User's Manual

32

SH72A2 Group, SH72A0 Group

User's Manual: Hardware

Renesas 32-Bit RISC Microcomputer
SuperH™ RISC Engine Family / SH72Ax Series

Renesas Electronics
www.renesas.com

Rev.1.00 Apr 2012

Notice

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas Electronics products listed herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas Electronics such as that disclosed through our website.
2. Renesas Electronics does not assume any liability for infringement of patents, copyrights, or other intellectual property rights of third parties by or arising from the use of Renesas Electronics products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
3. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part.
4. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
5. When exporting the products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations. You should not use Renesas Electronics products or the technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. Renesas Electronics products and technology may not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations.
6. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
7. Renesas Electronics products are classified according to the following three quality grades: "Standard", "High Quality", and "Specific". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below. You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application categorized as "Specific" without the prior written consent of Renesas Electronics. Further, you may not use any Renesas Electronics product for any application for which it is not intended without the prior written consent of Renesas Electronics. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for an application categorized as "Specific" or for which the product is not intended where you have failed to obtain the prior written consent of Renesas Electronics. The quality grade of each Renesas Electronics product is "Standard" unless otherwise expressly specified in a Renesas Electronics data sheets or data books, etc.

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

   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; safety equipment; and medical equipment not specifically designed for life support.

   "Specific": Aircraft; aerospace equipment; submersible repeaters; nuclear reactor control systems; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.
8. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system 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. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

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

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

General Precautions in the Handling of MPU/MCU Products

The following usage notes are applicable to all MPU/MCU products from Renesas. For detailed usage notes on the products covered by this manual, refer to the relevant sections of the manual. If the descriptions under General Precautions in the Handling of MPU/MCU Products and in the body of the manual differ from each other, the description in the body of the manual takes precedence.

1. Handling of Unused Pins

   Handle unused pins in accord 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 unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of 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. Unused pins should be handled as described under Handling of Unused Pins in the manual.

2. Processing at Power-on

   The state of the product is undefined at the moment 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 moment 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 moment 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 moment when power is supplied until the power reaches the level at which resetting has been specified.

3. Prohibition of Access to Reserved Addresses

   Access to reserved addresses is prohibited.

   — The reserved addresses are provided for the possible future expansion of functions. Do not access these addresses; the correct operation of LSI is not guaranteed if they are accessed.

4. Clock Signals

   After applying a reset, only release the reset line after the operating clock signal has become stable. When switching the clock signal during program execution, wait until the target clock signal has 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. Moreover, 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.

5. Differences between Products

   Before changing from one product to another, i.e. to one with a different part number, confirm that the change will not lead to problems.

   — The characteristics of MPU/MCU in the same group but having different part numbers may differ because of the differences in internal memory capacity and layout pattern. When changing to products of different part numbers, implement a system-evaluation test for each of the products.

How to Use This Manual

1. Purpose and Target Readers

This manual is designed to provide the user with an understanding of the hardware functions and electrical characteristics of the MCU. It is intended for users designing application systems incorporating the MCU. A basic knowledge of electric circuits, logical circuits, and MCUs is necessary in order to use this manual.
The manual comprises an overview of the product; descriptions of the CPU, system control functions, peripheral functions, and electrical characteristics; and usage notes.

Particular attention should be paid to the precautionary notes when using the manual. These notes occur within the body of the text, at the end of each section, and in the Usage Notes section.

The revision history summarizes the locations of revisions and additions. It does not list all revisions. Refer to the text of the manual for details.

The following documents apply to the SH72A2 Group, SH72A0 Group. Make sure to refer to the latest versions of these documents. The newest versions of the documents listed may be obtained from the Renesas Technology Web site.

Document Type	Description	Document Title	Document No.
Shortsheet	Hardware overview		
Hardware manual	Hardware specifications (pin assignments, memory maps, peripheral function specifications, electrical characteristics, timing charts) and operation description Note: Refer to the application notes for details on using peripheral functions.	SH72A2 Group, SH72A0 Group Hardware Manual	This hardware manual
Software manual	Description of CPU instruction set	SH-2A, SH2A-FPU Software Manual	REJ09B0051
Application note	Information on using peripheral functions and application examples Sample programs Information on writing programs in assembly language and C	Available from Renesas Technology Web site.	
Renesas technical update	Product specifications, updates on documents, etc.		

2. Notation of Numbers and Symbols

The notation conventions for register names, bit names, numbers, and symbols used in this manual are described below.

(1) Register Names, Bit Names, and Pin Names
Registers, bits, and pins are referred to in the text by symbols. The symbol is accompanied by the word “register,” “bit,” or “pin” to distinguish the three categories.
Examples the DEF bit in the ABC register
PA01 pin, VCC pin

(2) Notation of Numbers
Binary numbers are notated as B'nnnn (However, the “B” may be omitted when it is clear that the number is binary), hexadecimal numbers are notated as H'nnnn, and decimal numbers are notated as nnnn.
Examples Binary: B'11 or 11
Hexadecimal: H'EFA0
Decimal: 1234

(3) Notation for Low Active Signals
A sharp sign (#) is appended to the names of signals and pins which are Low active.
Example the ABC# pin

3. Register Configuration

Each section in this manual provides a table listing all the registers used by the corresponding module before the register descriptions in the section. The symbols and terms used in these tables are described below.

[Register Table]

Register Name	Symbol	After Reset	Address	Access Size
Interrupt priority level register 1	IPR1	H'0000	H'FFFD 941A	16, 32
Interrupt priority level register 2	IPR2	H'0000	H'FFFD 941C	16, 32
Interrupt priority level register 3	IPR3	H'0000	H'FFFD 941E	16, 32
Interrupt priority level register 4	IPR4	H'0000	H'FFFD 9420	16, 32
Interrupt request register 102	IR102	H'00	H'FFFD 9800	8
[illegible]est register 103		H'00		

(1) (2) (3) (4) (5)

Note. • The bit names and text in the figure above are examples that are unrelated to the content of this manual.

(1) Register Name
The name and abbreviation are shown for each register.

(2) Symbol
Gives the name of the register.

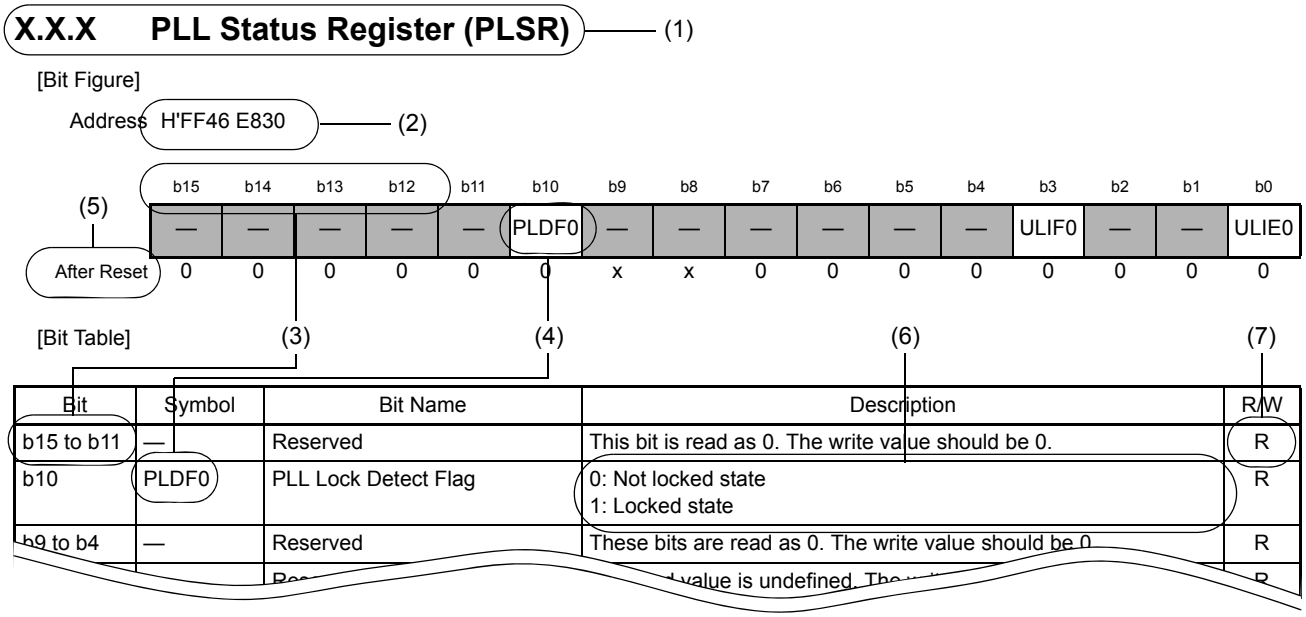
(3) After Reset
Indicates the values of each bit after a hardware reset in hexadecimal.

(4) Register Address
Indicates the location (address) of each register.

(5) Access Size
8-bit access is indicated as “8”, 16-bit access as “16”, and 32-bit access as “32”, respectively.
For registers that allow multiple accesses, each access size is indicated with a comma “,”.
If an access size is indicated without a comma “,”, only the indicated size is allowed.
- For 32-bit registers that can be accessed using 32-bit and 16-bit accesses
  The access size is indicated as “16/32”.
- For 8-bit registers that can be accessed using 8-bit access, and also using 16-bit access at the same time with the next aligned 8-bit register
  The access size is indicated as “8/16”.

4. Register Notation

Each register description includes both a bit figure that shows the bit sequence and a bit table that describes the content set with each bit. The symbols and terms used are described below.



Note. • The bit names and text in the figure above are examples that are unrelated to the content of this manual.

(1) Register Name
The name and abbreviation are shown for each register.

(2) Register Address
Indicates the location (address) of each register.

(3) Bit
Indicates the bit number.
The bits are shown in the order 31 to 0 for 32-bit registers and in the order 15 to 0 for 16-bit registers.

(4) Symbol
Indicates the name of the bit or field.
Reserved bits are indicated with a dash ("—") and are shaded in the bit figure.
Note that there are also cases, such as time counters, were the bit names are not given and the field is left blank.

(5) After Reset
Indicates the value of each bit after a hardware reset, power-on reset, voltage monitor reset, software reset, watchdog timer reset, or power-down mode wake-up reset, unless noted otherwise.
0 : 0
1 : 1
x : Undefined
— : Can not read

(6) Description
Describes the function of the bit or field.

(7) R/W
Indicates whether the bit(s) can be read, can be written, or can be neither read nor written. The following notations are used.
R/W : A bit or field that can be read or written.
R : A bit or field that can only be read.
Reserved bits are always notated as "R". If it is necessary to write to a reserved bit, the value specified in the bit table must be written.
W : A bit or field that can only be written.

4. List of Abbreviations and Acronyms

Abbreviation	Full Form
ACIA	Asynchronous Communication Interface Adapter
bps	bits per second
CRC	Cyclic Redundancy Check
DMA	Direct Memory Access
DMAC	Direct Memory Access Controller
GSM	Global System for Mobile Communications
Hi-Z	High Impedance
IEBus	Inter Equipment Bus
I/O	Input/Output
IrDA	Infrared Data Association
LSB	Least Significant Bit
MSB	Most Significant Bit
NC	Non-Connection
PLL	Phase Locked Loop
PWM	Pulse Width Modulation
SFR	Special Function Register
SIM	Subscriber Identity Module
UART	Universal Asynchronous Receiver/Transmitter
VCO	Voltage Controlled Oscillator

Table of Contents

1. Overview

1.1 Features

The SH72Ax Series incorporates the RISC SH2A-FPC core, which combines superscalar and Harvard architectures. With a multiplier and a floating point unit (FPU), the core achieves maximum code efficiency and high-speed processing. The SH72Ax Series integrates large-capacity ROM and RAM, and its various peripheral functions include a memory protection unit (MPU), an interrupt controller (INTC), DMAC, several types of timers, a serial communication interface (SCI), a serial bus interface (SBI), CAN, LIN, a 12-bit A/D converter and a 10-bit A/D converter, a CRC calculator, and I/O ports.

The SH72A2 group operates at frequencies up to 100 MHz. A 100-pin LQFP is available.

The SH72A0 group operates at frequencies up to 80 MHz. A 64-pin LQFP is available.

1.1.1 Applications

Automotive

1.1.2 Specifications

Tables 1.1 to 1.3 outline the specifications.

Table 1.1 Specifications Overview (1)

Type	Module/function	Description
CPU		SH2A-FPU core
	Central processing unit	• Maximum operating frequency: 100MHz • Minimum instruction execution time: 10 ns (One instruction in one cycle) • Number of instructions: 197 • Registers:General-purpose 32 bits × 16 registers Control 32 bits × 4 registers System 32 bits × 4 registers • Register banks: 15 • Operating mode: Single-chip mode • Superscala: Simultaneous execution of two instructions
	FPU (Optional)	• Conforms to IEEE-754 • Single-precision and double-precision supported • Number of instructions: 56 • Registers: 32 bits × 16 registers or 64 bits × 8 registers • Data formats: Single-precision floating point (32 bits) Double-precision floating point (64 bits)
Memory	ROM	Up to 512 Kbytes
	RAM	Up to 64 Kbytes
	Data flash	Up to 32 Kbytes
Memory protection unit (MPU) (Optional)		• Allows setting of 16 areas in the entire address space (H'0000 0000 to H'FFFF FFFF) for each of the bus masters (CPU and DMAC) and specifies the protection attributes for each area. • The protection attributes supported for the areas are as follows: Read enabled or disabled, write enabled or disabled, and instruction execution enabled or disabled (for CPU only)
Clock	Clock pulse generator	3 circuits: Main clock, PLL frequency synthesizer, and low-speed on-chip oscillator • Oscillation stop detection: Available (main clock monitor function) • Low-power consumption modes: CPU sleep mode, standby mode, and power-down mode
Reset		Hardware reset, software reset, watchdog timer reset, and power-down mode wake-up reset
Voltage monitor function (LVD) (Optional)		• Cold start-up/warm start-up function • Voltage moritor circuit 1 An interrupt can be generated when the VCC falls or rises (monitor voltage: Vdet1). Monitor voltage Vdet1 can be set in three levels.
Interrupts		• Number of interrupt vectors: Up to 512 • External interrupt input: NMI, INT × 14 (SH72A2 Group), and INT × 10 (SH72A0 Group) • Interrupt priority : 15 levels (levels 1 to 15)
Watchdog timer (WDT)		14 bits × 1
Task monitor timer		1 channel
Data transfer	DMAC	8 channels • DMA activation sources: software trigger, interrupt request from peripheral functions (CMT, MTU-III, TPU, SCI, SBI, or A/D converter)

Table 1.2 Specifications Overview (2)

Type	Module/function	Description
I/O ports	Programmable I/O ports	• Number of ports: 80 (SH72A2 Group) : 46 (SH72A0 Group) • CMOS I/O: Pull-up resistor selectable • Port output enable (POE)
Timer	Compare match timer (CMT)	16-bit timer × 6 (2 channels × 3 units)
	16-bit timer pulse unit (TPU)	16-bit timer × 16 channels (4 channels × 4 units) • Compare match • PWM waveform output • Input capture The above functions can be assigned to: Up to 16 pins (SH72A2 Group) Up to 11 pins (SH72A0 Group)
	Multifunction timer pulse unit (MTU-III)	16-bit timer × 8 channels Output compare: 24 channels (SH72A2 Group), 18 channels (SH72A0 Group) Input capture: 27 channels (SH72A2 Group), 18 channels (SH72A0 Group) • 3-phase motor control function × 2
Communication functions	Serial communication interface (SCI)	Clock synchronous/asynchronous × 4 channels (SH72A2 Group) Clock synchronous/asynchronous × 3 channels, clock asynchronous × 1 channel (SH72A0 Group)
	Serial bus interface (SBI)	4-wire clock synchronous × 4 channels (SH72A2 Group), 4-wire clock synchronous × 3 channels (SH72A0 Group) • Transfer bit length: Up to 32 bits • Transmit/receive buffers: Up to 128 bits The number of available CS pins is limited according to the channel in products of the SH72A0 group.
	LIN	2 channels (SH72A2 Group) 1 channel (SH72A0 Group) For single-channel products (products of the SH72A0 Group), the functions and the number of pins depend on the product. For details, see SH72A0 Group Pin Functions. • LIN protocol revisions 1.3, 2.0, and 2.1 supported • Dedicated for LIN master
	CAN	2 channels* The functions and the number of pins depend on the product. For details, see SH72A2 Group Pin Functions and SH72A0 Group Pin Functions. • Conforms to ISO 11898-1 Specification • 64 mailboxes
A/D converter	12-bit A/D converter	6 channels • Dedicated sample and hold function for each channel + common sample and hold function for all channels included
	10-bit A/D converter	24 channels (SH72A2 Group) 8 channels (SH72A0 Group) • Common sample and hold function for all channels included
CRC calculator		• CRC-CCITT ($X^{16} + X^{12} + X^5 + 1$) • CRC-32 ($X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X^1 + 1$)
Debugging interface	Advanced user debugger (AUD-II)	• RAM monitor mode AUDCK input clock: 10 MHz or less Module connected to internal bus is readable/writable.
	User break controller (UBC)	• An address, a data value, an access type, and a data size can be set as break conditions. • Sequential break function supported • 4 break channels
	User debugging interface (UDI)	• On-chip emulator interface supported
Operating frequency		80 to 100 MHz (SH72A2 Group) 64 to 80 MHz (SH72A0 Group)
Supply voltage		VCC = AVCC0 = VREFH0 = AVCC1 = 3.3 to 5.5V Note: VCC = AVCC0 = VREFH0 = AVCC1 = 3.3 to 4.5V (with some restrictions on characteristics)

Note: * The functions and number of pins depend on the product. For details, see List of SH72A2 Group Pin Functions and List of SH72A0 Group Pin Functions.

Table 1.3 Specifications Overview (3)

Type	Module/function	Description
Operating ambient temperature (°C)		-40 to 85°C (J version) -40 to 125°C (K version)
Package		SH72A2 Group : 100-pin LQFP (PLQP0100KB-A (Size: 14 × 14 mm, pitch: 0.5mm)) SH72A0 Group : 64-pin LQFP (PLQP0064KB-A (Size: 10 × 10 mm, pitch: 0.5 mm))

Table 1.4 Comparison of SH72Ax Series Functions

Group name		SH72A2 Group	SH72A0 Group
Number of pins		100 pins	64 pins
Voltage monitor function		Available	Available The functions and the number of pins depend on the product. For details, see List of SH72A0 Group Pin Functions.
Interrupt	NMI pin input	Available (1 pin)	
	INT pin input	Available (14 pins)	Available (10 pins)
Watchdog timer (WDT)		Available (14 bits × 1 channel)	
Task monitor timer		Available (1 channel)	
Data transfer	DMAC	Available (8 channels)	
Timer	Compare match timer (CMT)	Available (6 channels)	
	16-bit timer pulse unit (TPU)	Available (16 bits × 16 channels, 16 pins)	Available (16 bits × 16 channels, 11 pins)
	Multifunction timer pulse unit (MTU-III)	Available (16 bits × 8 channels) Compare match/PWM: 24 channels Input capture: 27 channels	Available (16 bits × 8 channels) Compare match/PWM: 18 channels Input capture: 18 channels
Communication functions	Serial communication interface (SCI)	Available (4 channels) The functions and the number of pins depend on the product. For details, see List of SH72A2 Group Pin Functions and List of SH72A0 Group Pin Functions.	
	Serial communiucation bus (SBI)	Available (4 channels)	Available (3 channels)
	LIN module (LIN)	Available (2 channels)	Available (1 channel) The functions and the number of pins depend on the product. For details, see List of SH72A0 Group Pin Functions.
	CAN module (CAN)	Available (2 channels)	Available (2 channels) The functions and the number of pins depend on the product. For details, see List of SH72A0 Group Pin Functions.
12-bit A/D converter (AD0)		6 channels	
10-bit A/D converter (AD1)		Available (24 channels)	Available (8 channels)
CRC calculator (CRC)		Available	
I/O ports	Input/Output	74 ports	40 ports
	Input	6 ports	6 ports
Advanced user debugger (AUD-II)		Available	
Maximum operating frequency		100 MHz	80 MHz
Package		100-pin LQFP (Size: 14 × 14 mm, pitch: 0.5 mm)	64-pin LQFP (Size: 10 × 10 mm, pitch: 0.5 mm)

[Legend]
Available: Supported
Not available: Not supported
Note: The functions and the number of pins depend on the product. For details, see SH72A2 Group Pin Functions and SH72A0 Group Pin Functions.

1.2 Product List

Table 1.5 is the product list for the SH72A2 group. Table 1.7 is the product list for the SH72A0 group. Figure 1.1 shows the correspondence of the part number, memory size, and package.

Table 1.5 SH72A2 Group Product List

Group	Part No.	ROM Capacity	RAM Capacity	Data Flash	Operating Frequency (max)	Operating Ambient Temperature	Operating Voltage VCC/ AVCC0/ AVCC1	MPU/ FPU	LVD Interrupt	Number of Channels in CAN/LIN
SH72A2	R5F72A28JADFP	512 Kbytes	64 Kbytes	32 Kbytes	100 MHz	-40 to +85°C	3.3 to 5.5 V *	Unavailable	Available	2 channels/ 2 channels
	R5F72A28KADFP					-40 to +125°C				
	R5F72A28JACFP				80 MHz	-40 to +85°C				
	R5F72A28KACFP					-40 to +125°C				
	R5F72A28JADFFP				100 MHz	-40 to +85°C		Available		
	R5F72A28KADFFP					-40 to +125°C				
	R5F72A28JACFFP				80 MHz	-40 to +85°C				
	R5F72A28KACFFP					-40 to +125°C				
	R5F72A26J3DFP	256 Kbytes	32 Kbytes	16 Kbytes	100 MHz	-40 to +85°C		Unavailable	Available	2 channels/ 2 channels
	R5F72A26K3DFP					-40 to +125°C				
	R5F72A26J3CFP				80 MHz	-40 to +85°C				
	R5F72A26K3CFP					-40 to +125°C				
	R5F72A26J3DFFP				100 MHz	-40 to +85°C		Available		
	R5F72A26K3DFFP					-40 to +125°C				
	R5F72A26J3CFFP				80 MHz	-40 to +85°C				
	R5F72A26K3CFFP					-40 to +125°C				
	R5F72A26J2DFP			—	100 MHz	-40 to +85°C		Unavailable	Available	2 channels/ 2 channels
	R5F72A26K2DFP					-40 to +125°C				
	R5F72A26J2CFP				80 MHz	-40 to +85°C				
	R5F72A26K2CFP					-40 to +125°C				
	R5F72A26J2DFFP				100 MHz	-40 to +85°C		Available		
	R5F72A26K2DFFP					-40 to +125°C				
	R5F72A26J2CFFP				80 MHz	-40 to +85°C				
	R5F72A26K2CFFP					-40 to +125°C				

Note: * VCC = AVCC0 = VREFH0 = AVCC1 = 3.3 to 4.5V (with some restrictions on characteristics)

Table 1.6 SH72A0 Group Product List

Group	Part No.	ROM Capacity	RAM Capacity	Data Flash	Operating Frequency (max)	Operating Ambient Temperature	Operating Voltage VCC/ AVCC0/ AVCC1	MPU/ FPU	LVD Interrupt	Number of Channels in CAN/LIN
SH72A0	R5F72A08JACFM	512 Kbytes	64 Kbytes	32 Kbytes	80 MHz	-40 to +85°C	3.3 to 5.5 V *	Unavailable	Available	2 channels/ 1 channel
	R5F72A08KACFM					-40 to +125°C				
	R5F72A08JABFM				64 MHz	-40 to +85°C				
	R5F72A08KABFM					-40 to +125°C				
	R5F72A08JACFFM				80 MHz	-40 to +85°C		Available		
	R5F72A08KACFFM					-40 to +125°C				
	R5F72A08JABFFM				64 MHz	-40 to +85°C				
	R5F72A08KABFFM					-40 to +125°C				
	R5F72A06J3CFM	256 Kbytes	32 Kbytes	16 Kbytes	80 MHz	-40 to +85°C		Unavailable	Available	2 channels/ 1 channel
	R5F72A06K3CFM					-40 to +125°C				
	R5F72A06J3BFM				64 MHz	-40 to +85°C				
	R5F72A06K3BFM					-40 to +125°C				
	R5F72A06J3CFFM				80 MHz	-40 to +85°C		Available		
	R5F72A06K3CFFM					-40 to +125°C				
	R5F72A06J3BFFM				64 MHz	-40 to +85°C				
	R5F72A06K3BFFM					-40 to +125°C				
	R5F72A06J2CFM			—	80 MHz	-40 to +85°C		Unavailable	Available	2 channels/ 1 channel
	R5F72A06K2CFM					-40 to +125°C				
	R5F72A06J2BFM				64 MHz	-40 to +85°C				
	R5F72A06K2BFM					-40 to +125°C				
	R5F72A06J2CFFM				80 MHz	-40 to +85°C		Available		
	R5F72A06K2CFFM					-40 to +125°C				
	R5F72A06J2BFFM				64 MHz	-40 to +85°C				
	R5F72A06K2BFFM					-40 to +125°C				
	R5F72A06K0BAFM	256 Kbytes	16 Kbytes	—	64 MHz	-40 to +125°C	4.5 to 5.5 V	Unavailable	Unavailable	1 channel/ unavailable

Note: * VCC = AVCC0 = VREFH0 = AVCC1 = 3.3 to 4.5V (with some restrictions on characteristics)

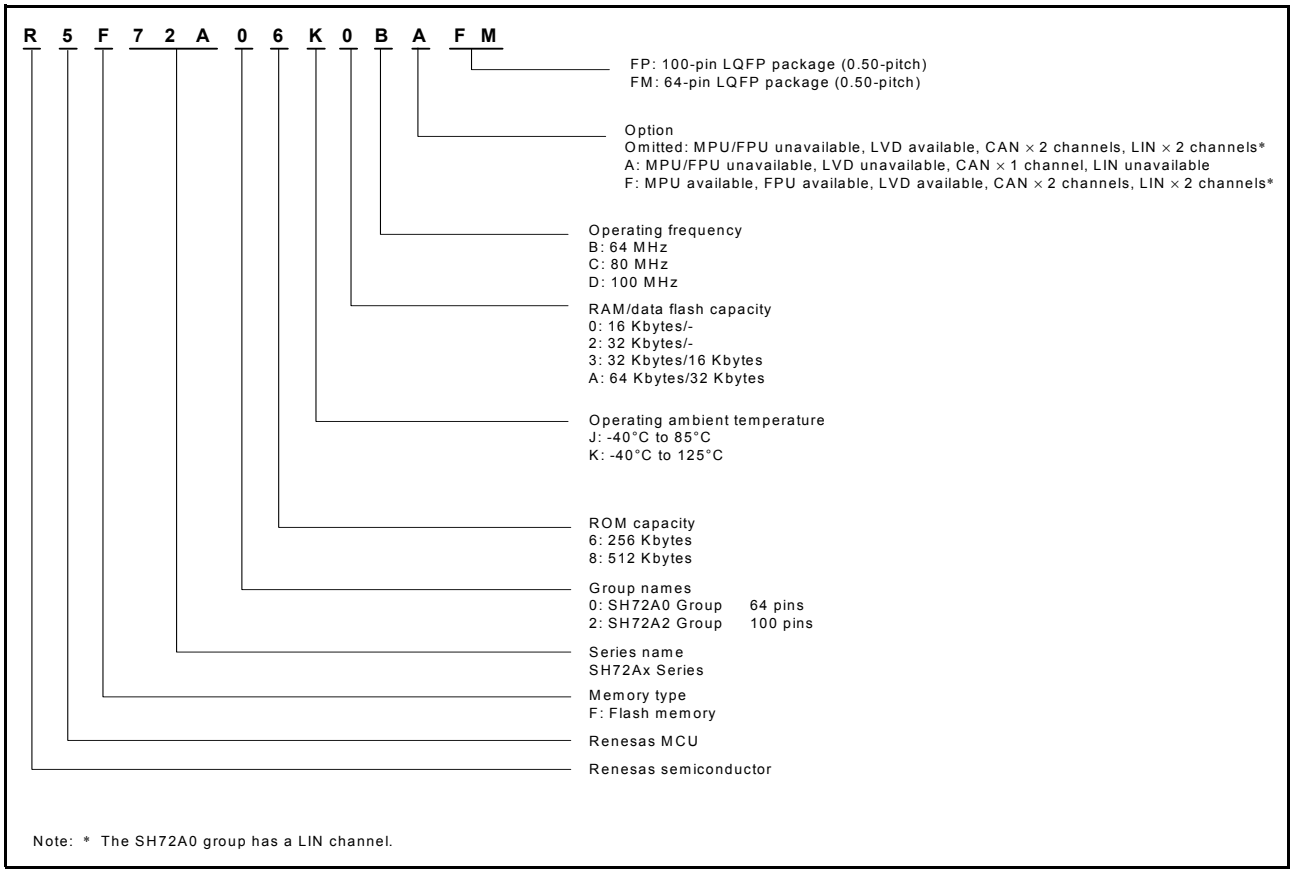


Figure 1.1 Part Number, Memory Size, and Package

1.3 Block Diagram

Figure 1.2 is a block diagram.

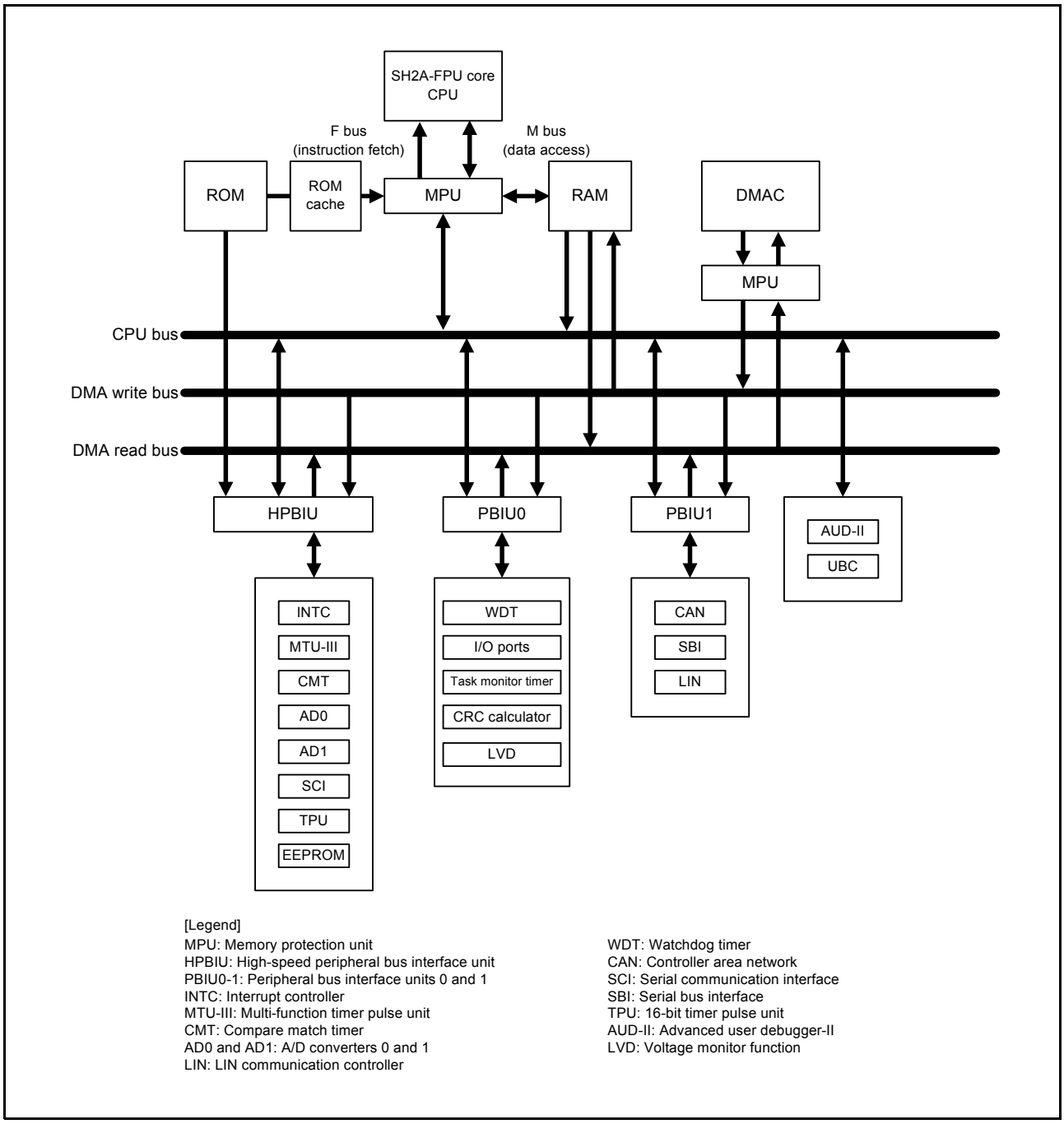
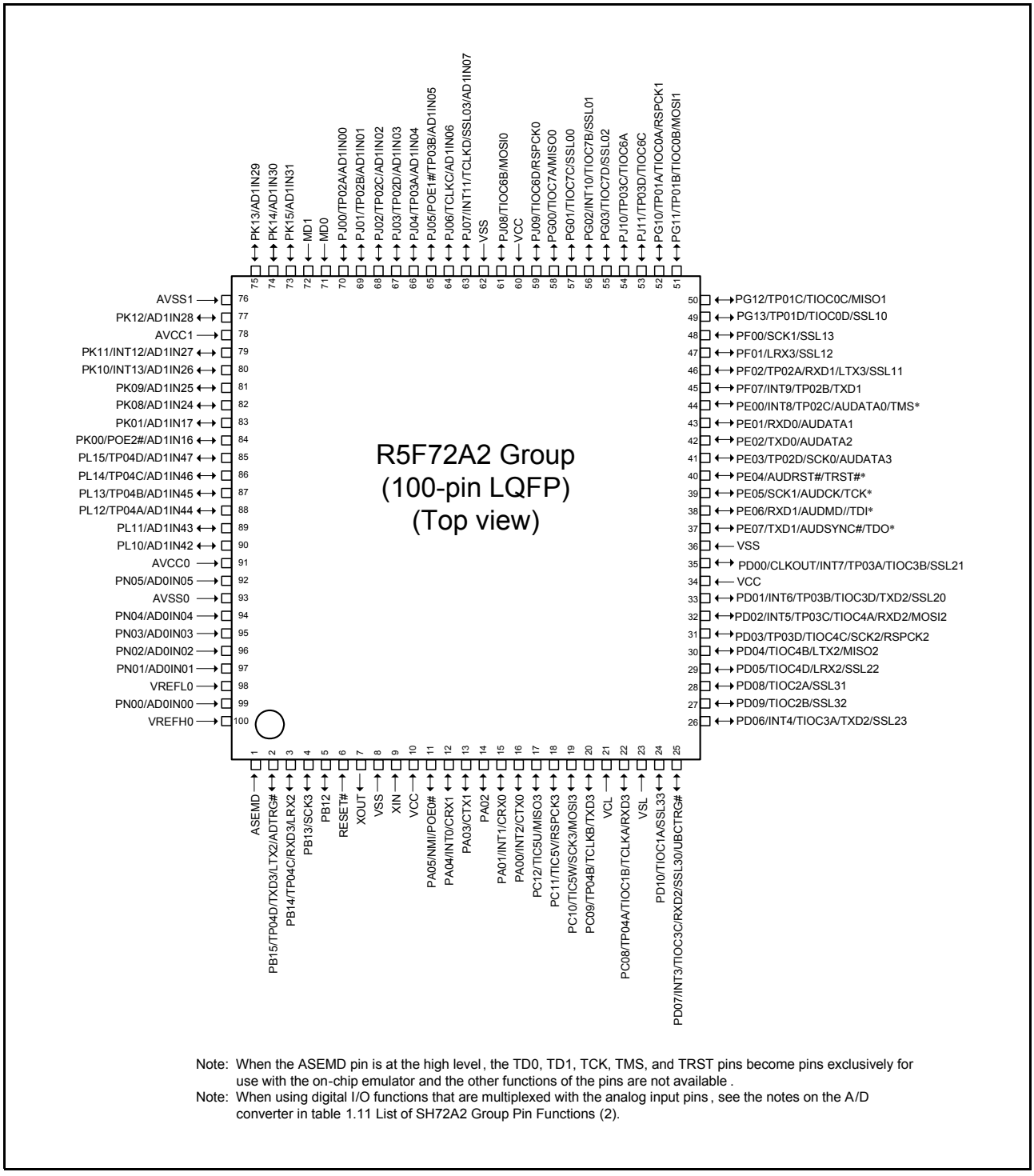


Figure 1.2 Block Diagram

1.4 Pin Description

1.4.1 SH72A2 Group Pin Assignment

Figure 1.3 shows the SH72A2 group pin assignment (top view). Tables 1.7 to 1.9 list the SH72A2 group pin assignment.



Note: When the ASEMD pin is at the high level, the TD0, TD1, TCK, TMS, and TRST pins become pins exclusively for use with the on-chip emulator and the other functions of the pins are not available.
Note: When using digital I/O functions that are multiplexed with the analog input pins, see the notes on the A/D converter in table 1.11 List of SH72A2 Group Pin Functions (2).

