of the Loader Program and Parameters for the Loader, note 3 changed

Rev.	Date	Description	
		Page	Summary
0.90	Sep. 07, 2015	57	Table 3.6 Status of the ARM CP15 Registers at the Time the Boot Finishes, the item “Setting Value at the Time the Boot Processing Finishes” for system control auxiliary register, changed
		4. Address Space	
		61	Figure 4.1 Memory Map, erroneous descriptions corrected (DMA0 → DMAC0, Area for MDIO (4KB → 4MB)), note 4 replaced with the statement in the map “mirror area of xxx”
		62	Figure 4.2 Memory Map (1-Mbyte Extended Internal SRAM), erroneous description corrected (Area for MDIO (4KB → 4MB)), note 4 replaced with the statement in the map “mirror area of xxx”
		14. DMA Controller (DMACAa)	
		268	14.2.12 Channel Control Register n (CHCTRL_n), description for b0 (SETEN), note added
		270 to 273	14.2.13 Channel Configuration Register n (CHCFG_n), description changed, description for b30 (REN) changed
		321	14.5.2 Setting Example 2 (Register Mode Software Request), Setting Example 2, setting of CHCFG changed
		17. Multi-Function Pin Controller (MPC)	
		375	17.2.5 P3n Pin Function Control Register (P3nPFS), value after reset changed Table 17.5 Register Settings for the Input/Output Function, PSEL[5:0]Setting, value after reset changed
		23. Management Data Input/Output Interface (MDIO)	
		552	23.1.1 Overview, description changed (Technical document regarding MDIO changed to “CFP MSA Management Interface Specification Version 2.4 r06b June 8, 2015”)
		553	Table 23.1 MDIO I/O Pins, note added
		554	23.1.3 Interrupt Functions, description added
		554	23.1.4 Restrictions, description added
		558	23.3.1.4 Mode Register (MODE) added
		562	23.3.2.2 Version Register (VER) added
		24. Serial Communications Interface with FIFO (SCIFA)	
		575	Table 24.4 Bit Rates and BRR Register Settings in Asynchronous Mode, Bit Rate 110 bps deleted, 115200 bps and 500000 bps, settings for SERICLK changed (to blank), note changed
		576	Table 24.6 Maximum Bit Rates for Various Frequencies with Baud Rate Generator (in Asynchronous Mode), the maximum bit rate changed, note changed
		576	Table 24.8 Maximum Bit Rates with External Clock Input (in Asynchronous Mode), the maximum bit rate changed, note changed
		577	Table 24.10 Maximum Bit Rates and BRR and MDDR Registers Settings in Asynchronous Mode, added
		613	24.8.4 Break Signal Transmission, erroneous bit names corrected (SPB2DT → SPB2IO)
		27. SPI Multi I/O Bus Controller (SPIBSC)	
		776	27.2.1 Common Control Register (CMNCR), b24 (SFDE), value after reset changed
		33. RAM (Product Option)	
		983	Table 33.1 Specifications of RAM, the item “Data retention function” deleted
		34. Electrical Characteristics	
		995	Table 34.1 Absolute Maximum Rating, Operating temperature (junction temperature), value changed
		1015	Table 34.19 SPIBSC Timing, the minimum values for tSU and tH changed
		1019	Table 34.20 RIICa Timing, symbols changed (tr ' tsr, tf ' tsf), note 4 added
		1021	Table 34.21 12-Bit A/D (Unit 0) Conversion Characteristics, “TBD” cells filled with values
		1022	Table 34.22 12-Bit A/D (Unit 1) Conversion Characteristics, “TBD” cells filled with values
		Appendix 1. Outer Dimensions Diagram	
		1026	Newly added

Rev.	Date	Description	
		Page	Summary
1.00	Feb 17, 2016	All	Module symbol changed (S12ADC → S12ADCa)
		Features	
		28	Management Data Input/Output Interface (MDIO) added
		28	Operating temperature range, changed
		1. Overview	
		30	Table 1.1 Outline of Specifications, I/O ports: values are added in the T.B.D cells, Event link controller (ELC): notation as (T.B.D.) deleted, 12-bit A/D converter (S12ADCa): conversion times are changed, Temperature sensor: relative precision changed, Operating temperature: the upper limit value changed
		40	Table 1.5 Pin Assignments: pin names of A4, A5, A6, B4, B5, C4, C5 changed
		43	Table 1.6 List of Pin and Pin Functions (112-Pin FBGA), pin names are added to the I/O ports A4, A5, A6, B4, B5, C4, C5
		5. I/O Registers	
		65, 66	Table 5.1 List of I/O Registers (Address Order), A000 000Ah (PDR), A000 0085h (PMR), A000 010Ah (PCR), A000 0228h (P50PFS), A000 0229h (P51PFS), A000 022Ah (P52PFS), A000 022Bh (P53PFS), A000 022Ch (P54PFS), A000 022Dh (P55PFS), and A000 022Eh (P56PFS) are added
		9. Low-Power Consumption Function	
		144	9.2.4 Module Stop Control Register E (MSTPCRE), b1 to b3, descriptions changed
		10. Debugging Interface	
		159	Table 10.8 Available Trace Functions, the module name changed
		12. Interrupt Controller (ICUA)	
		170	12.2.6 NMI Pin Interrupt Control Register (NMICR), the description changed
		214	12.4.2.13 Interrupt Address Register (HVA0), descriptions partially changed
		238	Figure 12.5 Initializing Registers of VIC: content changed, a note added
		242	Figure 12.7 IRQ Interrupt Operation (Level Operations), register name changed (HVA → HVA0)
		243	Figure 12.8 IRQ Interrupt Operation (Edge Interrupt), register name changed (HVA → HVA0)
		245	Figure 12.9 Concept of Multiple Interrupts (1 / 2), descriptions partially changed
		246	Figure 12.9 Concept of Multiple Interrupts (2 / 2), descriptions added
		248	12.4.6.3 Notes on Selecting Level Detection, descriptions partially changed, an example program added
		14. DMA Controller (DMACAa)	
		267	14.2.11 Common Control Register (CMNCR), b30, b29, b28, b26, b25, and b24 are changed to reserved bits
		276	14.2.17 Source Continuous Register n (SCNT_n), descriptions partially changed
		279	14.2.19 Destination Continuous Register n (DCNT_n), descriptions partially changed
		282	14.2.21 DMA Control Register (DCTRL_X (X = A or B)), b31-b1 (reserved bits) are changed
		319	Table 14.23 DMA Transfer Setting Example 1, start address changed, description of DMA transfer request partially changed, the item “Selection of the side which makes a DMA transfer request” added
		319	14.5.1 Setting Example 1 (Register Hardware Request), Setting Example: values for N0SA and CHCFG are changed
		320	Figure 14.19 Setting Example 1 changed
		16. I/O Ports	
		361	16.3.1 Port Direction Register (PDR), the address PORT5.PDR A000 000Ah added, the suffix m changed
		364	16.3.4 Port Mode Register (PMR), the address PORT5.PMR A000 0085h added, the suffix m changed
		365	16.3.5 Pull-Up/Pull-Down Resistor Control Register (PCR), the address PORT5.PCR A000 010Ah added, the suffix m changed

Rev.	Date	Description	
		Page	Summary
1.00	Feb 17, 2016	17. Multi-Function Pin Controller (MPC)	
		370	Table 17.1 List of Multiplexed Pin Configurations, the module management data input/output interface (MDIO) added
		375	17.2.5 P3n Pin Function Control Register (P3nPFS), values after reset changed, notes 1 and 2 added
		377	17.2.7 P5n Pin Function Control Register (P5nPFS) (n = 0 to 6) newly added
		23. Management Data Input/Output Interface (MDIO)	
		553	Figure 23.1 Block Diagram of the MDIO, contents changed
		554	Table 23.1 MDIO I/O Pins, note 1 for the PRTADR3 and PRTADR4 pins added
		555	23.1.4 Restrictions, descriptions for a case for CFP4 added
		561	23.2.2.1 Receive Register (RX), b15 to b0 (RXD[15:0]), descriptions changed
		564	23.3 Operational Outline newly added
		564	23.3.1 MDIO Startup Sequence added (moved from 23.2)
		565	23.3.2 Flowchart of MDIO Processing newly added
		24. Serial Communications Interface with FIFO (SCIFA)	
		585	24.2.12 Serial Port Register (SPTR), descriptions changed, a note added
		597	Figure 24.4 Sample Flowchart for Transmitting Serial Data in Asynchronous Mode, a note added
		615	24.8.4 Writing to the SPTR Register added
		26. Serial Peripheral Interface (RSPIa)	
		705	26.2.5 RSPI Data Register (SPDR), the bit map for 16 higher-order bits (H) added
		29. Error Control Module (ECM)	
		827	Figure 29.1 Block Diagram of ECM changed
		30. 12-Bit A/D Converter (S12ADCa)	
		884	Table 30.1 Specifications of 12-Bit A/D Converter (1 / 2), conversion time changed, numbering of notes changed
		970	30.5.9 Allowable Signal Source Impedance deleted
		34. Electrical Characteristics	
		997	Figure 34.1 Power On/Off Sequence, notes 1 and 3 are changed
		998	34.3 DC Characteristics, conditions changed
		998	Table 34.2 DC Characteristics (1), typical value for VCCQ12 changed
		998	Table 34.3 DC Characteristics (2) [Power Supply], values are added in the T.B.D. cells, Standby mode with all modules inactive (reference value): the unit for AV0Icc, AV1Icc, VRF0Icc, and VRF1Icc are changed, note 2 added
		999	Table 34.5, title changed (DC Characteristics (4) [MDIO] → DC Characteristics for 1.2-V pin), items are changed (permissible output high current → output high level current, permissible output low level current → output low level current), symbols are changed, values are added in the T.B.D. cells, prerequisites and target pins are respectively changed to notes 1 and 2.
		1001	34.4 AC Characteristics, conditions changed
		1001	1.2 V I/O clock cycle added
		1021	34.5 A/D Conversion Characteristics, conditions added
		1023	34.6 Temperature Sensor Characteristics, conditions added
		1025	Figure 34.33 Trace Interface Timing changed
1.10	Aug. 12, 2016	Features	
		28	Description on the data transfer and the clock, modified
		1. Overview	
		29	1.1 Outline of Specifications: As for the number of peripheral module channels, the reference modified

Rev.	Date	Description	
		Page	Summary
1.10	Aug. 12, 2016	29	Table 1.1 Outline of Specifications (1 / 4) Clock generation circuit: Functional description on the input clock, modified
		29	Table 1.1 Outline of Specifications (1 / 4) Clock generation circuit: External clock signal, deleted
		29	Table 1.1 Outline of Specifications (1 / 4) Vector interrupt controller (VIC): Number of peripheral function interrupts and external interrupts, modified
		30	Table 1.1 Outline of Specifications (2 / 4) General-purpose I/O ports: Number of I/O pins, modified. Open drain outputs, deleted.
		30	Table 1.1 Outline of Specifications (2 / 4) 16-bit timer pulse unit (TPUa): Pulse-input/output, modified
		31	Table 1.1 Outline of Specifications (3 / 4) Serial communication interface with FIFO (SCIFA): Note 2 added to the serial communications modes
		32	Table 1.1 Outline of Specifications (4 / 4) 12-bit A/D converter (S12ADCa): Conversion time, modified
		32	Table 1.1 Outline of Specifications (4 / 4) Error control module (ECM): Some of the operations for the error signal input from each module, deleted
		33	Table 1.1 Outline of Specifications (4 / 4) Error control module (ECM): Duplicated target in the master and the checker, modified
		33	Table 1.1 Outline of Specifications (4 / 4) : Power supply voltage, modified
		34	Table 1.2 List of Functions: Table title, modified
		34	Table 1.2 List of Functions Serial communications interface with FIFO (SCIFA): Note 2 added to ch4
		35	1.2 List of Products: Note 2 deleted
		35	Table 1.3 List of Products R7S910020CBG, R7S910021CBG: Description added to the security function
		36	Figure 1.1 Block Diagram SCIFA: Note 1 added
		36	Figure 1.1 Block Diagram 12-bit A/D converter (unit 1): Number of channels, modified
		37	Table 1.4 Pin Functions (1 / 3) Power supply VSSQ33, deleted
		37	Table 1.4 Pin Functions (1 / 3) XTAL: Description modified
		37	Table 1.4 Pin Functions (1 / 3) Debugging interface: Description on TRST#, TMS, TDI, TDO, and TCK, modified
		37	Table 1.4 Pin Functions (1 / 3) 16-bit timer pulse unit (TPUa): TIOCA2, TIOCA3, and TIOCA5, added
		37	Table 1.4 Pin Functions (1 / 3) Serial communication interface with FIFO (SCIFA): Note 1 added to RXD4 and TXD4
		39	Table 1.4 Pin Functions (3 / 3) I/O ports: P35 added. Functional description, modified.
		39	Table 1.4 Pin Functions (3 / 3) I/O ports: I/O and functional description of PC2, PC3, PC6, and PC7, modified
		40	Figure 1.2 Pin Arrangement (112-pin FBGA) (Top View): Figure title, modified
		40	Figure 1.2 Pin Arrangement (112-pin FBGA) (Top View): A-1, E-4, F-4,9, G-2,4, H-3,7, J-3, and K-3, modified
		41	Table 1.5 Pin Assignments (112-Pin FBGA) (1 / 3) Pin number A1: Pin name, modified
		41	Table 1.5 Pin Assignments (112-Pin FBGA) (1 / 3) Pin number A4, A5, A6, B4, B5, C4, and C5: Note 1 added to the pin names