Figure 1.3 SH72A2 Group Pin Assignment (Top View) (Two-Channel CAN, Two-Channel LIN)

Table 1.7 SH72A2 Group Pin Assignment List (1)

Pin No.	Power supply, clock, system control	I/O Port	Interrupt	Timer		Communication function				A/D Converter	Debugging
				TPU	MTU-III	SCI	CAN	LIN	SBI		
1	ASEMD										
2		PB15		TP04D		TXD3		LTX2		ADTRG#	
3		PB14		TP04C		RXD3		LRX2			
4		PB13				SCK3					
5		PB12									
6	RESET#										
7	XOUT										
8	VSS										
9	XIN										
10	VCC										
11		PA05	NMI/POE0#								
12		PA04	INT0				CRX1				
13		PA03					CTX1				
14		PA02									
15		PA01	INT1				CRX0				
16		PA00	INT2				CTX0				
17		PC12			TIC5U				MISO3		
18		PC11			TIC5V				RSPCK3		
19		PC10			TIC5W	SCK3			MOSI3		
20		PC09		TP04B	TCLKB	TXD3					
21	VCL										
22		PC08		TP04A	TIOC1B /TCLKA	RXD3					
23	VSL										
24		PD10			TIOC1A				SSL33		
25		PD07	INT3		TIOC3C	RXD2			SSL30		UBCTRG#
26		PD06	INT4		TIOC3A	TXD2			SSL23		
27		PD09			TIOC2B				SSL32		
28		PD08			TIOC2A				SSL31		
29		PD05			TIOC4D			LRX2	SSL22		
30		PD04			TIOC4B			LTX2	MISO2		
31		PD03		TP03D	TIOC4C	SCK2			RSPCK2		
32		PD02	INT5	TP03C	TIOC4A	RXD2			MOSI2		
33		PD01	INT6	TP03B	TIOC3D	TXD2			SSL20		
34	VCC										
35	CLKOUT	PD00	INT7	TP03A	TIOC3B				SSL21		
36	VSS										
37		PE07				TXD1					TDO/ AUDSYNC#
38		PE06				RXD1					TDI/ AUDMD
39		PE05				SCK1					TCK/ AUDCK
40		PE04									TRST#/ AUDRST#
41		PE03		TP02D		SCK0					AUDATA3
42		PE02				TXD0					AUDATA2
43		PE01				RXD0					AUDATA1

Table 1.8 SH72A2 Group Pin Assignment List (2)

Pin No.	Power supply, clock, system control	I/O Port	Interrupt	Timer		Communication function				A/D Converter	Debugging
				TPU	MTU-III	SCI	CAN	LIN	SBI		
44		PE00	INT8	TP02C							TMS/ AUDATA0
45		PF07	INT9	TP02B		TXD1					
46		PF02		TP02A		RXD1		LTX3	SSL11		
47		PF01						LRX3	SSL12		
48		PF00				SCK1			SSL13		
49		PG13		TP01D	TIOC0D				SSL10		
50		PG12		TP01C	TIOC0C				MISO1		
51		PG11		TP01B	TIOC0B				MOSI1		
52		PG10		TP01A	TIOC0A				RSPCK1		
53		PJ11		TP03D	TIOC6C						
54		PJ10		TP03C	TIOC6A						
55		PG03			TIOC7D				SSL02		
56		PG02	INT10		TIOC7B				SSL01		
57		PG01			TIOC7C				SSL00		
58		PG00			TIOC7A				MISO0		
59		PJ09			TIOC6D				RSPCK0		
60	VCC										
61		PJ08			TIOC6B				MOSI0		
62	VSS										
63		PJ07	INT11		TCLKD				SSL03	AD1IN07	
64		PJ06			TCLKC					AD1IN06	
65		PJ05	POE1#	TP03B						AD1IN05	
66		PJ04		TP03A						AD1IN04	
67		PJ03		TP02D						AD1IN03	
68		PJ02		TP02C						AD1IN02	
69		PJ01		TP02B						AD1IN01	
70		PJ00		TP02A						AD1IN00	
71	MD0										
72	MD1										
73		PK15								AD1IN31	
74		PK14								AD1IN30	
75		PK13								AD1IN29	
76	AVSS1										
77		PK12								AD1IN28	
78	AVCC1										
79		PK11	INT12							AD1IN27	
80		PK10	INT13							AD1IN26	
81		PK09								AD1IN25	
82		PK08								AD1IN24	
83		PK01								AD1IN17	
84		PK00	POE2#							AD1IN16	
85		PL15		TP04D						AD1IN47	
86		PL14		TP04C						AD1IN46	
87		PL13		TP04B						AD1IN45	
88		PL12		TP04A						AD1IN44	
89		PL11								AD1IN43	
90		PL10								AD1IN42	

Table 1.9 SH72A2 Group Pin Assignment List (3)

Pin No.	Power supply, clock, system control	I/O Port	Interrupt	Timer		Communication function				A/D Converter	Debugging
				TPU	MTU-III	SCI	CAN	LIN	SBI		
91	AVCC0										
92		PN05								AD0IN05	
93	AVSS0										
94		PN04								AD0IN04	
95		PN03								AD0IN03	
96		PN02								AD0IN02	
97		PN01								AD0IN01	
98	VREFL0										
99		PN00								AD0IN00	
100	VREFH0										

1.4.2 List of SH72A2 Group Pin Functions

Tables 1.10 to 1.12 list the SH72A2 group pin functions.

Table 1.10 List of SH72A2 Group Pin Functions (1)

Type	Pin Name	I/O	Description
Power supply	VCC	Input	Power supply pin. Connect to the system power supply.
	VSS	Input	Ground pin. Connect to the system power supply (0 V).
	VCL	Input	External capacitor pin for internal high-voltage power supply. To stabilize this power supply, connect this pin to VSL via a capacitor (0.1 uF). Place the capacitor close to pin.
	VSL	Input	Dedicated ground Pin for VCL. To stabilize this power supply, connect this pin to VCL via a capacitor (0.1 uF). Place the capacitor close to pin.
Clock	XIN	Input	I/O pins for the main clock oscillation circuit. Connect a crystal oscillator between XIN and XOUT. To use an external clock, input it to XIN and leave XOUT open.
	XOUT	Output	
	CLKOUT	Output	Clock output pin.
System control	ASEMD	Input	This pin is provided to enable the on-chip emulator function. In normal operation other than debugging mode, input a low-level signal to this pin (connect this pin to VSS via a resistor). In debugging mode, input a high-level signal to it on the user system board.
	MD0, MD1	Input	These pins are provided to set MCU operating mode. Input a low-level signal to these pins (connect these pins to VSS via a resistor).
	RESET#	Input	Reset pin. This LSI enters a reset state when this pin goes low.
Interrupt	NMI	Input	Non-maskable interrupt reqiest pin.
	INT13 to INT0	Input	Maskable interrupt reqiest pins.
16-bit timer pulse unit (TPU)	TP01A to TP04A	Input/ output	TP01A to TP04A input capture input/output compare output/PWM output pins.
	TP01B to TP04B	Input/ output	TP01B to TP04B input capture input/output compare output/PWM output pins.
	TP01C to TP04C	Input/ output	TP01C to TP04C input capture input/output compare output/PWM output pins.
	TP01D to TP04D	Input/ output	TP01D to TP04D input capture input/output compare output/PWM output pins.
Multi-function timer pulse unit (MTU-III)	TCLKA, TCLKB, TCLKC, TCLKD	Input	Timer external clock input pins.
	TIOC0A, TIOC0B, TIOC0C, TIOC0D	Input/ output	TIOC0A to TIOC0D input capture input/output compare output/PWM output pins.
	TIOC1A, TIOC1B	Input/ output	TIOC1A and TIOC1B input capture input/output compare output/PWM output pins.
	TIOC2A, TIOC2B	Input/ output	TIOC2A and TIOC2B input capture input/output compare output/PWM output pins.
	TIOC3A, TIOC3B, TIOC3C, TIOC3D	Input/ output	TIOC3A to TIOC3D input capture input/output compare output/PWM output pins.
	TIOC4A, TIOC4B, TIOC4C,TIOC4D	Input/ output	TIOC4A to TIOC4D input capture input/output compare output/PWM output pins.
	TIC5U, TIC5V, TIC5W	Input	TIC5U, TIC5V, and TIC5W input capture input pins.

Table 1.11 List of SH72A2 Group Pin Functions (2)

Type	Pin Name	I/O	Description
Multi-function timer pulse unit (MTU-III)	TIOC6A, TIOC6B, TIOC6C, TIOC6D	Input/ output	TIOC6A to TIOC6D input capture input/output compare output/PWM output pins.
	TIOC7A, TIOC7B, TIOC7C, TIOC7D	Input/ output	TIOC7A to TIOC7D input capture input/output compare output/PWM output pins.
Serial communication interface (SCI)	RXD0 to RXD3	Input	SCI receive data input pins.
	SCK0 to SCK3	Input/ output	SCI clock I/O pins.
	TXD0 to TXD3	Output	SCI transmit data input pins.
Serial bus interface (SBI)	MISO0 to MISO3	Input/ output	SBI slave transmit data I/O pins.
	MOSI0 to MOSI3	Input/ output	SBI master transmit data I/O pins.
	RSPCK0 to RSPCK3	Input/ output	SBI clock I/O pins.
	SSL00, SSL10, SSL20, SSL30	Input/ output	SBI slave select I/O pins.
	SSL01 to SSL03, SSL11 to SSL13, SSL21 to SSL23, SSL31 to SSL33	Output	SBI slave select output pins.
LIN	LRX2, LRX3	Input	Input pins for the LIN communication function.
	LTX2, LTX3	Output	Output pins for the LIN communication function.
CAN	CRX0, CRX1	Input	Receive data input pins for the CAN communication function.
	CTX0, CTX1	Output	Transmit data output pins for the CAN communication function.
A/D converter	AD0IN00 to AD0IN05	Input	Analog input pins for the 12-bit A/D converter.
	AD1IN00 to AD1IN07	Input	Analog input pins for the 10-bit A/D converter. (Analog pin group A)
	AD1IN16 and AD1IN17, AD1IN24 to AD1IN31, AD1IN42 to AD1IN47	Input	Analog input pins for the 10-bit A/D converter. (Analog pin group B)
	ADTRG#	Input	External trigger input pin for starting A/D conversion.
Analog power supply	AVCC0	Input	Analog power supply pin for the 12-bit A/D converter. Connect to the system power supply when the 12-bit A/D converter is not used.
	AVSS0	Input	Ground pin for the 12-bit A/D converter. Connect to the system power supply (0 V).
	VREFH0	Input	Reference power supply pin for the 12-bit A/D converter. Connect to the system power supply when the 12-bit A/D converter is not used.
	VREFL0	Input	Ground pin of the reference power supply pin for the 12-bit A/D converter. Connect to the system power supply (0 V).
	AVCC1	Input	Analog power supply pin for the 10-bit A/D converter. Connect to the system power supply when the 10-bit A/D converter is not used.
	AVSS1	Input	Ground pin for the 10-bit A/D converter. Connect to the system power supply (0 V).
Port output enable (POE)	POE0# to POE2#	Input	Request signal input pins for driving the MTU-III waveform output pin to the high-impedance state.

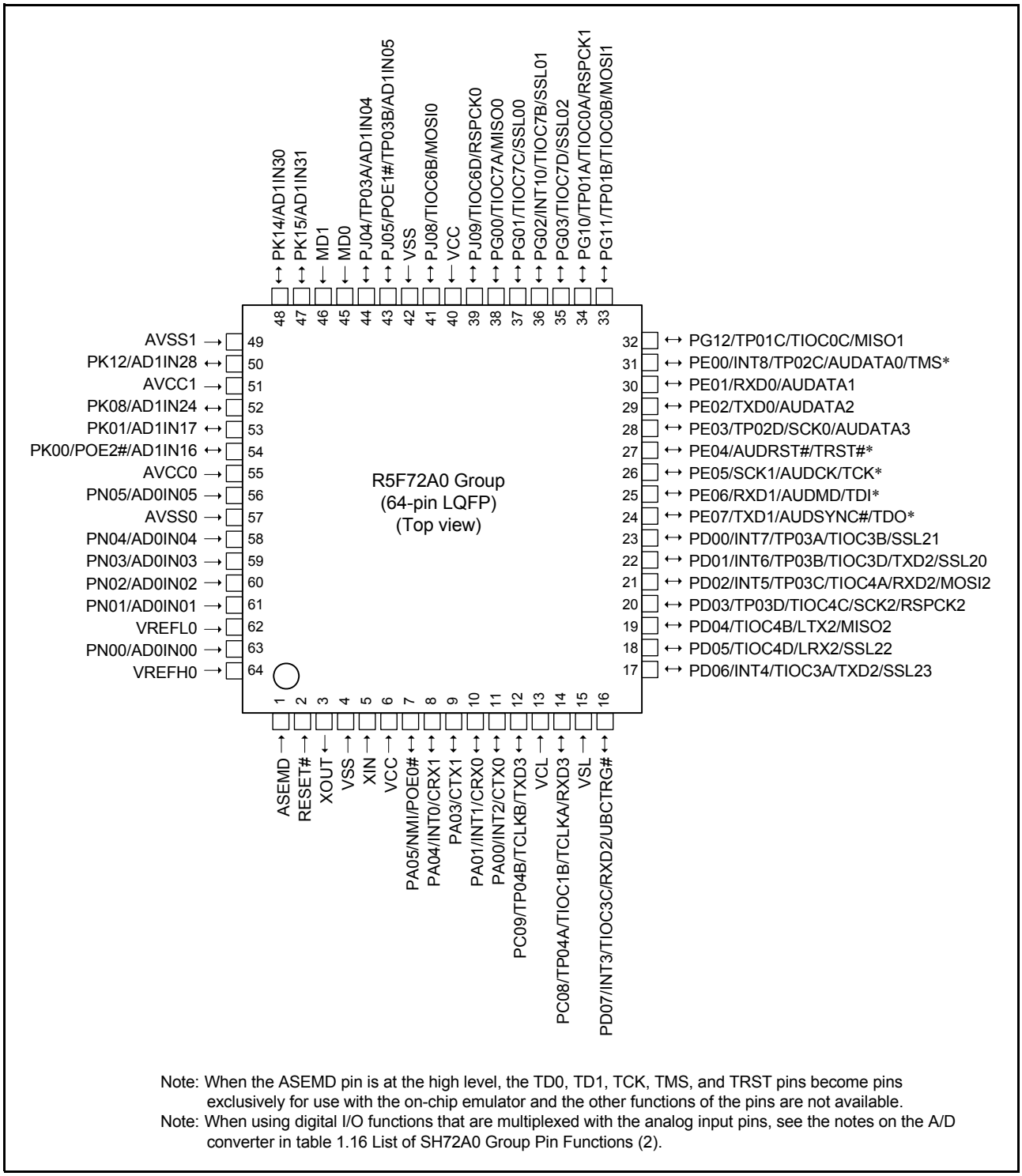
Notes: 1. When the AD01N00 to AD01N05 analog input pins are in use as digital inputs, the input values must be fixed.
2. Do not use the pins in analog pin group A (AD1IN00 to AD1IN07) as digital I/O pins. Only use pins in analog pin group A as analog input pins when this is required on top of pins in analog pin group B that are being used as analog input pins. When the pins in analog pin group B are used as digital I/O pins, the pins in analog pin group A must not be used as analog input pins.
3. Do not use the pins in analog pin group B (AD1IN16, AD1IN17, AD1IN24 to AD1IN31, AD1IN42 to AD1IN47) as digital I/O pins.

Table 1.12 List of SH72A2 Group Pin Functions (3)

Type	Pin Name	I/O	Description
I/O port	PA05 to PA00	Input/ output	6-bit general-purpose I/O pins.
	PB15 to PB12	Input/ output	4-bit general-purpose I/O pins.
	PC12 to PC08	Input/ output	5-bit general-purpose I/O pins.
	PD10 to PD00	Input/ output	11-bit general-purpose I/O pins.
	PE07 to PE00	Input/ output	8-bit general-purpose I/O pins.
	PF07, PF02 to PF00	Input/ output	4-bit general-purpose I/O pins.
	PG13 to PG10, PG03 to PG00	Input/ output	8-bit general-purpose I/O pins.
	PJ11 to PJ00	Input/ output	12-bit general-purpose I/O pins.
	PK15 to PK08, PK01 to PK00	Input/ output	10-bit general-purpose I/O pins.
	PL15 to PL10	Input/ output	6-bit general-purpose I/O pins.
	PN05 to PN00	Input	6-bit general-purpose I/O pins. (Only a fixed value can be input)
Advanced user debugger (AUD-II)	AUDATA3 to AUDATA0	Input/ output	RAM monitor mode: Monitor address input/data output pins.
	AUDRST#	Input	AUD reset input pin.
	AUDMD	Input	AUD mode selection pin. When it is set to a high level, RAM monitor mode is selected.
	AUDCK	Input/ output	RAM monitor mode: Sync clock input pin.
	AUDSYNC#	Input/ output	RAM monitor mode: Input pin for data start position identification signal.
User break controller (UBC)	UBCTRG#	Output	Trigger output pin for UBC condition match.
Debugging	TCK	Input	Dedicated pins for the on-chip emulator. When the ASEMD pin is set to a high level, these pins become dedicated pins for the on-chip emulator.
	TMS	Input	
	TDI	Input	
	TDO	Output	
	TRST	Input	

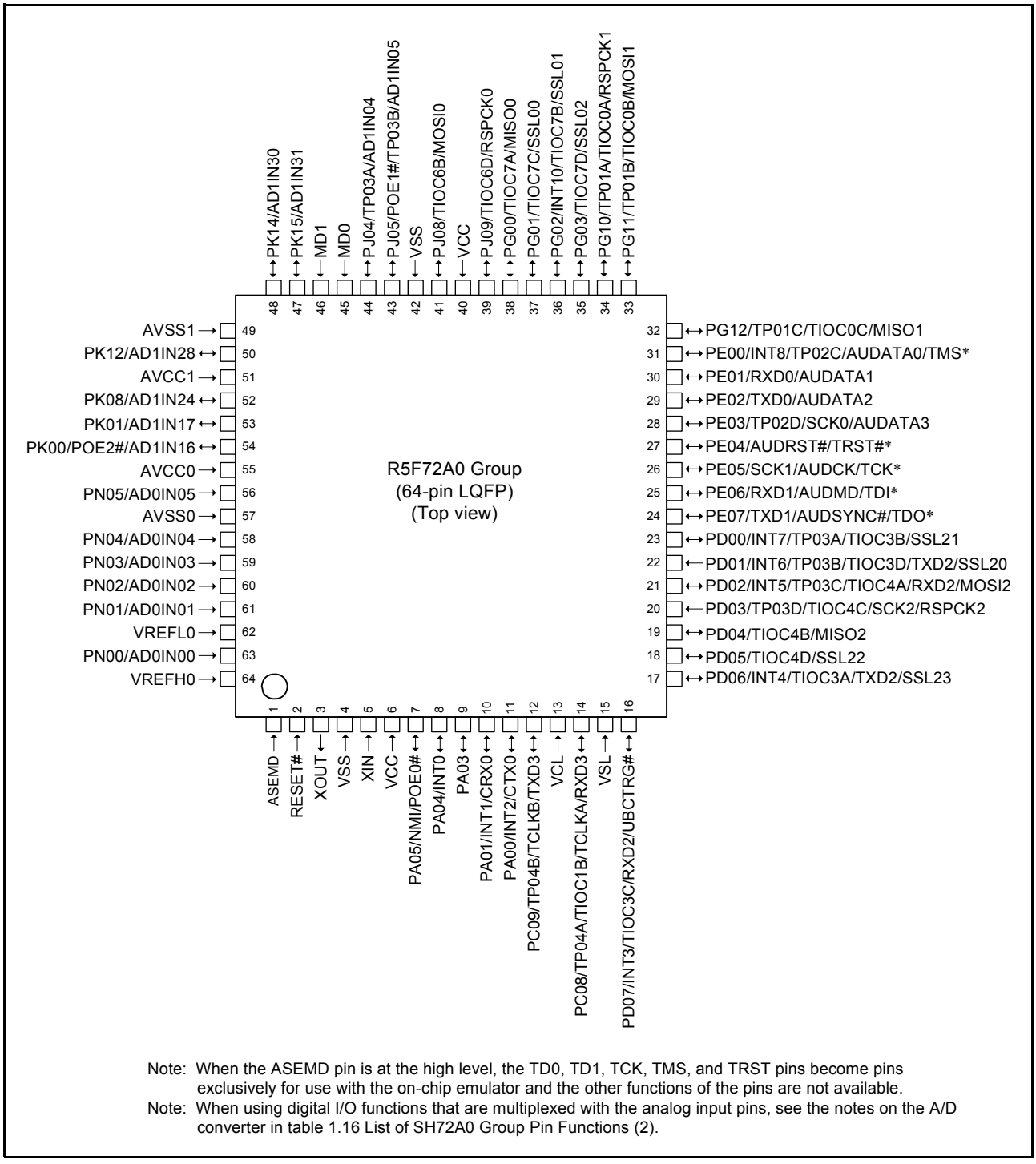
1.4.3 SH72A0 Group Pin Assignment

Figure 1.4 shows the SH72A0 group pin assignment (top view) (excluding R5F72A06K0BAF). Figure 1.5 shows the pin assignment (top view). Tables 1.13 and 1.14 list the SH72A0 group pin assignment.



Note: When the ASEMD pin is at the high level, the TD0, TD1, TCK, TMS, and TRST pins become pins exclusively for use with the on-chip emulator and the other functions of the pins are not available.
Note: When using digital I/O functions that are multiplexed with the analog input pins, see the notes on the A/D converter in table 1.16 List of SH72A0 Group Pin Functions (2).

Figure 1.4 SH72A0 Group Pin Assignment (Top View) (Two-Channel CAN, One-Channel LIN)



Note: When the ASEMD pin is at the high level, the TD0, TD1, TCK, TMS, and TRST pins become pins exclusively for use with the on-chip emulator and the other functions of the pins are not available.

Note: When using digital I/O functions that are multiplexed with the analog input pins, see the notes on the A/D converter in table 1.16 List of SH72A0 Group Pin Functions (2).

Figure 1.5 SH72A0 Group Pin Assignment (Top View) (One-Channel CAN, LIN Unavailable)

Table 1.13 SH72A0 Group Pin Assignment List (1)

Pin No.	Power supply, clock, system control	I/O Port	Interrupt	Timer		Communication function				A/D Converter	Debugging
				TPU	MTU-III	SCI	CAN	LIN	SBI		
1	ASEMD										
2	RESET#										
3	XOUT										
4	VSS										
5	XIN										
6	VCC										
7		PA05	NMI/POE0#								
8		PA04	INT0				CRX1*				
9		PA03					CTX1*				
10		PA01	INT1				CRX0				
11		PA00	INT2				CTX0				
12		PC09		TP04B	TCLKB	TXD3					
13	VCL										
14		PC08		TP04A	TIOC1B/ TCLKA	RXD3					
15	VSL										
16		PD07	INT3		TIOC3C	RXD2					UBCTRG#
17		PD06	INT4		TIOC3A	TXD2			SSL23		
18		PD05			TIOC4D			LRX2*	SSL22		
19		PD04			TIOC4B			LTX2*	MISO2		
20		PD03		TP03D	TIOC4C	SCK2			RSPCK2		
21		PD02	INT5	TP03C	TIOC4A	RXD2			MOSI2		
22		PD01	INT6	TP03B	TIOC3D	TXD2			SSL20		
23		PD00	INT7	TP03A	TIOC3B				SSL21		
24		PE07				TXD1					TDO/ AUDSYNC#
25		PE06				RXD1					TDI/ AUDMD
26		PE05				SCK1					TCK/ AUDCK
27		PE04									TRST#/ AUDRST#
28		PE03		TP02D		SCK0					AUDATA3
29		PE02				TXD0					AUDATA2
30		PE01				RXD0					AUDATA1
31		PE00	INT8	TP02C							TMS/ AUDATA0
32		PG12		TP01C	TIOC0C				MISO1		
33		PG11		TP01B	TIOC0B				MOSI1		
34		PG10		TP01A	TIOC0A				RSPCK1		
35		PG03			TIOC7D				SSL02		
36		PG02	INT10		TIOC7B				SSL01		
37		PG01			TIOC7C				SSL00		
38		PG00			TIOC7A				MISO0		
39		PJ09			TIOC6D				RSPCK0		
40	VCC										

Note: *Optional

Table 1.14 SH72A0 Group Pin Assignment List (2)

Pin No.	Power supply, clock, system control	I/O Port	Interrupt	Timer		Communication function				A/D Converter	Debugging
				TPU	MTU-III	SCI	CAN	LIN	SBI		
41		PJ08			TIOC6B				MOSI0		
42	VSS										
43		PJ05	POE1#	TP03B						AD1IN05	
44		PJ04		TP03A						AD1IN04	
45	MD0										
46	MD1										
47		PK15								AD1IN31	
48		PK14								AD1IN30	
49	AVSS1										
50		PK12								AD1IN28	
51	AVCC1										
52		PK08								AD1IN24	
53		PK01								AD1IN17	
54		PK00	POE2#							AD1IN16	
55	AVCC0										
56		PN05								AD0IN05	
57	AVSS0										
58		PN04								AD0IN04	
59		PN03								AD0IN03	
60		PN02								AD0IN02	
61		PN01								AD0IN01	
62	VREFL0										
63		PN00								AD0IN00	
64	VREFH0										

1.4.4 List of SH72A0 Group Pin Functions

Tables 1.15 to 1.17 list the SH72A0 group pin functions.

Table 1.15 List of SH72A0 Group Pin Functions (1)

Type	Pin Name	I/O	Description
Power supply	VCC	Input	Power supply pin. Connect to the system power supply.
	VSS	Input	Ground pin. Connect to the system power supply (0 V).
	VCL	Input	External capacitor pin for internal high-voltage power supply. To stabilize this power supply, connect this pin to VSL via a capacitor (0.1 uF). Place the capacitor close to pin.
	VSL	Input	Ground Pin for VCL. To stabilize this power supply, connect this pin to VCL via a capacitor (0.1 uF). Place the capacitor close to pin.
Clock	XIN	Input	I/O pins for the main clock oscillation circuit. Connect a crystal oscillator between XIN and XOUT. To use an external clock, input it to XIN and leave XOUT open.
	XOUT	Output	
System control	ASEMD	Input	This pin is provided to enable the on-chip emulator function. In normal operation other than debugging mode, input a low-level signal to this pin (connect this pin to VSS via a resistor). In debugging mode, input a high-level signal to it on the user system board.
	MD0, MD1	Input	These pins are provided to set MCU operating mode. Input a low-level signal to these pins (connect these pins to VSS via a resistor).
	RESET#	Input	Reset pin. This LSI enters a reset state when this pin goes low.
Interrupt	NMI	Input	Non-maskable interrupt reqiest pin.
	INT10, INT8 to INT0	Input	Maskable interrupt reqiest pins.
16-bit timer pulse unit (TPU)	TP01A, TP03A, TP04A	Input/ output	TP01A, TP03A, TP04A input capture input/output compare output/PWM output pins.
	TP01B, TP03B, TP04B	Input/ output	TP01B, TB03B, TP04B input capture input/output compare output/PWM output pins.
	TP01C, TP02C, TP03C	Input/ output	TP01C, TP02C, TP03C input capture input/output compare output/PWM output pins.
	TP02D, TP03D	Input/ output	TP02D and TP03D input capture input/output compare output/PWM output pins.
Multi-function timer pulse unit (MTU-III)	TCLKA, TCLKB	Input	Timer external clock input pins.
	TIOC0A, TIOC0B, TIOC0C	Input/ output	TIOC0A to TIOC0C input capture input/output compare output/PWM output pins.
	TIOC1B	Input/ output	TIOC1B input capture input/output compare output/PWM output pin.
	TIOC3A, TIOC3B, TIOC3C, TIOC3D	Input/ output	TIOC3A to TIOC3D input capture input/output compare output/PWM output pins.
	TIOC4A, TIOC4B TIOC4C, TIOC4D	Input/ output	TIOC4A to TIOC4D input capture input/output compare output/PWM output pins.

Table 1.16 List of SH72A0 Group Pin Functions (2)

Type	Pin Name	I/O	Description
Multi-function timer pulse unit (MTU-III)	TIOC6B, TIOC6D	Input/ output	TIOC6B and TIOC6D input capture input/output compare output/PWM output pins.
	TIOC7A, TIOC7B, TIOC7C, TIOC7D	Input/ output	TIOC7A to TIOC7D input capture input/output compare output/PWM output pins.
Serial communication interface (SCI)	RXD0 to RXD3	Input	SCI receive data input pins.
	SCK0 to SCK2	Input/ output	SCI clock I/O pins.
	TXD0 to TXD3	Output	SCI transmit data input pins.
Serial bus interface (SBI)	MISO0 to MISO2	Input/ output	SBI slave transmit data I/O pins.
	MOSI0 to MOSI2	Input/ output	SBI master transmit data I/O pins.
	RSPCK0 to RSPCK2	Input/ output	SBI clock I/O pins.
	SSL00, SSL20	Input/ output	SBI slave select I/O pins.
	SSL01 to SSL02, SSL21 to SSL23	Output	SBI slave select output pins.
LIN option*1	LRX2	Input	Input pins for the LIN communication function.
	LTX2	Output	Output pins for the LIN communication function.
CAN	CRX0, CRX1*1	Input	Receive data input pins for the CAN communication function.
	CTX0, CTX1*1	Output	Transmit data output pins for the CAN communication function.
A/D converter	AD0IN00 to AD0IN05	Input	Analog input pins for the 12-bit A/D converter.*2
	AD1IN04, AD1IN05	Input	Analog input pins for the 10-bit A/D converter. (Analog pin group A)*3
	AD1IN16, AD1IN17, AD1IN24, AD1IN28, AD1IN30, AD1IN31	Input	Analog input pins for the 10-bit A/D converter. (Analog pin group B)*4
Analog power supply	AVCC0	Input	Analog power supply pin for the 12-bit A/D converter. Connect to the system power supply when the 12-bit A/D converter is not used.
	AVSS0	Input	Ground pin for the 12-bit A/D converter. Connect to the system power supply (0 V).
	VREFH0	Input	Reference power supply pin for the 12-bit A/D converter. Connect to the system power supply when the 12-bit A/D converter is not used.
	VREFL0	Input	Ground pin of the reference power supply pin for the 12-bit A/D converter. Connect to the system power supply (0 V).
	AVCC1	Input	Analog power supply pin for the 10-bit A/D converter. Connect to the system power supply when the 10-bit A/D converter is not used.
	AVSS1	Input	Ground pin for the 10-bit A/D converter. Connect to the system power supply (0 V).
Port output enable (POE)	POE0# to POE2#	Input	Request signal input pins for driving the MTU-III waveform output pin to the high-impedance state.

Notes: 1. Optional
2. When the AD01N00 to AD01N05 analog input pins are in use as digital inputs, the input values must be fixed.
3. Do not use the pins in analog pin group A (AD1IN04 and AD1IN05) as digital I/O pins. Only use pins in analog pin group A as analog input pins when this is required on top of pins in analog pin group B that are being used as analog input pins. When the pins in analog pin group B are used as digital I/O pins, the pins in analog pin group A must not be used as analog input pins.
4. Do not use the pins in analog pin group B (AD1IN16, AD1IN17, AD1IN24, AD1IN28, AD1IN30, and AD1IN31 to AD1IN47) as digital I/O pins.

Table 1.17 List of SH72A0 Group Pin Functions (3)

Type	Pin Name	I/O	Description
I/O port	PA05 to PA03, PA01 to PA00	Input/ output	5-bit general-purpose I/O pins.
	PC09 to PC08	Input/ output	2-bit general-purpose I/O pins.
	PD07 to PD00	Input/ output	8-bit general-purpose I/O pins.
	PE07 to PE00	Input/ output	8-bit general-purpose I/O pins.
	PG12 to PG10, PG03 to PG00	Input/ output	7-bit general-purpose I/O pins.
	PJ09 to PJ08, PJ05 to PJ04	Input/ output	4-bit general-purpose I/O pins.
	PK15 to PK14, PK12, PK08, PK01 to PK00	Input/ output	6-bit general-purpose I/O pins.
	PN05 to PN00	Input	6-bit general-purpose I/O pins. (Only a fixed value can be input)
Advanced user debugger (AUD-II)	AUDATA3 to AUDATA0	Input/ output	RAM monitor mode: Monitor address input/data output pins.
	AUDRST#	Input	AUD reset input pin.
	AUDMD	Input	AUD mode selection pin. When it is set to a high level, RAM monitor mode is selected.
	AUDCK	Input/ output	RAM monitor mode: Sync clock input pin.
	AUDSYNC#	Input/ output	RAM monitor mode: Input pin for data start position identification signal.
User break controller (UBC)	UBCTRG#	Output	Trigger output pin for UBC condition match.
Debugging	TCK	Input	Dedicated pins for the on-chip emulator. When the ASEMD pin is set to a high level, these pins become dedicated pins for the on-chip emulator.
	TMS	Input	
	TDI	Input	
	TDO	Output	
	TRST	Input	

1.4.5 Handling of Unused Pins

Unused input pins should be fixed high or low.

2. CPU

2.1 Data Format

Figure 2.1 shows the data format supported by the SH2A-FPU.

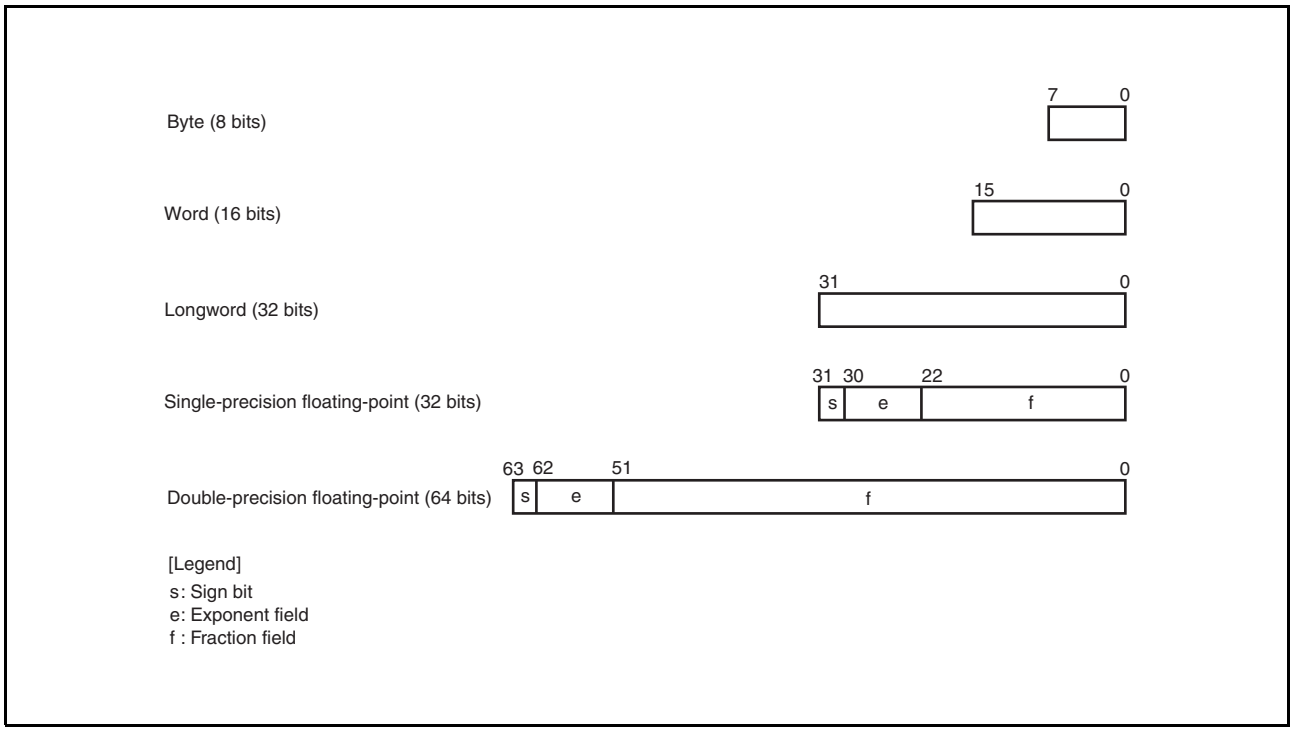


Figure 2.1 Data Format

2.2 Register Descriptions

2.2.1 General Registers

The general registers consist of sixteen 32-bit registers, numbered R0 to R15, and are used for data processing and address calculation. R0 is also used as an index register. Several instructions have R0 fixed as their only usable register. R15 is used as the hardware stack pointer (SP). Saving and restoring the status register (SR) and program counter (PC) in exception handling is accomplished by referencing the stack using R15. Figure 2.2 shows the general registers.

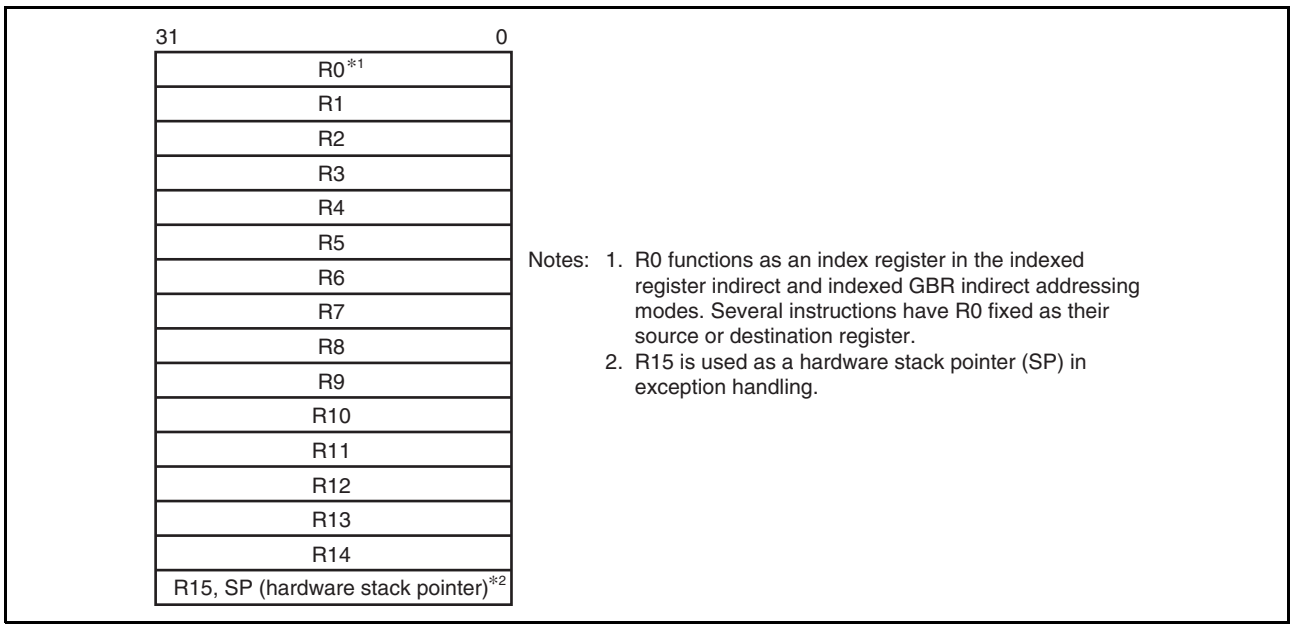


Figure 2.2 General Registers

2.2.2 Control Registers

The control registers consist of four 32-bit registers: the status register (SR), the global base register (GBR), the vector base register (VBR), and the jump table base register (TBR).

The status register indicates instruction processing states.

The global base register functions as a base address for the GBR indirect addressing mode to transfer data to the registers of on-chip peripheral modules.

The vector base register functions as the base address of the exception handling vector area (including interrupts).

The jump table base register functions as the base address of the function table area.

Figure 2.3 shows the control registers.

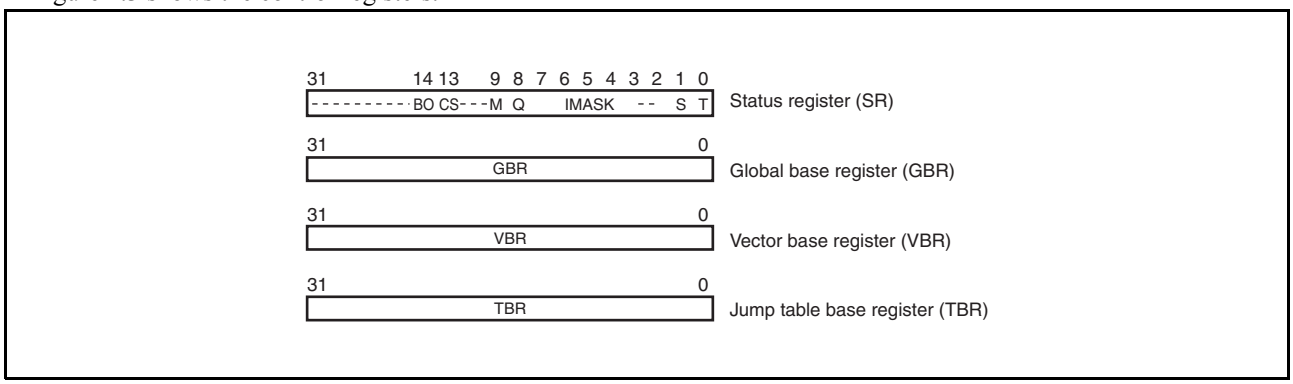


Figure 2.3 Control Registers

(1) Status Register (SR)

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
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
	—	BO	CS	—	—	—	M	Q	IMASK [3:0]				—	—	S	T
After Reset	0	0	0	0	0	0	x	x	1	1	1	1	0	0	x	x

Bit	Symbol	Bit Name	Description	R/W
b31 to b15	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b14	BO	BO Bit	Indicates the register bank has overflowed.	R/W
b13	CS	CS Bit	Indicates, in CLIP instruction execution, the value has exceeded the saturation upper-limit value or fallen below the saturation lower-limit value.	R/W
b12 to b10	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b9	M	M Bit	Used by the DIV0S, DIV0U, and DIV1 instructions.	R/W
b8	Q	Q Bit	Used by the DIV0S, DIV0U, and DIV1 instructions.	R/W
b7 to b4	IMASK [3:0]	IMASK Bits	These bits are 4-bit data and indicate the interrupt mask level.	R/W
b3, b2	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b1	S	S Bit	Specifies a saturation operation for a MAC instruction.	R/W
b0	T	T Bit	True/false condition or carry/borrow bit	R/W

(2) Global Base Register (GBR)

GBR is referenced as the base address in a GBR-referencing MOV instruction.

The value is undefined after a reset.

(3) Vector Base Register (VBR)

VBR is referenced as the branch destination base address when an exception or an interrupt occurs.

The value is H'0000 0000 after a reset.

(4) Jump Table Base Register (TBR)

TBR is referenced as the start address of a function table located in memory in a JSR/N@@(disp8,TBR) table-referencing subroutine call instruction. The value is undefined after a reset.

2.2.3 System Registers

The system registers consist of four 32-bit registers: the high and low multiply and accumulate registers (MACH and MACL), the procedure register (PR), and the program counter (PC). MACH and MACL store the results of multiply or multiply-and-accumulate operations. PR stores the return address from a subroutine procedure. PC indicates the program address being executed and controls the flow of the processing. Figure 2.4 shows the system registers.

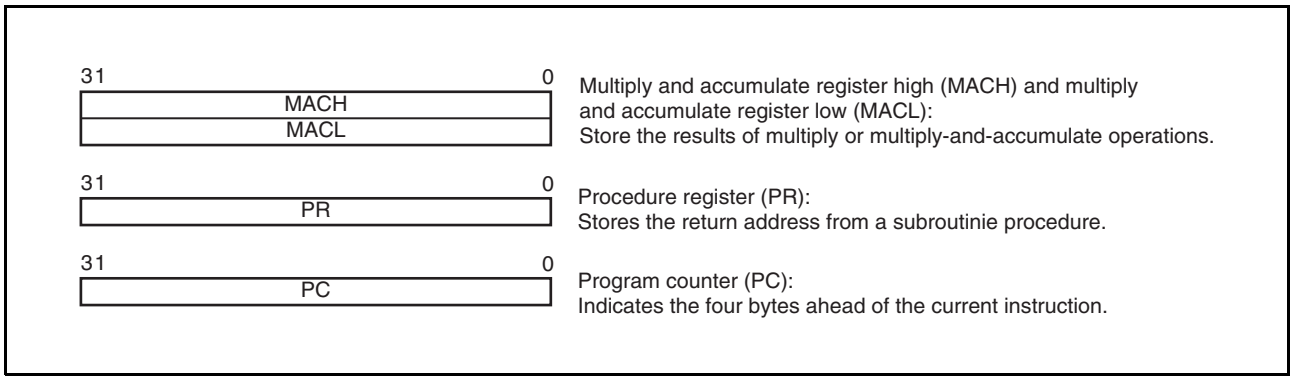


Figure 2.4 System Registers

(1) Multiply and Accumulate Register High (MACH) and Multiply and Accumulate Register Low (MACL)

MACH and MACL are used as the addition value in a MAC instruction, and store the result of a MAC or MUL instruction. The value is undefined after a reset.

(2) Procedure Register (PR)

PR stores the return address of a subroutine call using a BSR, BSRF, or JSR instruction, and is referenced by a subroutine return instruction (RTS). The value is undefined after a reset.

(3) Program Counter (PC)

PC indicates the address of the instruction being executed. The value after a reset is that of PC in the vector table.

2.2.4 Floating-Point Registers

There are sixteen 32-bit floating-point registers, FPR0 to FPR15. These sixteen registers are referenced as FR0 to FR15, DR0, DR2, DR4, DR6, DR8, DR10, DR12, and DR14. The correspondence between FPRn and the referenced name is determined by the PR and SZ bits in FPSCR. Figure 2.5 shows the floating-point registers.

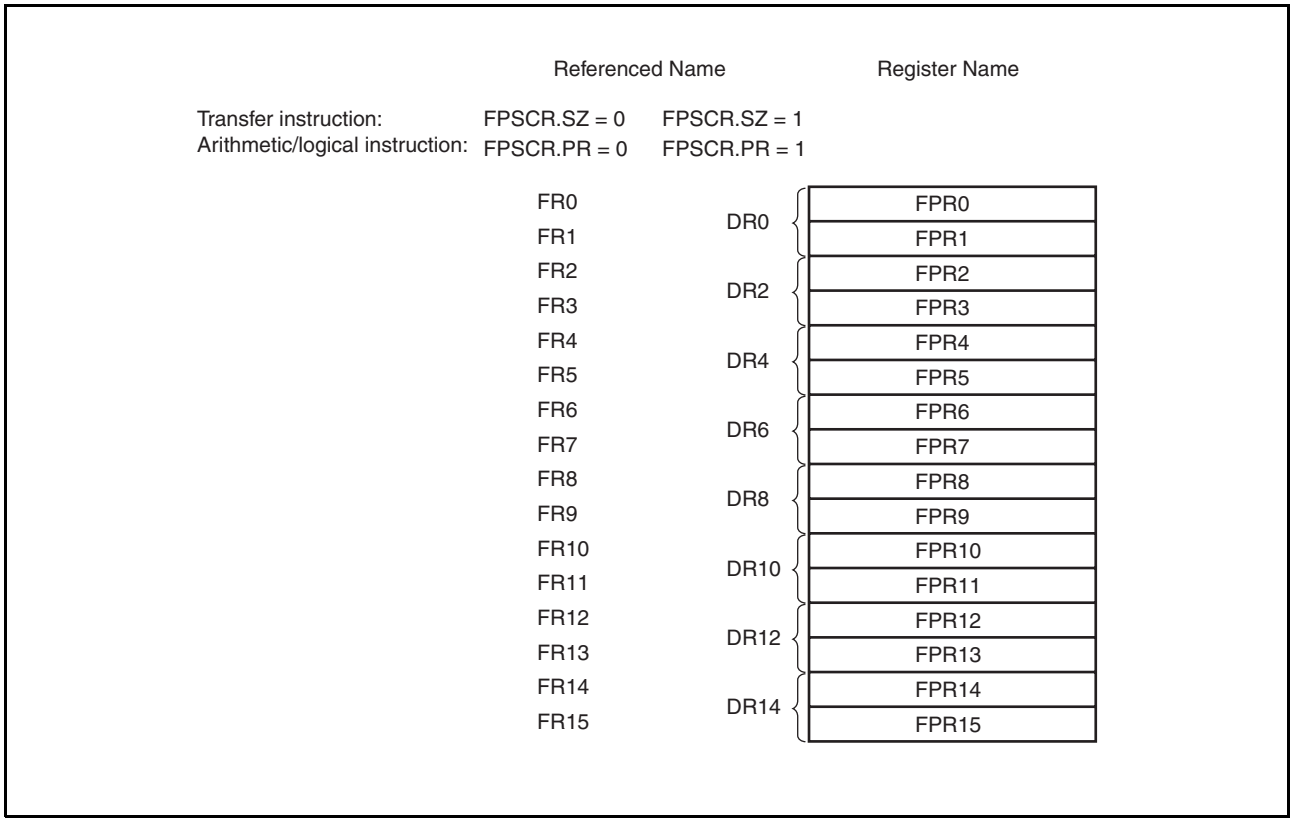


Figure 2.5 Floating-Point Registers

(1) Floating-Point Registers (FPRn: 16 registers)
FPR0, FPR1, FPR2, FPR3, FPR4, FPR5, FPR6, FPR7,
FPR8, FPR9, FPR10, FPR11, FPR12, FPR13, FPR14, and FPR15

(2) Single-Precision Floating-Point Registers (FRi: 16 registers)
FR0 to FR15 are allocated to FPR0 to FPR15.

(3) Double-Precision Floating-Point Registers or Single-Precision Floating-Point Register Pairs (DRi: 8 registers)
A DR register is composed of two FR registers.
DR0 = {FPR0, FPR1}, DR2 = {FPR2, FPR3},
DR4 = {FPR4, FPR5}, DR6 = {FPR6, FPR7},
DR8 = {FPR8, FPR9}, DR10 = {FPR10, FPR11},
DR12 = {FPR12, FPR13}, DR14 = {FPR14, FPR15}

Note: Programming Note
The values of FPR0 to FPR15 are undefined after a reset.

2.2.5 Floating-Point System Registers

(1) Floating-Point Communication Register (FPUL)

Data is transferred between an FPU register and a CPU register via FPUL.

The value is undefined after a reset.

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

The value is H'0004 0001 after a reset.

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	QIS	—	SZ	PR	DN	Cause [5:4]	
After Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	Cause [3:0]				Enable [4:0]					Flag [4:0]					RM [1:0]	
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
b31 to b23	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b22	QIS	QIS Bit	sNaN is treated as qNaN or ±∞. Valid only when the V bit in the FPU exception enable field (Enable) is set to 1. 0: Processed as qNaN or ±∞ 1: Exception generated (processed same as sNaN)	R/W
b21	—	Reserved	This bit is always read as 0. The write value should always be 0.	R
b20	SZ	SZ Bit	Transfer Size Mode 0: Sets the size of an FMOV instruction to 32 bits. 1: Sets the size of an FMOV instruction to 32-bit pair (64 bits).	R/W
b19	PR	PR Bit	Precision Mode 0: Executes floating-point instructions in single precision. 1: Executes floating-point instructions in double precision (the result of an instruction with no support for double-precision is undefined).	R/W
b18	DN	DN Bit	Denormalization Mode This bit is always set to 1. 1: A denormalized number is treated as zero.	R/W
b17 to b12	Cause [5:0]	FPU exception cause field	When an FPU operation instruction is first executed, the FPU exception cause field is set to 0; when an FPU exception next occurs, the corresponding bit in the FPU exception cause field and FPU exception flag field is set to 1. The FPU exception flag field retains the status of an exception generated after that field was last cleared. For bit allocation for each field, see table 2.1.	R/W
b11 to b7	Enable [4:0]	FPU exception enable field		R/W
b6 to b2	Flag [4:0]	FPU exception flag field		R/W
b1, b0	RM [1:0]	RM Bit	Round Mode b1 b0 0 0 : Round to nearest 0 1 : Round to zero 1 0 : Reserved 1 1 : Reserved	R/W

Table 2.1 Bit Allocation for FPU Exception Handling

		FPU Error (E)	Invalid Operation (V)	Division by 0 (Z)	Overflow (O)	Underflow (U)	Incorrect (I)
Cause	FPU exception cause field	Bit 17	Bit 16	Bit 15	Bit 14	Bit 13	Bit 12
Enable	FPU exception enable field	None	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7
Flag	FPU exception flag field	None	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2

2.2.6 Register Bank

Using a register bank, high-speed register saving and restoration can be achieved for the 19 32-bit registers: general registers R0 to R14, control register GBR, and system registers MACH, MACL, and PR. The register contents are automatically saved in the bank after the CPU accepts an interrupt that uses the bank. Restoration from the bank is executed by a RESBANK instruction issued in an interrupt processing routine.
For details, refer to the SH-2A, SH2A-FPU Software Manual.

2.2.7 Initial Values of Registers

Table 2.2 lists the initial values of the registers (values after a reset).

Table 2.2 Initial Values of Registers (Values after a Reset)

Classification	Register	Value after a Reset
General registers	R0 to R14	Undefined
	R15 (SP)	Value of the stack pointer in the vector table
Control registers	SR	Bits IMASK are 1111 (H'F), BO and CS are 0, reserved bits are 0, and others are undefined
	GBR, TBR	Undefined
	VBR	H'0000 0000
System registers	MACH, MACL, PR	Undefined
	PC	Value of the program counter in the vector table
Floating-point registers	FPR0 to FPR15	Undefined
Floating-point system registers	FPUL	Undefined
	FPSCR	H'0004 0001

2.3 Data Formats

2.3.1 Data Format in Registers

Register operands are always longwords (32 bits). If the size of a memory operand is a byte (8 bits) or a word (16 bits), it is changed into a longword through sign extension or zero extension when loaded into a register. Figure 2.6 shows the data format in registers.

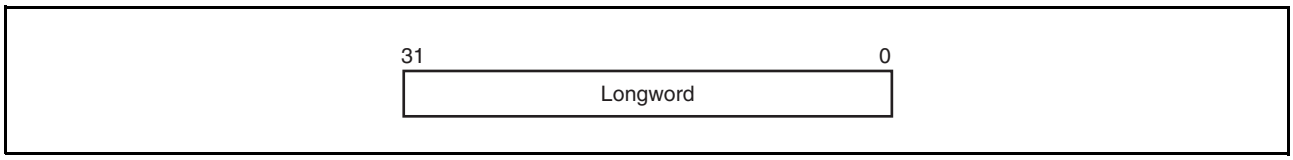


Figure 2.6 Data Format in Registers

2.3.2 Data Formats in Memory

Memory data formats are classified into bytes, words, and longwords. Memory can be accessed in 8-bit bytes, 16-bit words, or 32-bit longwords. A memory operand of fewer than 32 bits is stored in a register in sign-extended or zero-extended form.

A word operand should be accessed at a word boundary (an even address of multiple of two bytes: address 2n), and a longword operand at a longword boundary (an even address of multiple of four bytes: address 4n). Otherwise, an address error will occur. A byte operand can be accessed at any address.

Only big-endian byte order can be selected for the data format.

Data formats in memory are shown in figure 2.7

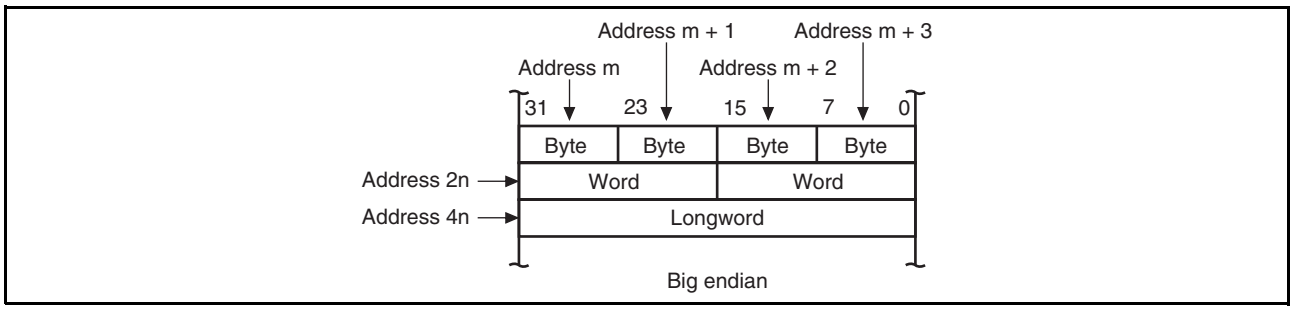


Figure 2.7 Data Formats in Memory

2.3.3 Immediate Data Format

Byte (8-bit) immediate data is located in an instruction code.

Immediate data accessed by the MOV, ADD, and CMP/EQ instructions is sign-extended and handled in registers as longword data. Immediate data accessed by the TST, AND, OR, and XOR instructions is zero-extended and handled as longword data. Consequently, AND instructions with immediate data always clear the upper 24 bits of the destination register.

20-bit immediate data is located in the code of a MOVI20 or MOVI20S 32-bit transfer instruction. The MOVI20 instruction stores immediate data in the destination register in sign-extended form. The MOVI20S instruction shifts immediate data by eight bits in the upper direction, and stores it in the destination register in sign-extended form.

Word or longword immediate data is not located in the instruction code, but rather is stored in a memory table. The memory table is accessed by an immediate data transfer instruction (MOV) using the PC relative addressing mode with displacement.

See examples given in section section 2.4.1, RISC-Type Instruction Set, (10) Immediate Data.

2.4 Instruction Features

2.4.1 RISC-Type Instruction Set

The CPU has a RISC-type instruction set, which features following functions.

(1) 16-Bit Fixed-Length Instructions

Basic instructions have a fixed length of 16 bits, improving program code efficiency.

(2) 32-Bit Fixed-Length Instructions

The SH2A-FPU additionally features 32-bit fixed-length instructions, improving performance and ease of use.

(3) One Instruction per Cycle

Each basic instruction can be executed in one cycle using the pipeline system.

(4) Data Length

The standard data length for all operations is a longword. Memory can be accessed in bytes, words, or longwords. Byte or word data in memory is sign-extended and handled as longword data. Immediate data is sign-extended for arithmetic operations or zero-extended for logic operations. It is also handled as longword data.

Table 2.3 shows sign extension of word data.

Table 2.3 Sign Extension of Word Data

SH2A-FPU CPU	Description	Example of Other CPU
MOV.W @(disp, PC), R1 ADD R1, R0 ..... .DATA.W H'1234	Data is sign-extended to 32 bits, and R1 becomes H'0000 1234. It is next operated upon by an ADD instruction.	ADD.W #H'1234,R0

Note: @(disp, PC) accesses the immediate data.

(5) Load-Store Architecture

Basic operations are executed between registers. For operations that involve memory access, data is loaded to the registers and executed (load-store architecture). Instructions such as AND that manipulate bits, however, are executed directly in memory.

(6) Delayed Branch Instructions

With the exception of some instructions, unconditional branch instructions, etc., are executed as delayed branch instructions. With a delayed branch instruction, the branch is taken after execution of the instruction immediately following the delayed branch instruction. This reduces disturbance of the pipeline control when a branch is taken.

In a delayed branch, the actual branch operation occurs after execution of the slot instruction. However, instruction execution such as register updating excluding the actual branch operation, is performed in the order of delayed branch instruction → delay slot instruction. For example, even though the contents of the register holding the branch destination address are changed in the delay slot, the branch destination address remains as the register contents prior to the change.

Table 2.4 shows delayed branch instructions.

Table 2.4 Delayed Branch Instructions

SH2A-FPU CPU	Description	Example of Other CPU
BRA TRGET ADD R1, R0	Executes the ADD before branching to TRGET.	ADD.W R1,R0 BRA TRGET

(7) Addition of Unconditional Branch Instructions with No Delay Slot

The SH2A-FPU additionally features unconditional branch instructions in which a delay slot instruction is not executed. This eliminates unnecessary NOP instructions, and so reduces the code size.

(8) Multiply/Multiply-and-Accumulate Operations

16-bit × 16-bit → 32-bit multiply operations are executed in one to two cycles. 16-bit × 16-bit + 64-bit → 64-bit multiply-and-accumulate operations are executed in two to three cycles. 32-bit × 32-bit → 64-bit multiply and 32-bit × 32-bit + 64-bit → 64-bit multiply-and-accumulate operations are executed in two to four cycles.

(9) T Bit

The T bit in the status register (SR) changes according to the result of the comparison. Whether a conditional branch is taken or not taken depends upon the T bit condition (true/false). The number of instructions that change the T bit is kept to a minimum to improve the processing speed.

Table 2.5 shows the description of the T bit.

Table 2.5 T Bit

SH2A-FPU CPU		Description	Example of Other CPU	
CMP/GE BT BF	R1, R0 TRGET0 TRGET1	T bit is set when R0 ≥ R1. The program branches to TRGET0 when R0 ≥ R1 and to TRGET1 when R0 < R1.	CMP.W BGE BLT	R1,R0 TRGET0 TRGET1
ADD CMP/EQ BT	# -1, R0 #0, R0 TRGET	T bit is not changed by ADD. T bit is set when R0 = 0. The program branches if R0 = 0.	SUB.W BEQ	#1,R0 TRGET

(10) Immediate Data

Byte immediate data is located in an instruction code. Word or longword immediate data is not located in instruction codes but in a memory table. The memory table is accessed by an immediate data transfer instruction (MOV) using the PC relative addressing mode with displacement.

With the SH2A-FPU, 17- to 28-bit immediate data can be located in an instruction code. However, for 21- to 28-bit immediate data, an OR instruction must be executed after the data is transferred to a register.

Table 2.6 shows immediate data accessing.

Table 2.6 Immediate Data Accessing

Classification	SH2A-FPU CPU		Example of Other CPU	
8-bit immediate	MOV	#H'12,R0	MOV.B	#H'12,R0
16-bit immediate	MOVI20	#H'1234,R0	MOV.W	#H'1234, R0
20-bit immediate	MOVI20	#H'12345,R0	MOV.L	#H'12345, R0
28-bit immediate	MOVI20S OR	#H'12345,R0 #H'67,R0	MOV.L	#H'1234567,R0
32-bit immediate	MOV.L ..... .DATA.L	@(disp,PC),R0 H'1234 5678	MOV.L	#H'12345678,R0

Note: @(disp, PC) accesses the immediate data.

(11) Absolute Address

When data is accessed by an absolute address, the absolute address value should be placed in the memory table in advance. That value is transferred to the register by loading the immediate data during the execution of the instruction, and the data is accessed in register indirect addressing mode.

With the SH2A-FPU, when data is referenced using an absolute address not exceeding 28 bits, it is also possible to transfer immediate data located in the instruction code to a register and to reference the data in register indirect addressing mode. However, when referencing data using an absolute address of 21 to 28 bits, an OR instruction must be used after the data is transferred to a register.

Table 2.7 shows absolute address accessing.

Table 2.7 Absolute Address Accessing

Classification	SH2A-FPU CPU	Example of Other CPU
Up to 20 bits	MOVI20 #H'12345,R1 MOV.B @R1,R0	MOV.B @H'12345,R0
21 to 28 bits	MOVI20S #H'12345,R1 OR #H'67,R1 MOV.B @R1,R0	MOV.B @H'1234567,R0
29 bits or more	MOV.L @(disp,PC), R1 MOV.B @R1,R0 ..... .DATA.L H'1234 5678	MOV.B @H'12345678,R0

(12) 16-Bit/32-Bit Displacement

When data is accessed by 16-bit or 32-bit displacement, the displacement value should be placed in the memory table in advance. That value is transferred to the register by loading the immediate data during the execution of the instruction, and the data is accessed in the indexed register indirect addressing mode.

Table 2.8 shows displacement accessing.

Table 2.8 Displacement Accessing

Classification	SH2A-FPU CPU	Example of Other CPU
16-bit displacement	MOV.W @(disp,PC),R0 MOV.W @(R0,R1),R2 ..... .DATA.W H'1234	MOV.W @(H'1234,R1),R2

2.4.2 Addressing Modes

The addressing modes and effective address calculation methods are listed in tables 2.9 to 2.12.

Table 2.9 Addressing Modes and Effective Addresses (1)

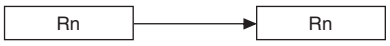
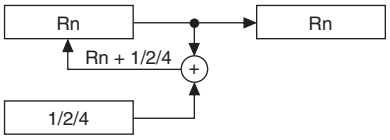
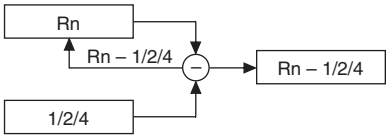
Addressing Mode	Instruction Format	Effective Address Calculation	Equation
Register direct	Rn	The effective address is register Rn. (The operand is the contents of register Rn.)	—
Register indirect	@Rn	The effective address is the contents of register Rn. 	Rn
Register indirect with post-increment	@Rn+	The effective address is the contents of register Rn.A constant is added to the contents of Rn after the instruction is executed. 1 is added for a byte operation, 2 for a word operation, and 4 for a longword operation. 	Rn (After instruction execution) Byte: Rn + 1 → Rn Word: Rn + 2 → Rn Longword: Rn + 4 → Rn
Register indirect with pre-decrement	@-Rn	The effective address is the value obtained by subtracting a constant from register Rn. 1 is subtracted for a byte operation, 2 for a word operation, and 4 for a longword operation. 	Byte: Rn - 1 → Rn Word: Rn - 2 → Rn Longword: Rn - 4 → Rn (Instruction is executed with Rn after this calculation)

Table 2.10 Addressing Modes and Effective Addresses (2)

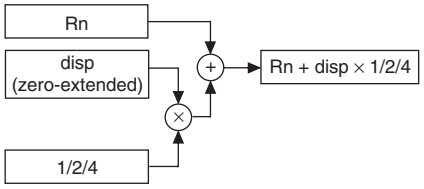
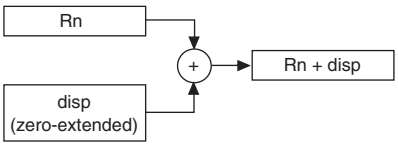
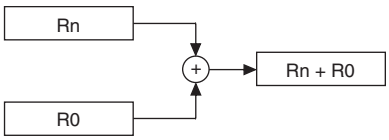
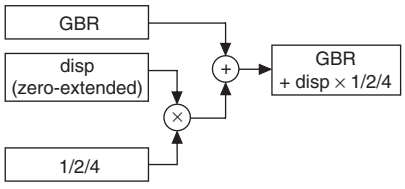
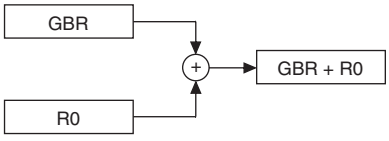
Addressing Mode	Instruction Format	Effective Address Calculation	Equation
Register indirect with displacement	@(disp:4,Rn)	The effective address is the sum of register Rn and a 4-bit displacement (disp). The value of disp is zero-extended, and remains unchanged for a byte operation, is doubled for a word operation, and is quadrupled for a longword operation. 	Byte: Rn + disp Word: Rn + disp × 2 Longword: Rn + disp × 4
	@(disp:12,Rn)	The effective address is the sum of register Rn and a 12-bit displacement (disp). The value of disp is zero-extended. 	Byte: Rn + disp Word: Rn + disp Longword: Rn + disp
Indexed register indirect	@(R0,Rn)	The effective address is the sum of register Rn and R0. 	Rn + R0
GBR indirect with displacement	@(disp:8,GBR)	The effective address is the sum of register GBR value and an 8-bit displacement (disp). The value of disp is zero-extended, and remains unchanged for a byte operation, is doubled for a word operation, and is quadrupled for a longword operation. 	Byte: GBR + disp Word: GBR + disp × 2 Longword: GBR + disp × 4
Indexed GBR indirect	@(R0,GBR)	The effective address is the sum of register GBR value and R0. 	GBR + R0

Table 2.11 Addressing Modes and Effective Addresses (3)

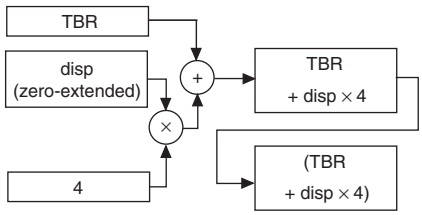
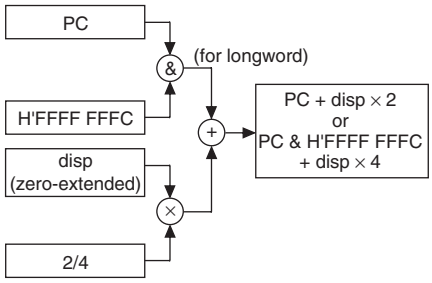
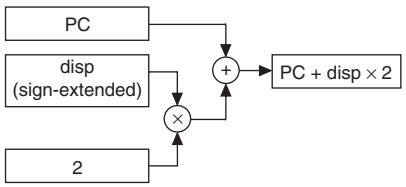
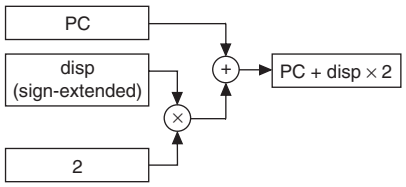
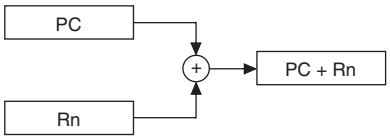
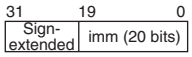
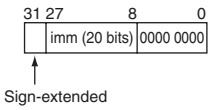
Addressing Mode	Instruction Format	Effective Address Calculation	Equation
TBR duplicate indirect with displacement	@@(disp:8,TBR)	The effective address is the sum of register TBR value and an 8-bit displacement (disp). The value of disp is zero-extended, and is multiplied by 4. 	Contents of address (TBR + disp × 4)
PC relative with displacement	@(disp:8,PC)	The effective address is the sum of register PC value and an 8-bit displacement (disp). The value of disp is zero-extended, and is doubled for a word operation, and quadrupled for a longword operation. For a longword operation, the lowest two bits of the PC value are masked. 	Word: PC + disp × 2 Longword: PC&H'FFFF FFFC + disp × 4

Table 2.12 Addressing Modes and Effective Addresses (4)

Addressing Mode	Instruction Format	Effective Address Calculation	Equation
PC relative	disp:8	The effective address is the sum of register PC value and the value that is obtained by doubling the sign-extended 8-bit displacement (disp). 	PC + disp × 2
	disp:12	The effective address is the sum of register PC value and the value that is obtained by doubling the sign-extended 12-bit displacement (disp). 	PC + disp × 2
	Rn	The effective address is the sum of register PC value and Rn. 	PC + Rn
Immediate	#imm:20	The 20-bit immediate data (imm) for the MOVI20 instruction is sign-extended. 	—
		The 20-bit immediate data (imm) for the MOVI20S instruction is shifted by eight bits to the left, the upper bits are sign-extended, and the lower bits are padded with zero. 	—
	#imm:8	The 8-bit immediate data (imm) for the TST, AND, OR, and XOR instructions is zero-extended.	—
	#imm:8	The 8-bit immediate data (imm) for the MOV, ADD, and CMP/EQ instructions is sign-extended.	—
	#imm:8	The 8-bit immediate data (imm) for the TRAPA instruction is zero-extended and then quadrupled.	—
	#imm:3	The 3-bit immediate data (imm) for the BAND, BOR, BXOR, BST, BLD, BSET, and BCLR instructions indicates the target bit location.	—

2.4.3 Instruction Format

The instruction formats and the meaning of source and destination operands are described below. The meaning of the operand depends on the instruction code.

The symbols used are as follows:

xxxx : Instruction code
mmmm : Source register
nnnn : Destination register
iiii : Immediate data
dddd : Displacement

Tables 2.13 to 2.15 show the instruction formats.

Table 2.13 Instruction Formats (1)

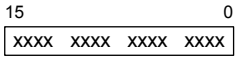
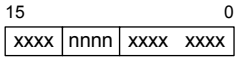
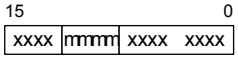
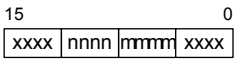
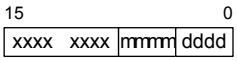
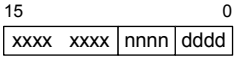
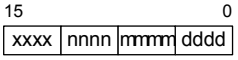
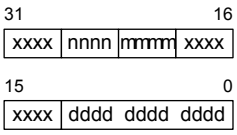
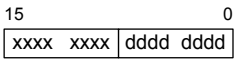
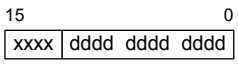
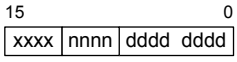
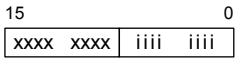
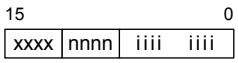
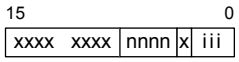
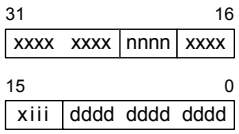
Instruction Formats		Source Operand	Destination Operand	Example
0 format		—	—	NOP
n format		—	nnnn: Register direct	MOV T Rn
		Control register or system register	nnnn: Register direct	STS MACH,Rn
		R0 (Register direct)	nnnn: Register direct	DIVU R0, Rn
		Control register or system register	nnnn: Register indirect with pre-decrement	STC.L SR,@-Rn
		mmmm: Register direct	R15 (Register indirect with pre-decrement)	MOVMU.L Rm,@-R15
		R15 (Register indirect with post-increment)	nnnn: Register direct	MOVMU.L @R15+,Rn
		R0 (Register direct)	nnnn: Register indirect with post-increment	MOV.L R0,@Rn+
m format		mmmm: Register direct	Control register or system register	LDC Rm,SR
		mmmm: Register indirect with post-increment	Control register or system register	LDC.L @Rm+,SR
		mmmm: Register indirect	—	JMP @Rm
		mmmm: Register indirect with pre-decrement	R0 (Register direct)	MOV.L @-Rm,R0
		mmmm: PC relative using Rm	—	BRAF Rm

Table 2.14 Instruction Formats (2)

Instruction Formats		Source Operand	Destination Operand	Example	
nm format		mmmm: Register direct	nnnn: Register direct	ADD	Rm,Rn
		mmmm: Register direct	nnnn: Register indirect	MOV.L	Rm,@Rn
		mmmm: Register indirect with post-increment (multiply-and-accumulate) nnnn*: Register indirect with post-increment (multiply-and-accumulate)	MACH , MACL	MAC.W	@Rm+,@Rn+
		mmmm: Register indirect with post-increment	nnnn: Register direct	MOV.L	@Rm+,Rn
		mmmm: Register direct	nnnn: Register indirect with pre-decrement	MOV.L	Rm,@-Rn
		mmmm: Register direct	nnnn: Indexed register indirect	MOV.L	Rm,@(R0,Rn)
md format		mmmmdddd: Register indirect with displacement	R0 (Register direct)	MOV.B	@(disp, Rm),R0
nd4 format		R0 (Register direct)	nnnndddd: Register indirect with displacement	MOV.B	R0,@(disp,Rn)
nmd format		mmmm: Register direct	nnnndddd: Register indirect with displacement	MOV.L	Rm,@(disp,Rn)
		mmmmdddd: Register indirect with displacement	nnnn: Register direct	MOV.L	@(disp,Rm),Rn
nmd12 format		mmmm: Register direct	nnnndddd: Register indirect with displacement	MOV.L	Rm ,@(disp12,Rn)
		mmmmdddd: Register indirect with displacement	nnnn: Register direct	MOV.L	@(disp12,Rm),Rn
d format		dddddddd: GBR indirect with displacement	R0 (Register direct)	MOV.L	@(disp,GBR),R0
		R0 (Register direct)	dddddddd: GBR indirect with displacement	MOV.L	R0,@(disp,GBR)
		dddddddd: PC relative with displacement	R0 (Register direct)	MOVA	@(disp,PC),R0
		dddddddd: TBR duplicate indirect with displacement	—	JSR/N	@@(disp8,TBR)
		dddddddd: PC relative	—	BF	label

Note: * In multiply-and-accumulate instructions, nnnn is the source register.

Table 2.15 Instruction Formats (3)

Instruction Formats		Source Operand	Destination Operand	Example
d12 format		ddddddddddd: PC relative	—	BRA label (label = disp + PC)
nd8 format		dddddddd: PC relative with displacement	nnnn: Register direct	MOV.L @(disp,PC),Rn
i format		iiiiiiii: Immediate	Indexed GBR indirect	AND.B #imm,@(R0,GBR)
		iiiiiiii: Immediate	R0 (Register direct)	AND #imm,R0
		iiiiiiii: Immediate	—	TRAPA #imm
ni format		iiiiiiii: Immediate	nnnn: Register direct	ADD #imm,Rn
ni3 format		nnnn: Register direct iii: Immediate	—	BLD #imm3,Rn
		—	nnnn: Register direct iii: Immediate	BST #imm3,Rn
ni20 format		iiiiiiiiiiiiiiii: Immediate	nnnn: Register direct	MOVI20 #imm20,Rn
nid format		nnnnddddddddddd: Register indirect with displacement iii: Immediate	—	BLD.B #imm3,@(disp12,Rn)
		—	nnnnddddddddddd: Register indirect with displacement iii: Immediate	BST.B #imm3,@ (disp12,Rn)

2.5 Instruction Set

2.5.1 Instruction Set by Classification

Tables 2.16 to 2.19 list classification of instructions.

Table 2.16 Classification of Instructions (1)

Classification	Types	Operation Code	Description	No. of Instructions
Data transfer	13	MOV	Data transfer Immediate data transfer Peripheral module data transfer Structure data transfer Reverse stack transfer	62
		MOVA	Effective address transfer	
		MOVI20	20-bit immediate data transfer	
		MOVI20S	20-bit immediate data transfer 8-bit left-shift	
		MOVML	R0-Rn register save/restore	
		MOVMU	Rn-R14 and PR register save/restore	
		MOVRT	T bit inversion and transfer to Rn	
		MOVT	T bit transfer	
		MOVU	Unsigned data transfer	
		NOTT	T bit inversion	
		PREF	Prefetch to operand cache	
		SWAP	Swap of upper and lower bytes	
		XTRCT	Extraction of the middle of registers connected	
Arithmetic operations	26	ADD	Binary addition	40
		ADDC	Binary addition with carry	
		ADDV	Binary addition with overflow check	
		CMP/cond	Comparison	
		CLIPS	Signed saturation value comparison	
		CLIPU	Unsigned saturation value comparison	
		DIVS	Signed division (32 ÷ 32)	
		DIVU	Unsigned division (32 ÷ 32)	
		DIV1	One-step division	
		DIV0S	Initialization of signed one-step division	
		DIV0U	Initialization of unsigned one-step division	
		DMULS	Signed double-precision multiplication	
		DMULU	Unsigned double-precision multiplication	
		DT	Decrement and test	
		EXTS	Sign extension	
		EXTU	Zero extension	
		MAC	Multiply-and-accumulate, double-precision multiply-and-accumulate operation	
		MUL	Double-precision multiply operation	
		MULR	Signed multiplication with result storage in Rn	
		MULS	Signed multiplication	
		MULU	Unsigned multiplication	

Table 2.17 Classification of Instructions (2)

Classification	Types	Operation Code	Description	No. of Instructions
Arithmetic operations	26	NEG	Negation	40
		NEGC	Negation with borrow	
		SUB	Binary subtraction	
		SUBC	Binary subtraction with borrow	
		SUBV	Binary subtraction with underflow	
Logic operations	6	AND	Logical AND	14
		NOT	Bit inversion	
		OR	Logical OR	
		TAS	Memory test and bit set	
		TST	Logical AND and T bit set	
		XOR	Exclusive OR	
Shift	12	ROTL	One-bit left rotation	16
		ROTR	One-bit right rotation	
		ROTCL	One-bit left rotation with T bit	
		ROTCR	One-bit right rotation with T bit	
		SHAD	Dynamic arithmetic shift	
		SHAL	One-bit arithmetic left shift	
		SHAR	One-bit arithmetic right shift	
		SHLD	Dynamic logical shift	
		SHLL	One-bit logical left shift	
		SHLLn	n-bit logical left shift	
		SHLR	One-bit logical right shift	
		SHLRn	n-bit logical right shift	
Branch	10	BF	Conditional branch, conditional delayed branch (branch when T = 0)	15
		BT	Conditional branch, conditional delayed branch (branch when T = 1)	
		BRA	Unconditional delayed branch	
		BRAF	Unconditional delayed branch	
		BSR	Delayed branch to subroutine procedure	
		BSRF	Delayed branch to subroutine procedure	
		JMP	Unconditional delayed branch	
		JSR	Branch to subroutine procedure Delayed branch to subroutine procedure	
		RTS	Return from subroutine procedure Delayed return from subroutine procedure	
		RTV/N	Return from subroutine procedure with Rm → R0 transfer	

Table 2.18 Classification of Instructions (3)

Classification	Types	Operation Code	Description	No. of Instructions
System control	14	CLRT	T bit clear	36
		CLRMAC	MAC register clear	
		LDBANK	Register restoration from specified register bank entry	
		LDC	Load to control register	
		LDS	Load to system register	
		NOP	No operation	
		RESBANK	Register restoration from register bank	
		RTE	Return from exception handling	
		SETT	T bit set	
		SLEEP	Transition to power-down mode	
		STBANK	Register save to specified register bank entry	
		STC	Store control register data	
		STS	Store system register data	
		TRAPA	Trap exception handling	
Floating-point instructions	19	FABS	Floating-point absolute value	48
		FADD	Floating-point addition	
		FCMP	Floating-point comparison	
		FCNVDS	Conversion from double-precision to single-precision	
		FCNVSD	Conversion from single-precision to double-precision	
		FDIV	Floating-point division	
		FLDI0	Floating-point load immediate 0	
		FLDI1	Floating-point load immediate 1	
		FLDS	Floating-point load into system register FPUL	
		FLOAT	Conversion from integer to floating-point	
		FMAC	Floating-point multiply-and-accumulate operation	
		FMOV	Floating-point data transfer	
		FMUL	Floating-point multiplication	
		FNEG	Floating-point sign inversion	
		FSCHG	SZ bit inversion	
		FSQRT	Floating-point square root	
		FSTS	Floating-point store from system register FPUL	
		FSUB	Floating-point subtraction	
		FTRC	Floating-point conversion with rounding to integer	
FPU-related CPU instructions	2	LDS	Load into floating-point system register	8
		STS	Store from floating-point system register	

Table 2.19 Classification of Instructions (4)

Classification	Types	Operation Code	Description	No. of Instructions
Bit manipulation	10	BAND	Bit AND	14
		BCLR	Bit clear	
		BLD	Bit load	
		BOR	Bit OR	
		BSET	Bit set	
		BST	Bit store	
		BXOR	Bit exclusive OR	
		BANDNOT	Bit NOT AND	
		BORNOT	Bit NOT OR	
		BLDNOT	Bit NOT load	
	Total: 112			253

The table below shows the format of instruction codes, operation, and execution states. They are described by using this format according to their classification.