Rev.	Date	Description	
		Page	Summary
1.10	Aug. 12, 2016	42	Table 1.5 Pin Assignments (112-Pin FBGA) (2 / 3) Pin number E4, F4, F9, G2, G4, H3, H7, and J3: Pin names, modified
		43	Table 1.5 Pin Assignments (112-Pin FBGA) (3 / 3) Pin number K3 and L5: Pin names, modified
		44	Table 1.6 List of Pins and Pin Functions (112-Pin FBGA) (1 / 3) Pin number A1, A9 and B11: Power supply clock system control, modified
		44	Table 1.6 List of Pins and Pin Functions (112-Pin FBGA) (1 / 3) Pin number A4, A5, A6, B4, B5, C4, and C5: Note 1 added to I/O port
		45	Table 1.6 List of Pins and Pin Functions (112-Pin FBGA) (2 / 3) Pin number E4, F4, F9, G2, G4, H3, and H7: Power supply clock system control, modified
		45	Table 1.6 List of Pins and Pin Functions (112-Pin FBGA) (2 / 3) Pin number J3: Power supply clock system control, modified
		46	Table 1.6 List of Pins and Pin Functions (112-Pin FBGA) (3 / 3) Pin number K3: Power supply clock system control, modified
		46	Table 1.6 List of Pins and Pin Functions (112-Pin FBGA) (3 / 3) Pin number L5: Description on the timer, added
		3. Operating Modes	
		53	Figure 3.2 Memory Assignment of the Loader Program and Parameters for the Loader Description of offset address at 0000 0004h and 0000 0008h, modified
		55	3.5.3 Loader Program: Description of the storage address in the external memory, modified
		59	Figure 3.4 Relationship Between the Memory Map Definition During the Boot Processing and the Default Memory Map of Cortex-R4F Address map: 3000 0000h to 3400 0000h, modified
		59	Figure 3.4 Relationship Between the Memory Map Definition During the Boot Processing and the Default Memory Map of Cortex-R4F Address map: 4000 0000h to 4400 0000h, deleted
		4. Address Space	
		61	4.1 Address Space: Some of the description, deleted (Figure 4.3)
		62	Figure 4.1 Memory Map, deleted
		62	Figure 4.1 Memory Map (1-Mbyte Extended Internal SRAM): Note 1 modified
		63	Figure 4.2 Memory Map (0-Kbyte Extended Internal SRAM): Note 1 modified
		63	Figure 4.2 Memory Map (0-Kbyte Extended Internal SRAM): Note 4 and Note 5 deleted
		5. I/O Registers	
		66	Table 5.1 List of I/O Registers (2 / 42) Address at A000 0142h, deleted
		98	Table 5.1 List of I/O Registers (34 / 42) Address at A007 D0A8h, A007 D0ACh, and A007 D0B0h, deleted
		101	Table 5.1 List of I/O Registers (37 / 42) Address at A008 0410h, deleted
		103	Table 5.1 List of I/O Registers (39 / 42) Address at A008 C012h, deleted
		6. Reset	
		107	Table 6.2 Targets to be Initialized for Each Reset Type Reset target: ECM error output clear disable setting register, deleted
		7. Clock Generation Circuit	
		113	Table 7.1 Specifications of Clock Generation Circuit Main clock oscillator: External clock input frequency, deleted
		115	Figure 7.1 Block Diagram of Clock Generation Circuit Input signal to the selector, controlled by CKSEL0.SCKCR2, modified
		116	Table 7.3 Pin Configuration of Clock Generation Circuit: Functional description on XTAL and EXTAL, modified
		116	Table 7.3 Pin Configuration of Clock Generation Circuit: Pin name OSCTH, deleted

Rev.	Date	Description	
		Page	Summary
1.10	Aug. 12, 2016	118	7.2.2 System Clock Control Register 2 (SCKCR2) Bit b0: External bus clock was deleted from the functional description
		124	7.3 Input to Main Clock Oscillator: Section title, modified
		124	7.3 Input to Main Clock Oscillator: Functional description, modified
		124	Table 7.5 Clock Input Mode Selected for OSCTH Pin, deleted
		125	7.3.2 External Clock Input: Description deleted
		129	7.9.1 Notes on Clock Generation Circuit (4) Channels of SCIFA, modified
		9. Low-Power Consumption Function	
		139	Table 9.2 Stopping Peripheral Modules and Exiting Module-Stop State Module: Management data input/output interface (MDIO), added
		139	Table 9.2 Stopping Peripheral Modules and Exiting Module-Stop State: Note 2 (in Rev. 1.00) deleted
		140	Table 9.2 Stopping Peripheral Modules and Exiting Module-Stop State: Note 2 modified
		141	9.2.1 Module Stop Control Register A (MSTPCRA) Bits b2 and b7: Symbol, functional description, modified
		143	9.2.3 Module Stop Control Register C (MSTPCRC) Bit b7: Value after reset, Note 1 deleted
		143	9.2.3 Module Stop Control Register C (MSTPCRC): Note 1 modified
		144	9.2.4 Module Stop Control Register E (MSTPCRE) Bit b0: Symbol, functional description, modified
		11. Register Write Protection Function	
		160	Table 11.1 Correspondence between PRCR Register Bits and Registers to be Protected PRCR register, PRC1 bit: MSTPCRD and MRCTLC were deleted from the registers to be protected
		12. Interrupt Controller (ICUA)	
		163	Figure 12.1 Block Diagram of Interrupt Controller: Arrow from the request source selection, deleted
		173	12.3.1 Selecting Interrupt Request Destinations: VIC/NVIC modified to VIC
		230	Table 12.3 Cortex-R4F/DMAC Interrupt Vector Table (1 / 8) Vector number 9: Detection type, CR4, and DMAC, modified
		232	Table 12.3 Cortex-R4F/DMAC Interrupt Vector Table (3 / 8) Vector number 88 to 91: Request source, source, detection type, CR4, and DMAC, modified
		236	Table 12.3 Cortex-R4F/DMAC Interrupt Vector Table (7 / 8) Vector number 248 to 250: Request source, source, detection type, CR4, and DMAC, modified
		236	Table 12.3 Cortex-R4F/DMAC Interrupt Vector Table (7 / 8) Vector number 267 to 272: Request source, source, detection type, CR4, and DMAC, modified
		237	Table 12.3 Cortex-R4F/DMAC Interrupt Vector Table (8 / 8) Vector number 273 to 292: Request source, source, detection type, CR4, and DMAC, modified
		248	12.4.6.2 Notes on Accessing HVA0 Register: modified
		13. Internal Buses	
		251	Table 13.1 Specifications of Internal Buses Internal bus type: Names of peripheral bus 1, peripheral bus 2, peripheral bus 3, peripheral bus 4, and peripheral bus 5, modified
		251	Figure 13.1 Bus Configuration: RAM1 and RAM2, added
		251	Figure 13.1 Bus Configuration: Names of peripheral bus 1, peripheral bus 2, peripheral bus 3, peripheral bus 4, and peripheral bus 5, modified
		251	Figure 13.1 Bus Configuration: Note 1, added
		252	Table 13.3 Internal Main Bus 2: Connection between Bus Master and Bus Slave Bus slave, extended internal RAM: Name modified
		252	Table 13.3 Internal Main Bus 2: Connection between Bus Master and Bus Slave Bus slave, data RAM: Description deleted

Rev.	Date	Description	
		Page	Summary
1.10	Aug. 12, 2016	252	Table 13.3 Internal Main Bus 2: Connection between Bus Master and Bus Slave Names of peripheral bus 1, peripheral bus 2, peripheral bus 3, peripheral bus 4, and peripheral bus 5, modified
		14. DMA Controller (DMACAa)	
		—	14.2.11 Common Control Register (CMNCR) (in Rev. 1.00): Description deleted
		281	Figure 14.2 Relationship between DSKP and DCNT: Figure title, modified
		295	14.3.1.2 Link Mode: Description modified
		295	Figure 14.7 Link Mode Overview: Description modified (memory)
		307	14.3.4.1 Specifying Detection Operation of DMA Transfer Requests for Each Source: Some of the description, deleted (external requests)
		318	Figure 14.18 Stop Timing for Responding to a Bus Error: DREQ[0] and DACK[0] signals, deleted
		—	Table 14.22 List of Transfer Conditions for the DMA Transfer Setting Examples Setting example: Setting example 1 (in Rev. 1.00), deleted
		319	Table 14.22 List of Transfer Conditions for the DMA Transfer Setting Examples Setting example: Setting example 1, setting example 2, and setting example 3 were modified from setting example 2, setting example 3, and setting example 4 (in Rev. 1.00).
		—	14.5.1 Setting Example 1 (Register Mode Software Request) (in Rev. 1.00): Description deleted
		320	14.5.1 Setting Example 1 (Register Mode Software Request): Section title, modified
		320	Table 14.23 DMA Transfer Setting Example 1: Start address of the transfer source, modified
		320	14.5.1 Setting Example 1 (Register Mode Software Request): N1SA address of setting example 1, modified
		321	Figure 14.19 Setting Example 1: Figure title, modified
		321	Figure 14.19 Setting Example 1: Description modified (Start (setting example 1), source address, N1SA_2 address, end (setting example 1))
		322	14.5.2 Setting Example 2 (Register Mode Continuous Execution): Section title, modified
		322	Table 14.24 DMA Transfer Setting Example 2: Start address of the transfer source, modified
		322	14.5.2 Setting Example 2 (Register Mode Continuous Execution): N0DA, N1SA, and N1DA addresses of setting example 2, modified
		323	Figure 14.20 Setting Example 2: Figure title, modified
		323	Figure 14.20 Setting Example 2: Description modified (Start (setting example 2), source address, N1SA_1 address, end (setting example 2))
		324	14.5.3 Setting Example 3 (Link Mode): Section title, modified
		324	14.5.3 Setting Example 3 (Link Mode): Setting example 3, modified
		324	Table 14.25 DMA Transfer Setting Example 3: Table title, modified
		324	Table 14.26 DMA Transfer Setting Example 3 (Descriptor 1): Table title, modified
		325	Table 14.27 DMA Transfer Setting Example 3 (Descriptor 2): Table title, modified
		325	Table 14.27 DMA Transfer Setting Example 3 (Descriptor 2): Start address of the transfer source, modified
		325	Table 14.28 DMA Transfer Setting Example 3 (Descriptor 3): Table title, modified
		325	Table 14.28 DMA Transfer Setting Example 3 (Descriptor 3): Start addresses of the transfer source and transfer destination, modified
		326	Table 14.29 Descriptor Settings: SA (source address) of Descriptor 2 and Descriptor 3, modified
		326	Table 14.29 Descriptor Settings: DA (destination address) of Descriptor 3, modified
		326	Figure 14.21 Setting Example 3 : Figure title, modified
		326	Figure 14.21 Setting Example 3: Description modified (start (setting example 3), end Z(setting example 3)
		328	Figure 14.23 Setting Example 4: Figure title, modified
		15. Event Link Controller (ELC)	
		346	Figure 15.3 Event Linkage Related to Single Ports (Port E): Figure title, modified

Rev.	Date	Description	
		Page	Summary
1.10	Aug. 12, 2016	346	Figure 15.3 Event Linkage Related to Single Ports (Port E): Port name, modified
		347	Figure 15.4 Event Linkage Related to Input Port Groups (Port E): Figure title, modified
		347	Figure 15.4 Event Linkage Related to Input Port Groups (Port E): Port name, modified
		349	Figure 15.5 Event Linkage Related to Output Port Groups (Port E): Figure title, modified
		349	Figure 15.5 Event Linkage Related to Output Port Groups (Port E): Port name, modified
		350	Figure 15.6 Bit-Rotating Operation of Output Port Groups (Port E): Figure title, modified
		350	Figure 15.6 Bit-Rotating Operation of Output Port Groups (Port E): Port name, modified
		16. I/O Ports	
		354	Table 16.2 Port Functions: Switching of driving ability for PORT1, modified
		365	16.3.6 Driving Ability Control Register (DSCR): Some of the description, deleted (P10)
		365	16.3.6 Driving Ability Control Register (DSCR) Address: Some of the description, deleted (PORT1.DSCR A000 0142h)
		365	16.3.6 Driving Ability Control Register (DSCR) Bit, b0 to b13: Functional description, modified
		366	Table 16.3 Handling of Unused Pins: EXTAL, XTAL, RES#, and port 33 (TDO), deleted
		366	Table 16.3 Handling of Unused Pins: Ports 0 to 4, 6, 7, 9, A, C, E, and G (except port 34), modified
		366	Table 16.3 Handling of Unused Pins: Port 5, added
		17. Multi-Function Pin Controller (MPC)	
		367	17.1 Overview: Some of the description, deleted
		370	17.2 Register Descriptions: Description deleted
		374	17.2.5 P3n Pin Function Control Register (P3nPFS) (n = 3 to 5): Description modified
		374	17.2.5 P3n Pin Function Control Register (P3nPFS) (n = 3 to 5) ISEL Bit (Interrupt Input Function Select): Description modified
		375	17.2.6 P4n Pin Function Control Register (P4nPFS) (n = 0, 2, 4) Bit b6: Symbol, bit name, and functional description, modified
		377	17.2.8 P6n Pin Function Control Register (P6nPFS) (n = 0 to 5) Bit b6: Symbol, bit name, and functional description, modified
		380	17.2.11 PAn Pin Function Control Register (PAnPFS) (n = 3 to 5) Bit b6: Symbol, bit name, and functional description, modified
		381	17.2.12 PCn Pin Function Control Register (PCnPFS) (n = 2, 3, 6, 7) Bit b6: Symbol, bit name, and functional description, modified
		385	Table 17.15 Register Settings Interrupt input (NMI, IRQ0 to IRQ4, IRQ6, IRQ7): Description modified
		18. 16-Bit Timer Pulse Unit (TPUa)	
		388	18. 16-Bit Timer Pulse Unit (TPUa): Description modified
		388	Table 18.1 Specifications of TPU Settable operations: Some of the description, deleted (Internal PWM feedback input select)
		389	Table 18.2 TPU Functions (1 / 2) DMAC activation of TPU0, TPU1, TPU2, TPU3, TPU4, and TPU5: Description added (TGRB)
		393	18.2.1 Timer Control Register (TCR): Description modified (6 TCR registers in total)
		445	18.4 Interrupt Sources, (2) Overflow Interrupt: Description modified
		446	18.5 DMAC Activation: Description modified
		19. Compare Match Timer (CMT)	
		469	Table 19.1 CMT Specifications: Description on the number of internal channels, modified
		469	Figure 19.1 Block Diagram of CMT (Unit 0, Unit 1): Figure title, modified
		20. Compare Match Timer W (CMTW)	
		485	Figure 20.1 Block Diagram of CMTW: Some of the description, deleted (ECMDMESLR)
		494	20.2.10 ECM Dynamic Mode Error Output Select Register (ECDMESLR): Description deleted

Rev.	Date	Description	
		Page	Summary
1.10	Aug. 12, 2016	503	20.3.10 ECM Dynamic Mode Error Output Selection: Description deleted
		24. Serial Communications Interface with FIFO (SCIFA)	
		564	Table 24.1 Specifications of SCIFA Serial communication method: Description added (Note 2)
		569	24.2.6 Serial Control Register (SCR) Bits b1-b0: Description added (Note 2)
		591	24.3.2 Operation in Asynchronous Mode, (2) Clock: Some of the description, deleted
		601	24.3.3 Operation in Clock Synchronous Mode, (2) Clock: Some of the description, deleted
		25. I^2C Bus Interface (RIICa)	
		617	Figure 25.1 RIIC Block Diagram: Some of the description, deleted (FMPE at the top right corner of the output control block)
		618	Figure 25.2 Connections to the External Circuit by the I/O Pins (I^2C Bus Configuration Example): Some of the description, modified (VCCQ33)
		619	25.2.1 I^2C Bus Control Register 1 (ICCR1): Description added ((n = 0, 1))
		659	25.3.5 Slave Transmit Operation, (2) and (6): Some of the description, deleted (HOA)
		662	25.3.6 Slave Receive Operation, (2) and (5): Some of the description, deleted (HOA)
		27. SPI Multi I/O Bus Controller (SPIBSC)	
		777, 778	27.2.1 Common Control Register (CMNCR) Bits b17-b16: Bit name and functional description, modified
		777, 778	27.2.1 Common Control Register (CMNCR) Bits b19-b18, b21-b20, b23-b22: Bit name, modified
		786	27.2.7 Data Read Option Setting Register (DROPR) Bits b7-b0: Bit name, modified
		29. Error Control Module (ECM)	
		824	Table 29.1 Specifications of ECM Self-diagnosis: Some of the description, deleted
		825	Figure 29.1 Block Diagram of ECM, modified
		827	Table 29.2 ECM Error Input (2 / 2) Error source number 96: Functional description, modified
		829	29.2.1 ECM Master/Checker Error Source Status Register 0 (ECMmESSTR0 (m = M or C)) Bits b23 and b24: Functional description, modified
		831	29.2.3 ECM Master/Checker Error Source Status Register 2 (ECMmESSTR2 (m = M or C)): Some of the description, modified or deleted
		831	29.2.3 ECM Master/Checker Error Source Status Register 2 (ECMmESSTR2 (m = M or C)) Bit b11: Symbol, bit name, and functional description, modified
		834, 835	29.2.5 ECM Maskable Interrupt Configuration Register 0 (ECMMICFG0) Bits b23 and b24: Functional description, modified
		835	29.2.5 ECM Maskable Interrupt Configuration Register 0 (ECMMICFG0) Bit b31: Description on the bit name, modified
		839, 840	29.2.8 ECM Non-maskable Interrupt Configuration Register 0 (ECMNMICFG0) Bits b23 and b24: Functional description, modified
		839, 840	29.2.11 ECM Internal Reset Configuration Register 0 (ECMIRCFG0) Bits b23 and b24: Functional description, modified
		840	29.2.11 ECM Internal Reset Configuration Register 0 (ECMIRCFG0) Bit b31: Description on the bit name, modified
		—	29.2.14 ECM Error Mask Register 0 (ECMEMK0) (in Rev. 1.00): Description deleted
		849, 850	29.2.14 ECM Error Source Status Clear Trigger Register 0 (ECMESSTC0) Bits b23 and b24: Functional description, modified
		—	29.2.15 ECM Error Mask Register 1 (ECMEMK1) (in Rev. 1.00): Description deleted
		—	29.2.16 ECM Error Mask Register 2 (ECMEMK2) (in Rev. 1.00): Description deleted
		853	29.2.18 ECM Protection Status Register (ECMPS): Description, R/W, modified

Rev.	Date	Description	
		Page	Summary
1.10	Aug. 12, 2016	855	29.2.19 ECM Pseudo Error Trigger Register 0 (ECMPE0) Bits b23 and b24: Functional description, modified
		862	29.2.25 ECM Delay Timer Configuration Register 0 (ECMDTMCFG0) Bits b23 and b24: Functional description, modified
		868	29.2.28 ECM Delay Timer Configuration Register 3 (ECMDTMCFG3) Bits b23 and b24: Functional description, modified
		873	Table 29.3 Example Assignment of Errors for Functional Safety to Error Sources: Note added
		873	29.3.2.1 Protection Unlock Sequence 1. Description modified or added
		874	29.3.3 Timeout Function for Interrupt Processing by Using the Delay Timer: Description modified
		875	29.4.1 Notes Regarding ECMCLK: Description modified
		30. 12-Bit A/D Converter (S12ADCa)	
		877	Table 30.1 Specifications of 12-Bit A/D Converter (1 / 2): Conversion time, modified
		877	Table 30.1 Specifications of 12-Bit A/D Converter (1 / 2) A/D conversion clock: Some of the description, deleted (CPG)
		879	Table 30.2 Functions of 12-Bit A/D Converter: Some of the description, deleted (Note 1)
		881	Table 30.3 I/O Pins of 12-Bit A/D Converter: Functional description on VREFH1, VREFL1 and AN100 to AN107, modified
		903	30.2.10 A/D Conversion Extended Input Control Register (ADEXICR): Section title, modified
		—	30.2.11 A/D Conversion Extended Input Control Register (ADEXICR) Unit1 (without Temperature Sensor and with Extended Analog Input) (in Rev. 1.00): Description deleted
		949	Table 30.11 Times for Conversion during Scanning (in Numbers of Cycles of the ADCLK and PCLKH): t_{SPLSH}, t_{CONV}, and Note 3, added
		962	30.5.10 Allowable Signal Source Impedance: Description added
		31. Temperature Sensor	
		963	Figure 31.1 Block Diagram of Temperature Sensor: Description modified (Internal peripheral bus 4)
		967	Figure 31.2 Procedure for Using the Temperature Sensor: Some of the description, deleted (Flow for setting the number of sampling states)
		33. RAM (Product Option)	
		985	33.3.1 Configuration of Memory Map: Description modified
		985	Figure 33.1 Configuration of Memory: Some of the description, modified ((1) Extended internal SRAM, (2) Extended internal SRAM)
		34. Electrical Characteristics	
		988	Table 34.1 Absolute Maximum Rating: V_{in1}, VREFH0, and VREFH1, modified
		988	Table 34.1 Absolute Maximum Rating: Input voltage (1.2 V I/O port), V_{in3}, added
		988	Table 34.1 Absolute Maximum Rating: Note 1, modified. Note 4 and Note 5, added.
		989	Figure 34.1 Power On/Off Sequence: Some of the description, modified (Timing, No.)
		990	34.3 DC Characteristics: Conditions modified
		990	Table 34.3 DC Characteristics (2) [Power Supply]: Description added to the test conditions for normal operation, VDD
		991	Table 34.4 DC Characteristics (3): Some of the description, deleted from the input pull-up MOS current and the input pull-down MOS current
		992	Table 34.6 DC Characteristics for 12-Bit A/D Converter, added
		994	34.4 AC Characteristics: Conditions modified
		—	Table 34.8 EXTAL Clock Timing (in Rev. 1.00): Description deleted
		—	Figure 34.3 EXTAL External Clock Input Timing (in Rev. 1.00): Description deleted
		998	Figure 34.10 TPUa Clock Input Timing: Description modified (TCLKA to TCLKD)
		1001	Figure 34.14 SCK Clock Input Timing: Description modified (n = 0 to 2)

Rev.	Date	Description	
		Page	Summary
1.10	Aug. 12, 2016	1002	Figure 34.15 SCIFA Input/Output Timing/Clock Synchronous Mode: Description modified (n = 0 to 2)
		1004	Figure 34.16 RSPIa Clock Timing: Description modified (n = 0, 1)
		1004	Figure 34.17 RSPIa Timing (Master, CPHA = 0): Description modified (SSL00, 01, 02, 03, SSL10, 11 output (n = 0, 1))
		1005	Figure 34.18 RSPIa Timing (Master, CPHA = 1): Description modified (SSL00, 01, 02, 03, SSL10, 11 output (n = 0, 1))
		1005	Figure 34.19 RSPI Timing (Slave, CPHA = 0): Description modified (SSL00, 01, 02, 03, SSL10, 11 input (n = 0, 1))
		1006	Figure 34.20 RSPI Timing (Slave, CPHA = 1): Description modified (SSL00, 01, 02, 03, SSL10, 11 input (n = 0, 1))
		1007	Table 34.18 SPIBSC Timing: Description, Test Conditions modified
		1012	Figure 34.28 RIICa Bus Interface Input/Output Timing: Description modified (SDA0, SDA1 and SCL0, SCL1)
		1013	34.4.3.9 Serial Management Interface, added
		1014	34.5 A/D Conversion Characteristics: Conditions modified
		1014	Table 34.21 12-Bit A/D (Unit 0) Conversion Characteristics 1: Table title, modified
		1015	Table 34.22 12-Bit A/D (Unit 1) Conversion Characteristics 1: Table title, modified
		1016	Table 34.23 12-Bit A/D (Unit 0) Conversion Characteristics 2, added
		1016	Table 34.24 12-Bit A/D (Unit 1) Conversion Characteristics 2, added
		1019	Figure 34.34 Trace Interface Timing: Description modified (TRACECTL)
1.20	Mar. 02, 2017	1. Overview	
		29	1.1 Outline of Specifications: Description modified
		32	Table 1.1 Outline of Specifications (4 / 4), 12-bit A/D converter: Conversion time modified
		42	Table 1.5 Pin Assignments (112-Pin FBGA) (2 / 3), Pin number E4: Pin name modified
		44	Table 1.6 List of Pins and Pin Functions (112-Pin FBGA) (1 / 3), Pin number B11: Pin name for power supply, clock, and system control, modified
		45	Table 1.6 List of Pins and Pin Functions (112-Pin FBGA) (2 / 3), Pin number H3: Pin name for power supply, clock, and system control, modified
		3. Operating Modes	
		56	3.5.4.1 Operation Settings in SPI Boot Mode: Operation settings, added
		61	3.5.7.2 Serial Flash Memory in SPI Boot Mode, added
		10. Debugging Interface	
		160	Figure 10.7 Example of Connection Circuit of an Emulator That Can Drive the nTRST Output to High: Waveforms when the emulator is not connected and when the emulator is connected, modified
		12. Interrupt Controller (ICUA)	
		169	12.2.4 Non-maskable Interrupt Status Register (NMISR), Description on the NMIST flag: [Setting condition] modified
		177	12.3.4 NMI Pin Interrupts: Description modified
		222	12.4.2.15 Interrupt Service Current Register n (ISCn) (n = 0 to 9), ISC1, Description on the ISCi bit: Suffix (n) modified
		223	12.4.2.15 Interrupt Service Current Register n (ISCn) (n = 0 to 9), ISC3, Description on the ISCi bit: Suffix (n) modified
		224	12.4.2.15 Interrupt Service Current Register n (ISCn) (n = 0 to 9), ISC5, Description on the ISCi bit: Suffix (n) modified
		225	12.4.2.15 Interrupt Service Current Register n (ISCn) (n = 0 to 9), ISC7, Description on the ISCi bit: Suffix (n) modified
		226	12.4.2.15 Interrupt Service Current Register n (ISCn) (n = 0 to 9), ISC9, Description on the ISCi bit: Suffix (n) modified

Rev.	Date	Description	
		Page	Summary
1.20	Mar. 02, 2017	227	12.4.2.16 Interrupt Address Store Register 0 (VADn) (n = 1 to 255) Interrupt Address Store Register 1 (VADn) (n = 256 to 294): Suffix deleted. Functional description on the VAD[31:0] bits in the table of bits, modified. Description on the VAD bit, modified.
		234	Table 12.3 Cortex-R4F/DMAC Interrupt Vector Table (4 / 9), Vector number 132: Source of INT_ETC, modified
		239	Table 12.3 Cortex-R4F/DMAC Interrupt Vector Table: Note 2. added
		242	Figure 12.7 Register Rewrite Flow, modified
		251	12.4.6.5 Using Falling-Edge Detection with the NMI Pin, added
		14. DMA Controller (DMACAa)	
		308	Table 14.18 Detection Operation Specification for Each Source of DMA Transfer Requests: Detection operation specification of DMA transfer requests for the external interrupt, modified
		16. I/O Ports	
		367	Table 16.3 Handling of Unused Pins: Handling of the TRST# and TCK pins, modified
		17. Multi-Function Pin Controller (MPC)	
		371	17.2.1 Write-Protect Register (PWPR): Register description modified (PFSW bit → PFSWE bit)
		19. Compare Match Timer (CMT)	
		472	19.2.3 Compare Match Timer Control Register (CMCR): b7 (reserved bit) was added to the table of bits.
		23. Management Data Input/Output Interface (MDIO)	
		552	23.1.1 Overview: Reference document, modified
		552	Figure 23.1 Block Diagram of the MDIO, modified
		553	Table 23.1 MDIO I/O Pins: Note 1, added. Note 2, modified.
		553	Table 23.2 Interrupt Sources, modified
		554	23.1.3 Interrupt Functions: Detailed description on interrupts, modified
		556	23.2.1.2 Interrupt Clear Register (CL) Bit chart: b7 (CLPHY), b2 (CLPHYNM), and b1 (CLDEVNM) were added. Table of bits: b1 (CLDEVNM), b2 (CLPHYNM), and b7 (CLPHY) were added. Bit name and functional description on b5 (CLDEV) were modified
		557	23.2.1.3 Control Register (CTL): Functional description on the STMDIO bit in the table of bits, modified
		558	23.2.1.4 Mode Register (MODE): Functional description on the PCM bit in the table of bits, modified
		559, 560	23.2.1.5 Interrupt Enable Register (INTE) Bit chart: b7 (PHY), b2 (PHYNM), and b1 (DEVNM) were added. Table of bits: b1 (DEVNM), b2 (PHYNM), and b7 (PHY) were added. Bit name and functional description on b5 (DEV) were modified. Functional description on b6 (END) was modified.
		560	23.2.1.6 Physical Port Address Register (PADR), added
		561	23.2.1.7 Device Address Register (DADR), added
		561	23.2.1.8 Enable Address Register (ENADR), added
		562, 563	23.2.2.1 Receive Register (RX) Bit chart: b31 (SSPHY), b26 (SSPHYNM), and b25 (SSDEVNM) were added. Table of bits: Functional description on b15 to b0 (RXD[15:0]) was modified. b25 (SSDEVNM), b26 (SSPHYNM), and b31 (SSPHY) were added. Bit name and functional description on b29 (SSDEV) were modified.
		564	23.2.2.2 Receive Register 2 (RX2), added
		565	23.2.2.3 Version Register (VER): Value after reset of b9, b8, and b0 in the bit chart, modified
		567	Figure 23.2 Flowchart of MDIO Processing, modified
		568	23.3.3 MDIO Operational Setting, added
		569	23.4 Usage Notes, added