Instruction	Instruction Code	Operation	Execution Cycles	T Bit
Indicated by mnemonic. [Legend] OP.Sz SRC, DEST OP : Operation code Sz : Size SRC : Source DEST : Destination Rm : Source register Rn : Destination register imm : Immediate data disp : Displacement*2	Indicated in MSB ↔ LSB order. [Legend] mmmm : Source register nnnn : Destination register 0000 : R0 0001 : R1 ..... 1111 : R15 iiii : Immediate data dddd : Displacement	Indicates summary of operation. [Legend] →, ← : Transfer direction (xx) : Memory operand M/Q/T : Flag bits in SR & : Logical AND of each bit | : Logical OR of each bit ^ : Exclusive logical OR of each bit ~ : Logical NOT of each bit <<n : n-bit left shift >>n : n-bit right shift	Value when no wait states are inserted.*1	Value of T bit after instruction is executed. [Legend] — : No change

Notes: 1. Instruction execution cycles: The execution cycles shown in the table are minimums. In practice, the number of instruction execution states will be increased in cases such as the following:
a.When there is a conflict between an instruction fetch and a data access
b.When the destination register of a load instruction (memory → register) is the same as the register used by the next instruction.
2. Depending on the operand size, displacement is scaled by 1, 2, or 4. For details, refer to the SH-2A, SH2A-FPU Software Manual.

2.5.2 Data Transfer Instructions

Tables 2.20 to 2.22 show data transfer instructions.

Table 2.20 Data Transfer Instructions (1)

Instruction		Instruction Code	Operation	Execution Cycles	T Bit	Compatibility		
						SH2E	SH4	SH-2A/ SH2A-FPU
MOV	#imm, Rn	1110nnnniiiiiiii	imm → sign extension → Rn	1	—	Yes	Yes	
MOV.W	@(disp, PC), Rn	1001nnnndddddddd	(disp × 2 + PC) → sign extension → Rn	1	—	Yes	Yes	
MOV.L	@(disp, PC), Rn	1101nnnndddddddd	(disp × 4 + PC) → Rn	1	—	Yes	Yes	
MOV	Rm, Rn	0110nnnnmmmm0011	Rm → Rn	1	—	Yes	Yes	
MOV.B	Rm, @Rn	0010nnnnmmmm0000	Rm → (Rn)	1	—	Yes	Yes	
MOV.W	Rm, @Rn	0010nnnnmmmm0001	Rm → (Rn)	1	—	Yes	Yes	
MOV.L	Rm, @Rn	0010nnnnmmmm0010	Rm → (Rn)	1	—	Yes	Yes	
MOV.B	@Rm, Rn	0110nnnnmmmm0000	(Rm) → sign extension → Rn	1	—	Yes	Yes	
MOV.W	@Rm, Rn	0110nnnnmmmm0001	(Rm) → sign extension → Rn	1	—	Yes	Yes	
MOV.L	@Rm, Rn	0110nnnnmmmm0010	(Rm) → Rn	1	—	Yes	Yes	
MOV.B	Rm, @-Rn	0010nnnnmmmm0100	Rn - 1 → Rn, Rm → (Rn)	1	—	Yes	Yes	
MOV.W	Rm, @-Rn	0010nnnnmmmm0101	Rn - 2 → Rn, Rm → (Rn)	1	—	Yes	Yes	
MOV.L	Rm, @-Rn	0010nnnnmmmm0110	Rn - 4 → Rn, Rm → (Rn)	1	—	Yes	Yes	
MOV.B	@Rm+, Rn	0110nnnnmmmm0100	(Rm) → sign extension → Rn, Rm + 1 → Rm	1	—	Yes	Yes	
MOV.W	@Rm+, Rn	0110nnnnmmmm0101	(Rm) → sign extension → Rn, Rm + 2 → Rm	1	—	Yes	Yes	
MOV.L	@Rm+, Rn	0110nnnnmmmm0110	(Rm) → Rn, Rm + 4 → Rm	1	—	Yes	Yes	
MOV.B	R0, @(disp, Rn)	10000000nnnndddd	R0 → (disp + Rn)	1	—	Yes	Yes	
MOV.W	R0, @(disp, Rn)	10000001nnnndddd	R0 → (disp × 2 + Rn)	1	—	Yes	Yes	
MOV.L	Rm, @(disp, Rn)	0001nnnnmmmmdddd	Rm → (disp × 4 + Rn)	1	—	Yes	Yes	
MOV.B	@(disp, Rm), R0	10000100mmmmdddd	(disp + Rm) → sign extension → R0	1	—	Yes	Yes	
MOV.W	@(disp, Rm), R0	10000101mmmmdddd	(disp × 2 + Rm) → sign extension → R0	1	—	Yes	Yes	
MOV.L	@(disp, Rm), Rn	0101nnnnmmmmdddd	(disp × 4 + Rm) → Rn	1	—	Yes	Yes	
MOV.B	Rm, @(R0, Rn)	0000nnnnmmmm0100	Rm → (R0 + Rn)	1	—	Yes	Yes	
MOV.W	Rm, @(R0, Rn)	0000nnnnmmmm0101	Rm → (R0 + Rn)	1	—	Yes	Yes	
MOV.L	Rm, @(R0, Rn)	0000nnnnmmmm0110	Rm → (R0 + Rn)	1	—	Yes	Yes	
MOV.B	@(R0, Rm), Rn	0000nnnnmmmm1100	(R0 + Rm) → sign extension → Rn	1	—	Yes	Yes	
MOV.W	@(R0, Rm), Rn	0000nnnnmmmm1101	(R0 + Rm) → sign extension → Rn	1	—	Yes	Yes	
MOV.L	@(R0, Rm), Rn	0000nnnnmmmm1110	(R0 + Rm) → Rn	1	—	Yes	Yes	
MOV.B	R0, @(disp, GBR)	11000000dddddddd	R0 → (disp + GBR)	1	—	Yes	Yes	
MOV.W	R0, @(disp, GBR)	11000001dddddddd	R0 → (disp × 2 + GBR)	1	—	Yes	Yes	
MOV.L	R0, @(disp, GBR)	11000010dddddddd	R0 → (disp × 4 + GBR)	1	—	Yes	Yes	
MOV.B	@(disp, GBR), R0	11000100dddddddd	(disp + GBR) → sign extension → R0	1	—	Yes	Yes	
MOV.W	@(disp, GBR), R0	11000101dddddddd	(disp × 2 + GBR) → sign extension → R0	1	—	Yes	Yes	

Table 2.21 Data Transfer Instructions (2)

Instruction	Instruction Code	Operation	Execution Cycles	T Bit	Compatibility		
					SH2E	SH4	SH-2A/ SH2A-FPU
MOV.L @(disp, GBR), R0	11000110dddddddd	(disp × 4 + GBR) → R0	1	—	Yes	Yes	
MOV.B R0, @Rn+	0100nnnn10001011	R0 → (Rn), Rn + 1 → Rn	1	—			Yes
MOV.W R0, @Rn+	0100nnnn10011011	R0 → (Rn), Rn + 2 → Rn	1	—			Yes
MOV.L R0, @Rn+	0100nnnn10101011	R0 → (Rn), Rn + 4 → Rn	1	—			Yes
MOV.B @-Rm, R0	0100mmmm11001011	Rm - 1 → Rm, (Rm) → sign extension → R0	1	—			Yes
MOV.W @-Rm, R0	0100mmmm11011011	Rm - 2 → Rm, (Rm) → sign extension → R0	1	—			Yes
MOV.L @-Rm, R0	0100mmmm11101011	Rm - 4 → Rm, (Rm) → R0	1	—			Yes
MOV.B Rm, disp12, Rn)	0011nnnnmmmm0001 0000dddddddddddd	Rm → (disp + Rn)	1	—			Yes
MOV.W Rm, @(disp12, Rn)	0011nnnnmmmm0001 0001dddddddddddd	Rm → (disp × 2 + Rn)	1	—			Yes
MOV.L Rm, @(disp12, Rn)	0011nnnnmmmm0001 0010dddddddddddd	Rm → (disp × 4 + Rn)	1	—			Yes
MOV.B @(disp12, Rm), Rn	0011nnnnmmmm0001 0100dddddddddddd	(disp + Rm) → sign extension → Rn	1	—			Yes
MOV.W @(disp12, Rm), Rn	0011nnnnmmmm0001 0101dddddddddddd	(disp × 2 + Rm) → sign extension → Rn	1	—			Yes
MOV.L @(disp12, Rm), Rn	0011nnnnmmmm0001 0110dddddddddddd	(disp × 4 + Rm) → Rn	1	—			Yes
MOVA @(disp, PC), R0	11000111dddddddd	disp × 4 + PC → R0	1	—	Yes	Yes	
MOVI20 #imm20, Rn	0000nnnniiii0000 iiiiiiiiiiiiiiii	imm → sign extension → Rn	1	—			Yes
MOVI20S #imm20, Rn	0000nnnniiii0001 iiiiiiiiiiiiiiii	imm << 8 → sign extension → Rn	1	—			Yes
MOVML.L Rm, @-R15	0100mmmm11110001	R15 – 4 → R15, Rm → (R15) R15 – 4 → R15, Rm – 1 → (R15) : R15 – 4 → R15 , R0 → (R15) Note: When Rm = R15, read Rm as PR	1 to 16	—			Yes
MOVML.L @R15+, Rn	0100nnnn11110101	(R15) → R0, R15 + 4 → R15 (R15) → R1, R15 + 4 → R15 : (R15) → Rn Note: When Rn = R15, read Rm as PR	1 to 16	—			Yes
MOVMU.L Rm, @-R15	0100mmmm11110000	R15 – 4 → R15, PR → (R15) R15 – 4 → R15, R14 → (R15) : R15 – 4 → R15, Rm → (R15) Note: When Rm = R15, read Rm as PR	1 to 16	—			Yes

Table 2.22 Data Transfer Instructions (3)

Instruction	Instruction Code	Operation	Execution Cycles	T Bit	Compatibility		
					SH2E	SH4	SH-2A/ SH2A-FPU
MOVMU.L @R15+, Rn	0100nnnn11110100	(R15) → Rn , R15 + 4 → R15 (R15) → Rn + 1, R15 + 4 → R15 : (R15) → R14, R15 + 4 → R15 (R15) → PR Note: When Rn = R15, read Rm as PR	1 to 16	—			Yes
MOVRT Rn	0000nnnn00111001	~ T → Rn	1	—			Yes
MOVT Rn	0000nnnn00101001	T → Rn	1	—	Yes	Yes	
MOVU.B @(disp12, Rm), Rn	0011nnnnmmmm0001 1000dddddddddddd	(disp + Rm) → zero extension → Rn	1	—			Yes
MOVU.W @(disp12, Rm), Rn	0011nnnnmmmm0001 1001dddddddddddd	(disp × 2 + Rm) → zero extension → Rn	1	—			Yes
NOTT	0000000001101000	~ T → T	1	Operation result			Yes
PREF @Rn	0000nnnn10000011	(Rn) → operand cache	1	—		Yes	
SWAP.B Rm, Rn	0110nnnnmmmm1000	Rm → swap lower 2 bytes → Rn	1	—	Yes	Yes	
SWAP.W Rm, Rn	0110nnnnmmmm1001	Rm → swap upper and lower words → Rn	1	—	Yes	Yes	
XTRCT Rm, Rn	0010nnnnmmmm1101	Middle 32 bits of Rm: Rn → Rn	1	—	Yes	Yes	

2.5.3 Arithmetic Operation Instructions

Tables 2.23 to 2.25 show arithmetic operation instructions.

Table 2.23 Arithmetic Operation Instructions (1)

Instruction		Instruction Code	Operation	Execution Cycles	T Bit	Compatibility		
						SH2E	SH4	SH-2A/ SH2A-FPU
ADD	Rm, Rn	0011nnnnmmmm1100	Rn + Rm → Rn	1	—	Yes	Yes	
ADD	#imm, Rn	0111nnnniiiiiiii	Rn + imm → Rn	1	—	Yes	Yes	
ADDC	Rm, Rn	0011nnnnmmmm1110	Rn + Rm + T → Rn, carry →T	1	Carry	Yes	Yes	
ADDV	Rm, Rn	0011nnnnmmmm1111	Rn + Rm → Rn, overflow→T	1	Over-flow	Yes	Yes	
CMP/EQ	#imm, R0	10001000iiiiiiii	When R0 = imm, 1 → T Otherwise, 0 → T	1	Com-parison result	Yes	Yes	
CMP/EQ	Rm, Rn	0011nnnnmmmm0000	When Rn = Rm, 1 → T Otherwise, 0 → T	1	Com-parison result	Yes	Yes	
CMP/HS	Rm, Rn	0011nnnnmmmm0010	When Rn ≥ Rm (unsigned), 1 → T Otherwise, 0 → T	1	Com-parison result	Yes	Yes	
CMP/GE	Rm, Rn	0011nnnnmmmm0011	When Rn ≥ Rm (signed), 1 → T Otherwise, 0 → T	1	Com-parison result	Yes	Yes	
CMP/HI	Rm, Rn	0011nnnnmmmm0110	When Rn > Rm (unsigned), 1 → T Otherwise, 0 → T	1	Com-parison result	Yes	Yes	
CMP/GT	Rm, Rn	0011nnnnmmmm0111	When Rn > Rm (signed), 1 → T Otherwise, 0 → T	1	Com-parison result	Yes	Yes	
CMP/PL	Rn	0100nnnn00010101	When Rn > 0, 1 → T Otherwise, 0 → T	1	Com-parison result	Yes	Yes	
CMP/PZ	Rn	0100nnnn00010001	When Rn ≥ 0, 1 → T Otherwise, 0 → T	1	Com-parison result	Yes	Yes	
CMP/STR	Rm, Rn	0010nnnnmmmm1100	When any bytes are equal, 1 → T Otherwise, 0 → T	1	Com-parison result	Yes	Yes	
CLIPS.B	Rn	0100nnnn10010001	When Rn > (H'0000 007F), (H'0000 007F) → Rn, 1 → CS When Rn < (H'FFFF FF80), (H'FFFF FF80) → Rn, 1 → CS	1	—			Yes

Table 2.24 Arithmetic Operation Instructions (2)

Instruction	Instruction Code	Operation	Execution Cycles	T Bit	Compatibility		
					SH2E	SH4	SH-2A/ SH2A-FPU
CLIPS.W Rn	0100nnnn10010101	When Rn > (H'0000 7FFF), (H'0000 7FFF) → Rn, 1 → CS When Rn < (H'FFFF 8000), (H'FFFF 8000) → Rn, 1 → CS	1	—			Yes
CLIPU.B Rn	0100nnnn10000001	When Rn > (H'0000 00FF), (H'0000 00FF) → Rn, 1 → CS	1	—			Yes
CLIPU.W Rn	0100nnnn10000101	When Rn > (H'0000 FFFF), (H'0000 FFFF) → Rn, 1 → CS	1	—			Yes
DIV1 Rm, Rn	0011nnnnmmmm0100	1-step division (Rn ÷ Rm)	1	Calculation result	Yes	Yes	
DIV0S Rm, Rn	0010nnnnmmmm0111	MSB of Rn → Q, MSB of Rm → M, M ^ Q → T	1	Calculation result	Yes	Yes	
DIV0U	0000000000011001	0 → M/Q/T	1	0	Yes	Yes	
DIVS R0, Rn	0100nnnn10010100	Signed operation of Rn ÷ R0 → Rn 32 ÷ 32 → 32 bits	36	—			Yes
DIVU R0, Rn	0100nnnn10000100	Unsigned operation of Rn ÷ R0 → Rn 32 ÷ 32 → 32 bits	34	—			Yes
DMULS.L Rm, Rn	0011nnnnmmmm1101	Signed operation of Rn × Rm → MACH, MACL 32 × 32 → 64 bits	2	—	Yes	Yes	
DMULU.L Rm, Rn	0011nnnnmmmm0101	Unsigned operation of Rn × Rm → MACH, MACL 32 × 32 → 64 bits	2	—	Yes	Yes	
DT Rn	0100nnnn00010000	Rn - 1 → Rn When Rn is 0, 1 → T When Rn is not 0, 0 → T	1	Comparison result	Yes	Yes	
EXTS.B Rm, Rn	0110nnnnmmmm1110	Byte in Rm is sign-extended → Rn	1	—	Yes	Yes	
EXTS.W Rm, Rn	0110nnnnmmmm1111	Word in Rm is sign-extended → Rn	1	—	Yes	Yes	
EXTU.B Rm, Rn	0110nnnnmmmm1100	Byte in Rm is zero-extended → Rn	1	—	Yes	Yes	
EXTU.W Rm, Rn	0110nnnnmmmm1101	Word in Rm is zero-extended → Rn	1	—	Yes	Yes	
MAC.L @Rm+, @Rn+	0000nnnnmmmm1111	Signed operation of (Rn) × (Rm) + MAC → MAC 32 × 32 + 64 → 64 bits	4	—	Yes	Yes	

Table 2.25 Arithmetic Operation Instructions (3)

Instruction		Instruction Code	Operation	Execu-tion Cycles	T Bit	Compatibility		
						SH2E	SH4	SH-2A/ SH2A-FPU
MAC.W	@Rm+, @Rn+	0100nnnnmmmm1111	Signed operation of (Rn) × (Rm) + MAC → MAC 16 × 16 + 64 → 64 bits	3	—	Yes	Yes	
MUL.L	Rm, Rn	0000nnnnmmmm0111	Rn × Rm → MACL 32 × 32 → 32 bits	2	—	Yes	Yes	
MULR	R0, Rn	0100nnnn10000000	R0 × Rn → Rn 32 × 32 → 32 bits	2	—			Yes
MULS.W	Rm, Rn	0010nnnnmmmm1111	Signed operation of Rn × Rm → MACL 16 × 16 → 32 bits	1	—	Yes	Yes	
MULU.W	Rm, Rn	0010nnnnmmmm1110	Unsigned operation of Rn × Rm → MACL 16 × 16 → 32 bits	1	—	Yes	Yes	
NEG	Rm, Rn	0110nnnnmmmm1011	0 - Rm → Rn	1	—	Yes	Yes	
NEGC	Rm, Rn	0110nnnnmmmm1010	0 - Rm - T → Rn, borrow → T	1	Borrow	Yes	Yes	
SUB	Rm, Rn	0011nnnnmmmm1000	Rn - Rm → Rn	1	—	Yes	Yes	
SUBC	Rm, Rn	0011nnnnmmmm1010	Rn - Rm - T → Rn, borrow → T	1	Borrow	Yes	Yes	
SUBV	Rm, Rn	0011nnnnmmmm1011	Rn - Rm → Rn , underflow → T	1	Under-flow	Yes	Yes	

2.5.4 Logic Operation Instructions

Table 2.26 shows logic operation instructions.

Table 2.26 Logic Operation Instructions

Instruction		Instruction Code	Operation	Execution Cycles	T Bit	Compatibility		
						SH2E	SH4	SH-2A/ SH2A-FPU
AND	Rm, Rn	0010nnnnmmmm1001	Rn & Rm → Rn	1	—	Yes	Yes	
AND	#imm, R0	11001001iiiiiiii	R0 & imm → R0	1	—	Yes	Yes	
AND.B	#imm, @(R0, GBR)	11001101iiiiiiii	(R0 + GBR) & imm → (R0 + GBR)	3	—	Yes	Yes	
NOT	Rm, Rn	0110nnnnmmmm0111	~ Rm → Rn	1	—	Yes	Yes	
OR	Rm, Rn	0010nnnnmmmm1011	Rn | Rm → Rn	1	—	Yes	Yes	
OR	#imm, R0	11001011iiiiiiii	R0 | imm → R0	1	—	Yes	Yes	
OR.B	#imm, @(R0, GBR)	11001111iiiiiiii	(R0 + GBR) | imm → (R0 + GBR)	3	—	Yes	Yes	
TAS.B	@Rn	0100nnnn00011011	When (Rn) is 0, 1 → T Otherwise, 0 → T, 1 → MSB of (Rn)	3	Test result	Yes	Yes	
TST	Rm, Rn	0010nnnnmmmm1000	Rn & Rm When the result is 0, 1 → T Otherwise, 0 → T	1	Test result	Yes	Yes	
TST	#imm, R0	11001000iiiiiiii	R0 & imm When the result is 0, 1 → T Otherwise, 0 → T	1	Test result	Yes	Yes	
TST.B	#imm, @(R0, GBR)	11001100iiiiiiii	(R0 + GBR) & imm When the result is 0, 1 → T Otherwise, 0 → T	3	Test result	Yes	Yes	
XOR	Rm, Rn	0010nnnnmmmm1010	Rn ^ Rm → Rn	1	—	Yes	Yes	
XOR	#imm, R0	11001010iiiiiiii	R0 ^ imm → R0	1	—	Yes	Yes	
XOR.B	#imm, @(R0, GBR)	11001110iiiiiiii	(R0 + GBR) ^ imm → (R0 + GBR)	3	—	Yes	Yes	

2.5.5 Shift Instructions

Table 2.27 shows shift instructions.

Table 2.27 Shift Instructions

Instruction		Instruction Code	Operation	Execution Cycles	T Bit	Compatibility		
						SH2E	SH4	SH-2A/ SH2A-FPU
ROTL	Rn	0100nnnn00000100	T ← Rn ← MSB	1	MSB	Yes	Yes	
ROTR	Rn	0100nnnn00000101	LSB → Rn → T	1	LSB	Yes	Yes	
ROTCL	Rn	0100nnnn00100100	T ← Rn ←T	1	MSB	Yes	Yes	
ROTCR	Rn	0100nnnn00100101	T → Rn → T	1	LSB	Yes	Yes	
SHAD	Rm, Rn	0100nnnnmmmm1100	When Rm ≥ 0, Rn << Rm → Rn When Rm < 0, Rn >> |Rm| → [MSB → Rn]	1	—		Yes	
SHAL	Rn	0100nnnn00100000	T ← Rn ← 0	1	MSB	Yes	Yes	
SHAR	Rn	0100nnnn00100001	MSB → Rn → T	1	LSB	Yes	Yes	
SHLD	Rm, Rn	0100nnnnmmmm1101	When Rm ≥ 0, Rn << Rm → Rn When Rm < 0, Rn >> |Rm| → [0 → Rn]	1	—		Yes	
SHLL	Rn	0100nnnn00000000	T ← Rn ← 0	1	MSB	Yes	Yes	
SHLR	Rn	0100nnnn00000001	0 → Rn → T	1	LSB	Yes	Yes	
SHLL2	Rn	0100nnnn00001000	Rn << 2 → Rn	1	—	Yes	Yes	
SHLR2	Rn	0100nnnn00001001	Rn >> 2 → Rn	1	—	Yes	Yes	
SHLL8	Rn	0100nnnn00011000	Rn << 8 → Rn	1	—	Yes	Yes	
SHLR8	Rn	0100nnnn00011001	Rn >> 8 → Rn	1	—	Yes	Yes	
SHLL16	Rn	0100nnnn00101000	Rn << 16 → Rn	1	—	Yes	Yes	
SHLR16	Rn	0100nnnn00101001	Rn >> 16 → Rn	1	—	Yes	Yes	

2.5.6 Branch Instructions

Table 2.28 shows branch instructions.

Table 2.28 Branch Instructions

Instruction	Instruction Code	Operation	Execution Cycles	T Bit	Compatibility		
					SH2E	SH4	SH-2A/ SH2A-FPU
BF label	10001011dddddddd	When T = 0, disp × 2 + PC → PC, When T = 1, nop	3/1*	—	Yes	Yes	
BF/S label	10001111dddddddd	Delayed branch When T = 0, disp × 2 + PC → PC, When T = 1, nop	2/1*	—	Yes	Yes	
BT label	10001001dddddddd	When T = 1, disp × 2 + PC → PC, When T = 0, nop	3/1*	—	Yes	Yes	
BT/S label	10001101dddddddd	Delayed branch When T = 1, disp × 2 + PC → PC, When T = 0, nop	2/1*	—	Yes	Yes	
BRA label	1010dddddddddddd	Delayed branch, disp × 2 + PC → PC	2	—	Yes	Yes	
BRAF Rm	0000mmmm00100011	Delayed branch, Rm + PC → PC	2	—	Yes	Yes	
BSR label	1011dddddddddddd	Delayed branch, PC → PR, disp × 2 + PC → PC	2	—	Yes	Yes	
BSRF Rm	0000mmmm00000011	Delayed branch, PC → PR, Rm + PC → PC	2	—	Yes	Yes	
JMP @Rm	0100mmmm00101011	Delayed branch, Rm → PC	2	—	Yes	Yes	
JSR @Rm	0100mmmm00001011	Delayed branch, PC → PR, Rm → PC	2	—	Yes	Yes	
JSR/N @Rm	0100mmmm01001011	PC - 2 → PR, Rm → PC	3	—			Yes
JSR/N @@(disp8, TBR)	10000011dddddddd	PC - 2 → PR , (disp × 4 + TBR) → PC	5	—			Yes
RTS	0000000000001011	Delayed branch, PR → PC	2	—	Yes	Yes	
RTS/N	0000000001101011	PR → PC	3	—			Yes
RTV/N Rm	0000mmmm01111011	Rm → R0 , PR → PC	3	—			Yes

Note: * One cycle when the program does not branch.

2.5.7 System Control Instructions

Tables 2.29 and 2.30 show system control instructions.

Table 2.29 System Control Instructions (1)

Instruction		Instruction Code	Operation	Execution Cycles	T Bit	Compatibility		
						SH2E	SH4	SH-2A/ SH2A-FPU
CLRT		0000000000001000	0 → T	1	0	Yes	Yes	
CLRMAC		0000000000101000	0 → MACH , MACL	1	—	Yes	Yes	
LDBANK	@Rm, R0	0100mmmm11100101	(Specified register bank entry) → R0	6	—			Yes
LDC	Rm, SR	0100mmmm00001110	Rm → SR	3	LSB	Yes	Yes	
LDC	Rm, TBR	0100mmmm01001010	Rm → TBR	1	—			Yes
LDC	Rm, GBR	0100mmmm00011110	Rm → GBR	1	—	Yes	Yes	
LDC	Rm, VBR	0100mmmm00101110	Rm → VBR	1	—	Yes	Yes	
LDC.L	@Rm+, SR	0100mmmm00000111	(Rm) → SR, Rm + 4 → Rm	5	LSB	Yes	Yes	
LDC.L	@Rm+, GBR	0100mmmm00010111	(Rm) → GBR, Rm + 4 → Rm	1	—	Yes	Yes	
LDC.L	@Rm+, VBR	0100mmmm00100111	(Rm) → VBR, Rm + 4 → Rm	1	—	Yes	Yes	
LDS	Rm, MACH	0100mmmm00001010	Rm → MACH	1	—	Yes	Yes	
LDS	Rm, MACL	0100mmmm00011010	Rm → MACL	1	—	Yes	Yes	
LDS	Rm, PR	0100mmmm00101010	Rm → PR	1	—	Yes	Yes	
LDS.L	@Rm+, MACH	0100mmmm00000110	(Rm) → MACH, Rm + 4 → Rm	1	—	Yes	Yes	
LDS.L	@Rm+, MACL	0100mmmm00010110	(Rm) → MACL, Rm + 4 → Rm	1	—	Yes	Yes	
LDS.L	@Rm+, PR	0100mmmm00100110	(Rm) → PR, Rm + 4 → Rm	1	—	Yes	Yes	
NOP		0000000000001001	No operation	1	—	Yes	Yes	
RESBANK		0000000001011011	Bank → R0 to R14, GBR, MACH, MACL, PR	9*	—			Yes
RTE		0000000000101011	Delayed branch, stack area → PC/SR	6	—	Yes	Yes	
SETT		0000000000011000	1 → T	1	1	Yes	Yes	
SLEEP		0000000000011011	Sleep	5	—	Yes	Yes	
STBANK	R0, @Rn	0100nnnn11100001	R0 → (specified register bank entry)	7	—			Yes
STC	SR, Rn	0000nnnn00000010	SR → Rn	2	—	Yes	Yes	
STC	TBR, Rn	0000nnnn01001010	TBR → Rn	1	—			Yes
STC	GBR, Rn	0000nnnn00010010	GBR → Rn	1	—	Yes	Yes	
STC	VBR, Rn	0000nnnn00100010	VBR → Rn	1	—	Yes	Yes	
STC.L	SR, @-Rn	0100nnnn00000011	Rn - 4 → Rn, SR → (Rn)	2	—	Yes	Yes	
STC.L	GBR, @-Rn	0100nnnn00010011	Rn - 4 →Rn, GBR → (Rn)	1	—	Yes	Yes	
STC.L	VBR, @-Rn	0100nnnn00100011	Rn - 4 → Rn, VBR → (Rn)	1	—	Yes	Yes	
STS	MACH, Rn	0000nnnn00001010	MACH → Rn	1	—	Yes	Yes	
STS	MACL, Rn	0000nnnn00011010	MACL → Rn	1	—	Yes	Yes	
STS	PR, Rn	0000nnnn00101010	PR → Rn	1	—	Yes	Yes	

Note: * Instruction execution cycles: The execution cycles shown in the table are minimums. In practice, the number of instruction execution cycles increases in cases such as the following:
a. When there is a conflict between an instruction fetch and a data access
b. When the destination register of a load instruction (memory → register) is the same as the register used by the next instruction.
Note that, in the event of bank overflow, the number of cycles is 19.

Table 2.30 System Control Instructions (2)

Instruction		Instruction Code	Operation	Execution Cycles	T Bit	Compatibility		
						SH2E	SH4	SH-2A/ SH2A-FPU
STS.L	MACH, @-Rn	0100nnnn00000010	Rn - 4 → Rn, MACH → (Rn)	1	—	Yes	Yes	
STS.L	MACL, @-Rn	0100nnnn00010010	Rn - 4 → Rn, MACL → (Rn)	1	—	Yes	Yes	
STS.L	PR, @-Rn	0100nnnn00100010	Rn - 4 → Rn, PR → (Rn)	1	—	Yes	Yes	
TRAPA	#imm	11000011iiiiiiii	PC/SR → stack area, (imm × 4 + VBR) → PC	5	—	Yes	Yes	

2.5.8 Floating-Point Instructions

Tables 2.31 and 2.32 show floating-point instructions.

Table 2.31 Floating-Point Instructions (1)

Instruction		Instruction Code	Operation	Execution Cycles	T Bit	Compatibility		
						SH2E	SH4	SH-2A/ SH2A-FPU
FABS	FRn	1111nnnn01011101	|FRn| → FRn	1	—	Yes	Yes	
FABS	DRn	1111nnn001011101	|DRn| → DRn	1	—		Yes	
FADD	FRm, FRn	1111nnnnmmmm0000	FRn + FRm → FRn	1	—	Yes	Yes	
FADD	DRm, DRn	1111nnn0mmm00000	DRn + DRm → DRn	6	—		Yes	
FCMP/EQ	FRm, FRn	1111nnnnmmmm0100	(FRn = FRm) ? 1 : 0 → T	1	Comparison result	Yes	Yes	
FCMP/EQ	DRm, DRn	1111nnn0mmm00100	(DRn = DRm) ? 1 : 0 → T	2	Comparison result		Yes	
FCMP/GT	FRm, FRn	1111nnnnmmmm0101	(FRn>FRm) ? 1 : 0 → T	1	Comparison result	Yes	Yes	
FCMP/GT	DRm, DRn	1111nnn0mmm00101	(DRn>DRm) ? 1: 0 → T	2	Comparison result		Yes	
FCNVDS	DRm, FPUL	1111mmm010111101	(float) DRm → FPUL	2	—		Yes	
FCNVSD	FPUL, DRn	1111nnn010101101	(double) FPUL → DRn	2	—		Yes	
FDIV	FRm, FRn	1111nnnnmmmm0011	FRn/FRm → FRn	10	—	Yes	Yes	
FDIV	DRm, DRn	1111nnn0mmm00011	DRn/DRm → DRn	23	—		Yes	
FLDI0	FRn	1111nnnn10001101	0 × 00000000 → FRn	1	—	Yes	Yes	
FLDI1	FRn	1111nnnn10011101	0 × 3F800000 → FRn	1	—	Yes	Yes	
FLDS	FRm, FPUL	1111mmmm00011101	FRm → FPUL	1	—	Yes	Yes	
FLOAT	FPUL, FRn	1111nnnn00101101	(float) FPUL → FRn	1	—	Yes	Yes	
FLOAT	FPUL, DRn	1111nnn000101101	(double) FPUL → DRn	2	—		Yes	
FMAC	FR0, FRm, FRn	1111nnnnmmmm1110	FR0 × FRm + FRn → FRn	1	—	Yes	Yes	
FMOV	FRm, FRn	1111nnnnmmmm1100	FRm → FRn	1	—	Yes	Yes	
FMOV	DRm, DRn	1111nnn0mmm01100	DRm → DRn	2	—		Yes	
FMOV.S	@(R0, Rm), FRn	1111nnnnmmmm0110	(R0 + Rm) → FRn	1	—	Yes	Yes	
FMOV.D	@(R0, Rm), DRn	1111nnn0mmmm0110	(R0 + Rm) → DRn	2	—		Yes	
FMOV.S	@Rm+, FRn	1111nnnnmmmm1001	(Rm) → FRn, Rm+ = 4	1	—	Yes	Yes	
FMOV.D	@Rm+, DRn	1111nnn0mmmm1001	(Rm) → DRn, Rm+ = 8	2	—		Yes	
FMOV.S	@Rm, FRn	1111nnnnmmmm1000	(Rm) → FRn	1	—	Yes	Yes	
FMOV.D	@Rm, DRn	1111nnn0mmmm1000	(Rm) → DRn	2	—		Yes	
FMOV.S	@(disp12, Rm), FRn	0011nnnnmmmm0001 0111dddddddddddd	(disp × 4 + Rm) → FRn	1	—			Yes
FMOV.D	@(disp12, Rm), DRn	0011nnn0mmmm0001 0111dddddddddddd	(disp × 8 + Rm) → DRn	2	—			Yes

Table 2.32 Floating-Point Instructions (2)

Instruction		Instruction Code	Operation	Execu-tion Cycles	T Bit	Compatibility		
						SH2E	SH4	SH-2A/ SH2A-FPU
FMOV.S	FRm, @(R0, Rn)	1111nnnnmmmm0111	FRm → (R0 + Rn)	1	—	Yes	Yes	
FMOV.D	DRm, @(R0, Rn)	1111nnnnmmm00111	DRm → (R0 + Rn)	2	—		Yes	
FMOV.S	FRm, @-Rn	1111nnnnmmmm1011	Rn- = 4, FRm → (Rn)	1	—	Yes	Yes	
FMOV.D	DRm, @-Rn	1111nnnnmmm01011	Rn- = 8, DRm → (Rn)	2	—		Yes	
FMOV.S	FRm, @Rn	1111nnnnmmmm1010	FRm → (Rn)	1	—	Yes	Yes	
FMOV.D	DRm, @Rn	1111nnnnmmm01010	DRm → (Rn)	2	—		Yes	
FMOV.S	FRm, @(disp12, Rn)	0011nnnnmmmm0001 0011dddddddddddd	FRm → (disp × 4 + Rn)	1	—			Yes
FMOV.D	DRm, @(disp12, Rn)	0011nnnnmmm00001 0011dddddddddddd	DRm → (disp × 8 + Rn)	2	—			Yes
FMUL	FRm, FRn	1111nnnnmmmm0010	FRn × FRm → FRn	1	—	Yes	Yes	
FMUL	DRm, DRn	1111nnn0mmm00010	DRn × DRm → DRn	6	—		Yes	
FNEG	FRn	1111nnnn01001101	-FRn → FRn	1	—	Yes	Yes	
FNEG	DRn	1111nnn001001101	-DRn → DRn	1	—		Yes	
FSCHG		1111001111111101	FPSCR.SZ = ~ FPSCR.SZ	1	—		Yes	
FSQRT	FRn	1111nnnn01101101	√FRn → FRn	9	—		Yes	
FSQRT	DRn	1111nnn001101101	√DRn → DRn	22	—		Yes	
FSTS	FPUL, FRn	1111nnnn00001101	FPUL → FRn	1	—	Yes	Yes	
FSUB	FRm, FRn	1111nnnnmmmm0001	FRn – FRm → FRn	1	—	Yes	Yes	
FSUB	DRm, DRn	1111nnn0mmm00001	DRn – DRm → DRn	6	—		Yes	
FTRC	FRm, FPUL	1111mmmm00111101	(long) FRm → FPUL	1	—	Yes	Yes	
FTRC	DRm, FPUL	1111mmm000111101	(long) DRm → FPUL	2	—		Yes	

2.5.9 FPU-Related CPU Instructions

Table 2.33 shows FPU-related CPU instructions.

Table 2.33 FPU-Related CPU Instructions

Instruction		Instruction Code	Operation	Execution Cycles	T Bit	Compatibility		
						SH2E	SH4	SH-2A/ SH2A-FPU
LDS	Rm,FPSCR	0100mmmm01101010	Rm → FPSCR	1	—	Yes	Yes	
LDS	Rm,FPUL	0100mmmm01011010	Rm → FPUL	1	—	Yes	Yes	
LDS.L	@Rm+,FPSCR	0100mmmm01100110	(Rm) → FPSCR, Rm+ = 4	1	—	Yes	Yes	
LDS.L	@Rm+,FPUL	0100mmmm01010110	(Rm) → FPUL, Rm+ = 4	1	—	Yes	Yes	
STS	FPSCR,Rn	0000nnnn01101010	FPSCR → Rn	1	—	Yes	Yes	
STS	FPUL,Rn	0000nnnn01011010	FPUL → Rn	1	—	Yes	Yes	
STS.L	FPSCR,@-Rn	0100nnnn01100010	Rn- = 4, FPCSR → (Rn)	1	—	Yes	Yes	
STS.L	FPUL,@-Rn	0100nnnn01010010	Rn- = 4, FPUL → (Rn)	1	—	Yes	Yes	

2.5.10 Bit Manipulation Instructions

Table 2.34 shows bit manipulation instructions.

Table 2.34 Bit Manipulation Instructions

Instruction		Instruction Code	Operation	Execution Cycles	T Bit	Compatibility		
						SH2E	SH4	SH-2A/ SH2A-FPU
BAND.B	#imm3,@(disp12,Rn)	0011nnnn0iii1001 0100dddddddddddd	(imm of (disp + Rn)) & T → T	3	Operation result			Yes
BANDNOT.B	#imm3,@(disp12,Rn)	0011nnnn0iii1001 1100dddddddddddd	~ (imm of (disp + Rn)) & T → T	3	Operation result			Yes
BCLR.B	#imm3,@(disp12,Rn)	0011nnnn0iii1001 0000dddddddddddd	0 → (imm of (disp + Rn))	3	—			Yes
BCLR	#imm3,Rn	10000110nnnn0iii	0 → imm of Rn	1	—			Yes
BLD.B	#imm3,@(disp12,Rn)	0011nnnn0iii1001 0011dddddddddddd	(imm of (disp + Rn)) → T	3	Operation result			Yes
BLD	#imm3,Rn	10000111nnnn1iii	imm of Rn → T	1	Operation result			Yes
BLDNOT.B	#imm3,@(disp12,Rn)	0011nnnn0iii1001 1011dddddddddddd	~ (imm of (disp + Rn)) → T	3	Operation result			Yes
BOR.B	#imm3,@(disp12,Rn)	0011nnnn0iii1001 0101dddddddddddd	(imm of (disp + Rn)) | T → T	3	Operation result			Yes
BORNOT.B	#imm3,@(disp12,Rn)	0011nnnn0iii1001 1101dddddddddddd	~ (imm of (disp + Rn)) | T → T	3	Operation result			Yes
BSET.B	#imm3,@(disp12,Rn)	0011nnnn0iii1001 0001dddddddddddd	1 → (imm of (disp + Rn))	3	—			Yes
BSET	#imm3,Rn	10000110nnnn1iii	1 → imm of Rn	1	—			Yes
BST.B	#imm3,@(disp12,Rn)	0011nnnn0iii1001 0010dddddddddddd	T → (imm of (disp + Rn))	3	—			Yes
BST	#imm3,Rn	10000111nnnn0iii	T → imm of Rn	1	—			Yes
BXOR.B	#imm3, @(disp12,Rn)	0011nnnn0iii1001 0110dddddddddddd	(imm of (disp + Rn)) ^ T → T	3	Operation result			Yes

2.6 Processing States

The CPU has four processing states: reset, exception handling, program execution, and power-down. Figure 2.8 shows the CPU state transitions.

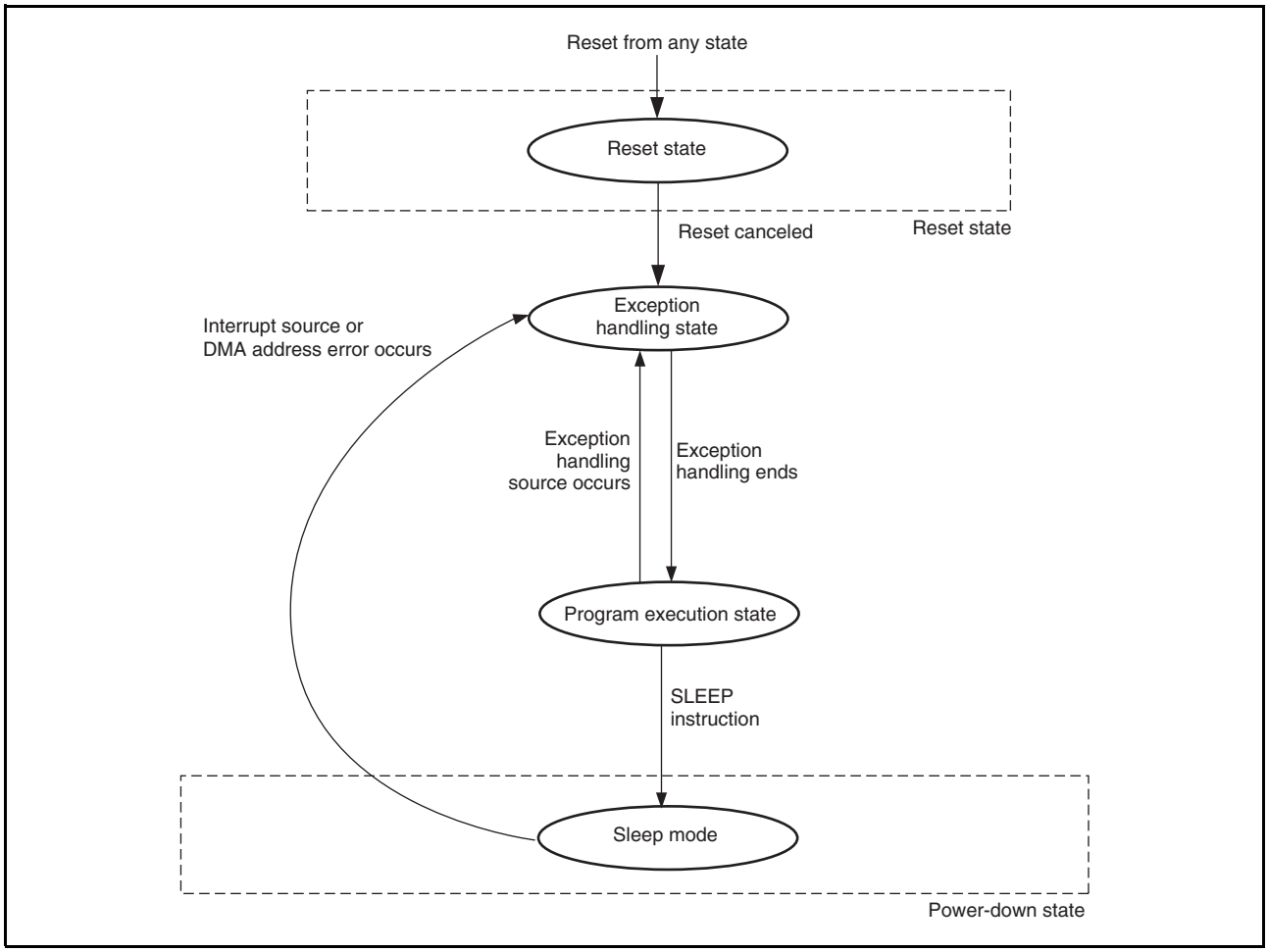


Figure 2.8 CPU State Transitions

(1) Reset State

In this state, the CPU is reset by a hardware reset, a software reset, a watchdog timer reset, and power-down mode wake-up reset.

(2) Exception Handling State

The exception handling state is a transient state that occurs when exception handling sources such as resets or interrupts alters the CPU's processing state flow.

For a reset, the initial values of the program counter (PC) (execution start address) and stack pointer (SP) are fetched from the exception handling vector table and stored; the CPU then branches to the execution start address and execution of the program begins.

For an interrupt, the stack pointer (SP) is accessed and the program counter (PC) and status register (SR) are saved to the stack area. The exception service routine start address is fetched from the exception handling vector table; the CPU then branches to that address and the program starts executing, thereby entering the program execution state.

(3) Program Execution State

In the program execution state, the CPU sequentially executes the program.

(4) Power-Down State

In the power-down state, the CPU stops operating to conserve power. Sleep mode is entered by executing a SLEEP instruction.

3. Address Space

Figures 3.1 to 3.5 show the address space of this LSI. Addresses are allocated to each area as follows:

- H'0000 0000 to H'0007 FFFF: On-chip ROM (read from user MAT) [512 Kbytes]
  (H'0000 0000 to H'0000 7FFF) *
- H'0040 2000 to H'0040 3FFF: FCU firmware area
- H'8010 0000 to H'8010 7FFF: EEPROM [32 Kbytes]
- H'8080 0000 to H'8087 FFFF: On-chip ROM (write to user MAT) [512 Kbytes]
  (H'8080 0000 to H'8080 7FFF) *
- H'80FF 8000 to H'80FF 9FFF: FCU RAM area
- H'FF40 0000 to H'FF7F FFFF: Peripheral function register
- H'FFF8 0000 to H'FFF8 FFFF: On-chip RAM [64 Kbytes]
- H'FFFC 0000 to H'FFFF FFFF: Peripheral function register

Note: * When the user boot MAT of the on-chip ROM is selected. For details, refer to section 24, ROM.

Do not access reserved areas; otherwise, operation cannot be guaranteed.
The ROM has two types of memory areas for reading and programming/erasing (hereafter referred to as memory MATs) in the same address space. For details, refer to section 24, ROM.

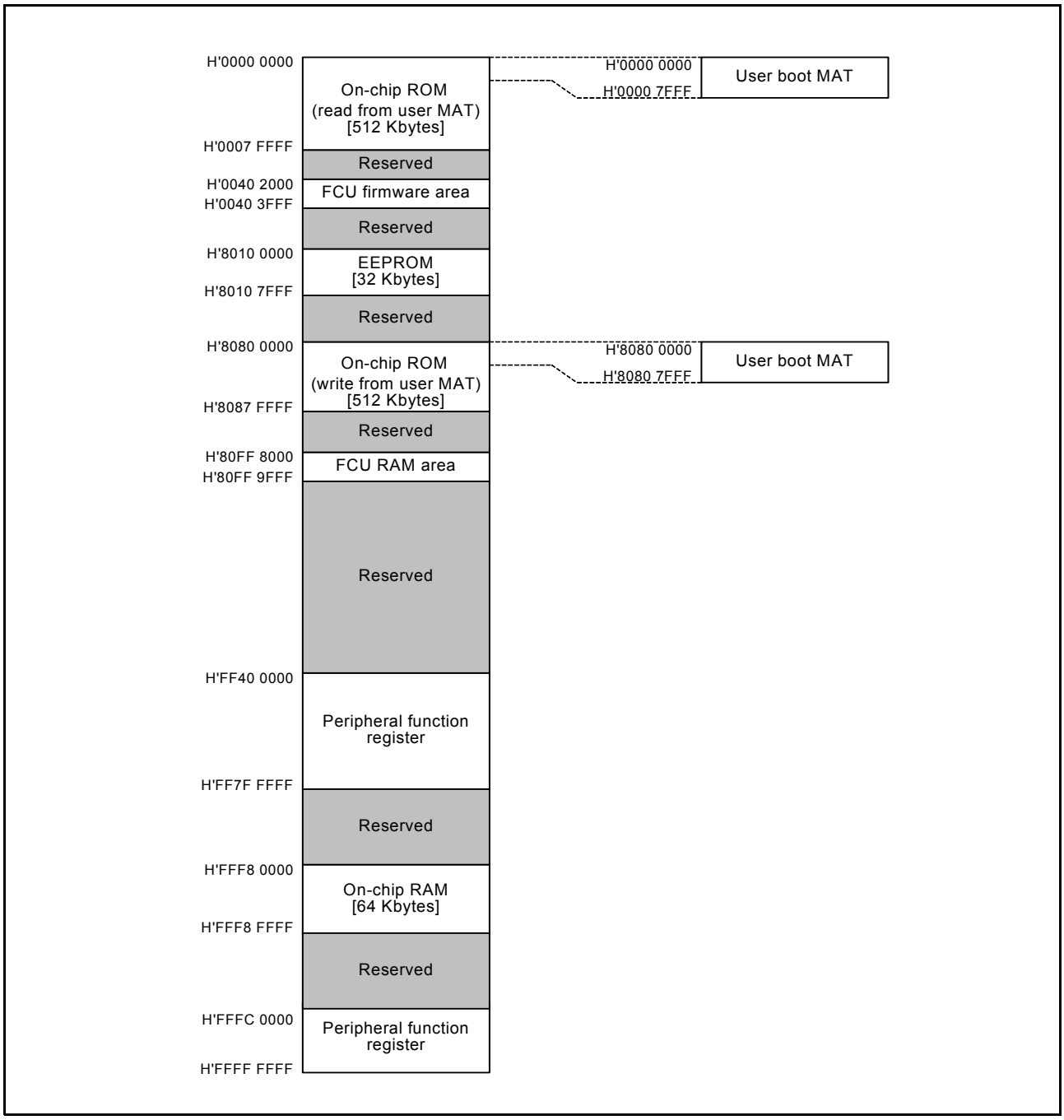


Figure 3.1 Address Space (1) (ROM: 512 KB, EERPROM: 32 KB, and RAM: 64 KB)

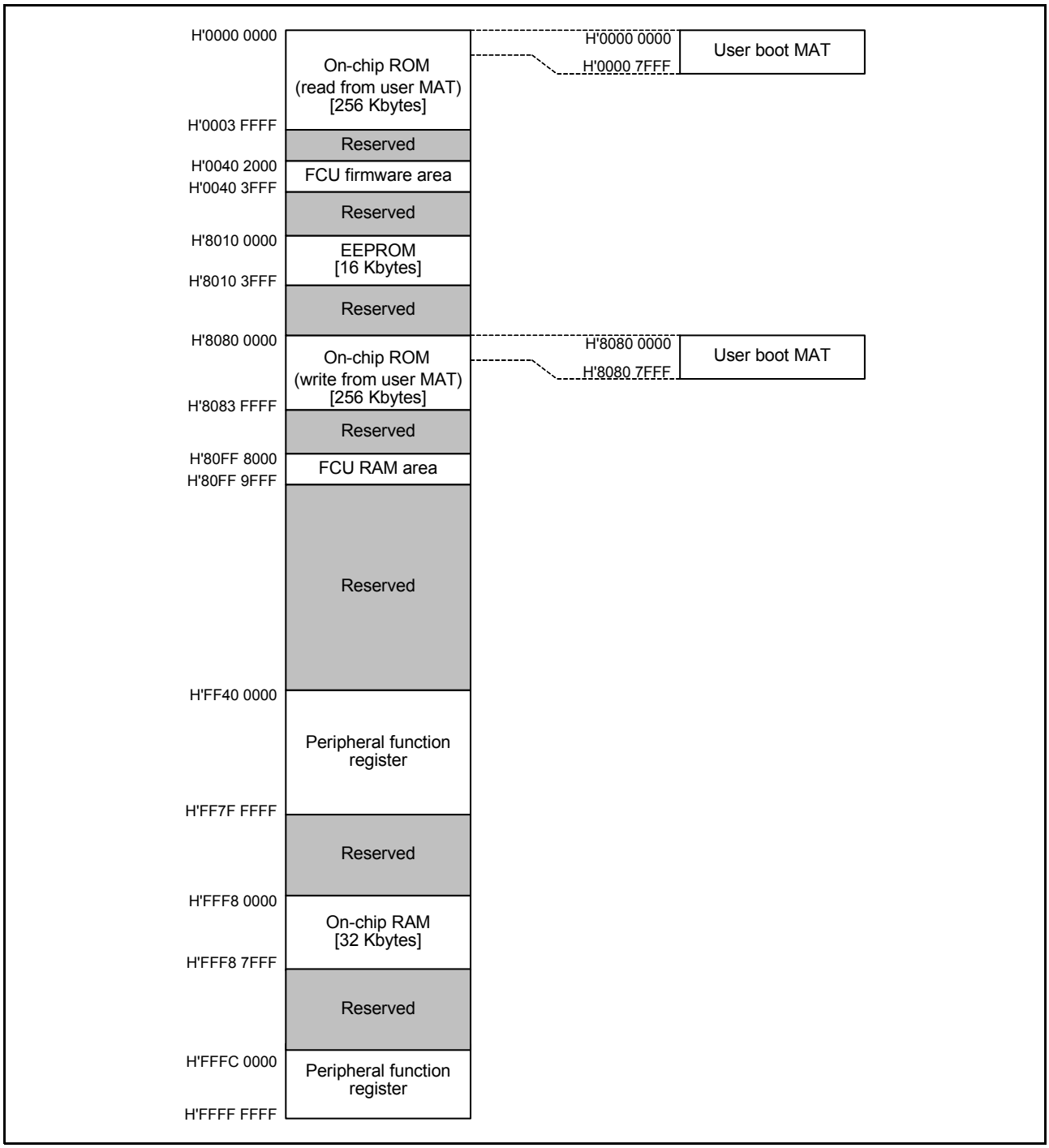


Figure 3.2 Address Space (2) (ROM: 256 KB, EERPROM: 16 KB, and RAM: 32 KB)

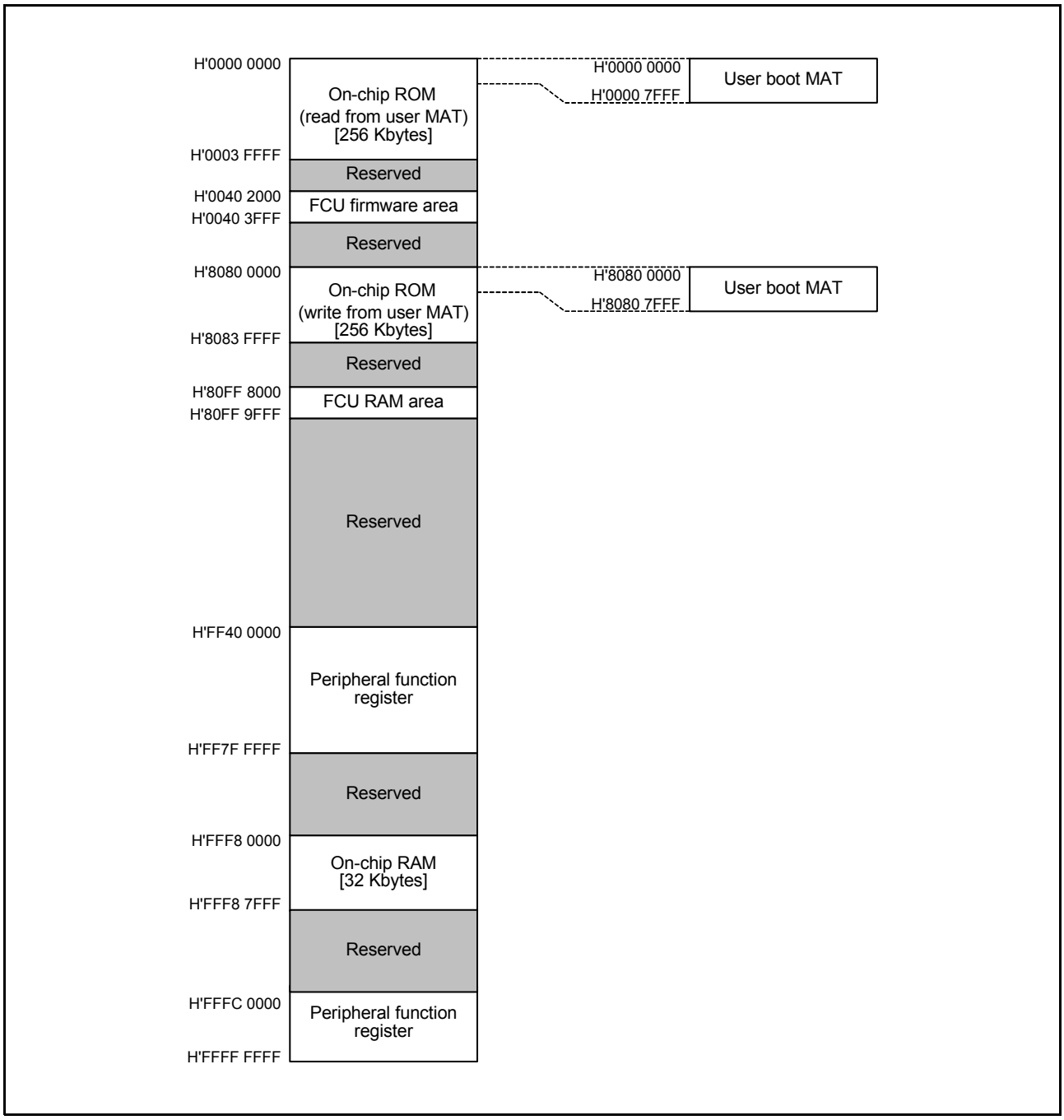


Figure 3.3 Address Space (3) (ROM: 256 KB and RAM: 32 KB)

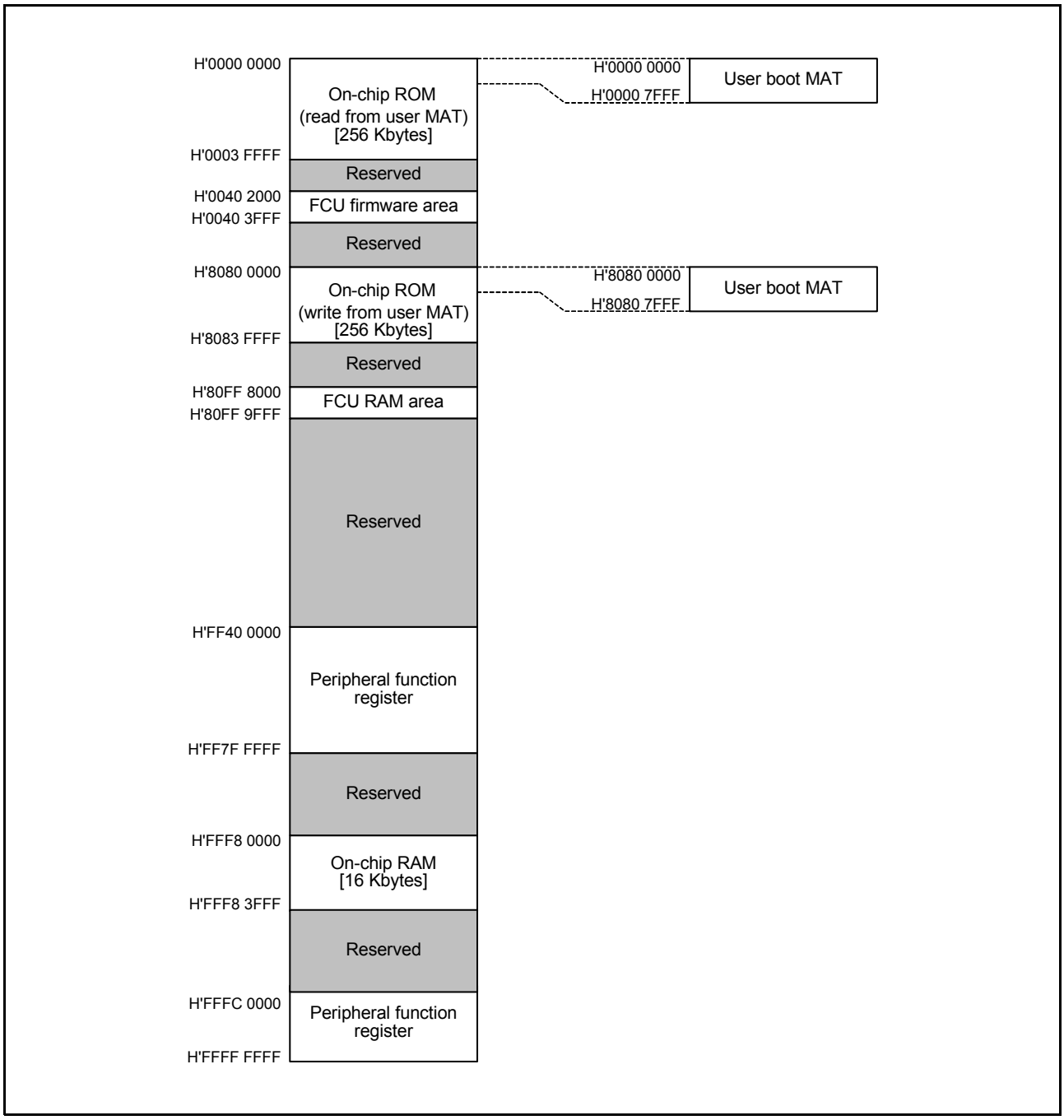
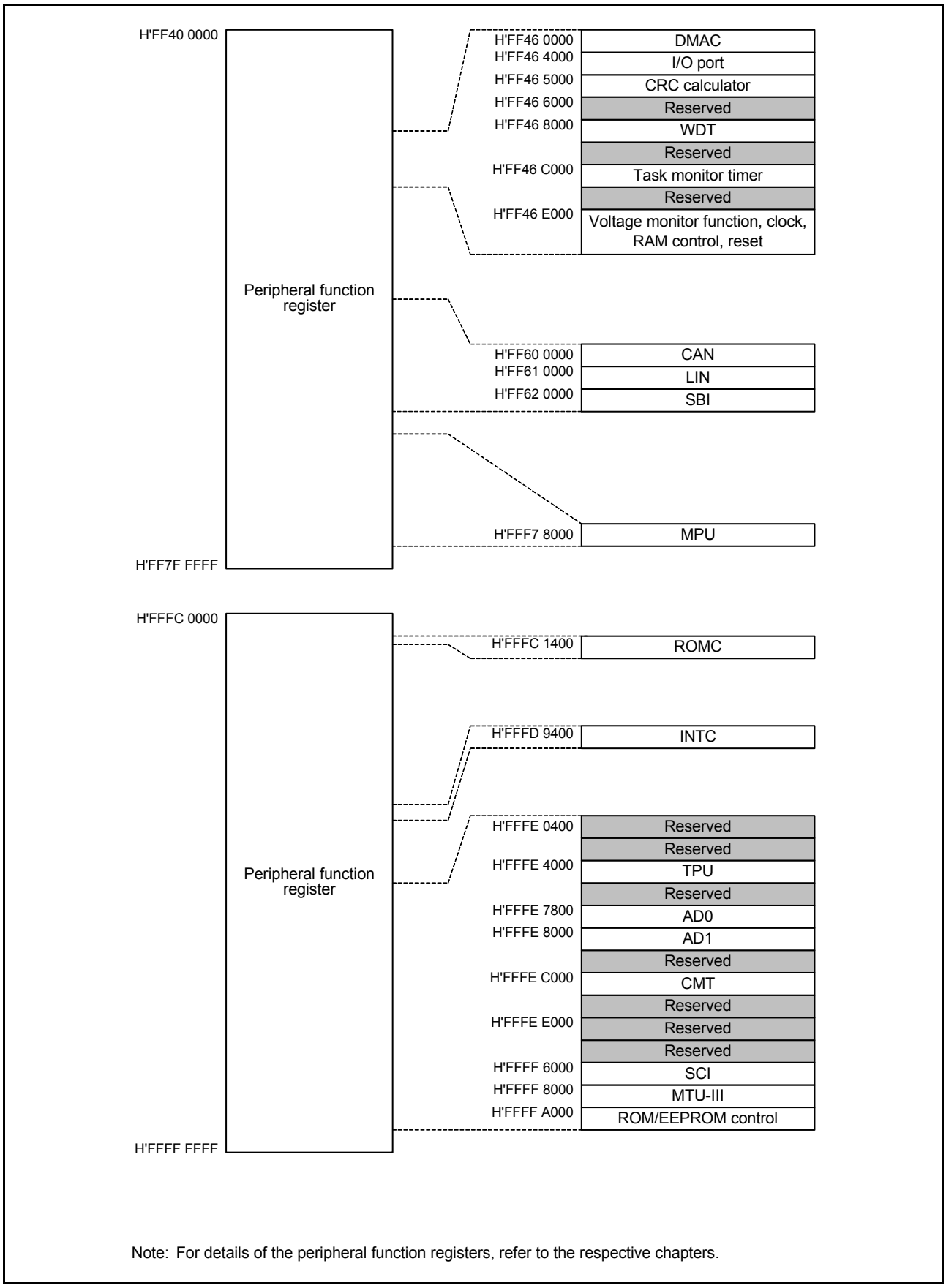


Figure 3.4 Address Space (4) (ROM: 256 KB and RAM: 16 KB)



Note: For details of the peripheral function registers, refer to the respective chapters.

Figure 3.5 Address Space (5)

4. Operating Modes

4.1 Types and Selection of Operating Modes

MCU operating modes include single-chip mode, user boot mode, and boot mode.

(1) Single-Chip Mode

After a reset, the program is executed from the address indicated by the reset vector in the user program area. Programs usually run in this mode.

(2) User Boot Mode

After a reset, the program is executed from the address indicated by the reset vector in the user boot area, and then the user's flash reprogrammer runs. For details, see section 24, ROM.

(3) Boot Mode

After a reset, the program is executed from the address indicated by the reset vector in the boot area, and then the Renesas flash reprogrammer runs. The general flash programmer can be used to write to or erase the user area and user boot area. SCI channel 1 is used for serial communication between the general flash programmer and the MCU. For details, see section 24, ROM.

Table 4.1 lists the pin settings for operating modes.

Table 4.1 Pin Settings for Operating Modes

Mode Name	Pin Settings*1		
	MD0	MD1	ASEMD*2
Single-chip mode	Low	Low	Low
User boot mode	Low	High	Low
Boot mode	High	Low	Low

Notes: 1. Each pin must be pulled up (connected to VCC via a resistor) or pulled down (connected to VSS via a resistor) on the board. Do not specify a combination of input levels that are not shown in the table.
As safety measures in case when the pull-down resistor is detached from the board, these mode pins are pulled down within the chip. For details of resistor values, see section 30, Electrical Characteristics.
2. The ASEMD pin is used only for the debugging function. In single-chip and user boot modes, when the ASEMD pin is set high, the emulator can be connected. If the ASEMD pin is set high without connecting an emulator, the operation is not guaranteed.

5. Clocks

5.1 Introduction

This LSI integrates the following three clock generators: a main clock oscillator, PLL frequency synthesizer, and low-speed on-chip oscillator. The clocks listed in Table 5.2 are generated from these clock generators. Other functions are a main clock monitor function that detects when the main clock stops oscillating, and a clock output function for the CLKOUT pin.

Table 5.1 lists the Clock Generators and Table 5.2 lists the Clock Specifications.
Figure 5.1 shows the Clock Block Diagram (1) and Figure 5.2 shows the Clock Block Diagram (2).

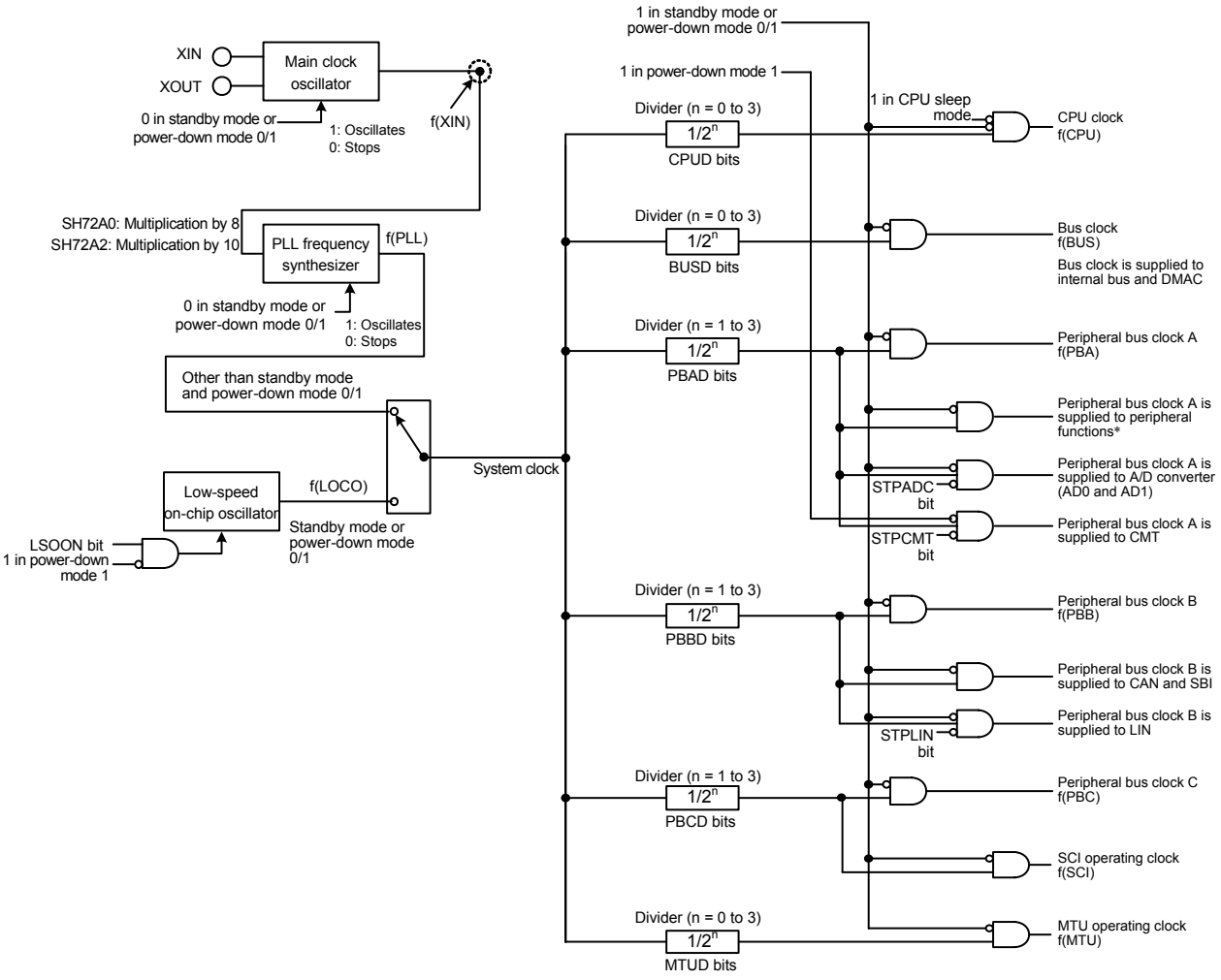
Table 5.1 Clock Generators

Clock Generator	Description
Main clock oscillator	• f(XIN): 8 or 10 MHz • Additional circuit: Connect a crystal oscillator between XIN and XOUT, or input an external clock from the XIN pin • Oscillation control: Oscillation “starts/stops” can be switched • After reset: Oscillates
PLL frequency synthesizer	• f(PLL): Maximum 100 MHz • Oscillation control: Oscillation • After reset: Oscillates
Low-speed on-chip oscillator	• f(LOCO): 125 kHz • Oscillation control: Oscillation • After reset: Oscillates

Table 5.2 Clock Specifications

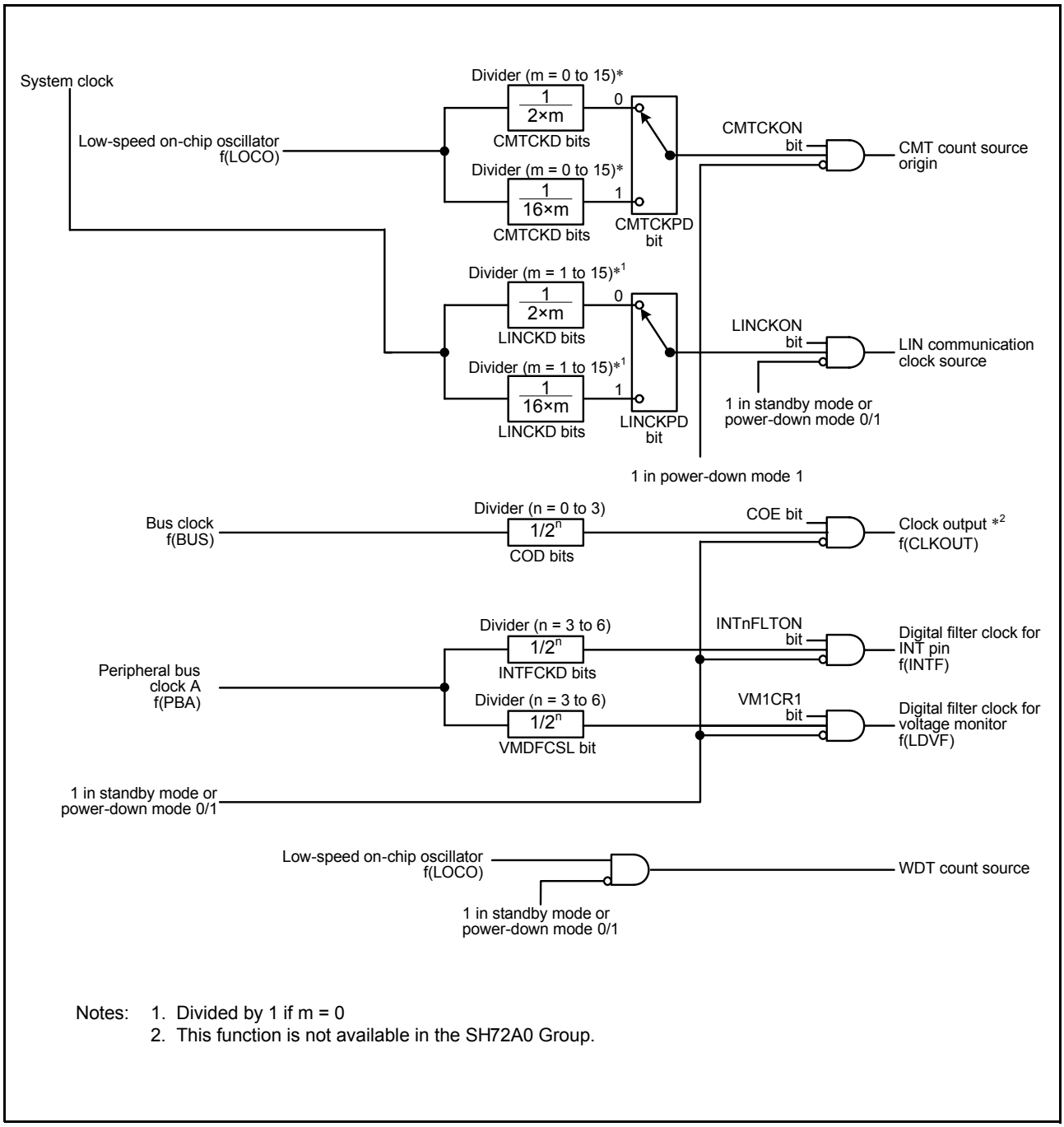
Clock Name [Legend Indicating Frequency]		Description
System clock [f(SYS)]		• Clock source: f(PLL) • After reset: f(PLL) clock divided by 1
CPU clock [f(CPU)]		• Division ratio: System clock divided by 1, 2, 4, or 8 selectable • After reset: System clock divided by 4
Bus clock [f(BUS)]		• Division ratio: System clock divided by 1, 2, 4, or 8 selectable • After reset: System clock divided by 4
Peripheral bus clocks	Peripheral bus clock A [f(PBA)]	• Division ratio: System clock divided by 2, 4, or 8 selectable • After reset: System clock divided by 8
	Peripheral bus clock B [f(PBB)]	• Division ratio: System clock divided by 2, 4, or 8 selectable • After reset: System clock divided by 8
	Peripheral bus clock C [f(PBC)]	• Division ratio: System clock divided by 2, 4, or 8 selectable • After reset: System clock divided by 8
	SCI operating clock [f(SCI)]	• Division ratio: System clock divided by 2, 4, or 8 selectable • After reset: System clock divided by 8
	MTU operating clock [f(MTU)]	• Division ratio: System clock divided by 1, 2, 4, or 8 selectable • After reset: System clock divided by 4
Peripheral function clocks	CMT count source origin [f(CMT)]	• Division ratio:f(LOCO) clock divided by 2n or 16n selectable (n = 0 to 15) • Oscillation control: Oscillation “starts/stops” can be switched • After reset: Stops
	LIN communication clock source [f(LIN)]	• Clock source: System clock • Division ratio:Clock source divided by 2n or 16n selectable (n = 1 to 15) • Oscillation control: Oscillation “starts/stops” can be switched • After reset: Stops
	WDT count source [f(WDT)]	• Clock source: f(LOCO) • Oscillation control: Oscillation starts • After reset: Oscillates
Clock output* [f(CLKOUT)]		• Clock output from the CLKOUT pin • Division ratio: Bus clock divided by 1, 2, 4, or 8 selectable • Output control: Clock “output/not output” can be switched
Digital filter clock for voltage monitor [f(LDVF)]		• Division ratio: Peripheral bus clock A divided by 8, 16, 32, or 64 selectable • Function control: Enabled/disabled can be controlled
Digital filter clock for INT pin [f(INTF)]		• Division ratio: Peripheral bus clock A divided by 8, 16, 32, or 64 selectable • Function control: Enabled/disabled can be controlled

Note: * This function is not available in the SH72A0 Group.