Rev.	Date	Description	
		Page	Summary
1.20	Mar. 02, 2017	24. Serial Communications Interface with FIFO (SCIFA)	
		602	Figure 24.5 Example of Transmit Operation in Asynchronous Mode (8-Bit Data with Parity and One Stop Bit when LSB-First Transfer is Selected), modified (ICU.IRn → ICU.RAISn)
		605	Figure 24.9 Example of SCIFA Receive Operation in Asynchronous Mode (8-Bit Data with Parity and One Stop Bit when LSB-First Transfer is Selected), modified (ICU.IRn → ICU.RAISn)
		610	Figure 24.14 Example of SCIFA Transmit Operation in Clock Synchronous Mode (when LSB-First Transfer is Selected), modified (ICU.IRn → ICU.RAISn)
		612	Figure 24.16 Example of SCIFA Receive Operation (when LSB-First Transfer is Selected), modified (ICU.IRn → ICU.RAISn)
		26. Serial Peripheral Interface (RSPIa)	
		709	26.2.5 RSPI Data Register (SPDR): Bit chart of the SPDR register when accessing in words (the SPLW bit is 0), modified (b31 to b16 → b15 to b0)
		27. SPI Multi I/O Bus Controller (SPIBSC)	
		783, 784	27.2.1 Common Control Register (CMNCR): Description on the IO0FV[1:0], IO2FV[1:0], IO3FV[1:0], MOIIO0[1:0], MOIIO1[1:0], MOIIO2[1:0], and MOIIO3[1:0] bits in the table of bits, modified
		800	27.2.14 SPI Mode Read Data Register 0 (SMRDR0): Register description, modified
		800	27.2.15 SPI Mode Write Data Register 0 (SMWDR0): Register description, modified
		806	Figure 27.4 Data Alignment in External Address Space Read Mode, modified
		806	Figure 27.5 Data Alignment in SPI Operating Mode, modified
		29. Error Control Module (ECM)	
		832	Table 29.2 ECM Error Input (1 / 2): Functions of error source numbers 5 and 6, modified
		834	29.2.1 ECM Master/Checker Error Source Status Register 0 (ECMmESSTR0 (m = M or C)): Description of bits ECMmSSE004 and ECMmSSE005, modified
		839	29.2.5 ECM Maskable Interrupt Configuration Register 0 (ECMMICFG0): Description of bits ECMMIE004 and ECMMIE005, modified
		844	29.2.8 ECM Non-maskable Interrupt Configuration Register 0 (ECMNMICFG0): Description of bits ECMNMIE004 and ECMNMIE005, modified
		849	29.2.11 ECM Internal Reset Configuration Register 0 (ECMIRCFG0): Description of bits ECMIRE004 and ECMIRE005, modified
		854	29.2.14 ECM Error Source Status Clear Trigger Register 0 (ECMESSTC0): Description of bits ECMCLSSE004 and ECMCLSSE005, modified
		859	29.2.17 ECM Protection Command Register (ECMPCMD1): Symbol of b7 to b0 in the table of bits, modified (ECM1REG7 to ECM1REG0 → ECM1REG[7:0])
		860	29.2.19 ECM Pseudo Error Trigger Register 0 (ECMPE0): Description of bits ECMPE004 and ECMPE005, modified
		866	29.2.25 ECM Delay Timer Configuration Register 0 (ECMDTMCFG0): Description of bits ECMTE004 and ECMTE005, modified
		872	29.2.28 ECM Delay Timer Configuration Register 3 (ECMDTMCFG3): Description of bits ECMTE304 and ECMTE305, modified
		30. 12-Bit A/D Converter (S12ADCa)	
		883	Table 30.1 Specifications of 12-Bit A/D Converter (1 / 2): Description on the conversion time, modified
		33. RAM (Product Option)	
		982	Table 33.1 Specifications of RAM: Description on the error checking, modified
		983	33.2.1 Protect Command Register (RAMPCMD): Description on step 2, modified

Rev.	Date	Description	
		Page	Summary
1.20	Mar. 02, 2017	34. Electrical Characteristics	
		997	Table 34.4 DC Characteristics (3) Item modified: "Input pull-up MOS current and resistance" and "Input pull-down MOS current and resistance" R_{pu1}, R_{pu2}, R_{pd1}, and R_{pd2} were added. Test conditions for "Input pull-down MOS current and resistance" were modified. Note 1 was added. Note 4 was modified. Description on ports P50 to P56 was deleted.
		1014	Table 34.18 SPIBSC Timing, Test conditions: Reference figures added
1.30	Apr. 25, 2017	3. Operating Modes	
		50	3.1 Overview: The description, modified
		50	3.2 Types and Selection of Operating Modes: The description, modified
		9. Low-Power Consumption Function	
		142	9.2.1 Module Stop Control Register A (MSTPCRA): The description, added
		143	9.2.2 Module Stop Control Register B (MSTPCRB): The description, added
		144	9.2.3 Module Stop Control Register C (MSTPCRC): The description, added
		146	9.2.4 Module Stop Control Register D (MSTPCRD): The description, added
		146	9.2.5 Module Stop Control Register E (MSTPCRE): The description, added
		147, 148	9.3.1 Module-Stop Function: The description (procedure, description of procedures, table 9.3), added
		12. Interrupt Controller (ICUA)	
		245	Figure 12.7 Register Rewrite Flow: The processing, modified
		13. Internal Buses	
		255	Table 13.1 Specifications of Internal Buses: The contents of external serial flash bus, modified (PCLKD → ICLK)
		255	Figure 13.1 Bus Configuration: Serial Flash (PCLKD) → Serial Flash, modified
		14. DMA Controller (DMACAa)	
		334	14.6 Usage Notes: The note, added
		27. SPI Multi I/O Bus Controller (SPIBSC)	
		783	Table 27.1 SPIBSC Specifications: The contents of the bit rate, modified (PCLKA → ICLK)
		784	Figure 27.1 Block Diagram of SPIBSC: PCLKA → ICLK, modified
		789	27.2.3, (1) Bit Rate: PCLKA → ICLK, modified
		790	Table 27.3 Relationship between SPBR[7:0] and BRDV[1:0] Settings: Bit Rate, PCLKA → ICLK, modified. Setting of bits SPBR[7:0] and BRDV[1:0]: 0,1, 0,2, and 0,3, added.
		29. Error Control Module (ECM)	
		860	The functional description of the ECMCLSSE102 to ECMCLSSE108 bits in the table of bits, modified (the bit error of the ECMmESSTR register, corrected)
		34. Electrical Characteristics	
		1017	Figure 34.24 SPIBSC Transmit/Receive Timing (CPHAT = 0, CPHAR = 1): Modified
1.40	Jul. 24, 2017	All	"Cortex-R4F" changed to "Cortex-R4"
		Features	
		28	Management data input/output interface (MDIO), changed
		1. Overview	
		29	Table 1.1 Outline of Specifications (1/4): General-purpose I/O ports: Number of I/O pins changed
		31	Table 1.1 Outline of Specifications (3/4): Management data input/output interface (MDIO): Description changed
		33	Table 1.1 Outline of Specifications: Note 3 added
		34	Table 1.2 List of Functions: MDIO changed to MDIO master/MDIO slave, Note 1 added
		35	Table 1.3 List of Products: Products added, option added

Rev.	Date	Description	
		Page	Summary
1.40	Jul. 24, 2017	36	Figure 1.1 Block Diagram, Note 1 deleted
		38	Table 1.4 Pin Functions (2/3): Management data input/output interface Management data input/ output interface (MDIOM/MDIO): MDIO changed to MDIOM/MDIO, functions of MDC and MDIO added, MMDC0, MMDC1, MMDIO0, and MMDIO1 pins added, functions of PRTADR0 to PRTADR4 added
		39	Table 1.4 Pin Functions (3/3): I/O ports: P50 to P56 pins added, Note 2 added
		40	Figure 1.2 Pin Arrangement (112-pin FBGA) (Top View): Pin numbers C4, C5, E10, and F11 changed
		41	Table 1.5 Pin Assignments (112-Pin FBGA) (1/3): Pin names of pin numbers C4 and C5 changed
		42	Table 1.5 Pin Assignments (112-Pin FBGA) (2/3): Pin names of pin numbers E10 and F11 changed
		43	Table 1.5 Pin Assignments (112-Pin FBGA): Note 1 deleted
		44	Table 1.6 List of Pins and Pin Functions (112-Pin FBGA) (1/3): Communications pins of pin numbers C4 and C5 changed
		45	Table 1.6 List of Pins and Pin Functions (112-Pin FBGA) (2/3): Communications pins of pin numbers E10 and F11 changed
		46	Table 1.6 List of Pins and Pin Functions (112-Pin FBGA): Note 1 deleted
		4. Address Space	
		63	Figure 4.1 Memory Map (1-Mbyte Extended Internal SRAM): Address changed
		5. I/O Registers	
		107	Table 5.1 List of I/O Registers (Address Order) (42/42): MDIOM0 (BR, TX, VER, and RX registers), MDIOM1 (BR, TX, VER, and RX registers), and Note 1 added
		9. Low-Power Consumption Function	
		141	Table 9.2 Stopping Peripheral Modules and Exiting Module-Stop State: Module symbol changed (MDIO → MDIOM/MDIO)
		146	9.2.4 Module Stop Control Register E (MSTPCRE): Functional description of the MSTPCRE0 bit in the table of bits changed (MDIO → MDIOM/MDIO), Note 1 added
		149	Table 9.3 Peripheral Functions Requiring the above Procedure for Release from the Module-Stop State (2/2): Module symbol changed (MDIO → MDIOM/MDIO), Note 1 added
		12. Interrupt Controller (ICUA)	
		237	Table 12.3 Cortex-R4F/DMAC Interrupt Vector Table (4/8): Vector numbers 135 (MDIOM0) and 143 (MDIOM1) added
		241	Table 12.3 Cortex-R4F/DMAC Interrupt Vector Table: Note 3 added
		16. I/O Ports	
		356	Table 16.1 Specifications of I/O Ports: Note 1 added
		363	16.3.1 Port Direction Register (PDR): Description on setting of the PORTC.PDR.Bn bits for the PC2, PC3, PC6, and PC7 pins deleted
		364	16.3.2 Port Output Data Register (PODR): Description on setting of the PORTC.PDR.Bn bits for the PC2, PC3, PC6, and PC7 pins deleted
		368	16.3.6 Driving Ability Control Register (DSCR): Note added to the table of bits
		17. Multi-Function Pin Controller (MPC)	
		372	Table 17.1 List of Multiplexed Pin Configurations (3/3): Pin function of the management data input/ output interface changed (MMDIO0, MMDC0, MMDC1, MMDIO1 pins added)
		378	17.2.6 P4n Pin Function Control Register (P4nPFS): Description of the ISEL bit deleted
		379	Table 17.7 Register Settings for the Input/Output Function in the 320-pin FBGA Pin: Pin function assigned to P50 and P51 for the setting 101011b changed, Note 1 added
		380	17.2.8 P6n Pin Function Control Register (P6nPFS): Description of the ISEL bit deleted
		383	17.2.11 PAn Pin Function Control Register (PAnPFS): Description of the ISEL bit deleted
		383	17.2.11 PAn Pin Function Control Register (PAnPFS): Description on switching of the pin function added

Rev.	Date	Description	
		Page	Summary
1.40	Jul. 24, 2017	383	Table 17.11 Register Settings for the Input/Output Function: The setting 101011b added, Note 1 added
		384	17.2.12 PCn Pin Function Control Register (PCnPFS): Description of the ISEL bit deleted
		23. Management Data Input/Output Interface (MDIO)	
		554	23. Management Data Input/Output Interface (MDIO): Body added
		554 to 569	23.1 Management Data Input/Output Interface (MDIO Slave), Title changed (MDIO → MDIO Slave)
		570 to 585	23.2 Management Data Input/Output Interface (MDIO Mater) added
		34. Electrical Characteristics	
		1012	Table 34.3 DC Characteristics (2) [Power Supply]: Characteristics in normal operation (VDD) changed
		1013	Table 34.4 DC Characteristics (3): Applicable pins for the output high level voltage (VOH) changed
		1014	Table 34.5 DC Characteristics for 1.2-V Pin: Input pull-up MOS current and resistance and Input pull-down MOS current and resistance added
		1034	34.4.3.9 (1) Serial Management Interface (Slave) Timing: Title added
		1034	Table 34.20 Serial Management Interface (Slave) Timing, changed
		1034	Figure 34.30 Serial Management Interface Output Timing, deleted
		1035	34.4.3.9 (2) Serial Management Interface (Master, Channel 0) Timing, added
		1036	34.4.3.9 (3) Serial Management Interface (Master, Channel 1) Timing, added
1.50	Dec. 27, 2017	All	Cortex-R4F changed to Cortex-R4
		29	Table 1.1 Outline of Specifications (1 / 4): Description of the clock generation circuit modified
		30	Table 1.1 Outline of Specifications (2 / 4): Description of the independent watchdog timer (IWDTa) modified (120 MHz → 120 kHz)
		37	Table 1.4 Pin Functions (2 / 3): CTS0# to CTS2#: I/O and functional description changed; RTS0# to RTS2#: Functional description changed
		3. Operating Modes	
		53	Table 3.3 Parameter Information for the Loader in SPI Boot Mode: Offset address for DEST_ADDR_NML changed; Note 3 added
		54	3.5.3 Loader Program: Storage address in the external memory changed (LDR_ADDR_NML → LDR_ADDR_NMI)
		54	Figure 3.3 Connection Diagram of This LSI with a Serial Flash Memory: Pins in serial flash memory changed (SI/SIO0 → SO/SIO1, SI/SIO1 → SI/SIO0)
		58	Figure 3.4 Relationship Between the Memory Map Definition During the Boot Processing and the Default Memory Map of Cortex-R4: Changed
		4. Address Space	
		63	Figure 4.2 Memory Map (0-Kbyte Extended Internal SRAM): Address of the area for MDIO (4KB) changed (B051 C700h → B100 0000h)
		6. Reset	
		107	6.1 Overview: Description added
		107	Table 6.2 Targets to be Initialized for Each Reset Type, added
		7. Clock Generation Circuit	
		124	7.3 Input to Main Clock Oscillator: Description added
		10. Debugging Interface	
		151	Figure 10.1 Block Diagram of CoreSight: Table for reference changed (Table 10.5 → Table 10.4)
		12. Interrupt Controller (ICUA)	
		165	Table 12.1 Specifications of Interrupt Controller: Note 2 modified (CR4F → CR4)
		170	12.2.4 Non-maskable Interrupt Status Register (NMISR): Bit name of ECMST in the bit table corrected (NMI Error Status Flag → ECM Error Status Flag)