Notes: • f( ) is used as a legend for indicating the frequency of a clock.
• CPU sleep mode is the mode to be transited when the SLEEP instruction is executed while the SLCR0.SSBY bit is 0.
• Standby mode is the mode to be transited when the SLEEP instruction is executed while the SLCR0.SSBY bit is 1 and the SLCR1.DPSBY bit is 0.
• Power-down 0 is the mode to be transited when the SLEEP instruction is executed while the SLCR0.SSBY bit is 1, the SLCR1.DPSBY bit is 1, and the SLCR1.CMTCUT bit is 0.
• Power-down 1 is the mode to be transited when the SLEEP instruction is executed while the SLCR0.SSBY bit is 1, the SLCR1.DPSBY bit is 1, and the SLCR1.CMTCUT bit is 1.
* Peripheral functions: INTC, WDT, task monitor timer, I/O ports, TPU, and CRC calculator.

Figure 5.1 Clock Block Diagram (1)



Notes: 1. Divided by 1 if m = 0
2. This function is not available in the SH72A0 Group.

Figure 5.2 Clock Block Diagram (2)

Figure 5.3 shows the Mode Transitions by CPU State. Table 5.3 lists the Power Control Specifications, Table 5.4 lists the Clock Oscillation States by CPU State.

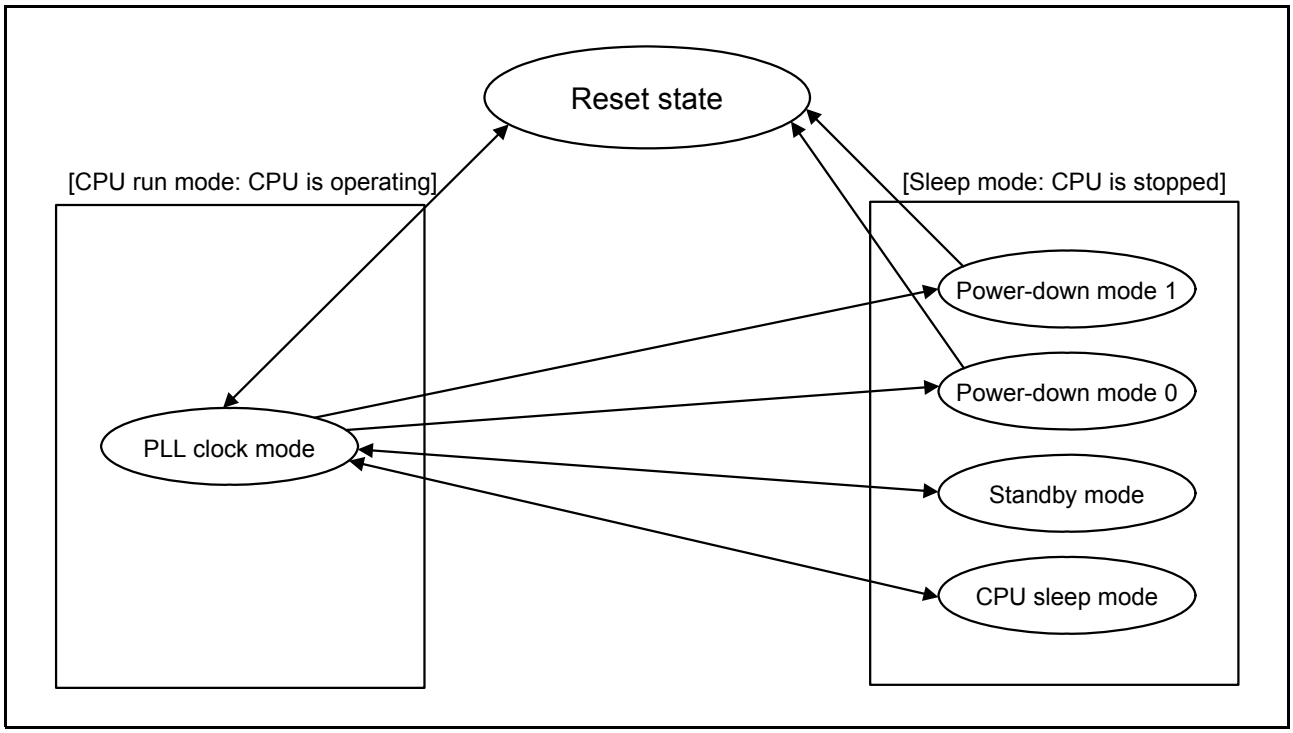


Figure 5.3 Mode Transitions by CPU State

Table 5.3 Power Control Specifications

Item		Description
CPU run mode		• The CPU clock, bus clock, peripheral bus clock, and peripheral function clock are supplied and the peripheral functions operate. • The frequency is decided by dividing the system clock.
Sleep modes	CPU sleep mode	• When the SLEEP instruction is executed while the SLCR0.SSBY bit is 0, the CPU clock stops. • All interrupts except software interrupt can be used for wake-up to the CPU run mode.
	Standby mode	• When the SLEEP instruction is executed while the SLCR0.SSBY bit is 1 and the SLCR1.DPSBY bit in is 0, the CPU clock, bus clock, and peripheral bus clock stop, and the power supply for the PLL is shut off. • A wake-up interrupt source* can be used for wake-up to the CPU run mode.
	Power-down mode 0	• When the SLEEP instruction is executed while the SLCR0.SSBY bit is 1, the SLCR1.DPSBY bit is 1, and the SLCR1.CMTCUT bit is 0, the main clock and PLL clock stop, and the power supply for the functions other than the CMT and WDT is shut off. • A wake-up interrupt source* can be used to generate a power-down mode wake-up reset and wake up using a PLL clock divided by 4.
	Power-down mode 1	• When the SLEEP instruction is executed while the SLCR0.SSBY bit is 1, the SLCR1.DPSBY bit is 1, and the SLCR1.CMTCUT bit is 1, all the clocks stop and the power supply for the functions other than the CMT and WDT is shut off. • A wake-up interrupt source* can be used to generate a power-down mode wake-up reset and wake up using a PLL clock divided by 4.
Man clock monitor function		• The system clock is switched to f(free-running PLL) by detecting the stop of the main clock. • Function control: “Function used/not used” can be controlled. • Interrupt: Main clock oscillation stop detection interrupt (disabled after reset) • After reset: Monitor function not used

Note: * For details on wake-up interrupt sources, refer to section 5.5, Power Control.

Table 5.4 Clock Oscillation States by CPU State

	Reset State		Sleep Mode				CPU Run Mode
	Reset Asserted *1 VDD = On RESET# = Low	Reset Sequence (VDD = On RESET# = High)	CPU Sleep Mode	Standby Mode	Power-Down Mode 0 *2	Power-Down Mode 1 *2	PLL Clock Mode f (CPU) = f (PLL)/n
Main clock oscillator	Oscillating	Oscillating	Oscillating	Stopped	Stopped	Stopped	Oscillating
Low-speed on-chip oscillator	Oscillating	Oscillating	Stopped/ Oscillating	Stopped/ Oscillating	Oscillating	Stopped	Stopped/ Oscillating
PLL frequency synthesizer	Oscillating	Oscillating	Oscillating	Power supply shut off	Power supply shut off	Power supply shut off	Oscillating
System clock	f(PLL)	f(PLL)	f(PLL)	f(LOCO)	f(LOCO)	Stopped	f(PLL)

Notes: 1. When RESET# = Low, the port goes to Hi-Z state.
2. The mode transits to power-down mode through standby mode.

Table 5.5 lists the Clock I/O Pin.

Table 5.5 Clock I/O Pin

Pin Name	I/O	Description
CLKOUT	Output	Clock output

Note: This function is not available in the SH72A0 Group.

5.2 Registers

Table 5.6 and Table 5.7 list the Clock Registers.

Table 5.6 Clock Registers (1)

Register Name	Symbol	After Reset	Address	Access Size
System protect register 0	SPR0	H'00	H'FF46 E063	8
System protect register 2	SPR2	H'00	H'FF46 E263	8
Low-speed on-chip oscillator control register	LOCR	H'01	H'FF46 E826	8
Main clock monitor function control register	MCMCR	H'00	H'FF46 E838	8
Clock protect register	CPR	H'00	H'FF46 E860	8
CPU clock divide register	CCDR	H'22	H'FF46 E804	8
Peripheral bus clock divide register	PBCDR	H'3333 0002	H'FF46 E808	8, 16, 32
Peripheral bus clock control register	PBCCR	H'0000	H'FF46 E81C	8, 16
LIN clock control register	LINCCR	H'02	H'FF46 E811	8
CMT clock control register	CMTCCR	H'00	H'FF46 E812	8
INT input digital filter control register 0	INTDFCR0	H'0000	H'FF46 E890	8, 16
INT input digital filter control register 1	INTDFCR1	H'0300	H'FF46 E892	8, 16
Sleep mode control register 0	SLCR0	H'0F00	H'FF46 E840	8, 16
Clock output function control register*	COCR	H'00	H'FF46 E800	8
Sleep mode control register 1	SLCR1	H'00	H'FF46 EC03	8
Power-down mode wake-up source select register 0	PDWSSL0	H'00	H'FF46 EC0B	8
Power-down mode wake-up source select register 2	PDWSSL2	H'00	H'FF46 EC13	8
Power-down mode wake-up source select register 3	PDWSSL3	H'00	H'FF46 EC17	8
Power-down mode wake-up source select register 4	PDWSSL4	H'00	H'FF46 EC1B	8
Power-down mode wake-up source select register 5	PDWSSL5	H'00	H'FF46 EC1F	8
Power-down mode wake-up source status register 0	PDWSSR0	H'00	H'FF46 EC23	8
Power-down mode wake-up source status register 2	PDWSSR2	H'00	H'FF46 EC2B	8
Power-down mode wake-up source status register 3	PDWSSR3	H'00	H'FF46 EC2F	8
Power-down mode wake-up source status register 4	PDWSSR4	H'00	H'FF46 EC33	8
Power-down mode wake-up source status register 5	PDWSSR5	H'00	H'FF46 EC37	8
Power-down mode wake-up source edge select register 0	PDWSESL0	H'0A	H'FF46 EC3B	8
Power-down mode wake-up source edge select register 1	PDWSESL1	H'00	H'FF46 EC3F	8

Note: * This function is not available in the SH72A0 Group.

Table 5.7 Clock Registers (2)

Register Name	Symbol	After Reset	Address	Access Size
Power-down mode wake-up source edge select register 2	PDWSESL2	H'00	H'FF46 EC43	8
Reset status register 0	RSTSR0	H'00	H'FF46 EC47	8
Backup register 0	BUR0	Undefined	H'FF46 ED03	8
Backup register 1	BUR1	Undefined	H'FF46 ED07	8
Backup register 2	BUR2	Undefined	H'FF46 ED0B	8
Backup register 3	BUR3	Undefined	H'FF46 ED0F	8
Backup register 4	BUR4	Undefined	H'FF46 ED13	8
Backup register 5	BUR5	Undefined	H'FF46 ED17	8
Backup register 6	BUR6	Undefined	H'FF46 ED1B	8
Backup register 7	BUR7	Undefined	H'FF46 ED1F	8
Backup register 8	BUR8	Undefined	H'FF46 ED23	8
Backup register 9	BUR9	Undefined	H'FF46 ED27	8
Backup register 10	BUR10	Undefined	H'FF46 ED2B	8
Backup register 11	BUR11	Undefined	H'FF46 ED2F	8
Backup register 12	BUR12	Undefined	H'FF46 ED33	8
Backup register 13	BUR13	Undefined	H'FF46 ED37	8
Backup register 14	BUR14	Undefined	H'FF46 ED3B	8
Backup register 15	BUR15	Undefined	H'FF46 ED3F	8
Backup register 16	BUR16	Undefined	H'FF46 ED43	8
Backup register 17	BUR17	Undefined	H'FF46 ED47	8
Backup register 18	BUR18	Undefined	H'FF46 ED4B	8
Backup register 19	BUR19	Undefined	H'FF46 ED4F	8
Backup register 20	BUR20	Undefined	H'FF46 ED53	8
Backup register 21	BUR21	Undefined	H'FF46 ED57	8
Backup register 22	BUR22	Undefined	H'FF46 ED5B	8
Backup register 23	BUR23	Undefined	H'FF46 ED5F	8
Backup register 24	BUR24	Undefined	H'FF46 ED63	8
Backup register 25	BUR25	Undefined	H'FF46 ED67	8
Backup register 26	BUR26	Undefined	H'FF46 ED6B	8
Backup register 27	BUR27	Undefined	H'FF46 ED6F	8
Backup register 28	BUR28	Undefined	H'FF46 ED73	8
Backup register 29	BUR29	Undefined	H'FF46 ED77	8
Backup register 30	BUR30	Undefined	H'FF46 ED7B	8
Backup register 31	BUR31	Undefined	H'FF46 ED7F	8

5.2.1 System Protect Register 0 (SPR0)

Address H'FF46 E063

	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	0	0	0	0	0	0	0	0

Bit	Description	R/W
b7 to b0	When written B'1111 0001: Protection unlocked Other than B'1111 0001: Protection locked When read Bit 0 (b0) 0: Protection locked 1: Protection unlocked Bit 7 to bit 1 (b7 to b1) are always read as 0.	R/W

The SPR0 register is used to set the protect function that protects registers LOCR, SLCR0, VMCR, VD1LSL, RAMECC, RAMACYC, and AUDEN from being rewritten easily. To change the values of these registers, perform the following procedure:

(1) Write H'F1 to the SPR0 register (writing to the registers enabled).
(2) Change the values of registers LOCR, SLCR0, VMCR, VD1LSL, RAMECC, RAMACYC, and AUDEN.
(3) Write a value other than H'F1 to the SPR0 register (writing to the registers disabled).

5.2.2 System Protect Register 2 (SPR2)

Address H'FF46 E263

	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	0	0	0	0	0	0	0	0

Bit	Description	R/W
b7 to b0	When written B'1111 0001: Protection unlocked Other than B'1111 0001: Protection locked When read Bit 0 (b0) 0: Protection locked 1: Protection unlocked Bit 7 to bit 1 (b7 to b1) are always read as 0.	R/W

The SPR2 register is used to set the protect function that protects registers SWRR, RSDR, SLCR1, PDWSSLi (i = 0, 2 to 5), PDWSSRi, PDWSESLj (j = 0 to 2), RSTSR0, and BURm (m = 0 to 31) from being rewritten easily. To change the values of these registers, perform the following procedure:

(1) Write H'F1 to the SPR2 register (writing to the registers enabled).
(2) Change the values of registers SWRR, RSDR, SLCR1, PDWSSLi, PDWSSRi, PDWSESLj, RSTSR0, and BURm.
(3) Write a value other than H'F1 to the SPR2 register (writing to the registers disabled).

5.2.3 Low-Speed On-Chip Oscillator Control Register (LOCR)

Address H'FF46 E826

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	LSOON
After Reset	0	0	0	0	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R
b0	LSOON	Low-Speed On-Chip Oscillator Start Bit	0: Stops 1: Oscillation starts	R/W

This register is used to control the low-speed on-chip oscillator. To change the value of the LOCR register, set the SPR0 register to unlock the protection beforehand.

LSOON Bit

When the LSOON bit is set to 1, the low-speed on-chip oscillator starts oscillating. After reset, this bit is set to 1 (low-speed on-chip oscillator starts oscillating).

5.2.4 Main Clock Monitor Function Control Register (MCMCR)

Address H'FF46 E838

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	LXDF	LXIE	XME
After Reset *	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R
b2	LXDF	Main Clock Oscillation Stop Detect Flag	0: Not detected 1: Detected	R/W
b1	LXIE	Main Clock Oscillation Stop Detection Interrupt Enable Bit	0: Main clock stop detection interrupt disabled 1: Main clock stop detection interrupt enabled	R/W
b0	XME	Main Clock Monitor Function Enable Bit	0: Main clock monitor function not used 1: Main clock monitor function used	R/W

Note: * If a power-down mode wake-up reset is used for reset operation from power-down mode 0, this register retains the value in power-down mode 0.

This register is used to control the main clock monitor function.

LXDF Bit

The LXDF flag becomes 1 when the stop of the main clock oscillation is detected. This flag can be set to 0 by a program, but cannot be set to 1. If the RESET# signal goes to the high level while oscillation of the main clock is not stable, this bit will have the value 1 after reset.

LXIE Bit

If the XME bit is 1 and the LXIE bit is 1, a main clock oscillation stop detection interrupt is generated when the stop of the main clock oscillation is detected. At this time, the system clock is automatically switched to the free-running PLL clock.

XME Bit

The XME bit is used to control the main clock monitor function. After reset, this bit is set to 0 (monitor function not used).

5.2.5 Clock Protect Register (CPR)

Address H'FF46 E860

	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	0	0	0	0	0	0	0	0

Bit	Description	R/W
b7 to b0	When written B'1111 0001: Protection unlocked Other than B'1111 0001: Protection locked When read Bit 0 (b0) 0: Protection locked 1: Protection unlocked Bit 7 to bit 1 (b7 to b1) are always read as 0.	R/W

The CPR register is used to set the protect function that protects registers CCDR, PBCCR, COCR, PBCDR, LINCCR, and CMTCCR from being rewritten easily. To change the values of registers CCDR, PBCCR, COCR, PBCDR, LINCCR, and CMTCCR, perform the following procedure:

(1) Write H'F1 to the CPR register (writing to the registers enabled).
(2) Change the values of registers CCDR, PBCCR, COCR, PBCDR, LINCCR, and CMTCCR.
(3) Write a value other than H'F1 to the CPR register (writing to the registers disabled).

5.2.6 CPU Clock Divide Register (CCDR)

Address H'FF46 E804

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	BUSD[1:0]		—	—	CPUD[1:0]	
After Reset	0	0	1	0	0	0	1	0

Bit	Symbol	Bit Name	Description	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R
b5, b4	BUSD[1:0]	Bus Clock Division Ratio Select Bits	b5 b4 0 0 : Divided by 1 0 1 : Divided by 2 1 0 : Divided by 4 1 1 : Divided by 8 Do not set other than the above.	R/W
b3, b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b1, b0	CPUD[1:0]	CPU Clock Division Ratio Select Bits	b1 b0 0 0 : Divided by 1 0 1 : Divided by 2 1 0 : Divided by 4 1 1 : Divided by 8 Do not set other than the above.	R/W

This register is used to set the CPU clock and the bus clock. To change the value of the CCDR register, set the CPR register to unlock the protection beforehand.

When setting the CCDR register, execute a CCDR register read instruction immediately after a CCDR register write instruction.

BUSD Bits

These bits are used to set a division ratio of the bus clock. The system clock divided by the BUSD bits is used as the bus clock. After reset, the BUSD bits are set to B'10 (divided by 4). Set the BUSD bits to set the bus clock to the same frequency as the CPU clock or the CPU clock divided by 2. Also set the frequency of the bus clock to 100 MHz or below.

CPUD Bits

These bits are used to set a division ratio of the CPU clock. The system clock divided by the CPUD bits is used as the CPU clock. After reset, the CPUD bits are set to B'10 (divided by 4).

5.2.7 Peripheral Bus Clock Divide Register (PBCDR)

Address H'FF46 E808

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	PBBD[1:0]		—	—	PBAD[1:0]		—	—	PBCD[1:0]		—	—	—	—
After Reset	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MTUD[1:0]	
After Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

Bit	Symbol	Bit Name	Description	R/W
b31, b30	—	Reserved	These bits are read as 0. The write value should be 0.	R
b29, b28	PBBD[1:0]	Peripheral Bus Clock B Division Ratio Select Bits	b29b28 0 1 : Divided by 2 1 0 : Divided by 4 1 1 : Divided by 8 Do not set other than the above.	R/W
b27, b26	—	Reserved	These bits are read as 0. The write value should be 0.	R
b25, b24	PBAD[1:0]	Peripheral Bus Clock A Division Ratio Select Bits	b25b24 0 1 : Divided by 2 1 0 : Divided by 4 1 1 : Divided by 8 Do not set other than the above.	R/W
b23, b22	—	Reserved	These bits are read as 0. The write value should be 0.	R
b21, b20	PBCD[1:0]	Peripheral Bus Clock C Division Ratio Select Bits (SCI operating clock division ratio select bits)	b21b20 0 1 : Divided by 2 1 0 : Divided by 4 1 1 : Divided by 8 Do not set other than the above.	R/W
b19, b18	—	Reserved	These bits are read as 0. The write value should be 0.	R
b17, b16	—	Reserved	These bits are read as 1. The write value should be 1.	R
b15 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b1, b0	MTUD[1:0]	MTU Operating Clock Division Ratio Select Bits	b1 b0 0 0 : Divided by 1 0 1 : Divided by 2 1 0 : Divided by 4 1 1 : Divided by 8 Do not set other than the above.	R/W

This register is used to set a division ratio of the peripheral function bus clock. This register can be accessed in longword, word, or byte units. To change the value of the PBCDR register, set the CPR register to unlock the protection beforehand.

When setting the PBCDR register, execute a PBCDR register read instruction immediately after a PBCDR register write instruction.

PBBD Bits

These bits are used to set a division ratio of peripheral bus clock B (f(PBB)). The system clock divided by the PBBD bits is used as peripheral bus clock B. Set the PBBD bits to set peripheral bus clock B to the bus clock divided by 2. Also set the frequency of peripheral bus clock B to 50 MHz or below.

PBAD Bits

These bits are used to set a division ratio of peripheral bus clock A (f(PBA)). The system clock divided by the PBAD bits is used as peripheral bus clock A. Set the PBAD bits to set peripheral bus clock A to the bus clock divided by 2, and the same frequency as the MTU operating clock (f(MTU)) or the MTU operating clock divided by 2. Also set the frequency of peripheral bus clock A to 50 MHz or below.

PBCD Bits

These bits are used to set a division ratio of peripheral bus clock C (f(PBC)). The system clock divided by the PBCD bits is used as peripheral bus clock C. Set the PBCD bits to set peripheral bus clock C to the bus clock divided by 2. Also set the frequency of peripheral bus clock C to 50 MHz or below.
To access the SCI peripheral function register, set peripheral clock A and the peripheral bus clock C to the same frequency.

MTUD Bits

These bits are used to set a division ratio of the MTU operating clock (f(MTU)). The system clock divided by the MTUD bits is used as the MTU operating clock. Set the frequency of the MTU operating clock to 100 MHz or below.

5.2.8 Peripheral Bus Clock Control Register (PBCCR)

Address H'FF46 E81C

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	STPADC	STPCMT	—	STPLIN	—	—	—	—	—	—	—	—	—	—	—
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	—	Reserved	This bit is read as 0. The write value should be 0.	R
b14	STPADC	ADC Bus Clock Stop Bit	Controls the supply or stop of the ADC bus clock 0: Clock supplied 1: Clock stopped	R/W
b13	STPCMT	CMT Bus Clock Stop Bit	Controls the supply or stop of the CMT bus clock 0: Clock supplied 1: Clock stopped	R/W
b12	—	Reserved	This bit is read as 0. The write value should be 0.	R
b11	STPLIN	LIN Bus Clock Stop Bit	Controls the supply or stop of the LIN bus clock 0: Clock supplied 1: Clock stopped	R/W
b10 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

Note: * If a power-down mode wake-up reset is used for reset operation from power-down mode 0 or 1, this register retains the value in power-down mode 0 or 1.

This register is used to control the supply or stop of the peripheral bus clock. To change the value of the PBCCR register, set the CPR register to unlock the protection beforehand.
When setting the PBCCR register, execute a PBCCR register read instruction immediately after a PBCCR register write instruction.

STPADC Bit

This bit is used to control the supply or stop of the peripheral bus clock for the A/D converter. When the A/D converter function is not used, setting the STPADC bit to 1 stops the clock supply, decreasing the power consumption.
When the STPADC bit is set to 1, do not access the peripheral function registers of the A/D converter.

STPCMT Bit

This bit is used to control the supply or stop of the peripheral bus clock for the CMT. When the CMT function is not used, setting the STPCMT bit to 1 stops the clock supply, decreasing the power consumption.
When the STPCMT bit is set to 1, do not access the peripheral function registers of the CMT.

STPLIN Bit

This bit is used to control the supply or stop of the peripheral bus clock for the LIN. When the LIN function is not used, setting the STPLIN bit to 1 stops the clock supply, decreasing the power consumption.
When the STPLIN bit is set to 1, do not access the peripheral function registers of the LIN.

5.2.9 LIN Clock Control Register (LINCCR)

Address H'FF46 E811

	b7	b6	b5	b4	b3	b2	b1	b0
	LINCKON	LINCKPD	—	—	LINCKD [3:0]			
After Reset *	0	0	0	0	0	0	1	0

Bit	Symbol	Bit Name	Description	R/W
b7	LINCKON	LIN Communication Clock Source Oscillation Start Bit	0: Stops 1: Oscillation starts	R/W
b6	LINCKPD	LIN Communication Clock Source Division Ratio Control Bit	Selects divided by 2n or 16n for the LIN clock.	R/W
b5, b4	—	Reserved	These bits are read as 0. The write value should be 0.	R
b3 to b0	LINCKD [3:0]	LIN Communication Clock Source Division Ratio Select Bits	[LINCKPD bit is 0] b3 b2 b1 b0 0 0 0 0 : Setting prohibited 0 0 0 1 : Divided by 2 0 0 1 0 : Divided by 4 0 0 1 1 : Divided by 6 : 1 1 1 0 : Divided by 28 1 1 1 1 : Divided by 30 [LINCKPD bit is 1] b3 b2 b1 b0 0 0 0 0 : Setting prohibited 0 0 0 1 : Divided by 16 0 0 1 0 : Divided by 32 0 0 1 1 : Divided by 48 : 1 1 1 0 : Divided by 224 1 1 1 1 : Divided by 240	R/W

Note: * If a power-down mode wake-up reset is used for reset operation from power-down mode 0, this register retains the value in power-down mode 0.

This register is used to control the LIN clock. The LIN communication clock source (f(LIN)) is used as the operating clock for the LIN. To change the value of the LINCCR register, set the CPR register to unlock the protection beforehand.

When setting the LINCCR register, execute a LINCCR register read instruction immediately after a LINCCR register write instruction.

LINCKON Bit

When the LINCKON bit is set to 1, the LIN communication clock source starts oscillating. To oscillate the LIN communication clock source, set bits LINCKD and LINCKPD and then set the LINCKON bit to 1. After reset, the LIN communication clock source is stopped.

LINCKPD Bit

This bit is used to select either divided by 2n or 16n as the setting value of the LINCKD bits. When this bit is set to 0, divided by 2n is selected. When this bit is set to 1, divided by 16n is selected.

LINCKD Bits

These bits are used to set a division ratio of the LIN communication clock source (f(LIN)). The system clock divided by the LINCKD bits is used as the LIN communication clock source. Set the frequency of the LIN communication clock source to 50 MHz or below.

5.2.10 CMT Clock Control Register (CMTCCR)

Address H'FF46 E812

	b7	b6	b5	b4	b3	b2	b1	b0
	CMTCKON	CMTCKPD	—	—	CMTCKD [3:0]			
After Reset *	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	CMTCKON	CMT Count Source Origin Oscillation Start Bit	0: Stops 1: Oscillation starts	R/W
b6	CMTCKPD	CMT Count Source Origin Division Ratio Control Bit	Selects divided by 2n or 16n for the CMT count source origin.	R/W
b5, b4	—	Reserved	These bits are read as 0. The write value should be 0.	R
b3 to b0	CMTCKD [3:0]	CMT Count Source Origin Division Ratio Select Bits	[CMTCKPD bit is 0] b3 b2 b1 b0 0 0 0 0 : Divided by 1 0 0 0 1 : Divided by 2 0 0 1 0 : Divided by 4 0 0 1 1 : Divided by 6 : 1 1 1 0 : Divided by 28 1 1 1 1 : Divided by 30 [CMTCKPD bit is 1] b3 b2 b1 b0 0 0 0 0 : Divided by 1 0 0 0 1 : Divided by 16 0 0 1 0 : Divided by 32 0 0 1 1 : Divided by 48 : 1 1 1 0 : Divided by 224 1 1 1 1 : Divided by 240	R/W

Note: * If a power-down mode wake-up reset is used for reset operation from power-down mode 0, this register retains the value in power-down mode 0.

This register is used to control the CMT clock source origin. The CMT count source origin is used as the count source for compare match timer CMT0. To change the value of the CMTCCR register, set the CPR register to unlock the protection beforehand.

When setting the CMTCCR register, execute a CMTCCR register read instruction immediately after a CMTCCR register write instruction.

CMTCKON Bit

When the CMTCKON bit is set to 1, f(CMT) starts oscillating. To oscillate f(CMT), set bits CMTCKD, CMTCKPD, and CMTCKS, and then set the CMTCKON bit to 1. After reset, f(CMT) is stopped.

CMTCKPD Bit

This bit is used to select either divided by 2n or 16n as the setting value of the CMTCKD bits. When this bit is set to 0, divided by 2n is selected. When this bit is set to 1, divided by 16n is selected.

CMTCKD Bits

These bits are used to set a division ratio of the CMT clock. The low-speed on-chip oscillator clock divided by these CMTCKD bits is used as the CMT count source origin. Set the frequency of the CMT clock source origin to be slower than the peripheral function bus clock A frequency divided by 3. Also set the frequency of f(CMT) to 50 MHz or below.

5.2.11 INT Input Digital Filter Control Register 0 (INTDFCR0)

Address H'FF46 E890

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	INT13 FLTON	INT12 FLTON	INT11 FLTON	INT10 FLTON	INT9 FLTON	INT8 FLTON	INT7 FLTON	INT6 FLTON	INT5 FLTON	INT4 FLTON	INT3 FLTON	INT2 FLTON	INT1 FLTON	INT0 FLTON
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, b14	—	Reserved	These bits are read as 0. The write value should be 0.	R
b13	INT13 FLTON*	INT13 Pin Digital Filter Enable Bit	0: Digital filter disabled 1: Digital filter enabled	R/W
b12	INT12 FLTON*	INT12 Pin Digital Filter Enable Bit	0: Digital filter disabled 1: Digital filter enabled	R/W
b11	INT11 FLTON*	INT11 Pin Digital Filter Enable Bit	0: Digital filter disabled 1: Digital filter enabled	R/W
b10	INT10 FLTON	INT10 Pin Digital Filter Enable Bit	0: Digital filter disabled 1: Digital filter enabled	R/W
b9	INT9 FLTON*	INT9 Pin Digital Filter Enable Bit	0: Digital filter disabled 1: Digital filter enabled	R/W
b8	INT8 FLTON	INT8 Pin Digital Filter Enable Bit	0: Digital filter disabled 1: Digital filter enabled	R/W
b7	INT7 FLTON	INT7 Pin Digital Filter Enable Bit	0: Digital filter disabled 1: Digital filter enabled	R/W
b6	INT6 FLTON	INT6 Pin Digital Filter Enable Bit	0: Digital filter disabled 1: Digital filter enabled	R/W
b5	INT5 FLTON	INT5 Pin Digital Filter Enable Bit	0: Digital filter disabled 1: Digital filter enabled	R/W
b4	INT4 FLTON	INT4 Pin Digital Filter Enable Bit	0: Digital filter disabled 1: Digital filter enabled	R/W
b3	INT3 FLTON	INT3 Pin Digital Filter Enable Bit	0: Digital filter disabled 1: Digital filter enabled	R/W
b2	INT2 FLTON	INT2 Pin Digital Filter Enable Bit	0: Digital filter disabled 1: Digital filter enabled	R/W
b1	INT1 FLTON	INT1 Pin Digital Filter Enable Bit	0: Digital filter disabled 1: Digital filter enabled	R/W
b0	INT0 FLTON	INT0 Pin Digital Filter Enable Bit	0: Digital filter disabled 1: Digital filter enabled	R/W

Note: * INT13, INT12, INT11, and INT9 pins are not usable on the SH72A0.

This register is used to switch the digital filter enabled or disabled for the INTn pin. Enabled or disabled can be set individually for pins INT0 to INT13.

When any of the INTn pins is used to enable the digital filter (the INTDFCR0 register is set to a value other than H'0000), the supply of a clock with its division ratio set by the INTDFCR1 register is started.

5.2.12 INT Input Digital Filter Control Register 1 (INTDFCR1)

Address H'FF46 E892

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	INTFCKD [2:0]			—	—	—	—	—	—	—	—
After Reset	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b11	—	Reserved	These bits are read as 0. The write value should be 0.	R
b10 to b8	INTFCKD [2:0]	INTn Pin Digital Filter Clock Division Ratio Select Bits	b10 b9 b8 0 1 1 : Divided by 8 1 0 0 : Divided by 16 1 0 1 : Divided by 32 1 1 0 : Divided by 64 Do not set other than the above.	R/W
b7 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

This register is used to control the digital filter clock (f(INTF)) for the INTn pin.

INTFCKD Bits

These bits are used to set a division ratio of the digital filter clock for the INTn pin. Peripheral bus clock A divided by the INTFCKD bits is used as the digital filter clock for the INTn pin.

5.2.13 Sleep Mode Control Register 0 (SLCR0)

Address H'FF46 E840

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SSBY	—	—	STS[4:0]					—	—	—	—	—	—	—	—
After Reset *	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15	SSBY	Standby Mode Bit	0: Transition to CPU sleep mode when the SLEEP instruction is executed 1: Transition to standby mode or power-down mode when the SLEEP instruction is executed	R/W
b14, b13	—	Reserved	These bits are read as 0. The write value should be 0.	R
b12 to b8	STS[4:0]	Standby Timer Select Bits	b12b11b10b9b8 0 0 1 01 : Waiting time = 64 states 0 0 1 10 : Waiting time = 512 states 0 0 1 11 : Waiting time = 1024 states 0 1 0 00 : Waiting time = 2048 states 0 1 0 01 : Waiting time = 4096 states 0 1 0 10 : Waiting time = 16384 states 0 1 0 11 : Waiting time = 32768 states 0 1 1 00 : Waiting time = 65536 states 0 1 1 01 : Waiting time = 131072 states 0 1 1 10 : Waiting time = 262144 states 0 1 1 11 : Waiting time = 524288 states Do not set other than the above.	
b7 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

Note: * If a power-down mode wake-up reset is used for reset operation from power-down mode 0, this register retains the value in power-down mode 0.

To change the value of the SLCR0 register, set the SPR0 register to unlock the protection beforehand.

SSBY Bit

This bit is used to set a mode to be transited after the SLEEP instruction is executed. When the SLEEP instruction is executed with the SSBY bit set to 0, the mode transits to CPU sleep mode. When the SLEEP instruction is executed with the SSBY bit set to 1, the mode transits to standby mode or power-down mode.

Each mode of sleep modes is set according to the combination of the SSBY bit and bits DPSBY and CMTCUT in the SLCR1 register. For details, see Table 5.3 “Power Control Specifications”.

STS[4:0] (Standby Timer Select) Bits

These bits are used to select a period of waiting for the clock signal to become stable when an external interrupt initiates release from standby mode. In the case of a crystal oscillator, select a value for waiting time according to the operating frequency that is at least as long as the entry for oscillation stabilization time in Table 5.10. In the case of an external clock, time for the PLL circuit to become stable is required. Set the waiting time with reference to Table 5.10. Counting during the oscillation stabilization time is at the frequency of peripheral bus clock A (f(PBA)).

5.2.14 Clock Output Function Control Register (COCR)

Address H'FF46 E800

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	COD[1:0]		—	—	—	COE
After Reset *	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R
b5, b4	COD[1:0]	Clock Output Division Ratio Select Bits	b5 b4 0 0 : Divided by 1 0 1 : Divided by 2 1 0 : Divided by 4 1 1 : Divided by 8	R/W
b3 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R
b0	COE	Clock Output Enable Bit	0: Clock not output 1: Clock output	R/W

Note: This function is not available in the SH72A0 Group.
* If a power-down mode wake-up reset is used for reset operation from power-down mode 0, this register retains the value in power-down mode 0.

To change the value of the COCR register, set the CPR register to unlock the protection beforehand.
When setting the COCR register, execute a COCR register read instruction immediately after a COCR register write instruction.

COD Bits

The selected clock source is divided by the COD bits to output from the CLKOUT pin. Set the frequency of the clock signal output from the CLKOUT pin to 25 MHz or below.

COE Bit

When the COE bit is set to 1, the bus clock (f(BUS)) divided by the COD bits is output from the CLKOUT pin.

5.2.15 Sleep Mode Control Register 1 (SLCR1)

Address H'FF46 EC03

	b7	b6	b5	b4	b3	b2	b1	b0
	DPSBY	IOKEEP	—	—	CMTCUT	—	—	—
After Reset *1	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	DPSBY	Power-Down Bit	Selects standby mode or power-down mode when the SLEEP instruction is executed. 0: Standby mode 1: Power-down mode*2	R/W
b6	IOKEEP	I/O Port Keep Bit	0: The retained state of I/O ports is released at the same time as power-down mode is released 1: The retained state of I/O ports is released when 0 is written to this bit after power-down mode is released	R/W
b5, b4	—	Reserved	These bits are read as 0. The write value should be 0.	R
b3	CMTCUT	CMT Cut Bit	0: Clock is supplied to CMT on transition to power-down mode 0 1: Clock is stopped to CMT on transition to power-down mode 1 Refer to the DPSBY bit for mode transitions.	R/W
b2 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

Notes: 1. If a power-down mode wake-up reset is used for reset operation from power-down mode 0 or 1, this register retains the value in power-down mode 0 or 1.
2. Check the value of the CMTCUT bit to see whether the mode is power-down mode 0 or 1.

To change the value of the SLCR1 register, set the SPR2 register to unlock the protection beforehand.

DPSBY Bit

When the SLEEP instruction is executed while the SSBY bit in the SLCR0 register is set to 1, the mode transits to standby mode. At this time, if the DPSBY bit is set to 1 and there is no source for releasing standby mode, the mode transits to power-down mode. When a power-down mode wake-up reset is triggered by an interrupt request, the DPSBY bit remains set to 1. Write 0 to clear this bit.

IOKEEP Bit

In power-down mode, I/O ports retain the same state as in standby mode. This bit is used to set the port state that retained in power-down mode to be kept retained or released.

Table 5.8 Settings to Place the CPU in CPU Sleep Mode, Standby Mode, Power-down Mode 0, and Power-down Mode 1

SSBY	DPSBY	CMTCUT	Mode
0	0	0	CPU sleep mode
		1	CPU sleep mode
	1	0	CPU sleep mode
		1	CPU sleep mode
1	0	0	Standby mode
		1	Standby mode
	1	0	Power-down mode 0
		1	Power-down mode 1

5.2.16 Power-Down Mode Wake-Up Source Select Register i (PDWSSLi) (i = 0, 2 to 5)

The PDWSSLi register is used to enable or disable an interrupt request for reset operation with a power-down mode wake-up reset. The setting of this register is not affected by enabling or disabling normal interrupt requests.
To change the value of the PDWSSLi register, set the SPR2 register to unlock the protection beforehand.
Follow the procedure below when making the setting of the PDWSSLi register to enable resetting as the source for wake-up from power-down modes. Deviating from this procedure creates the possibility of unintended edges of interrupt signals.

1. Enable input on one of the pins on which the reset to act as the trigger for wake-up from power-down mode is multiplexed by writing a value other than B'00 to the corresponding bits in registers PVSR0 to PVSR3.
2. Enable the reset to act as the trigger for wake-up from power-down mode by setting the corresponding bit in the PDWSSLi registers to 1.

For details on registers PVSR0 to PVSR3, see section 13.2.6, Input Threshold Value Select Register 0 (PVSR0) to section 13.2.9, Input Threshold Value Select Register 3 (PVSR3), in section 13, I/O Ports.

Address PDWSSL0: H'FF46 EC0B

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	DCAN 1IE	DCAN 0IE
After Reset *	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b1	DCAN1IE	CAN1 Wake-Up Interrupt Enable Bit	0: Disabled as power-down mode wake-up reset source 1: Enabled as power-down mode wake-up reset source	R/W
b0	DCAN0IE	CAN0 Wake-Up Interrupt Enable Bit		R/W

Note: * If a power-down mode wake-up reset is used for reset operation from power-down mode 0 or 1, this register retains the value in power-down mode 0 or 1.

.

Address PDWSSL2: H'FF46 EC13

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	DLIN3IE *2	DLIN2IE	—	—
After Reset *1	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R
b3	DLIN3IE*2	LIN3 Low Detection Interrupt Enable Bit	0: Disabled as power-down mode wake-up reset source 1: Enabled as power-down mode wake-up reset source	R/W
b2	DLIN2IE	LIN2 Low Detection Interrupt Enable Bit		R/W
b1, b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

Notes: 1. If a power-down mode wake-up reset is used for reset operation from power-down mode 0 or 1, this register retains the value in power-down mode 0 or 1.
2. LIN3 is not usable on the SH72A0.

Address PDWSSL3: H'FF46 EC17

	b7	b6	b5	b4	b3	b2	b1	b0
	DNMIIE	—	DCMTIE	DLVD 1IE	—	—	—	—
After Reset *1	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	DNMIIE	NMI Interrupt Enable Bit	0: Disabled as power-down mode wake-up reset source 1: Enabled as power-down mode wake-up reset source	R/W *2
b6	—	Reserved	This bit is read as 0. The write value should be 0.	R
b5	DCMTIE	CMT0 Interrupt Enable Bit	0: Disabled as power-down mode wake-up reset source 1: Enabled as power-down mode wake-up reset source	R/W
b4	DLVD1IE	Voltage Monitor Interrupt 1 Enable Bit		R/W
b3 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

Notes: 1. If a power-down mode wake-up reset is used for reset operation from power-down mode 0 or 1, this register retains the value in power-down mode 0 or 1.
2. The value 1 is only writable once. Subsequent write access is ineffective.

Address PDWSSL4: H'FF46 EC1B

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	DINT13IE*2	DINT12IE*2	DINT11IE*2	DINT10IE	DINT9IE*2	DINT8IE
After Reset *1	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R
b5	DINT13IE*2	INT13 Pin Interrupt Enable Bit	0: Disabled as power-down mode wake-up reset source 1: Enabled as power-down mode wake-up reset source	R/W
b4	DINT12IE*2	INT12 Pin Interrupt Enable Bit		R/W
b3	DINT11IE*2	INT11 Pin Interrupt Enable Bit		R/W
b2	DINT10IE	INT10 Pin Interrupt Enable Bit		R/W
b1	DINT9IE*2	INT9 Pin Interrupt Enable Bit		R/W
b0	DINT8IE	INT8 Pin Interrupt Enable Bit		R/W

Notes: 1. If a power-down mode wake-up reset is used for reset operation from power-down mode 0 or 1, this register retains the value in power-down mode 0 or 1.
2. INT13, INT12, INT11, and INT9 pins are not usable on the SH72A0.

Address PDWSSL5: H'FF46 EC1F

	b7	b6	b5	b4	b3	b2	b1	b0
	DINT7IE	DINT6IE	DINT5IE	DINT4IE	DINT3IE	DINT2IE	DINT1IE	DINT0IE
After Reset *	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	DINT7IE	INT7 Pin Interrupt Enable Bit	0: Disabled as power-down mode wake-up reset source 1: Enabled as power-down mode wake-up reset source	R/W
b6	DINT6IE	INT6 Pin Interrupt Enable Bit		R/W
b5	DINT5IE	INT5 Pin Interrupt Enable Bit		R/W
b4	DINT4IE	INT4 Pin Interrupt Enable Bit		R/W
b3	DINT3IE	INT3 Pin Interrupt Enable Bit		R/W
b2	DINT2IE	INT2 Pin Interrupt Enable Bit		R/W
b1	DINT1IE	INT1 Pin Interrupt Enable Bit		R/W
b0	DINT0IE	INT0 Pin Interrupt Enable Bit		R/W

Note: * If a power-down mode wake-up reset is used for reset operation from power-down mode 0 or 1, this register retains the value in power-down mode 0 or 1.

The PDWSSLi register is used to enable or disable an interrupt for releasing power-down mode.
This register is not initialized by an internal reset signal applied when releasing power-down mode.

5.2.17 Power-Down Mode Wake-Up Source Status Register i (PDWSSRi) (i = 0, 2 to 5)

The PDWSSRi register is used to identify the source used for wake-up with a power-down mode wake-up reset. The corresponding bit in the PDWSSRi register becomes 1 when a source enabled in the PDWSSLi register is generated. As the PDWSSRi register is enabled even not in power-down mode, clear the bits in this register before the transition to power-down mode.

To change the value of the PDWSSRi register, set the SPR2 register to unlock the protection beforehand.

Address PDWSSR0: H'FF46 EC23

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	DCAN 1IF	DCAN 0IF
After Reset *1	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b1	DCAN1IF	CAN1 Wake-Up Interrupt Flag	[Condition to become 1] When an interrupt request is generated while the corresponding bit in the PDWSSLi register is 1 [Condition to become 0] When 0 is written to after reading as 1	R/W *2
b0	DCAN0IF	CAN0 Wake-Up Interrupt Flag		R/W *2

Notes: 1. If a power-down mode wake-up reset is used for reset operation from power-down mode 0 or 1, this register retains the value in power-down mode 0 or 1.
2. The flag can be cleared only by writing 0 to it after reading it as 1.

Address PDWSSR2: H'FF46 EC2B

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	DLIN3IF *3	DLIN2IF	—	—
After Reset *1	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R
b3	DLIN3IF*3	LIN3 Low Detection Interrupt Flag	[Condition to become 1] When an interrupt request is generated while the corresponding bit in the PDWSSLi register is 1 [Condition to become 0] When 0 is written to after reading as 1	R/W *2
b2	DLIN2IF	LIN2 Low Detection Interrupt Flag		R/W *2
b1, b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

Notes: 1. If a power-down mode wake-up reset is used for reset operation from power-down mode 0 or 1, this register retains the value in power-down mode 0 or 1.
2. The flag can be cleared only by writing 0 to it after reading it as 1.
3. LIN3 is not usable on the SH72A0.

Address PDWSSR3: H'FF46 EC2F

	b7	b6	b5	b4	b3	b2	b1	b0
	DNMIIF	—	DCMTIF	DLVD 1IF	—	—	—	—
After Reset *1	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	DNMIIF	NMI Interrupt Flag	[Condition to become 1] When an interrupt request is generated while the corresponding bit in the PDWSSLi register is 1 [Condition to become 0] When 0 is written to after reading as 1	R/W *2
b6	—	Reserved	This bit is read as 0. The write value should be 0.	R
b5	DCMTIF	CMT0 Interrupt Flag	[Condition to become 1] When an interrupt request is generated while the corresponding bit in the PDWSSLi register is 1 [Condition to become 0] When 0 is written to after reading as 1	R/W *2
b4	DLVD1IF	Voltage Monitor Interrupt 1 Flag		R/W *2
b3 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

Notes: 1. If a power-down mode wake-up reset is used for reset operation from power-down mode 0 or 1, this register retains the value in power-down mode 0 or 1.
2. The flag can be cleared only by writing 0 to it after reading it as 1.

Address PDWSSR4: H'FF46 EC33

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	DINT13IF *3	DINT12IF *3	DINT11IF *3	DINT10IF	DINT9IF *3	DINT8IF
After Reset *1	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R
b5	DINT13IF *3	INT13 Pin Interrupt Flag	[Condition to become 1] When an interrupt request is generated while the corresponding bit in the PDWSSLi register is 1 [Condition to become 0] When 0 is written to after reading as 1	R/W *2
b4	DINT12IF *3	INT12 Pin Interrupt Flag		R/W *2
b3	DINT11IF *3	INT11 Pin Interrupt Flag		R/W *2
b2	DINT10IF	INT10 Pin Interrupt Flag		R/W *2
b1	DINT9IF *3	INT9 Pin Interrupt Flag		R/W *2
b0	DINT8IF	INT8 Pin Interrupt Flag		R/W *2

Notes: 1. If a power-down mode wake-up reset is used for reset operation from power-down mode 0 or 1, this register retains the value in power-down mode 0 or 1.
2. The flag can be cleared only by writing 0 to it after reading it as 1.
3. INT13, INT12, INT11, and INT9 pins are not usable on the SH72A0.

Address PDWSSR5: H'FF46 EC37

	b7	b6	b5	b4	b3	b2	b1	b0
	DINT7IF	DINT6IF	DINT5IF	DINT4IF	DINT3IF	DINT2IF	DINT1IF	DINT0IF
After Reset *1	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	DINT7IF	INT7 Pin Interrupt Flag	[Condition to become 1] When an interrupt request is generated while the corresponding bit in the PDWSSLi register is 1 [Condition to become 0] When 0 is written after reading as 1	R/W *2
b6	DINT6IF	INT6 Pin Interrupt Flag		R/W *2
b5	DINT5IF	INT5 Pin Interrupt Flag		R/W *2
b4	DINT4IF	INT4 Pin Interrupt Flag		R/W *2
b3	DINT3IF	INT3 Pin Interrupt Flag		R/W *2
b2	DINT2IF	INT2 Pin Interrupt Flag		R/W *2
b1	DINT1IF	INT1 Pin Interrupt Flag		R/W *2
b0	DINT0IF	INT0 Pin Interrupt Flag		R/W *2

Notes: 1. If a power-down mode wake-up reset is used for reset operation from power-down mode 0 or 1, this register retains the value in power-down mode 0 or 1.
2. The flag can be cleared only by writing 0 to it after reading it as 1.

5.2.18 Power-Down Mode Wake-Up Source Edge Select Register j (PDWSESLj) (j = 0 to 2)

The PDWSESLi register is used to select an edge of an interrupt used for a power-down mode wake-up reset.
To change the value of the PDWSESLj register, set the SPR2 register to unlock the protection beforehand.

Address PDWSESL0: H'FF46 EC3B

	b7	b6	b5	b4	b3	b2	b1	b0
	DNMIEG	—	—	—	—	—	—	—
After Reset *	0	0	0	0	1	0	1	0

Bit	Symbol	Bit Name	Description	R/W
b7	DNMIEG	NMI Interrupt Edge Select Bit	0: An interrupt request is generated at the falling edge of the NMI pin 1: An interrupt request is generated at the rising edge of the NMI pin	R/W
b6 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R
b3	—	Reserved	This bit is read as 1. The write value should be 1.	R
b2	—	Reserved	This bit is read as 0. The write value should be 0.	R
b1	—	Reserved	This bit is read as 1. The write value should be 1.	R
b0	—	Reserved	This bit is read as 0. The write value should be 0.	R

Note: * If a power-down mode wake-up reset is used for reset operation from power-down mode 0 or 1, this register retains the value in power-down mode 0 or 1.

Address PDWSESL1: H'FF46 EC3F

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	DINT13EG *2	DINT12EG *2	DINT11EG *2	DINT10EG	DINT9EG *2	DINT8EG
After Reset *1	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R
b5	DINT13EG *2	INT13 Pin Edge Select Bit	0: An interrupt request is generated at the falling edge of the INTi pin 1: An interrupt request is generated at the rising edge of the INTi pin	R/W
b4	DINT12EG *2	INT12 Pin Edge Select Bit		R/W
b3	DINT11EG *2	INT11 Pin Edge Select Bit		R/W
b2	DINT10EG	INT10 Pin Edge Select Bit		R/W
b1	DINT9EG *2	INT9 Pin Edge Select Bit		R/W
b0	DINT8EG	INT8 Pin Edge Select Bit		R/W

Notes: 1. If a power-down mode wake-up reset is used for reset operation from power-down mode 0 or 1, this register retains the value in power-down mode 0 or 1.
2. INT13, INT12, INT11, and INT9 pins are not usable on the SH72A0.

Address PDWSESL2: H'FF46 EC43

	b7	b6	b5	b4	b3	b2	b1	b0
	DINT7EG	DINT6EG	DINT5EG	DINT4EG	DINT3EG	DINT2EG	DINT1EG	DINT0EG
After Reset *	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	DINT7EG	INT7 Pin Edge Select Bit	0: An interrupt request is generated at the falling edge of the INTi pin 1: An interrupt request is generated at the rising edge of the INTi pin	R/W
b6	DINT6EG	INT6 Pin Edge Select Bit		R/W
b5	DINT5EG	INT5 Pin Edge Select Bit		R/W
b4	DINT4EG	INT4 Pin Edge Select Bit		R/W
b3	DINT3EG	INT3 Pin Edge Select Bit		R/W
b2	DINT2EG	INT2 Pin Edge Select Bit		R/W
b1	DINT1EG	INT1 Pin Edge Select Bit		R/W
b0	DINT0EG	INT0 Pin Edge Select Bit		R/W

Note: * If a power-down mode wake-up reset is used for reset operation from power-down mode 0 or 1, this register retains the value in power-down mode 0 or 1.

5.2.19 Reset Status Register 0 (RSTSR0)

The RSTSR0 register indicates the reset source for a power-down mode wake-up reset, and the voltage monitor (VCC falls: voltages to be monitored are Vdet1). To make a transition to power-down mode, clear each bit in this register beforehand.

To change the value of the RSTSR0 register, set the SPR2 register to unlock the protection beforehand.

Address RSTSR0: H'FF46 EC47

	b7	b6	b5	b4	b3	b2	b1	b0
	DPSBYF	—	—	—	—	—	LVD1F	—
After Reset *1	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	DPSBYF	Power-Down Reset Flag	[Condition to become 1] When a power-down mode wake-up reset is generated by an interrupt request [Condition to become 0] When 0 is written to after reading as 1	R/W *2
b6 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b1	LVD1F	LVD1 Flag	[Condition to become 1] When Vcc of Vdet1 or lower level is detected [Condition to become 0] When 0 is written to after reading as 1	R/W *2
b0	—	Reserved	This bit is read as 0. The write value should be 0.	R

Notes: 1. If a power-down mode wake-up reset is used for reset operation from power-down mode 0 or 1, this register retains the value in power-down mode 0 or 1.
2. The flag can be cleared only by writing 0 to it after reading it as 1.

DPSBYF Bit

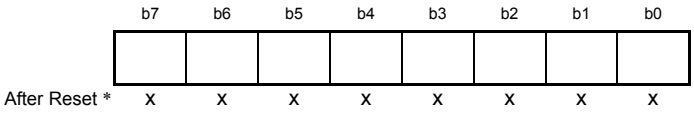
The DPSBYF bit indicates that a power-down mode wake-up reset has been generated.

LVD1F Bit

This bit indicates that a low voltage (VCC has reached Vdet1 or below) has been detected in the voltage monitor circuit.

5.2.20 Backup Register m (BURm) (m = 0 to 31)

Address: BUR0: H'FF46 ED03, BUR1: H'FF46 ED07, BUR2: H'FF46 ED0B, BUR3: H'FF46 ED0F,
BUR4: H'FF46 ED13, BUR5: H'FF46 ED17, BUR6: H'FF46 ED1B, BUR7: H'FF46 ED1F,
BUR8: H'FF46 ED23, BUR9: H'FF46 ED27, BUR10: H'FF46 ED2B, BUR11: H'FF46 ED2F,
BUR12: H'FF46 ED33, BUR13: H'FF46 ED37, BUR14: H'FF46 ED3B, BUR15: H'FF46 ED3F,
BUR16: H'FF46 ED43, BUR17: H'FF46 ED47, BUR18: H'FF46 ED4B, BUR19: H'FF46 ED4F,
BUR20: H'FF46 ED53, BUR21: H'FF46 ED57, BUR22: H'FF46 ED5B, BUR23: H'FF46 ED5F,
BUR24: H'FF46 ED63, BUR25: H'FF46 ED67, BUR26: H'FF46 ED6B, BUR27: H'FF46 ED6F,
BUR28: H'FF46 ED73, BUR29: H'FF46 ED77, BUR30: H'FF46 ED7B, BUR31: H'FF46 ED7F



Bit	Description	R/W
b7 to b0	This register is used to save data during power-down mode.	R/W

Note: * If a power-down mode wake-up reset is used for reset operation from power-down mode 0 or 1, this register retains the value in power-down mode 0 or 1.

The BURm register is a 1-byte writable/readable register that used to save data during power-down mode. The value of this register is retained even in power mode where data in the internal RAM is not retained.

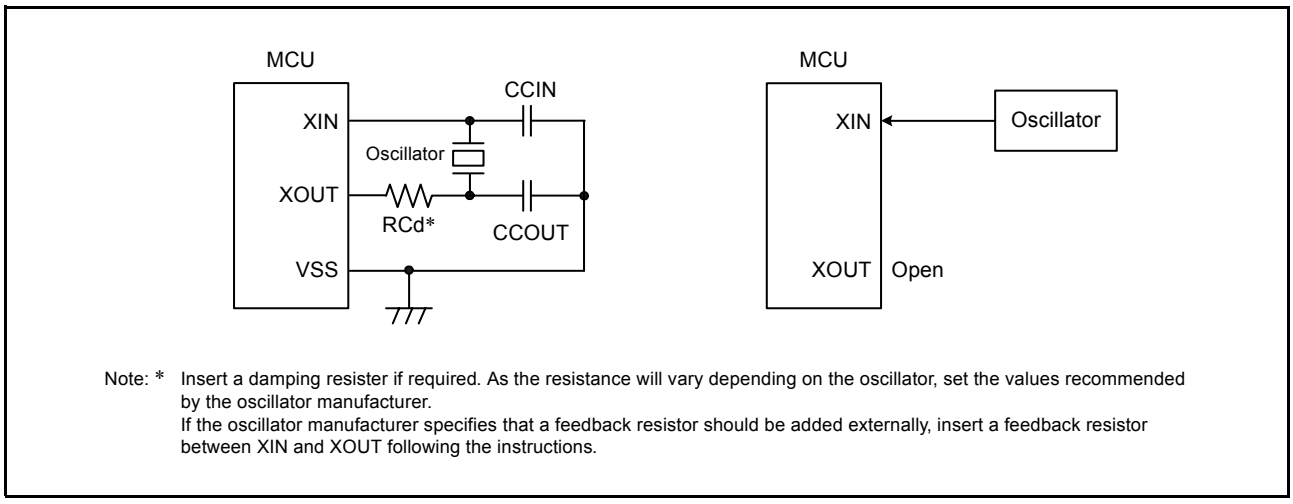
To change the value of the BURm register, set the SPR2 register to unlock the protection beforehand.

5.3 Clock Generators

5.3.1 Main Clock Oscillator

The main clock can be supplied by connecting an oscillator or inputting a clock to the XIN pin. Figure 5.4 shows an Example of Main Clock Connection Circuit.

The main clock oscillator integrates a feedback resistor. This resistor is disconnected from the oscillator in standby mode, power-down mode 0, or power-down mode 1 to decrease the power consumption. As the circuit constants will vary depending on the oscillator, set the values recommended by the oscillator manufacturer.



Note: * Insert a damping resister if required. As the resistance will vary depending on the oscillator, set the values recommended by the oscillator manufacturer.
If the oscillator manufacturer specifies that a feedback resistor should be added externally, insert a feedback resistor between XIN and XOUT following the instructions.

Figure 5.4 Example of Main Clock Connection Circuit

5.3.2 Low-Speed On-Chip Oscillator

The low-speed on-chip oscillator clock (f(LOCO)) at 125 kHz is supplied. After reset, this clock is oscillating.

5.3.3 PLL Frequency Synthesizer

The PLL clock (f(PLL)) is generated from the main clock.
The PLL clock is oscillating after reset.
The frequency of the PLL clock is calculated using the following formula:

PLL clock frequency (f(PLL)) = PLL multiplication ratio × f(XIN)
The PLL multiplication ratio is 8 in the SH72A0 group and 10 in the SH72A2 group.

5.4 System Clock (f(SYS))

The system clock is used as the clock source for the CPU clock, bus clock, peripheral bus clock, and peripheral function clock. The clock (f(PLL)) is used as the system clock.

5.4.1 CPU Clock (f(CPU))

The CPU clock is an operating clock for the CPU. The system clock divided by 1, 2, 4, or 8 is used as the CPU clock. After reset, the clock is set to the system clock divided by 4.

5.4.2 Bus Clock (f(BUS))

The system clock divided by 1, 2, 4, or 8 is used as the bus clock. Set the frequency of the bus clock to the same frequency as the CPU clock or divided by 2, and 80 MHz or below in the SH72A0 Group and 100 MHz or below in the SH72A2 Group. After reset, the bus clock is set to the system clock divided by 4.

5.4.3 Peripheral Bus Clocks

Peripheral clock A (f(PBA)), peripheral clock B (f(PBB)), peripheral clock C (f(PBC)), and MTU operating clock (f(MTU)) are peripheral bus clocks. These clocks are used for the peripheral functions. The system clock divided by 1, 2, 4, or 8 is used as a peripheral bus clock.

Set the frequency of f(PBA), f(PBB), and f(PBC) to the bus clock divided by 2 and 40 MHz or below in the SH72A0 Group and 50 MHz or below in the SH72A2 Group. After reset, the bus clock is set to divided by 2.

Set the frequency of f(MTU) to the same frequency as peripheral clock A or peripheral clock A multiplied by 2, and 80 MHz or below in the SH72A0 Group and 100 MHz or below in the SH72A2 Group. After reset, the f(MTU) is set to the same frequency as the bus clock.

The peripheral bus clock can be supplied or stopped to the peripheral functions through the setting of the PBCCR register. For details, refer to section 5.2.8, Peripheral Bus Clock Control Register (PBCCR).

5.4.4 Peripheral Function Clocks

The CMT count source origin (f(CMT)), LIN communication clock source (f(LIN)), WDT count source (f(LOCO)) are available as peripheral function clocks.

The CMT count source origin is used as the CMT0 count source. f(LOCO) divided by 2n or 16n is used as the CMT count source origin (n = 0 to 15; divided by 1 if n = 0). Set the frequency of the CMT count source origin to 40 MHz or below in the SH72A0 Group and 50 MHz or below in the SH72A2 Group. After reset, it is set to the same frequency as f(LOCO).

The LIN communication clock source (f(LIN)) is used as the operating clock for the LIN. The system clock divided by 2n or 16n is used as the LIN communication clock source (n = 1 to 15). Set the frequency of the LIN communication clock source to 40 MHz or below in the SH72A0 Group and 50 MHz or below in the SH72A2 Group. After reset, it is set to the system clock divided by 4.

The WDT count source (f(LOCO)) is used as the operating clock for the WDT.

5.4.5 Clock Divide Register Setting Procedure

The clock divide registers should be set according to the following procedure.

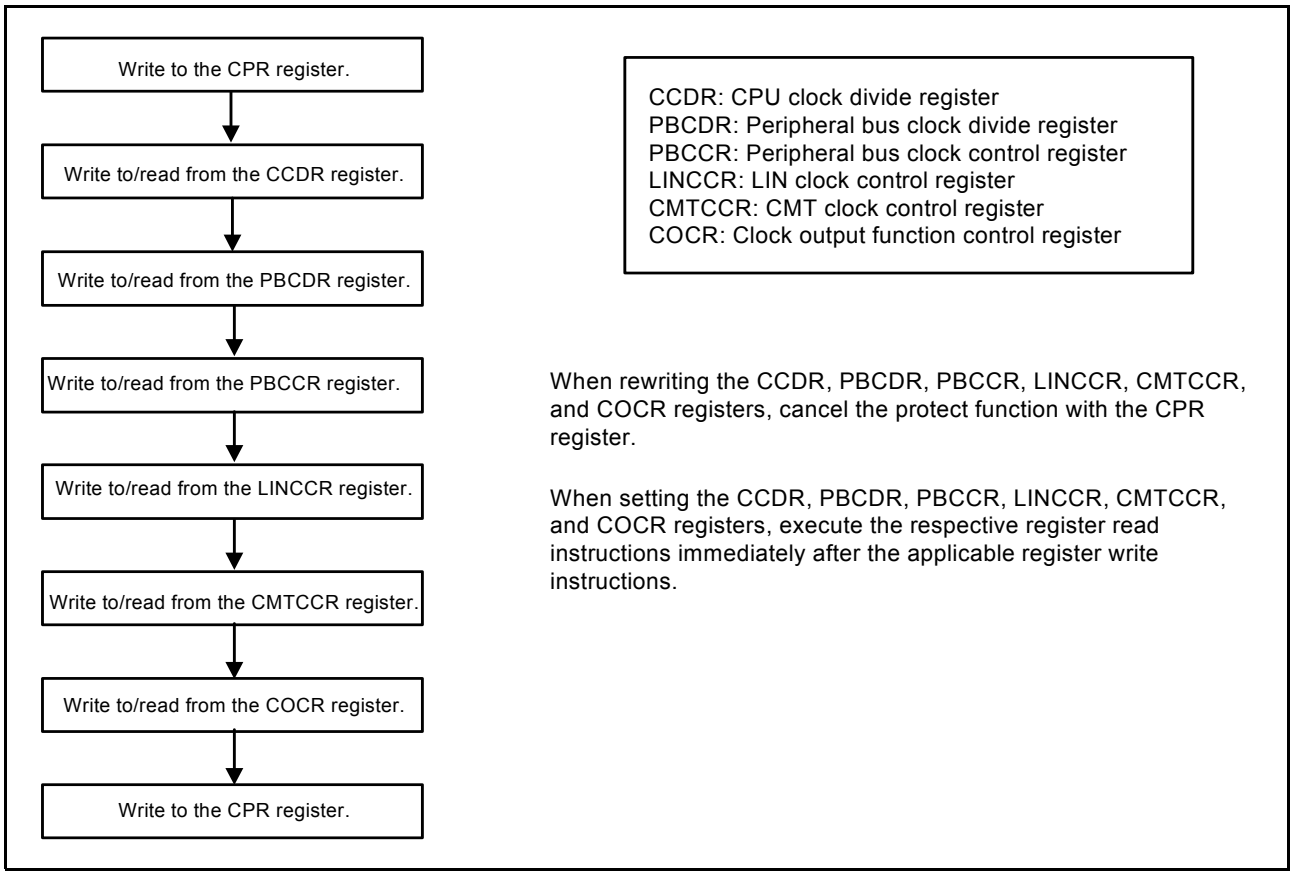


Figure 5.5 Clock Divide Register Setting Procedure

5.5 Power Control

Power consumption control is achieved by controlling the CPU clock frequency. The higher the frequency, the more the processing power increases. The lower the frequency, the more the power consumption decreases. Depending on the CPU operation, CPU run mode and sleep modes (CPU sleep mode, standby mode, power-down mode 0, and power-down mode 1) are available to control power consumption.

In CPU run mode, the CPU clock, bus clock, peripheral bus clock, and peripheral function clock are supplied.
In sleep mode, the supply of the CPU clock is stopped.

For details on specifications of the power control, see Table 5.3. Table 5.9 lists the Operating States for Each Mode. For details on mode transitions by CPU state, see Figure 5.3.

Table 5.9 Operating States for Each Mode

Operating State	CPU Sleep Mode	Standby Mode	Power-Down Mode 0 (CMT operation)	Power-Down Mode 1 (No CMT operation)
Main clock oscillator	Oscillating	Stopped	Stopped	Stopped
Low-speed on-chip oscillator	Stopped/Oscillating	Stopped/Oscillating	Oscillating	Stopped
PLL frequency synthesizer	Oscillating	Power supply shut off	Power supply shut off	Power supply shut off
System clock	f (PLL)/n	f(LOCO)	f(LOCO)	Stopped
ROM	Operating	Stopped	Power supply shut off	Power supply shut off
RAM	Operating	Stopped	Power supply shut off	Power supply shut off
EEPROM	Operating	Stopped	Power supply shut off (data retained)	Power supply shut off (data retained)
CPU	Stopped	Stopped	Power supply shut off	Power supply shut off
DMAC, task monitor timer, CRC calculator, SBI, SCI, A/D converter, TPU, MTU-III, debug-related	Operating	Stopped	Power supply shut off	Power supply shut off
CMT	Operating	Operating	Stopped (only CMT0 is operating)	Stopped
WDT	Operating	Stopped	Stopped	Stopped
Interrupt controller	Operating	Operating	Power supply shut off (only NMI and INT interrupts acceptable)	Power supply shut off (only NMI and INT interrupts acceptable)
CAN, LIN	Operating	Stopped	Power supply shut off (only wake-up interrupt acceptable)	Power supply shut off (only wake-up interrupt acceptable)
I/O ports	Operating	Retained	Retained	Retained
Wake-up sources from mode (reset sources)	• Hardware reset • Watchdog timer reset	• Hardware reset	• Hardware reset	• Hardware reset
Wake-up sources from mode (interrupt sources) *	• All interrupts except software interrupt	• CANi wake-up interrupt • LINi Low detection interrupt • NMI interrupt • INTi interrupt • Voltage monitor interrupt 1 • CMTi interrupt	• CANi wake-up interrupt request • LINi Low detection interrupt request • NMI interrupt request • INTi interrupt request • Voltage monitor interrupt 1 request • CMT0 interrupt request	• CANi wake-up interrupt request • LINi Low detection interrupt request • NMI interrupt request • INTi interrupt request • Voltage monitor interrupt 1 request

Note: * For wake-up from power-down mode 0 or power-down mode 1, an automatic reset (power-down wake-up reset) is used.

5.5.1 Modes

5.5.1.1 CPU Sleep Mode

(1) Transition to CPU Sleep Mode

When the SLEEP instruction is executed while the SSBY bit in the SLCR0 register is 0, the CPU enters CPU sleep mode. In CPU sleep mode, the CPU operation stops but the contents of the CPU internal registers are retained. The peripheral functions other than the CPU do not stop.

(2) CPU Sleep Mode State and Wake-UP from CPU Sleep Mode

For details on the CPU sleep mode operation state and wake-up sources, see Table 5.9.

[For wake-up using an interrupt source]

When an interrupt is generated, CPU sleep mode is released and interrupt exception handling starts. At this time, the CPU wakes up in the CPU run mode before the transition. If interrupts are disabled or interrupts other than NMI are masked in the CPU, the CPU does not wake up from the CPU sleep mode.

[For wake-up using a reset source]

The LSI enters the reset state and the CPU starts reset exception handling.

5.5.1.2 Standby Mode

(1) Transition to Standby Mode

To make a transition to standby mode, set beforehand the CPU, bus, and MTU operating clocks to the system clock divided by 4, and the peripheral bus clocks other than the MTU operating clock to the system clock divided by 8.
When the SLEEP instruction is executed while the SSBY bit in the SLCR0 register is 1 and the DPSBY bit in the SLCR1 register is 0, the mode transits to standby mode. Make the following settings for the watchdog timer before the transition to standby mode.

1. Set the CSS bits in the WDTCR0 register to B'11.
2. Set the TOPS bits in the WDTCR0 register to B'11.
3. Refresh the watchdog timer.

Restore the previous values to the watchdog timer after wake-up from standby mode.

(2) Standby Mode State and Wake-Up from Standby Mode

For details on the standby mode state and wake-up sources, see Table 5.9.

[For wake-up using an interrupt source]

When an interrupt is generated, standby mode is released and interrupt exception handling starts. At this time, the CPU wakes up in the CPU run mode before the transition. If interrupts are disabled or interrupts other than NMI are masked in the CPU, the CPU does not wake up from the standby mode.

[For wake-up using a reset source]

The LSI enters the reset state and the CPU starts reset exception handling.

(3) Setting the time for oscillation to become stable after release from standby mode

Set the STS[4:0] bits in the SLCR0 register as follows.

1. For a crystal oscillator
   Set the STS[4:0] bits for a waiting time that is at least as long as the oscillation stabilization time.
   Waiting times for the settings of the STS[4:0] bits and operating frequency are listed in Table 5.10.
2. For an external clock
   Time for the PLL circuit to become stable is required. Set the waiting time with reference to Table 5.10.

Table 5.10 Setting of Oscillation Stabilization Time

STS4	STS3	STS2	STS1	STS0	Waiting Time (State)	f (PBA)*1 [MHz]							Unit
						50	32	25	16	13	10	8	
0	0	0	0	0	(Reserved)	—	—	—	—	—	—	—	µs
				1	(Reserved)	—	—	—	—	—	—	—	
			1	0	(Reserved)	—	—	—	—	—	—	—	
				1	(Reserved)	—	—	—	—	—	—	—	
		1	0	0	(Reserved)	—	—	—	—	—	—	—	
				1	64	1.30	2.00	2.60	4.00	4.90	6.40	8.00	
			1	0	512	10.25	16.00	20.50	32.00	39.40	51.20	64.00	
				1	1024	20.50	32.00	41.00	64.00	78.80	102.40	128.00	
	1	0	0	0	2048	40.95	64.00	81.90	128.00	157.50	204.80	256.00	
				1	4096	0.08	0.13	0.16	0.26	0.32	0.41	0.51	ms
			1	0	16384	0.33	0.51	0.66	1.02	1.26 *2	1.64 *2	2.05 *2	
				1	32768	0.66	1.02	1.31 *2	2.05 *2	2.52	3.28	4.10	
		1	0	0	65536	1.31 *2	2.05 *2	2.62	4.10	5.04	6.55	8.19	
				1	131072	2.62	4.10	5.24	8.19	10.08 *3	13.11 *3	16.38 *3	
			1	0	262144	5.25	8.19	10.49 *3	16.38 *3	20.16	26.21	32.77	
				1	524288	10.49 *3	16.38 *3	20.97	40.33	40.33	52.43	65.54	
1	x	x	x	x	(Reserved)	—	—	—	—	—	—	—	

Notes: 1. f(PBA) is the frequency of the output from the frequency divider for the peripheral-module clock. The oscillation stabilization waiting time includes a period over which oscillation by the oscillator will not be stable, and this is affected by the characteristics of the oscillator. The values given above are for reference.
2. This interval is recommended if the external clock is in use.
3. This interval is recommended if a crystal oscillator is in use.

5.5.1.3 Power-Down Mode 0

(1) Transition to Power-Down Mode 0

To make a transition to power-down mode 0, set beforehand the CPU, bus, and MTU operating clocks to the system clock divided by 4, and the peripheral bus clocks other than the MTU operating clock to the system clock divided by 8. When the SLEEP instruction is executed while the SSBY bit in the SLCR0 register is 1, the DPSBY bit in the SLCR1 register is 1, and the CMTCUT bit in the SLCR1 register is 0, the mode transits to power-down mode 0.

Clear the PDWSSRi register before entering power-down mode 0. If a power-down wake-up reset is used for reset operation from power-down mode 0, the PDWSSRi register retains its value.

When transiting to power-down mode 0, the mode passes through standby mode internally. If there is an interrupt source for wake-up from standby mode during the transition, standby mode is released halfway and interrupt handling is started.

(2) Power-Down Mode 0 State and Wake-Up from Power-Down Mode 0

For details on the power-down mode 0 state and wake-up sources, see Table 5.9.

[For wake-up using an interrupt source (wake-up using a power-down mode wake-up reset)]

When an interrupt source for wake-up set in the PDWSSLi register is accepted, the corresponding bit in this register is set to 1, and the LSI automatically enters the reset state while the supply of the internal power is started. At this time, the DPSBYF flag in the RSTSR0 register becomes 1. The state of the I/O ports depends on the setting of the IOKEEP bit in the SLCR1 register. When the IOKEEP bit is set to 0, the retained state of the I/O ports is released at the same time as power-down mode is released. When the IOKEEP bit is set to 0, the state of the I/O ports is retained until 0 is written to this bit. The functions that have been running or stopped are not initialized. The other functions will be the same state as in reset handling.

[For wake-up using a reset source]

The LSI enters the reset state and the CPU starts reset exception handling.

5.5.1.4 Power-Down Mode 1

(1) Transition to Power-Down Mode 1

To make a transition to power-down mode 1, set beforehand the CPU, bus, and MTU operating clocks to the system clock divided by 4, and the peripheral bus clocks other than the MTU operating clock to the system clock divided by 8. When the SLEEP instruction is executed while the SSBY bit in the SLCR0 register is 1, the DPSBY bit in the SLCR1 register is 1, and the CMTCUT bit in the SLCR1 register is 1, the mode transits to power-down mode 1.

Clear the PDWSSRi register before entering power-down mode 1. If a power-down wake-up reset is used for reset operation from power-down mode 1, the PDWSSRi register retains its value.

When transiting to power-down mode 1, the mode passes through standby mode internally. If there is an interrupt source for wake-up from standby mode during the transition, standby mode is released halfway and interrupt handling is started.

(2) Power-Down Mode 1 State and Wake-Up from Power-Down Mode 1

For details on the power-down mode 1 state and wake-up sources, see Table 5.9.

[For wake-up using an interrupt source (wake-up using a power-down mode wake-up reset)]

When an interrupt source for wake-up set in the PDWSSLi register is accepted, the corresponding bit in this register is set to 1, and the LSI automatically enters the reset state while the supply of the internal power is started. At this time, the DPSBYF flag in the RSTSR0 register becomes 1. The state of the I/O ports depends on the setting of the IOKEEP bit in the SLCR1 register. When the IOKEEP bit is set to 0, the retained state of the I/O ports is released at the same time as power-down mode is released. When the IOKEEP bit is set to 1, the state of the I/O ports is retained until 0 is written to this bit. The functions that have been running or stopped are not initialized. The other functions will be the same state as in reset handling.

[For wake-up using a reset source]

The LSI enters the reset state and the CPU starts reset exception handling.

5.6 Main Clock Monitor Function

If the main clock is stopped due to external sources when the XME bit in the MCMCR register is 1, the system clock source is automatically switched to the free-running PLL clock.
When the main clock stops, the bits in the corresponding registers are set as follows:

- The LXDF flag in the MCMCR register = 1 (main clock oscillation stop detected)

The LXDF flag does not automatically become 0 once it has become 1. To set the LXDF flag to 0, set this flag to 0 by a program.

When the main clock stops while the LXIE bit is 1 (main clock oscillation stop detection interrupt enabled), a main clock oscillation stop detection interrupt request is generated. The main clock oscillation stop detection interrupt shares a vector with voltage monitor interrupt 1. When using the main clock oscillation stop detection interrupt and this interrupt at the same time, read the LXDF flag in the interrupt routine to confirm the main clock oscillation stop detection interrupt request has been generated.

This function is provided for cases when the main clock is stopped due to external sources.

5.7 Clock Output Function

When the COE bit in the COCR register is set to 1, a clock can be output from the CLKOUT pin. The clock to be output will be the bus clock (f(BUS)) divided by 1, 2, 4, or 8.
This function is not avaiable in the SH72A0 Group.

5.8 Usage Notes

5.8.1 Board Design Notes

Locate the crystal resonator and the load capacitors as close as possible to the XIN and XOUT pins. To assure that the circuit is not influenced by noise and operates correctly, do not allow the XIN pin and XOUT pin lines to cross any other signal lines as shown in figure 5.6.