Rev.	Date	Description	
		Page	Summary
1.50	Dec. 27, 2017	173	12.2.8 NMI Pin Digital Noise Filter Setting Register (NMIFLTC): Bit number in the bit table corrected (b7 → b31)
		232	Table 12.3 Cortex-R4/DMAC Interrupt Vector Table (1 / 8): Request source for vector numbers 1 to 3 modified (System (CR4F) → System (CR4))
		240	Figure 12.5 Initializing Registers of VIC: Corrected (Setting example 3 → Setting example 2)
		251	12.4.6.3 Notes on Selecting Level Detection: Description when level interrupt A is generated added
		252	12.4.6.5 Notes on Vector Settings, added
		14. DMA Controller (DMACAa)	
		310	Table 14.20 DMA Transfer Request Detection Operation Setting Table (1 / 2): DMA transfer request source name corrected (S12ADC → S12ADCa)
		311	Table 14.20 DMA Transfer Request Detection Operation Setting Table (2 / 2): Value of TM for TPUa Unit 0 changed (0/1 → 0)
		325	Figure 14.20 Setting Example 2: Changed
		15. Event Link Controller (ELC)	
		333	Figure 15.1 ELC Block Diagram (n = 7, 15, 18 to 27, 33, 35 to 38, 45): Changed (Port B or Port E → Port E)
		339	15.2.5 Port Group Control Register n (PGCn) (n = 2): Symbol of bits 1 and 0 in the bit chart and bit table corrected (PGCInI[1:0] → PGCIn[1:0])
		16. I/O Ports	
		368	Table 16.3 Handling of Unused Pins: Note 1 added
		17. Multi-Function Pin Controller (MPC)	
		376	17.2.5 P3n Pin Function Control Register (P3nPFS): Description of the ISEL bit added
		382	17.2.11 PAn Pin Function Control Register (PAnPFS): Reserved bits corrected (b7 → b7, b6)
		18. 16-Bit Timer Pulse Unit (TPUa)	
		431	18.3.3, (2) (a) When TPUm.TGRy is an output compare register: Register name in the title corrected (TPUm.TCNT → TPUm.TGRy)
		433	18.3.4 Cascaded Operation: Note 2 added
		434	18.3.4, (2) Examples of Cascaded Operation: Description added
		434	Figure 18.18 Example of Cascaded Operation (1): Waveforms of TPU1.TCNT and TPU2.TCNT clock signals changed
		436	Table 18.23 PWM Output Registers and Output Pins: Description of note modified
		448	18.4, (3) Underflow Interrupt: Index number of m for the TCImU interrupt changed (0 to 5 → 0 to 4)
		458	Figure 18.48 Conflict between Reading from the TPUm.TGRy Register and Input Capture: Register name in the title corrected (TPUm.TCNT → TPUm.TGRy)
		459	Figure 18.49 Conflict between Writing to the TPUm.TGRy Register and Input Capture: Register name in the title corrected (TPUm.TCNT → TPUm.TGRy)
		462	18.8.14 Continuous Output of Interrupt Signal in Response to Input Capture: Clock signal name corrected (PCLK → PCLKD)
		464	18.8.16 Input Capture Operation in Cascaded Operation, added
		19. Compare Match Timer (CMT)	
		474	19.2.3 Compare Match Timer Control Register (CMCR): Value of b7 in the bit chart corrected (x → 0)
		475	19.2.4 Compare Match Timer Counter (CMCNT): Title corrected
		475	19.2.5 Compare Match Timer Constant Register (CMCOR): Title corrected; description added
		20. Compare Match Timer W (CMTW)	
		494	20.2.5 Compare Match Timer Constant Register (CMWCOR): Title corrected; description added
		21. Watchdog Timer (WDTA)	
		524	Table 21.3 Relationship between Timeout Period and Window Start/End Counter Values: “b1” and “b0” added to the table header under “TOPS[1:0] Bits”

Rev.	Date	Description	
		Page	Summary
1.50	Dec. 27, 2017	527	21.2.4 WDT Reset Control Register (WDTRCR): Entry under “R/W” for reserved bits in the bit table corrected (R → R/W)
		24. Serial Communications Interface with FIFO (SCIFA)	
		596, 597	24.2.8 Bit Rate Register (BRR): Register symbol corrected (SMER → SEMR)
		600	24.2.9 Modulation Duty Register (MDDR): • When the baud rate generator is in double-speed mode (SEMR.BGDM = 1): Value in the formula under (When operating on the base clock with a frequency 16 times the bit rate (SEMR.ABCS0 = 0)) corrected (64 → 32)
		636	24.8.6 Receive Data Sampling Timing and Receive Margin in Asynchronous Mode: Description of the receive margin deleted
		26. Serial Peripheral Interface (RSPIa)	
		738	26.2.14 RSPI Command Registers 0 to 7 (SPCMD0 to SPCMD7): b6 to b4 (SSLy[2:0] bits) in the bit table: Description of 010 and 011 modified
		742	26.3.2 Controlling RSPI Pins: Note added
		27. SPI Multi I/O Bus Controller (SPIBSC)	
		815	27.2.13 SPI Mode Enable Setting Register (SMENR): Description of bit 15 in the bit table modified (Note 1 added, Note → Note 2)
		816	27.2.14 SPI Mode Read Data Register 0 (SMRDR0): Description added
		816	27.2.15 SPI Mode Write Data Register 0 (SMWDR0): Description added
		34. Electrical Characteristics	
		1026	Table 34.17 RSPIa Timing: Note 2 added; Note 2 changed to Note 3
1.60	May 31, 2019	All	
		—	"ARM" was modified to "Arm"
		Features	
		28	Management data input/output interface (MDIO): Interface for DSP control, modified
		1. Overview	
		29	Table 1.1 Outline of Specifications (1/4): On-chip extended SRAM with ECC: "Operating frequency", added
		29	Table 1.1 Outline of Specifications (1/4): Direct memory access controller (DMAC): Activation sources, modified
		30	Table 1.1 Outline of Specifications (2/4): Compare match timer (CMT): Event linking, modified
		36	Table 1.4 Pin Functions (1/3): 16-bit timer pulse unit (TPUa): The descriptions of the individual pins, modified
		3. Operating Modes	
		51	3.5.1 Boot Function: Step (3), modified
		51	Figure 3.1 Operating Overview of Boot Processing: Step (3), modified
		52	Figure 3.2 Memory Assignment of the Loader Program and Parameters for the Loader: Loader program, modified
		53	Table 3.3 Parameter Information for the Loader in SPI Boot Mode: Note 2, added; Note 3, modified; Note 2, changed to Note 4
		54	3.5.3 Loader Program: The description for setting the loader program, modified: The description of the storage address in the external memory, deleted; The description of the storage address in the external memory in SPI boot mode, added
		55	Table 3.4 Setting Values of the Individual Peripheral Modules and Registers at the Time SPI Boot Mode Finishes: The setting values of the MSTPCRC, PORT6.PMR, and MPC.PmnPFS registers, modified
		58	Figure 3.4 Relationship Between the Memory Map Definition During the Boot Processing and the Default Memory Map of Cortex-R4: The setting for the area from 4000 0000h to 5FFF FFFFh was added to the default memory map
		4. Address Space	
		62	Figure 4.1 Memory Map (1-Mbyte Extended Internal SRAM): "Mirror area of extended internal SRAM", modified; Note 3, added to "Area for MDIO"

Rev.	Date	Description	
		Page	Summary
1.60	May 31, 2019	63	Figure 4.2 Memory Map (0-Kbyte Extended Internal SRAM): "Area for MDIO", modified (4 KB → 4 MB); Note 3, added
		5. I/O Registers	
		66	Table 5.1 List of I/O Registers (Address Order) (2/42): The entry under "Access Size" for A000 014Ah/PORT5, modified
		6. Reset	
		107	Table 6.2 Targets to be Initialized for Each Reset Type: Note 3, modified
		111	6.3.1 RES# Pin Reset: The description of waiting time for PLL0 oscillation stabilization (tPL0WT), deleted
		111	6.3.2 ECM Reset: The description of waiting time for PLL0 oscillation stabilization (tPL0WT), deleted
		111	6.3.3 Software Reset: The description of waiting time for PLL0 oscillation stabilization (tPL0WT), deleted
		113	6.4 Usage Note: 6.4.1 Connection of Reset Output Pin (RSTOUT#), added
		7. Clock Generation Circuit	
		115	Table 7.2 Specifications of Clock Generation Circuit (Internal Clock): The frequency of PCLKE, CLMAPLCLK1, and TCLK, modified; Note 1, added
		118	7.2.1 System Clock Control Register: In the table of bits, the description of b20 (TCLK), modified
		120	7.2.3 PLL1 Control Register: Table of bits: Note 1, modified
		122	7.2.4 PLL1 Control Register 2: Table of bits: Note, modified (PLL1CR1 register → PLL1CR register)
		123	7.2.5 Low-Speed On-Chip Oscillator Control Register: The description, modified (The low-speed on-chip oscillator control register → LOCOCR)
		124	7.2.6 Oscillation Stop Detection Control Register: Table of bits: Note 1, modified
		128	7.8 Internal Clock: The CLMA clocks in (5), modified
		129	7.8.7 CLMA Clocks (CLMAMCLKA, CLMAMCLKB, CLMALCLK, CLMAPLCLK0, and CLMAPLCLK1): The title and description, modified
		129	7.8.9 ECM Clock (ECMCLK): The description, added
		129	7.8.11 Trace Interface Clock (TCLK): The description, added
		130	7.9.1 Notes on Clock Generation Circuit: The descriptions from (2) to (4), modified
		8. Clock Monitor Circuit (CLMA)	
		139	8.3.2, (2) Method of calculating threshold values, CLMAnCMPL.CLMAnCMPL[11:0] and CLMAnCMPH.CLMAnCMPH[11:0]: Example: For CLMA0: Nmax, modified; CLMAnCMPL modified to CLMAnCMPH
		9. Low-Power Consumption Function	
		142	9.2 Register Descriptions: The description, modified
		10. Debugging Interface	
		152	10.1 Overview: In the description, the table for reference (Table 10.4), added
		152	Table 10.1 CoreSight Specifications: The specification of "Trace port interface", modified
		156	Table 10.6 Configuration of Pins for the Debugging Interface: Note 1, modified
		160	10.3.3 Trace Port Interface: The description of 75 MHz, deleted
		11. Register Write Protection Function	
		166	11.2.1 Protect Register (PRCR): The index (i) of the PRCi Bits, modified
		12. Interrupt Controller (ICUA)	
		167	12.1 Overview: The description, modified
		167	Table 12.1 Specifications of Interrupt Controller: Note 3, added
		169	12.2.1 IRQ Control Register i: Note, added
		171	12.2.3 IRQ Pin Digital Noise Filter Setting Register: The description of the FCLKSELi[1:0] bits, modified

Rev.	Date	Description	
		Page	Summary
1.60	May 31, 2019	176	12.3.1 Selecting Interrupt Request Destinations: The description, modified; Note, added
		176	Figure 12.2 DMAC as the Interrupt Request Destination: The title, modified; the description, added
		177	Figure 12.3 CPU (Interrupt Controller) as the Interrupt Request Destination: The title, modified; the description, added
		178	Figure 12.4 Digital Noise Filter Operation Example: Text, modified
		179	12.3.3 External Pin Interrupts: The description, modified
		181	12.4 Cortex-R4 Vector Interrupt Controller (VIC), 12.4.1 Overview: The description, modified
		213	12.4.2.8 Interrupt Priority Level Mask Register 1: In the table of bits, the description of the value 1 of b15 to b0 (PRLM[15:0]), modified; the description of the PRLMi bit, modified
		230	12.4.2.16 Interrupt Address Store Register 0, Interrupt Address Store Register 1: The description of the VAD bit, modified
		246	12.4.4.3, (2) IRQ Interrupt (Level interrupt): Index modified (m → n)
		253	12.4.6.5 Notes on Vector Settings: The description, modified ("offset addresses" → "addresses")
		254	12.5 Usage Notes, 12.5.1 Using "Falling-Edge" or "Rising and Falling Edges" Detection with the External Pin Interrupts, added
		254	12.5.2 Using Falling-Edge Detection with the NMI Pin: Moved from 12.4.6.5 to 12.5.2
		13. Internal Buses	
		256	Table 13.3 Internal Main Bus 2: Connection between Bus Master and Bus Slave: Peripheral bus 3, added
		14. DMA Controller (DMACAa)	
		257	Table 14.1 Specifications of DMAC: The description of "DAM mode" (Register mode, Link mode) and "Skip function", modified
		263	14.2.5 Current Destination Address Register: Address(es): The register symbol, modified (CRDA_C → CRDA_8)
		265	14.2.7 Channel Status Register n: In the table of bits, the description of b10 (DER) and b16 (INTM), modified
		270	14.2.8 DMAC Unit 0 Source Select Register i: The description, modified; Address(es), modified (ICU.DMA0SELx → DMA0.DMA0SELx)
		270	14.2.9 DMAC Unit 1 Source Select Register i: The description, modified; Address(es), modified (ICU.DMA1SELx → DMA1.DMA1SELx)
		271	14.2.10 DMAC Software Activation Register: Address(es), modified (ICU.DMASTG → DMAC.DMASTG)
		272	14.2.11 Channel Control Register n: In the table of bits, the symbol and description of b7, modified; the description of b12 (SETREN), modified
		274	14.2.12 Channel Configuration Register n: In the table of bits, the description of b20 (SAD) and b21 (DAD), modified; Note 1, added
		284	14.2.19 Destination Skip Register n: Address(es), modified (DSKP_15: A006 36E8 → DSKP_15: A006 36ECh)
		287	14.2.21 Descriptor Interval Register n: The table of bits, modified ("b7 to b1" (reserved) → "b7 to b0" (reserved))
		299	14.3.1.2 Link Mode: The description, modified
		302	Table 14.11 Descriptor Format: The "Channel Extension" column, deleted
		303	Table 14.12 Description of activation in Table 14.11 Descriptor Format: "Channel Extension", deleted
		303	Table 14.13 Descriptor Placement Example: The entry under address "+18h" for DSCFM 1h, modified ("Extension" → "—")
		303	Figure 14.9 Header Area: The description, modified
		305	Figure 14.10 Header Area: Text, modified (External memory → Memory, DMA transaction → DMA transfer)
		306	14.3.1.2, (3) Descriptor settings, Notes on descriptors: The description, modified ("LV = 0" → "LV = 1")

Rev.	Date	Description	
		Page	Summary
1.60	May 31, 2019	307	Figure 14.11 Header Area: Text, modified (External memory → Memory)
		309	14.3.3.1 Fixed Priority Mode: The description, modified
		309	Figure 14.12 Priority Levels Immediately after a Reset and Transfer through DMA Channel 0, added
		310	14.3.3.2 Round-Robin Mode: The description, modified
		310	Figure 14.13 Priority Levels Immediately after a Reset and Transfer through DMA Channel 2, added
		311	14.3.4.1 Specifying Detection Operation of DMA Transfer Requests for Each Source: The bit symbols, modified (HEN → HIEN, LEN → LOEN)
		312	Table 14.20 DMA Transfer Request Detection Operation Setting Table (1/2): The TM bit of S12ADCa Unit 0 and S12ADCa Unit1, modified
		314	Figure 14.14 Software Forced Ejection Timing: Text, modified (Sweep write → Forced ejection (write))
		315	14.3.7.1 When the Transfer Data Size on the Transfer Source is Small: The description, modified
		318	14.3.10.1 Aborting a Transfer (No Buffer Flush: SBE = 0): The description, added
		321	Table 14.21 Interrupt Sources of DMAC: The error, corrected (Descriptor inbound → Descriptor invalid)
		324	Table 14.23 DMA Transfer Setting Example 1: The item of "AHB setting", deleted
		326	Table 14.24 DMA Transfer Setting Example 2: The item of "AHB setting", deleted
		332	Figure 14.25 Setting Example of Next Register Continuous Execution: The title, modified
		334	14.6 Usage Notes: The notation, modified (SKIP transfer → skip transfer)
		15. Event Link Controller (ELC)	
		All	The expressions indicating "restart counting" were modified appropriately to indicate "clear counting".
		336	15.2.2 Event Link Setting Register n: In the table of bits, the description of b7 to b0 (ELS[7:0]), modified
		342	15.2.6 Port Buffer Register n: In the table of bits, the description, modified
		347	Figure 15.2 Relation between Interrupt Handling and ELC: Text, modified (ICU → ICUA)
		348	15.3.2 Event Linkage: The description, modified
		348	Table 15.5 Operations of Modules When Event is Input: The descriptions of the CMT, CMTW, and TPU modules, I/O ports (output), and I/O ports (output) under "Operations When Event is Input", modified; Encoder I/F, deleted
		349	15.3.3, (2) Counting Clear Operation: The description, modified
		349	15.3.3, (5) Stopping Counting, deleted
		349	15.3.5, (1) Single ports and Port Groups: The register symbols, modified (PEL0 to PEL3 → PELn, PGCn → PGRn)
		350	15.3.5, (2) Single Input Port Operation upon Event Generation: The tile and description, modified
		350	15.3.5, (3) Single Output Port Operation upon Event Input: The description, modified
		350	Figure 15.3 Event Linkage Related to Single Ports (Port E): The figure, modified
		351	15.3.5, (4) Input Port Group Operation upon Event Generation: The tile and description, modified
		351	15.3.5, (5) Input Port Group Operation upon Event Input, added
		351	Figure 15.4 Input Port Group Operation upon Event Input (Port E): The title and figure, modified
		352	15.3.5, (6) Output Port Group Operation upon Event Input: The title and description, modified
		352	Figure 15.5 Event Linkage Related to Output Port Groups (Port E): The figure, modified
		353	Figure 15.6 Bit-Rotating Operation of Output Port Groups (Port E): The figure, modified
		—	15.3.5, (6) Operation of Port Buffer Registers, deleted
		353	15.3.5, (7) Restrictions on Writing to PODR and PDBFn Registers by a CPU: The title and description, modified

Rev.	Date	Description	
		Page	Summary
1.60	May 31, 2019	354	15.3.6 Example of Procedure for Linking Events: In the description of 2., the headings, added; the description of 4.,modified
		355	15.4.1 Setting ELSR18 and ELSR19 Registers: The description, modified
		16. I/O Ports	
		361	Figure 16.4 I/O Port Configuration (4): ISEL bit, deleted; Note 1, deleted
		362	Figure 16.5 I/O Port Configuration (5): Note 1, modified
		17. Multi-Function Pin Controller (MPC)	
		378	17.2.6 P4n Pin Function Control Register: The description, modified
		380	17.2.8 P6n Pin Function Control Register: The description, modified
		382	17.2.10 P9n Pin Function Control Register: ISEL Bit: The description, added
		383	17.2.11 PAn Pin Function Control Register: The description, modified
		384	17.2.12 PCn Pin Function Control Register: The description, modified
		386	17.2.14 PGn Pin Function Control Register: The description, modified
		386	Table 17.14 Register Settings for the Input/Output Function: The title, corrected
		388	Table 17.15 Register Settings: Note on "Analog inputs", modified
		18. 16-Bit Timer Pulse Unit (TPUa)	
		All	The expressions indicating "restart counting" were modified appropriately to indicate "clear counting".
		393	Table 18.2 TPU Functions (2/ 2): Module-stop setting: The bit symbol, corrected; Note 1, added (former Note 1 and Note 2 were changed to Note 2 and Note 3)
		412	18.2.4 Timer Interrupt Enable Register: Table of bits: Note 3, added
		417	18.2.6 Timer Counter: Index (m), modified
		434	18.3.4 Cascaded Operation: Index, modified (TPUm.TCNT → TPUn.TCNT): Note 3, added
		436	18.3.5 PWM Modes: The description of the indices n and m, moved; the descriptions of "1. PWM mode 1" and "2. PWM mode 2", modified
		440	Figure 18.23 Example of PWM Mode Operation (3): Cases from (a) to (c) and the descriptions, added
		446	18.3.6.1 Phase Counting Mode Application Example: The description, modified
		447	18.3.7 Noise Filters: The description, modified
		449	18.5 DMAC Activation: The notation, modified (TGRB → TPUm.TGRB)
		451	Figure 18.34 Input Capture Signal Timing: The timing lines of rising edges of the input capture signal, modified
		459	Figure 18.48 Conflict between Reading from the TPUm.TGRy Register and Input Capture: Text, modified (buffer register → TPUm.TGRy register)
		460	Figure 18.49 Conflict between Writing to the TPUm.TGRy Register and Input Capture: Text, modified (TPUm.TCNT → TPUm.TGRy)
		466	Table 18.30 Ability of Interrupt Sources to Serve as Event Signals for Transmission to the ELC: Channel 4 and Channel 5, deleted; Note 2, modified
		466	18.9.2, (1) Start Counting: The wording "by the ELOPF register of the ELC", added
		467	18.9.2, (2) Clear Counting: The title and description, modified
		467	Figure 18.57 Restart Counting on Reception of the Event Signal: The description added to the timing with which the TCNT register changes from "0000h" to "0001h"
		468	18.9.2, (3) Input Capture Operation: The description, modified
		469	18.9.3, (2) Clear Counting: The description, modified
		19. Compare Match Timer (CMT)	
		All	The register symbols, modified: CMCNT0 → CMT0.CMCNT, CMCNT1 → CMT1.CMCNT, CMCNT2 → CMT2.CMCNT, CMCNT3 → CMT3.CMCNT, CMCOR0 → CMT0.CMCOR, CMCOR1 → CMT1.CMCOR, CMCOR2 → CMT2.CMCOR, CMCOR3 → CMT3.CMCOR

Rev.	Date	Description	
		Page	Summary
1.60	May 31, 2019	473	Table 19.1 CMT Specifications: Event link function, modified ("(channel 1 only)" → "(only channel 1 of unit 0)"
		475	19.2.3 Compare Match Timer Control Register: In the table of bits, the description of "b1, b0" (CKS[1:0]), modified (CMCNTn counter → CMTn.CMCNT counter; CMSTR.STRn bit → CMSTRm.STRn bit); index (m, n), modified)
		476	19.2.4 Compare Match Timer Counter: The register symbol, modified (CMSTRn.STRm → CMSTRm.STRn); index (m, n), modified
		477	19.3.1 Periodic Count Operation: The register symbol, modified (CMSTRn.STRm → CMSTRm.STRn); index (m, n), modified
		479	19.5.1 Event Issuance to ELC: The register symbol, modified (CMCSRn.CMIE → CMT1.CMCR.CMIE)
		480	19.5.2, (1) Count Start: The register name and bit symbol, modified
		480	Figure 19.6 Count Start Operation at Reception of an Event: Text in the figure, modified (function select n → operation select; Event input signal n → Event input signal; CMSTR.STRn → CMSTR0.STR1; CMCNTn → CMT1.CMCNT)
		481	19.5.2, (2) Event Count: The register symbols and the register name, modified
		481	Figure 19.7 Event Count Operation at Reception of an Event: Text in the figure, modified (function select n → operation select; Event input signal n → Event input signal; CMCNTn → CMT1.CMCNT)
		482	19.5.2, (3) Count Clear: The description, modified; the register symbol and the register name, modified
		482	Figure 19.8 Count Clear Operation at Reception of an Event: Text in the figure, modified (function select n → operation select; Event input signal n → Event input signal; CMCNTn → CMT1.CMCNT)
		483	Figure 19.9 Conflict between Event Reception and Register Access at Count Start Operation: Text in the figure, modified (function select m → operation select; Event input signal m → Event input signal; CMSTRn.STRm → CMSTR0.STR1)
		484	Figure 19.10 Conflict between Event Reception and Register Access at Event Count Operation: Text in the figure, modified (function select → operation select; CMCNT → CMT1.CMCNT)
		485	Figure 19.11 Conflict between Event Reception and Register Access at Count Clear Operation: Text in the figure, modified (function select → operation select; CMCNT → CMT1.CMCNT)
		487	Table 19.3 Summary of Conflicted Operations among Event Link Operation, Register Access, and Counter Status: The register symbol, modified (CMSTRn.STRm → CMSTR0.STR1)
		20. Compare Match Timer W (CMTW)	
		All	The expressions indicating "restart counting" were modified appropriately to indicate "clear counting".
		488	Table 20.1 Specifications of CMTW: Event link: The description, corrected
		489	Figure 20.1 Block Diagram of CMTW (Unit 0): The title, modified; the register symbols, corrected; the interrupt request signals, modified
		491	20.2.2 Timer Control Register: In the table of bits, the description of b6 (OC0IE) and b7 (OC1IE), modified (output capture → output compare)
		505	Figure 20.9 Count Timing (PCLKD/8): "N-1" added to CMWCNT
		507	20.3.9 Digital Noise Filtering: The description of sampling, modified
		508	20.4.1 CMTW Interrupt Sources and DMAC Transfer Requests: Index n added to the individual interrupt source names
		511	20.5.2 Actions on Acceptance of Event Signals from ELC: The description, corrected (four actions → three actions)
		512	20.5.2, (3) Clear a Counter: The description, modified
		513	20.6.1 Module-Stop Function: The description, modified
		515	Figure 20.23 Contention between CMWOCR Register Writing and Compare Match: Text, modified (CMWCOR → CMWOCR, CMWCR → CMWOCR)
		—	20.6.8, (4) Input Capture Operation, deleted

Rev.	Date	Description	
		Page	Summary
1.60	May 31, 2019	—	Figure 20.29 Contention between Counter Clearing on Event Acceptance and Register Access in Input Capture Operation, deleted
		519	Table 20.4 Summary of Contention between Operations Due to the Event Link, Access to Registers, and Changes to the Counter's State: The register symbols, modified; the item "Input capture", deleted
		21. Watchdog Timer (WDTA)	
		520	Table 21.1 WDT Specifications: Event link function, deleted
		521	Figure 21.1 WDT Block Diagram: WDTRCR register, added; event link output and event link controller circuit, deleted
		527	21.2.3 WDT Status Register: The description, modified
		529	21.3.1.1 Register Setting: The description, modified
		530	Figure 21.3 Operation Example in Register Start Mode: Error notification to ECM, modified (Active: Low → Active: High)
		531	21.3.2 Control over Writing to the WDTCR and WDTRCR Registers: The description, modified
		532	21.3.3 Refresh Operation: [Sample sequences of writing that are not valid for refreshing the counter]: The error, corrected
		535	21.4.1 Watchdog Timer Operations in Low-Power Consumption Mode Transition: The typo, corrected (Coretex-R4 → Cortex-R4)
		535	Table 21.4 WDT Operations during Transition to Low-Power Consumption Mode <in Low-Power Consumption Mode Transition>: The "VDD", "WDT1 Clock Supply", and "WDT1 Operation" columns, deleted
		—	21.5 Link Operation by the Event Link (ELC) Function, deleted
		22. Independent Watchdog Timer (IWDTa)	
		536	Table 22.1 IWDT Specifications: Event link function, deleted
		537	Figure 22.1 IWDT Block Diagram: IWDTRCR register, added; event link output and event link controller circuit, deleted
		539	22.2.2 IWDT Control Register: The description of the CKS[3:0] bits, modified
		543	22.2.3 IWDT Status Register: The description, modified
		546	Figure 22.3 Operation Example in Register Start Mode: Error notification to ECM, modified (Active: Low → Active: High)
		547	22.3.2 Control Over Writing to the IWDTCR and IWDTRCR Registers: The description, modified
		548	22.3.3 Refresh Operation: [Sample sequences of writing that are not valid for refreshing the counter]: The error, corrected
		552	22.3.6 Reading the Down-Counter Value: The description, modified
		552	Figure 22.7 Processing for Reading IWDT Counter Value (IWDTCR.CKS[3:0] = 0000b, IWDTCR.TOPS[1:0] = 11b), changed (replaced)
		—	Figure 22.8 Processing for Reading IWDT Counter Value (IWDTCR.CKS[3:0] = 0010b, IWDTCR.TOPS[1:0] = 11b), deleted
		553	22.4.1 Watchdog Timer Operations in Low-Power Consumption Mode Transition: The typo, corrected (Coretex-R4 → Cortex-R4)
		553	Table 22.4 IWDT Operations during Transition to Low-Power Consumption Mode <in Low-Power Consumption Mode Transition>: The "VDD" column, deleted
		—	22.5 Usage Notes, deleted
		—	22.6 Link Operation by the Event Link (ELC) Function, deleted
		24. Serial Communications Interface with FIFO (SCIFA)	
		586	Table 24.1 Specifications of SCIFA: The description of "Interrupt source" and "Asynchronous communication mode" (Receive error detection), modified
		587	Figure 24.1 Block Diagram of SCIFA: Text, modified
		588	24.2.1 Receive Shift Register: The description, modified
		588	24.2.2 Receive FIFO Data Register: The description, added
		593	24.2.7 Serial Status Register: The description of the TEND bit, modified

Rev.	Date	Description	
		Page	Summary
1.60	May 31, 2019	597	Table 24.3 SMR Register Setting: Table heading, modified (SMR Register Settings → Setting of SMR.CKS [1:0] Bits, CKS1 → b1, CKS0 → b0)
		600	24.2.9 Modulation Duty Register: MDDR register setting, modified (128 ≤ MDDR ≤ 256 → 128 ≤ MDDR ≤ 255)
		601	Table 24.10 Bit Rates and Settings of BRR and MDDR Registers in Asynchronous Mode: The combinations of the bit rates and register settings were modified to minimize an error.
		602	24.2.10 FIFO Control Register: The description of the TTRG[1:0], RTRG[1:0], and RSTRG[2:0] bits, modified
		604	24.2.11 FIFO Data Count Register: The description, modified
		605	24.2.12 Serial Port Register: The description of the bits, modified (multi-port controller (MPC) → multi-function pin controller (MPC))
		611	24.3 Operation, 24.3.1 Overview: The description, modified
		612	Table 24.15 SMR, SCR, and SPTR Register Settings and SCIFA Clock Source Selection: The heading, modified (CKE1, CKE0 → CKE[1:0])
		616	Figure 24.3 Sample Flowchart for SCIFA Initialization in Asynchronous Mode: Processing, modified
		619	Figure 24.7 Sample Flowchart for Receiving Serial Data in Asynchronous Mode (1): The description, modified (the receive FIFO threshold → the receive data trigger number)
		622	Figure 24.10 Example of SCIFA Receive Operation in Asynchronous Mode Using Model Control Function (RTS#): Text, modified (RTSn pin → RTS# pin)
		624	Figure 24.12 Sample Flowchart for SCIFA Initialization in Clock Synchronous Mode: Processing, modified
		627	Figure 24.15 Sample Flowchart for Receiving Serial Data in Clock Synchronous Mode: The description, modified (the receive FIFO threshold → the receive data trigger number)
		629	Figure 24.17 Sample Flowchart for Simultaneous Transmitting/Receiving Serial Data in Clock Synchronous Mode: Note 1, modified (the receive FIFO threshold → the receive data trigger number)
		630	24.4 Bit Modulation: The bit symbol, modified
		631	24.5 Interrupt Sources: The description, modified
		637	24.8.11 Notes on Initialization of the SCIFA, added
		25. I^2C Bus Interface (RIICa)	
		638	Table 25.1 RIIC Specifications (1/2): The description of "Arbitration", modified
		648	25.2.4 I^2C Bus Mode Register 2: The description, modified (SDA output delay function, added); the address of RIIC1.ICMR2, modified
		650, 651	25.2.5 I^2C Bus Mode Register 3: The description, modified (the settings for acknowledgement, added); In the table of bits, the description of the value "1" of b5 (RDRFS), modified; the description of the ACKBR, ACKBT, and RDRFS bits, modified
		652, 653	25.2.6 I^2C Bus Function Enable Register: The description of the MALE, NALE, SALE, and SCLE bits, modified
		660	25.2.10 I^2C Bus Status Register 2: The description of the AL flag, modified (NACK arbitration-lost detection → NACK transmission arbitration-lost detection)
		673	Figure 25.6 Example of Master Transmission Flowchart: The description in [6], modified
		677	Figure 25.10 Example of Master Reception (7-Bit Address Format, 1 or 2 bytes): The description in [8], modified
		678	Figure 25.11 Example of Master Reception (7-Bit Address Format, 3 Bytes or More): The description in [8], modified
		681	25.3.5 Slave Transmit Operation: The description in (1) and (2), modified
		682	Figure 25.15 Example of Slave Transmission Flowchart: The description in [5], modified
		684	25.3.6 Slave Receive Operation: The description in (1), (3) and (4), modified
		687	Figure 25.21 Generation and Synchronization of the SCL Signal from the RIIC: Text, modified (SCL0n line → SCLn line)

Rev.	Date	Description	
		Page	Summary
1.60	May 31, 2019	689	Figure 25.23 Block Diagram of Digital Noise Filter Circuit: Text, modified (SCL0n/SDA0n → SCLn/SDAn)
		690	Figure 25.24 AASy Flag Set Timing with 7-Bit Address Format Selected: [7-bit address format: Slave reception]: The description, added
		691	Figure 25.25 AASy Flag Set Timing with 10-Bit Address Format Selected: [10-bit address format: Slave reception]: The description, added
		692	Figure 25.26 AASy Flag Set/Clear Timing with 7-Bit/10-Bit Address Formats Mixed: Case (1), modified (the location of "Address mismatch", moved; Case (2), modified (ICSAR1L L → ICSARL1)
		702	Figure 25.34 Example of Arbitration-Lost Detection during Transmission of NACK (NALE = 1): Text, modified (Release SCLn/SDA → Release SCL/SDA)
		703	Figure 25.35 Example of Slave Arbitration-Lost Detection (SALE = 1): Text, modified (Release SCLn/SDA → Release SCL/SDA)
		704	Figure 25.36 Start Condition/Restart Condition Issue Timing (ST and RS Bits): [Restart condition issuing operation]: The width at high level of the SCL line, modified (8 → 9)
		707	Figure 25.38 Timeout Function (TMOE, TMOS, TMOH, and TMOL Bits): [Example of operation when TMOH = 1 and TMOL = 1]: The setting of the TMOS bit, corrected
		26. Serial Peripheral Interface (RSPIa)	
		716	Figure 26.1 RSPI Block Diagram: Index "y" was added to interrupt names
		717	Table 26.2 RSPI Pin Configuration: The function of the SSL00, SSL01, SSL02, SSL03, SSL10, and SSL11 pins, modified; the index range in "Note", modified
		718	26.2.1 RSPI Control Register: The description of the SPTIE bit, modified ((i.e. the SPSR.SPTEF bit = 1), deleted)
		723	26.2.4 RSPI Status Register: In the table of bits, the description and R/W of b7 to b4 (reserved), modified
		725	Figure 26.2 Configuration of SPDR: Shift register, modified
		733	26.2.10 RSPI Clock Delay Register: The description, modified
		738	26.2.14 RSPI Command Registers 0 to 7: In the table of bits, the description of "x", deleted
		743	Figure 26.5 Single-Master/Single-Slave Configuration Example (This LSI = Master): Note 1, changed; Note 2, deleted
		771	Figure 26.31 Procedure for Determining the Form of Serial Transfer in Master Mode: Text, modified (SSLA[2:0] → SSLy[2:0])
		781	Figure 26.39 Example of Initialization Flowchart in Slave Mode (SPI Operation): Text, modified (SSLA0 input signal → SSL signals)
		785	Figure 26.43 Procedure for Determining the Form of Serial Transmission in Master Mode: Text, modified (SSLA[2:0] → SSLy[2:0])
		792	26.3.14 Interrupt Sources: The description of the transmission buffer empty and reception buffer full interrupts, deleted
		795	26.5.2 Note on Low-Power Consumption Functions: The description, modified
		27. SPI Multi I/O Bus Controller (SPIBSC)	
		All	The notation of the clock signal, modified (ICLK → PCLKA)
		796	27.1 Overview: The description, modified
		824	Figure 27.7 Burst Read Operation: The description in (1), modified
		833	Table 27.5 Data Registers: The description of "Dummy cycle", modified (DRDMC → DRDMCR, SMDMC → SMDMCR)
		833	27.3.10, (2) Data Enable: The description, modified
		837	Table 27.8 Pin Status (3): The pin names, corrected (SPBMI0/SPBIO10 → SPBMI/SPBIO1)
		28. CRC Operation Units (CRC)	
		840	Figure 28.1 Block Diagram of a CRC Operation Unit (CRC): Text, corrected (CRCCR.DCR0POL[1:0] → CRCCR.DCRA0POL[1:0])
		843	28.2.3 CRC Control Register (CRCCR): In the table of bits, the bit symbol, corrected (CRCCR → CRCDIR)

Rev.	Date	Description	
		Page	Summary
1.60	May 31, 2019	29. Error Control Module (ECM)	
		843	Table 29.2 ECM Error Input (1/2): The description of error source number 1, modified (Cortex-RF4 → Cortex-R4)
		850	29.2.1 ECM Master/Checker Error Source Status Register 0: In the table of bits, the description of b4 (ECMmSSE004) and b5 (ECMmSSE005), modified
		855	29.2.5 ECM Maskable Interrupt Configuration Register 0: In the table of bits, the description of b4 (ECMMIE004) and b5 (ECMMIE005), modified
		860	29.2.8 ECM Non-maskable Interrupt Configuration Register 0: In the table of bits, the description of b4 (ECMNMIE004) and b5 (ECMNMIE005), modified
		865	29.2.11 ECM Internal Reset Configuration Register 0: In the table of bits, the description of b4 (ECMIRE004) and b5 (ECMIRE005), modified
		870	29.2.14 ECM Error Source Status Clear Trigger Register 0: In the table of bits, the description of b4 (ECMCLSSE004) and b5 (ECMCLSSE005), modified
		876	29.2.19 ECM Pseudo Error Trigger Register 0: In the table of bits, the description of b4 (ECMPE004) and b5 (ECMPE005), modified
		882	29.2.25 ECM Delay Timer Configuration Register 0: In the table of bits, the description of b4 (ECMTE004) and b5 (ECMTE005), modified
		888	29.2.28 ECM Delay Timer Configuration Register 3: In the table of bits, the description of b4 (ECMTE304) and b5 (ECMTE305), modified
		30. 12-Bit A/D Converter (S12ADCa)	
		920	30.2.6 A/D-Converted Value Addition/Average Mode Select Register: The description of the ADS[7:0] bits, corrected (ADANSA0.ANSA0[7:0] = 00FFh → ADANSA.ANSA[7:0] = FFh)
		929	30.2.12 A/D Sample and Hold Circuit Control Register: The table of bits, modified (b15 to b11 (reserved) →b15 to b12 (reserved))
		941	30.2.22 A/D Compare Status Extended Register: Address(es), modified
		943	30.2.25 A/D Error Clear Register: Address(es): The module symbol, modified
		944	30.2.26 A/D Overwrite Error Register: In the description of the OWE[7:0] bits, the symbol, corrected (OWE0[7]→ OWE [7])
		971	30.3.5 Comparison: The description in (2), modified (ADCMPANSRy register → ADCMPANSR register)
		982	30.4.1 Interrupt Requests: The description, modified (ADI and GBADI interrupts → S12ADI and S12GBADI interrupts)
		982	Table 30.12 Relationship between Mode Setting and S12ADI Interrupt Output, modified (ADI interrupt → S12ADI interrupt, GBADI interrupt → S12GBADI interrupt)
		986	30.5.6 Notes on Entering Low-Power Consumption States: The erroneous description, corrected
		31. Temperature Sensor	
		990	31.3.1 Preparation for Using the Temperature Sensor: The section for reference, modified; o How to obtain the value for voltage to be output from the temperature sensor (when AD-converted value addition mode is not selected), added
		33. RAM (Product Option)	
		1001	33.1 Overview: Note, modified
		1003	33.2.2 ECC Decoder Configuration Register: The description, modified (the description of the "1-bit ECC error" and "2-bit ECC error" cases, added)
		1006	33.2.4 2-Bit ECC Error Status Register: The description, modified
		1008	33.2.5 2-Bit ECC Error Address Register: Table of bits: The erroneous description in Note, corrected
		1011	Figure 33.2 Example of ECC Error-Injection Setting Procedure: The title, added
		1012	Figure 33.3 Procedure of Checking ECC Operation: The title, added; the processing, modified (DBE_IRQ interrupt → interrupt); Note 1, added
		34. Electrical Characteristics	
		1039	Table 34.22 Serial Management Interface (Master, Channel 1) Timing: Note 1 added to the title

Rev.	Date	Description	
		Page	Summary
1.70	Oct 23, 2020	All	Registered trademark symbol added (Arm → Arm®)
		1. Overview	
		35	Figure 1.1 Block Diagram: Unnecessary bus lines, deleted
		37	Table 1.4 Pin Functions (2 / 3): MDIOM/MDIO: Pins MMDC0 and MMDIO0, deleted
		39	Figure 1.2 Block Diagram: Pins MMDC0 and MMDIO0, deleted
		40	Table 1.5 Pin Assignments (112-Pin FBGA) (1 / 3): Pins MMDC0 and MMDIO0, deleted
		43	Table 1.6 List of Pins and Pin Functions (112-Pin FBGA) (1 / 3): Pins MMDC0 and MMDIO0, deleted
		3. Operating Modes	
		52	Figure 3.2 Memory Assignment of the Loader Program and Parameters for the Loader: Note 3 added (renumbering the subsequent note numbers: Note 3 → Note 4, Note 4 → Note 5)
		53	Table 3.3 Parameter Information for the Loader in SPI Boot Mode: Note 3 modified
		54	3.5.3 Loader Program: The description of storage address in the external memory in SPI boot mode, modified
		5. I/O Registers	
		106	Table 5.1 List of I/O Registers (Address Order) (42 / 42): Addresses B083 0012h, B091 C100h. B091 C500h, B091 CD04h, deleted
		6. Reset	
		109	6.2.1 Reset Status Register 0 (RSTSR0): R/W in the table of bits, modified (R(/W) → R/(W))
		7. Clock Generation Circuit	
		128	7.8.2 System Clock (ICLK): The description modified
		128	7.8.3 High-Speed Peripheral Module Clock (PCLKA): The description modified
		128	7.8.4 Low-Speed Peripheral Module Clock (PCLKB): The description modified
		128	7.8.5 Low-Speed Peripheral Module Clocks (PCLKD, PCLKE, PCLKF, PCLKG, and PCLKH): The description modified
		129	7.8.7 CLMA Clocks (CLMAMCLKA, CLMAMCLKB, CLMALCLK, CLMAPLCLK0, and CLMAPLCLK1): The description modified
		8. Clock Monitor Circuit (CLMA)	
		136	8.3.1 CLMAn Operation, (1) Enabling operations: The description in step 3, modified
		9. Low-Power Consumption Function	
		All	RIICa Unit → RIICa channel, SCIFA Unit → SCIFA channel, RSPIa Unit → RSPIa channel
		12. Interrupt Controller (ICUA)	
		168	Figure 12.1 Block Diagram of Interrupt Controller: Text modified: DMA0SELi, DMA1SELi, CDSEL → DMA0SELi, DMA1SELi (CDSEL deleted)
		181	12.4.1 Overview: The description modified
		192	12.4.2.3 Interrupt Enable Register n (IENn) (n = 0 to 9): The title modified (Interrupt Enable Register 0 → Interrupt Enable Register n)
		230	12.4.2.16 Interrupt Address Store Register 0 (VADn), Interrupt Address Store Register 1 (VADn): Bit chart: VADn[31:0] entered
		230	12.4.2.16 Interrupt Address Store Register 0 (VADn), Interrupt Address Store Register 1 (VADn): Table of bits: The symbol and functional description, modified
		230	12.4.2.16 Interrupt Address Store Register 0 (VADn), Interrupt Address Store Register 1 (VADn): The description of "VADn[31:0] Bit": The bit symbol, index, and description, modified
		235	Table 12.3 Cortex-R4/DMAC Interrupt Vector Table (2 / 8): The request source modified: RSPI unit 0 → RSPI ch0
		236	Table 12.3 Cortex-R4/DMAC Interrupt Vector Table (3 / 8): The request sources modified: RSPI unit 1, SCIFA unit 0 to SCIFA unit 2, SCIFA unit 4, RIIC unit 0, RIIC unit 1 → RSPI ch1, SCIFA ch0 to SCIFA ch2, SCIFA ch4, RIIC ch0, RIIC ch1

Rev.	Date	Description	
		Page	Summary
1.70	Oct 23, 2020	237	Table 12.3 Cortex-R4/DMAC Interrupt Vector Table (4 / 8): The source of vector number 135 (Reserved), modified
		240	Table 12.3 Cortex-R4/DMAC Interrupt Vector Table (7 / 8): The sources of vector numbers 242 and 243, modified
		242	Figure 12.5 Initializing Registers of VIC: Indices (m, q) modified
		244	Figure 12.7 Register Rewrite Flow: Text modified (IECm, IENm → IECn, IENn)
		247	12.4.4.3 Detecting Interrupts, (3) IRQ Interrupt (Edge Interrupt): The description modified
		247	Figure 12.9 IRQ Interrupt Operation (Edge Interrupt): Text modified (PICm → PICn)
		251	Figure 12.11 IRQ Interrupt Operation by Polling (Edge/Level Detection): Text modified (IENm → IENn)
		14. DMA Controller (DMACAa)	
		312	Table 14.20 DMA Transfer Request Detection Operation Setting Table (1 / 2): The DMA transfer request sources modified: RSPI Unit 0, RSPI Unit 1 → RSPI Channel 0, RSPI Channel 1, SCIFA Unit 0, SCIFA Unit 1 → SCIFA Channel 0, SCIFA Channel 1
		313	Table 14.20 DMA Transfer Request Detection Operation Setting Table (2 / 2): The DMA transfer request sources modified: SCIFA Unit 2, SCIFA Unit 4, RIIC Unit 0, RIIC Unit 1 → SCIFA Channel 2, SCIFA Channel 4, RIIC Channel 0, RIIC Channel 1; the DMA transfer sources of ELC, modified
		15. Event Link Controller (ELC)	
		337	Table 15.2 Correspondence between the ELSRn Register and the Peripheral Functions: The entries under "Peripheral Function (Module)" for ELSR18 and ELSR19, modified
		348	Table 15.5 Operations of Modules When Event is Input: CMT, CMTW, TPU modules: The description under "Operations When Event is Input" modified
		349	15.3.3 Operation of Peripheral Timer Modules When Event is Input: The description modified
		16. I/O Ports	
		All	"Hi-z" was modified to "Hi-Z"
		365	16.3.3 Port Input Data Register (PIDR): The title modified
		367	16.3.5 Pull-Up/Pull-Down Control Register (PCR): The title modified
		368	16.3.6 Driving Ability Control Register (DSCR): Table of bits: Bit name index modified (m → 5)
		369	16.4 Handling of Unused Pins: The description modified
		17. Multi-Function Pin Controller (MPC)	
		372	Table 17.1 List of Multiplexed Pin Configurations (3 / 3): Management data input/output interface: Pin functions MMDIO0 (input/output) and MMDC0 (output), deleted
		379	17.2.7 P5n Pin Function Control Register (P5nPFS), Table 17.7 Register Settings for the Input/ Output Function in the 320-pin FBGA Pin: 101011b: Pins MMDIO0 and MMDC0, deleted; Note 1 deleted
		380	17.2.8 P6n Pin Function Control Register (P6nPFS): The description modified
		383	17.2.11 PAn Pin Function Control Register (PAnPFS), Table 17.11 Register Settings for the Input/ Output Function: Note 1 modified
		18. 16-Bit Timer Pulse Unit (TPUa)	
		419	18.2.9 Timer Synchronous Register (TSYRA): The description modified
		424	Figure 18.5 Example of Setting Procedure for Waveform Output by Compare Match: The description in [1]: Index modified
		449	18.5 DMAC Activation: The description modified
		470	Figure 18.61 Conflict between TGRy Read Cycle and Input Capture Operation: Text modified
		19. Compare Match Timer (CMT)	
		479	Figure 19.5 Timing of Event Issuance: Text modified (CMCNTn → CMT1.CMCNT, CMCORn → CMT1.CMCOR)
		481	19.5.2 CMT Operation When Receiving an Event from ELC, (2) Event Count: The register name, modified
		483	19.5.3 Notes on CMT Event Link Operation, (1) Count Start: The register name, modified

Rev.	Date	Description	
		Page	Summary
1.70	Oct 23, 2020	484	19.5.3 Notes on CMT Event Link Operation, (2) Event Count: The register symbol, modified
		485	19.5.3 Notes on CMT Event Link Operation, (3) Count Clear: The register symbol, modified
		20. Compare Match Timer W (CMTW)	
		488	Table 20.1 Specifications of CMTW, Low-power consumption function: The functional description modified
		495	20.2.5 Compare Match Constant Register (CMWCOR): The description modified
		498	20.2.9 Digital Noise Filter Control Register 1 (NFCR1): The description modified
		21. Watchdog Timer (WDTA)	
		521	21.1 Overview: The redundant description deleted
		527	21.2.3 WDT Status Register (WDTSR): Table of bits: R/W of b14, b15, modified
		23. Management Data Input/Output Interface (MDIO)	
		554	23. Management Data Input/Output Interface (MDIO): The description modified
		570	Table 23.3 Specification of the Management Data Input/Output Interface (MDIO Master): The description of the number of channels, modified
		571	23.2.1.1 I/O Pins: The description modified
		571	Table 23.4 I/O Pins of the MDIO Master: Channel MDIOM0, deleted; Note 1 deleted
		571	Table 23.5 I/O Pins and the Corresponding I/O Ports: Channel MDIOM0, deleted
		572	Table 23.6 List of I/O Registers: Addresses B083 0012h, B091 C100h, B091 C500h, and B091 CD04h, deleted
		572	23.2.2.1 Bit Rate Register (BR): Address(es): MDIOM0.BR B083 0012h, deleted
		573	23.2.2.2 Transmission Register (TX): Address(es): MDIOM0.TX B091 C100h. deleted
		575	23.2.2.3 Version Register (VER): Address(es): MDIOM0.VER B091 C500h, deleted
		576	23.2.2.4 Reception Register (RX): Address(es): MDIOM0.RX B091 CD04h, deleted
		582	23.2.4.1 Initialization, 1. Configuration of the MDIO Master: The description in step 2, modified; step 8, deleted; step 9 changed to step 8
		24. Serial Communications Interface with FIFO (SCIFA)	
		611	24.3.1 Overview: The description modified
		617	Figure 24.4 Sample Flowchart for Transmitting Serial Data in Asynchronous Mode: The description in [2], modified
		622	Figure 24.10 Example of SCIFA Receive Operation in Asynchronous Mode Using Modem Control Function (RTS#): The typo in the title, corrected
		625	Figure 24.13 Sample Flowchart for Transmitting Serial Data in Clock Synchronous Mode: The description in [2], modified
		629	Figure 24.17 Sample Flowchart for Simultaneous Transmitting/Receiving Serial Data in Clock Synchronous Mode: The description in [2], modified
		25. I^2C Bus Interface (RIICa)	
		681	25.3.5 Slave Transmit Operation: The description in (1), modified (slave mode → slave reception mode)
		684	25.3.6 Slave Receive Operation: The description in (1), modified (slave mode → slave reception mode)
		688	Figure 25.22 SDA Output Delay Facility: [Transmission mode]: SCLn modified
		710	Table 25.6 Interrupt Sources: Note 1 and Note 2 switched; a part of the description in Note 1 and Note 2, deleted
		26. Serial Peripheral Interface (RSPIa)	
		733	26.2.10 RSPI Clock Delay Register (SPCKD): The description modified
		29. Error Control Module (ECM)	
		All	29.2 Registers: Table of bits: The functional descriptions of reserved bits (R/W) and (W), modified

Rev.	Date	Description	
		Page	Summary
1.70	Oct 23, 2020	30. 12-Bit A/D Converter (S12ADCa)	
		915	Table 30.4 Relationship between DBLANS Bit Settings and Double Trigger Enabled Channels: Note added
		917	Table 30.5 Selectable Targets for A/D Conversion Depending on the Settings of Scan Mode and Double Trigger Mode: "Group scan": The entry under "Self-Diagnosis" when DBLE = 1, modified
		952	30.3.2.6 A/D Conversion in Double Trigger Mode: The description modified
		952	Figure 30.9 Example of Operation in Single Scan Mode (Double Trigger Mode Selected: AN003 Duplicated; TRG4ABN Selected as Trigger; Self-Diagnosis Deselected): The title modified
		982	Table 30.12 Relationship between Mode Setting and S12ADI Interrupt Output: "Single scan mode": The entries when DBLE = 1, modified; "Group scan mode": The entries when DBLE = 0 and DBLE = 1, modified: Note 1 modified
		987	30.5.9 Caution When Using Self-Diagnosis, added (the subsequent section numbers changed)
		31. Temperature Sensor	
		991	31.3.1 Preparation for Using the Temperature Sensor: The description of the output voltage and the description in "Remark", modified
		993	Table 31.2 Time until Completion of A/D Conversion after the Start of Temperature-Sensor Operation: The times of tSAM and tED, modified
		34. Electrical Characteristics	
		1016	Table 34.4 DC Characteristics (3) The footnote number added: Three-state leakage current (off state) / 5-V tolerant pins / $\lvert I_{TSI} \rvert$ → Three-state leakage current (off state) / 5-V tolerant pins*2 / $\lvert I_{TSI} \rvert$
		1035	Table 34.19 RIICa Timing: Note 4 modified
		—	34.4.3.9 Serial Management Interface Timing, (2) Serial Management Interface (Master, Channel 0) Timing, deleted Table 34.21 Serial Management Interface (Master, Channel 0) Timing, deleted Figure 34.30 Serial Management Interface (Master, Channel 0) Timing, deleted
		1038	Table 34.21 Serial Management Interface (Master, Channel 1) Timing: Note 1 deleted
1.80	Jan. 31, 2022	12. Interrupt Controller (ICUA)	
		240	Table 12.3 Cortex-R4/DMAC Interrupt Vector Table (7/8): RIIC Unit0 → RIIC ch0, RIIC Unit1 → RIIC ch1
		14. DMA Controller (DMACAa)	
		277	14.2.12 Channel Configuration Register n (CHCFG_n): Note 2, added
		24. Serial Communications Interface with FIFO (SCIFA)	
		594	24.2.7 Serial Status Register (FSR): The description in "DR Bit (Receive Data Ready Flag)", modified. Note 2, added.
		634	24.7 Noise Cancellation: base clock (the clock with a frequency 16 or 8 times the transfer rate) → base clock (the clock with a frequency 16, 8, or 4 times the transfer rate) Note 1, modified

RZ/T1-M Group User's Manual: Hardware

Publication Date: Rev.0.60 Nov. 14, 2014
Rev.1.80 Jan. 31, 2022

Published by: Renesas Electronics Corporation

RZ/T1-M Group

Renesas Electronics Corporation

R01UH0521EJ0180