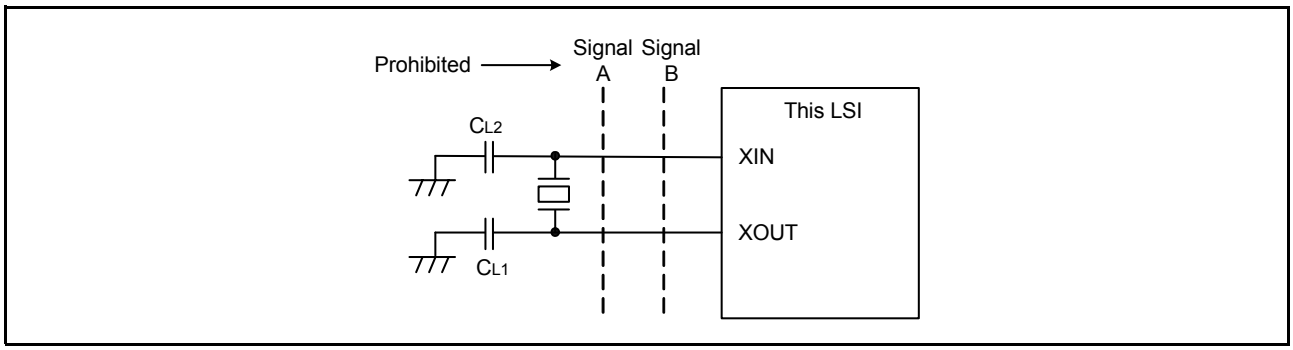


Figure 5.6 Notes on Oscillation Circuit Block Board Design

Figure 5.7 shows a connection example of the bypass capacitors. The Vcc and Vss pins must be isolated from the board power supply source. Also, the bypass capacitors CB must be inserted as close as possible to the pins. The capacitance of the bypass capacitors must be fully considered in user's system board design.

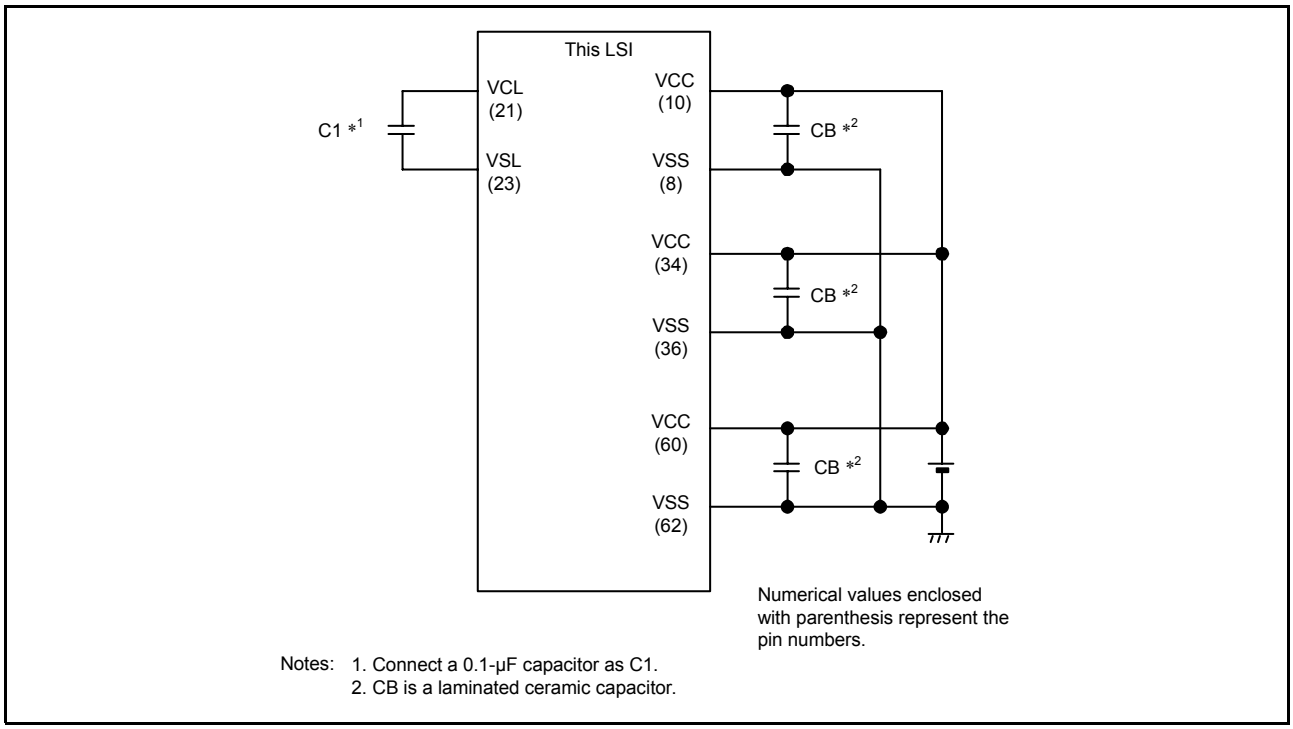


Figure 5.7 Connection Example of Bypass Capacitors (SH72A2 Group)

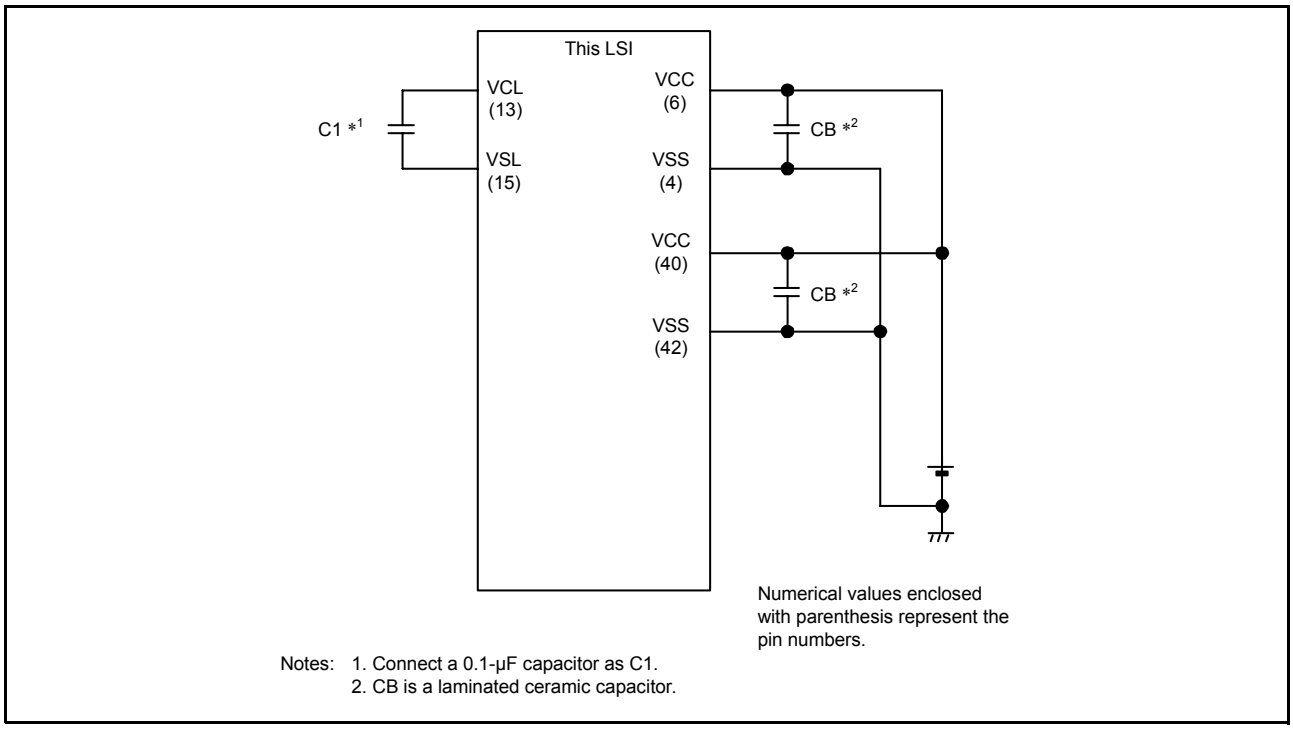


Figure 5.8 Connection Example of Bypass Capacitors (SH72A0 Group)

6. Exception Handling

6.1 Introduction

6.1.1 Types of Exception Handling and Priority

Exception handling is started by sources, such as a reset, address errors, register bank errors, interrupts, and instructions. When several exception sources occur at once, they are processed according to the priority shown below. Table 6.1 lists the Types of Exception Sources and Priority.

Table 6.1 Types of Exception Sources and Priority

Exception Source		Priority
Reset	Hardware reset, watchdog timer reset, software reset, power-down mode wake-up reset	High ▲
Address error	CPU address error	
CPU operand-access MPU error	CPU operand-access MPU error	
Instruction	FPU exception	
	Integer division exception (division by zero)	
	Integer division exception (overflow)	
Register bank error	Bank underflow	
	Bank overflow	
Interrupt	Voltage monitor 1, main clock oscillation stop	
	NMI	
	User break	
	INT	
	Memory error (RAM error/ROM error)	
	Software interrupt (SINT)	
	Peripheral functions | Refer to section 8, Interrupt Controller (INTC).	
Instruction	Trap instruction (TRAPA instruction)	
	General illegal instructions (undefined code)	
	Slot illegal instructions (undefined code placed directly after a delayed branch instruction *1 (including FPU instructions and FPU-related CPU instructions in FPU module standby status), instructions that rewrite the PC *2, 32-bit instructions *3, RESBANK instruction, DIVS instruction, DIVU instruction, and instructions that lead to CPU instruction-access MPU errors)	▼ Low

Notes: 1. Delayed branch instructions: JMP, JSR, BRA, BSR, RTS, RTE, BF/S, BT/S, BSRF, and BRAF.
2. Instructions that rewrite the PC: JMP, JSR, BRA, BSR, RTS, RTE, BT, BF, TRAPA, BF/S, BT/S, BSRF, BRAF, JSR/N, and RTV/N.
3. 32-bit instructions: BAND.B, BANDNOT.B, BCLR.B, BLD.B, BLDNOT.B, BOR.B, BORNOT.B, BSET.B, BST.B, BXOR.B, FMOV.S@disp12, FMOV.D@disp12, MOV.B@disp12, MOV.W@disp12, MOV.L@disp12, MOVI20, MOVI20S, MOVU.B, and MOVU.W.

6.1.2 Exception Handling Operations

The exception handling sources are detected and exception handling starts according to the timing shown in Table 6.2. Table 6.2 lists the Exception Source Detection and Exception Handling Start Timing

Table 6.2 Exception Source Detection and Exception Handling Start Timing

Exception Source		Source Detection and Handling Start Timing
Resets	Hardware reset	Starts when the RESET# pin changes from low to high.
	Watchdog timer reset	• Starts when the watchdog timer underflows. • Starts when a write is performed to the WDTRR register during a non-refresh period.
	Software reset	Starts when H'FF is written to the SWRR register.
	Power-down mode wake-up reset	Starts with an interrupt request source in power-down mode
Address error	CPU address error	Detected when an instruction is decoded and starts when the previous executing instruction finishes executing.
CPU operand-access MPU errors		
Interrupts		
Register bank errors	Bank underflow	Starts upon attempted execution of a RESBANK instruction when saving has not been performed to register banks.
	Bank overflow	In the state where saving has been performed to all register bank areas, starts when acceptance of register bank overflow exception has been set by the interrupt controller (the BOVE bit in the BNR register of the INTC is 1) and an interrupt that uses a register bank has occurred and been accepted by the CPU.
Instructions	Trap instruction	Starts from the execution of a TRAPA instruction.
	General illegal instructions	Starts from the decoding of undefined code anytime except immediately after a delayed branch instruction (delay slot) .
	Slot illegal instructions	Starts from the decoding of undefined code placed directly after a delayed branch instruction (delay slot) (including FPU instructions and FPU-related CPU instructions in FPU module standby status), of instructions that rewrite the PC, of 32-bit instructions, of the RESBANK instruction, of the DIVS instruction, or of the DIVU instruction.
	Integer division exception	Starts when detecting division-by-zero exception or overflow exception caused by division of the negative maximum value (H'8000 0000) by -1.
	FPU exception	Starts when detecting invalid operation exception defined by IEEE standard 754, division-by-zero exception, overflow, underflow, or inexact exception. Also starts when qNaN or ±∞ is input to the source for a floating point operation instruction when the QIS bit in the FPSCR register is set.

When exception handling starts, the CPU operates as follows.

(1) Exception Handling Triggered by Reset

The initial values of the program counter (PC) and stack pointer (SP) are fetched from the exception handling vector table (PC and SP are respectively the H'0000 0000 and H'0000 0004 addresses at a reset). Refer to section 6.1.3, Exception Handling Vector Table, for the exception handling vector table. The vector base register (VBR) is then initialized to H'0000 0000, the IMASK bits in the status register (SR) are initialized to H'F (B'1111), and the BO and CS bits are initialized to 0. The the BNR register of the interrupt controller (INTC) is also initialized to 0. The FPSCR register is also initialized to H'0004 0001 by a reset. The program starts running from the PC address fetched from the exception handling vector table.

(2) Exception Handling Triggered by Address Errors, CPU Operand-Access MPU Errors, Register Bank Errors, Interrupts, and Instructions

The program counter (PC) and status register (SR) are saved to the stack indicated by general register R15.

In the case of interrupt exception handling other than voltage monitor 1, main clock oscillation stop, NMI, or user break with usage of the register banks enabled, general registers R0 to R14, control register GBR, system registers MACH, MACL, and PR, and the vector number of the interrupt exception handling to be executed are saved to the register banks.

In the case of exception handling due to an address error, CPU operand-access MPU error, register bank error, voltage monitor interrupt 1, main clock oscillation stop detection interrupt, NMI interrupt, user break interrupt, or instruction, saving to a register bank is not performed.

When saving is performed to all register banks, automatic saving to the stack is performed instead of register bank saving. In this case, an interrupt controller setting must have been made so that a register bank overflow exception is not accepted (the BOVE bit in the BNR register is 0).

If a setting to accept a register bank overflow exception has been made (the BOVE bit in the BNR register is 1), the register bank overflow exception will be generated.

In the case of interrupt exception handling, the interrupt priority level is written to the IMASK bits in the SR register. In the case of exception handling due to an address error, CPU operand-access MPU error, register bank error, or instruction, the IMASK bits are not affected.

The start address is then fetched from the exception service routine corresponding to the exception handling vector table and the program starts running from that address.

6.1.3 Exception Handling Vector Table

Before exception handling starts running, the exception handling vector table must be set in memory. The exception handling vector table stores the start addresses of exception service routines. (The reset exception handling table holds the initial values of the program counter (PC) and stack pointer (SP).)
All exception sources are given different vector numbers and vector table address offsets, from which the vector table addresses are calculated. During exception handling, the start addresses of the exception service routines are fetched from the exception handling vector table, which is indicated by this vector table address.

Table 6.3 lists the Vector Numbers and Vector Table Address Offsets. Table 6.4 lists the Calculation of Exception Handling Vector Table Addresses.

Table 6.3 Vector Numbers and Vector Table Address Offsets

Exception Source			Vector No.	Vector Table Address Offset
Resets	Hardware reset, power-on reset, voltage monitor reset, watchdog timer reset, software reset, power-down mode wake-up reset	PC	0	H'0000 0000 to H'0000 0003
		SP	1	H'0000 0004 to H'0000 0007
(Reserved for system use)			2	H'0000 0008 to H'0000 000B
			3	H'0000 000C to H'0000 000F
Instruction	General illegal instruction *1		4	H'0000 0010 to H'0000 0013
(Reserved for system use)			5	H'0000 0014 to H'0000 0017
Instruction	Slot illegal instruction		6	H'0000 0018 to H'0000 001B
Interrupts	Voltage monitor 1, main clock oscillation stop *2		7	H'0000 001C to H'0000 001F
(Reserved for system use)			8	H'0000 0020 to H'0000 0023
Address error	CPU address error		9	H'0000 0024 to H'0000 0027
(Reserved for system use)			10	H'0000 0028 to H'0000 002B
Interrupts	NMI		11	H'0000 002C to H'0000 002F
	User break		12	H'0000 0030 to H'0000 0033
Instruction	FPU exception		13	H'0000 0034 to H'0000 0037
(Reserved for system use)			14	H'0000 0038 to H'0000 003B
Register bank errors	Bank overflow		15	H'0000 003C to H'0000 003F
	Bank underflow		16	H'0000 0040 to H'0000 0043
Instructions	Integer division exception (division by zero)		17	H'0000 0044 to H'0000 0047
	Integer division exception (overflow)		18	H'0000 0048 to H'0000 004B
CPU operand-access MPU errors			19	H'0000 004C to H'0000 004F
(Reserved for system use)			20 : 31	H'0000 0050 to H'0000 0053 : H'0000 007C to H'0000 007F
Instruction	Trap instruction (user vector)		32 : 63	H'0000 0080 to H'0000 0083 : H'0000 00FC to H'0000 00FF
Instructions	INT, peripheral functions *3		64 : 511	H'0000 0100 to H'0000 0103 : H'0000 07FC to H'0000 07FF

Notes: 1. General illegal instructions and slot illegal instructions include instructions that lead to CPU instruction-access MPU errors. To determine the source of an exception of either type, do so from within the given exception service routine.
2. Voltage monitor interrupt 1, main clock oscillation stop detection interrupt share a vector. To use these interrupts, determine which interrupt source is used to request the interrupt in the interrupt routine.
3. Refer to Table 8.11 to Table 8.16 “Interrupt Sources and Vectors” in section 8, Interrupt Controller (INTC), for more information on the vector numbers and vector table offsets of peripheral function interrupts.

Table 6.4 Calculation of Exception Handling Vector Table Addresses

Exception Source	Vector Table Address Calculation
Resets	Vector table address = (vector table address offset) = (vector number) × 4
Address errors, register bank errors, interrupts, instructions	Vector table address = VBR + (vector table address offset) = VBR + (vector number) × 4

Note: • VBR: Vector base register
• Vector table address offset: See Table 6.3.
• Vector number: See Table 6.3.

6.2 Resets

6.2.1 Types of Reset

A reset is the highest-priority exception handling source.
Table 6.5 lists the exception source detection and exception handling start timing.

Table 6.5 Exception Source Detection and Exception Handling Start Timing

Type	Internal State	
	CPU, FPU	Peripheral Functions
Hardware reset	Initialized	Initialized
Watchdog timer reset	Initialized	Initialized
Software reset	Initialized	Initialized
Power-down mode wake-up reset	Initialized	Peripheral functions that their power has been shut off are initialized

6.2.2 Hardware Reset

When the RESET# pin is driven low, this LSI enters the hardware reset state. To ensure that this LSI is reset, keep the RESET# pin low at least during oscillation stabilization time at power-on.
In the hardware reset state, the internal state of the CPU and all the peripheral function registers are initialized.

In the hardware reset state, hardware reset exception handling starts when the RESET# pin is first driven low for a fixed period and then returned to high. The CPU operates as follows:

1. The initial value (execution start address) of the program counter (PC) is fetched from the exception handling vector table.
2. The initial value of the stack pointer (SP) is fetched from the exception handling vector table.
3. The vector base register (VBR) is cleared to H'0000 0000, the IMASK bits in the status register (SR) are initialized to H'F (B'1111), and the BO and CS bits are initialized to 0. The BN bit in the BNR register of the INTC is also initialized to 0.
4. The values fetched from the exception handling vector table are set in the PC and SP, and the program starts execution.

Be sure to always perform hardware reset processing when turning the system power on.

6.2.3 Watchdog Timer Reset, Software Reset

Reset processing is performed the same as hardware reset. Refer to section 7, Resets for details.

6.2.4 Power-Down Mode Wake-Up Reset

Reset processing is performed in the same as hardware reset. Refer to section 7, Resets for details.
As for the initialization of peripheral functions, only the peripheral functions that their power has been shut off are initialized. Refer to section 5, Clocks for details.

6.3 Address Errors

6.3.1 Address Error Sources

Address errors occur when instructions are fetched or data is read or written to.
Table 6.6 lists Bus Cycles and Address Errors.

Table 6.6 Bus Cycles and Address Errors

Bus Cycle		Bus Cycle Description	Address Error Occurrence
Type	Bus Master		
Instruction fetch *2	CPU	Instruction fetched from even address	None (normal)
		Instruction fetched from odd address	Address error occurs
		Instruction fetched from other than addresses (H'0200 0000 to H'1FFF FFFF, H'2200 0000 to H'7FFF FFFF, and H'F000 0000 to H'F5FF FFFF) of peripheral function register space *1 and reserved areas *1	None (normal)
		Instruction fetched from addresses (H'0200 0000 to H'1FFF FFFF, H'2200 0000 to H'7FFF FFFF, and H'F000 0000 to H'F5FF FFFF) of peripheral function register space *1 and reserved areas *1	Address error occurs
		Instruction fetched from external memory space in single-chip mode	Address error occurs
Data read/write	CPU or DMAC	Word data accessed from even address	None (normal)
		Word data accessed from odd address	Address error occurs
		Longword data accessed from a longword boundary	None (normal)
		Longword data accessed from other than a longword boundary	Address error occurs
		Word or byte data accessed in peripheral function register space *1	None (normal)
		Longword data accessed in 16-bit peripheral function register space *1	None (normal)
		Longword data accessed in 8-bit peripheral function register space *1	None (normal)
		External memory space accessed in single-chip mode	Address error occurs

Notes: 1. Refer to section 3, Address Space for details on the peripheral function register space and the on-chip RAM space.
2. If an instruction is placed within 10 bytes from the last address of the on-chip RAM space, the CPU accesses beyond the boundary to fetch the instruction, causing an address error.

6.3.2 Address Error Exception Handling

When an address error occurs, the bus cycle in which the address error occurred ends. On completion of the instruction in progress, address error exception handling starts. The CPU operates as follows:

1. The start address of exception service routine which corresponds to the address error that occurred is fetched from the exception handling vector table.
2. The status register (SR) is saved to the stack.
3. The program counter (PC) is saved to the stack. The PC value saved is the start address of the instruction to be executed after the last executed instruction.
4. After jumping to the start address of exception service routine fetched from the exception handling vector table, program execution starts. This jump is not a delayed branch.

6.4 CPU Operand-Access MPU Errors

6.4.1 CPU Operand-Access MPU Error Sources

A CPU operand-access MPU error occurs when operand access by the CPU violates the protection attributes which have been set for the area being accessed. For details, see section 29, Memory Protection Unit (MPU).

6.4.2 CPU Operand-Access MPU Error Exception Handling

When a CPU operand-access MPU error occurs, the bus cycle in which the CPU operand-access MPU error occurred ends. On completion of the instruction for which execution was in progress, CPU operand-access MPU error exception handling starts. The CPU operates as follows:

1. The start address of the exception service routine which corresponds to the CPU operand-access MPU error that occurred is fetched from the exception handling vector table.
2. The status register (SR) is saved to the stack.
3. The program counter (PC) is saved to the stack. The PC value saved is the start address of the instruction to be executed after the last executed instruction.
4. After jumping to the address fetched from the exception handling vector table, program execution starts. This jump is not a delayed branch.

6.5 Register Bank Errors

6.5.1 Register Bank Error Sources

(1) Bank Overflow

In the state where saving has already been performed to all register bank areas, bank overflow occurs when acceptance of register bank overflow exception has been set by the interrupt controller (the BOVE bit in the BNR register is set to 1) and an interrupt that uses a register bank has occurred and been accepted by the CPU.

(2) Bank Underflow

Bank underflow occurs when an attempt is made to execute a RESBANK instruction while saving has not been performed to register banks.

6.5.2 Register Bank Error Exception Handling

When a register bank error occurs, register bank error exception handling starts.
The CPU operates as follows:

1. The start address of exception service routine which corresponds to the register bank error that occurred is fetched from the exception handling vector table.
2. The status register (SR) is saved to the stack.
3. The program counter (PC) is saved to the stack. The PC value saved is the start address of the instruction to be executed after the last executed instruction for a bank overflow, and the start address of the executed RESBANK instruction for a bank underflow. To prevent multiple interrupts from occurring at a bank overflow, the interrupt priority level that caused the bank overflow is written to the interrupt mask level bits (IMASK) in the status register (SR).
4. After jumping to the start address fetched from the exception handling vector table, program execution starts. This jump is not a delayed branch.

6.6 Interrupts

6.6.1 Interrupt Sources

Table 6.7 lists the Interrupt Sources to start interrupt exception handling.

Table 6.7 Interrupt Sources

Type	Request Source	Number of Sources
Voltage monitor 1	Voltage monitor circuit 1	1
Main clock oscillation stop	Main clock oscillator	1
NMI	NMI pin (external input)	1
User break	User break controller	1
INT	SH72A2 group: INT0 to INT13 pins (external input) SH72A0 group: INT0 to INT8 and INT10 pins (external input)	SH72A2 group: 14 SH72A0 group: 10
Memory error	RAM/ROM	4
DMAC access MPU error	MPU	1
SINT	Software interrupt	8
Peripheral function	Refer to section 8, Interrupt Controller (INTC).	SH72A2 group: 155 SH72A0 group: 134

Each interrupt source has a different vector number and vector table offset.

Refer to Table 8.11 to Table 8.16 Interrupt Sources and Vectors in section 8, Interrupt Controller (INTC), for more information on vector numbers and vector table address offsets.

6.6.2 Interrupt Priority Level

The priority of interrupt sources is predetermined. When multiple interrupts occur simultaneously, the interrupt controller (INTC) determines their relative priorities and starts processing according to the results.
Table 6.8 lists the Interrupt Priority.

Table 6.8 Interrupt Priority

Type	Priority Level	Comment
Voltage monitor 1	16	Fixed priority level. Cannot be masked.
Main clock oscillation stop	16	Fixed priority level. Cannot be masked.
NMI	16	Fixed priority level. Cannot be masked.
User break	15	Fixed priority level.
INT	0 to 15	Set using registers IPR1 to IPR4 of the INTC.
Memory error	15	Fixed priority level.
DMAC access MPU error	15	Fixed priority level.
SINT8 to 1	8 to 1	Fixed priority level.
Peripheral function	0 to 15	Set using the IPR bits in registers ICR102 to ICR511 of the INTC.

The priority of interrupt sources is expressed as priority levels 0 to 16, with priority 0 the lowest and priority 16 the highest. Voltage monitor 1, main clock oscillation stop, and NMI interrupts have priority 16 and cannot be masked, thus they are always accepted. The priority levels of the user break interrupt, the memory error interrupt, and the DMAC access MPU error interrupt are 15. The priority levels of software interrupts (SINT) are fixed to priority 8 to 1 for each source of SINT8 to SINT1.
The priority levels of INT interrupts, registers IPR1 to IPR4 of the INTC, and on-chip peripheral function interrupts can be set freely using the IPR bits in registers ICR102 to ICR511 of the INTC as shown in Table 6.8. The priority levels that can be set are 0 to 15. Level 16 cannot be set.

Refer to section 8.3.1, Interrupt Priority Level Register i (IPRi) (i = 1 to 4), for more information on registers IPR1 to IPR4. Refer to section 8.3.10, Interrupt Control Register i (ICRi) (i = 102 to 109, 134 to 136, 142 to 156, 165 to 166, 173 to 180, 191 to 196, 214 to 215, 221 to 240, 297 to 300, 312 to 339, 366 to 367, 378 to 379, 403 to 418, 427 to 464, 477), for more information on registers ICR102 to ICR511.

6.6.3 Interrupt Exception Handling

When an interrupt occurs, its priority level is ascertained by the interrupt controller (INTC). Voltage monitor, main clock oscillation stop, and NMI interrupt exception handling is always accepted, but other interrupts are only accepted if they have a priority level higher than the priority level set in the IMASK bits in the status register (SR).

When an interrupt is accepted, interrupt exception handling starts. In interrupt exception handling, the CPU fetches the exception service routine start address which corresponds to the accepted interrupt from the exception handling vector table, and saves the status register (SR) and the program counter (PC) to the stack.
In the case of interrupt exception handling other than voltage monitor 1, main clock oscillation stop, NMI, or user break with usage of the register banks enabled, general registers R0 to R14, control register GBR, system registers MACH, MACL, and PR, and the vector number of the interrupt exception handling to be executed are saved in the register banks.
In the case of exception handling due to an address error, CPU operand-access MPU error, voltage monitor 1 interrupt, main clock oscillation stop detection interrupt, NMI interrupt, user break interrupt, or instruction, saving is not performed to the register banks.
When saving has been performed to all register banks (0 to 14), automatic saving to the stack is performed instead of register bank saving. In this case, an interrupt controller setting must have been made so that a register bank overflow exception is not accepted (the BOVE bit in the BNR register of the INTC is 0).
If a setting to accept a register bank overflow exception has been made (the BOVE bit in the BNR register of the INTC is 1), the register bank overflow exception occurs. Next, the priority level value of the accepted interrupt is written to the IMASK bits in the status register (SR). For voltage monitor, main clock oscillation stop, and NMI interrupt exception handling, however, the priority level is 16, but the value set in the IMASK bits is H'F (level 15).
Then, after jumping to the start address fetched from the exception handling vector table, program execution starts. The jump that occurs is not a delayed branch.

Refer to section 8.4.11, Interrupt Operation Sequence for more details on interrupt exception handling.

6.7 Instructions

6.7.1 Types of Exceptions Triggered by Instructions

Exception handling can be triggered by trap instructions, slot illegal instructions, general illegal instructions, integer division exceptions, and FPU exception.

Table 6.9 lists the Types of Exceptions Triggered by Instructions.

Table 6.9 Types of Exceptions Triggered by Instructions

Type	Source Instruction	Comment
Trap instruction	TRAPA	
Slot illegal instruction	Undefined code placed immediately after a delayed branch instruction (delay slot) (including FPU instructions and FPU-related CPU instructions in FPU module standby state), instructions that rewrite the PC, 32-bit instructions, RESBANK instruction, DIVS instruction, and DIVU instruction, or instructions that lead to CPU instruction-access MPU errors	Delayed branch instructions: JMP, JSR, BRA, BSR, RTS, RTE, BF/S, BT/S, BSRF, and BRAF Instructions that rewrite the PC: JMP, JSR, BRA, BSR, RTS, RTE, BT, BF, TRAPA, BF/S, BT/S, BSRF, BRAF, JSR/N, and RTV/N 32-bit instructions: BAND.B, BANDNOT.B, BCLR.B, BLD.B, BLDNOT.B, BOR.B, BORNOT.B, BSET.B, BST.B, BXOR.B, FMOV.S@disp12, FMOV.D@disp12, MOV.B@disp12, MOV.W@disp12, MOV.L@disp12, MOVI20, MOVI20S, MOVU.B, MOVU.W.
General illegal instruction	Undefined code anywhere besides in a delay slot (including FPU instructions and FPU-related CPU instructions in FPU module standby status) or instructions that lead to CPU instruction-access MPU errors	
Integer division exception	Division by zero	DIVU, DIVS
	Negative maximum value ÷ (-1)	DIVS
FPU exception	Starts when detecting invalid operation exception defined by IEEE754, division-by-zero exception, overflow, underflow, or inexact exception.	FADD, FSUB, FMUL, FDIV, FMAC, FCMP/EQ, FCMP/GT, FNEG, FABS, FLOAT, FTRC, FCNVDS, FCNVSD, FSQRT

6.7.2 Trap Instructions

When a TRAPA instruction is executed, trap instruction exception handling starts. The CPU operates as follows:

1. The start address of exception service routine which corresponds to the vector number specified in the TRAPA instruction is fetched from the exception handling vector table.
2. The status register (SR) is saved to the stack.
3. The program counter (PC) is saved to the stack. The PC value saved is the start address of the instruction to be executed after the TRAPA instruction.
4. After jumping to the start address fetched from the exception handling vector table, program execution starts. This jump is not a delayed branch.

6.7.3 Slot Illegal Instructions

An instruction placed immediately after a delayed branch instruction is said to be placed in a delay slot. When the instruction placed in the delay slot is undefined code (including FPU instructions and FPU-related CPU instructions in FPU module standby state), an instruction that rewrites the PC, a 32-bit instruction, an RESBANK instruction, a DIVS instruction, a DIVU instruction, or a instruction that leads to CPU instruction-access MPU errors, slot illegal instruction exception handling starts when such kind of instruction is decoded. The CPU operates as follows:

1. The start address of the exception service routine corresponding to the slot illegal instruction that occurred is fetched from the exception handling vector table.
2. The status register (SR) is saved to the stack.
3. The program counter (PC) is saved to the stack. The PC value saved is the jump address of the delayed branch instruction immediately before the undefined code, the instruction that rewrites the PC, the 32-bit instruction, the RESBANK instruction, the DIVS instruction, the DIVU instruction, or the instruction that leads to CPU instruction-access MPU errors.
4. After jumping to the address fetched from the exception handling vector table, program execution starts. This jump is not a delayed branch.

6.7.4 General Illegal Instructions

When undefined code placed anywhere other than immediately after a delayed branch instruction (i.e., in a delay slot) (including FPU instructions and FPU-related CPU instructions in FPU module standby state) or the instruction that leads to CPU instruction-access MPU errors is decoded, general illegal instruction exception handling starts. The CPU handles general illegal instructions in the same way as slot illegal instructions. Unlike processing of slot illegal instructions, however, the program counter (PC) value stored is the start address of the undefined code or the instruction that leads to CPU instruction-access MPU errors.

6.7.5 Integer Division Exceptions

When an integer division instruction performs division by zero or the result of integer division overflows, integer division instruction exception occurs. The instructions that may become the source of division-by-zero exception are DIVU and DIVS. The only source instruction of overflow exception is DIVS, and overflow exception occurs only when the negative maximum value is divided by.-1. The CPU operates as follows:

1. The start address of exception service routine which corresponds to the integer division instruction exception that occurred is fetched from the exception handling vector table.
2. The status register (SR) is saved to the stack.
3. The program counter (PC) is saved to the stack. The PC value saved is the start address of the integer division instruction at which the exception occurred.
4. After jumping to the start address of exception service routine fetched from the exception handling vector table, program execution starts. This jump is not a delayed branch.

6.7.6 FPU Exceptions

An FPU exception handling occurs when the V, Z, O, U or I bit in the FPU enable field (Enable) of the floating point status/control register (FPSCR) is set to 1. This indicates the occurrence of an invalid operation exception defined by the IEEE standard 754, a division-by-zero exception, overflow (in the case of an instruction for which this is possible), underflow (in the case of an instruction for which this is possible), or inexact exception (in the case of an instruction for which this is possible).
The instructions that may cause FPU exception handling are FADD, FSUB, FMUL, FDIV, FMAC, FCMP/EQ, FCMP/GT, FLOAT, FTRC, FCNVDS, FCNVSD, and FSQRT.

An FPU exception handling occurs only when the corresponding enable bit (Enable) is set. When the FPU detects an exception source resulting from floating-point operations, FPU operation is suspended and occurrence of an FPU exception handling is reported to the CPU. When exception handling is started, the CPU operations are as follows.

1. The start address of the exception service routine corresponding to the FPU exception handling that occurred is fetched from the exception handling vector table.
2. The status register (SR) is saved to the stack.
3. The program counter (PC) is saved to the stack. The PC value saved is the start address of the instruction to be executed after the last executed instruction.
4. After jumping to the address fetched from the exception handling vector table, program execution starts.
   This jump is not a delayed branch.

The FPU exception flag field (Flag) of the FPSCR register is always updated regardless of whether or not an FPU exception handling has been accepted, and remains set until explicitly cleared by the user through an instruction. The FPU exception source field (Cause) of the FPSCR register changes each time a floating-point operation instruction is executed.
When the V bit in the FPU exception enable field (Enable) of the FPSCR register is set and the QIS bit in the FPSCR register is also set to 1, inputting qNaN or ±∞ to a floating point operation instruction source occurs FPU exception handling.

6.8 When Exception Handling is not Accepted

When an address error, FPU exception, CPU operand-access MPU error, register bank error (overflow), or interrupt occurs immediately after a delayed branch instruction, it is sometimes not accepted immediately but stored instead. When this happens, it will be accepted when an instruction that can accept the exception is decoded.
Table 6.10 shows the Exception Source Generation Immediately after Delayed Branch Instruction.

Table 6.10 Exception Source Generation Immediately after Delayed Branch Instruction

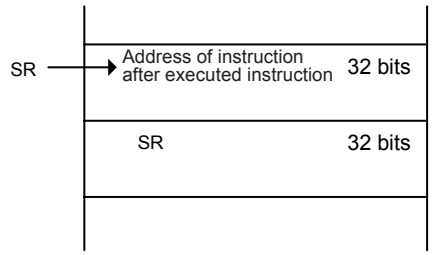
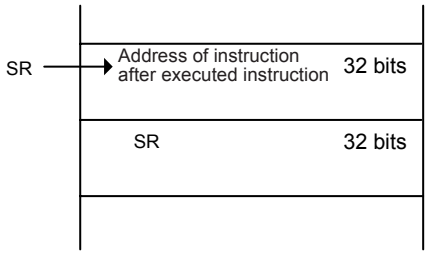
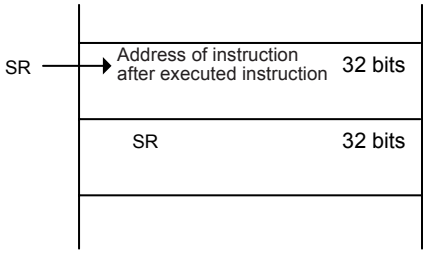
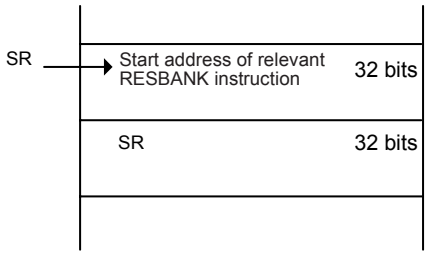
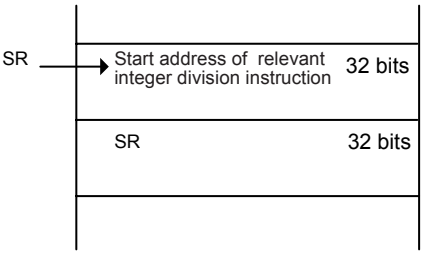
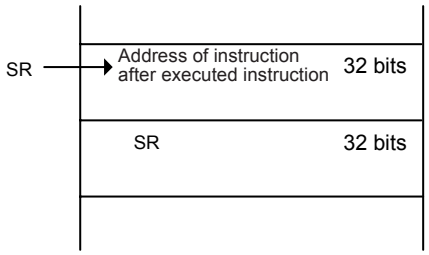
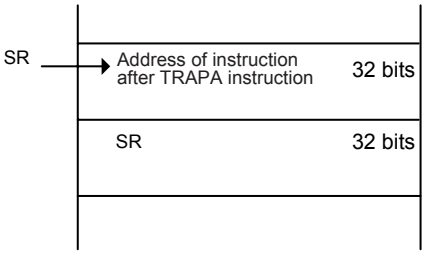
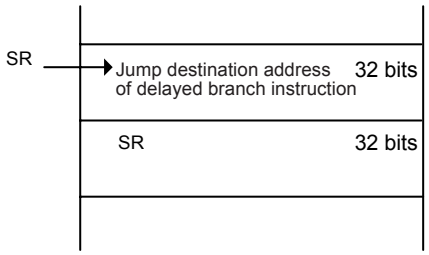
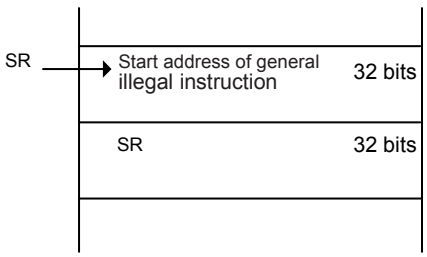
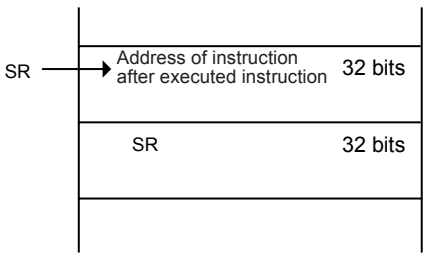
Point of Occurrence	Exception Source			
	Address Error	FPU Exception or CPU Operand-Access MPU Errors	Register Bank Error (Overflow)	Interrupt
Immediately after a delayed branch instruction *	Not accepted	Not accepted	Not accepted	Not accepted

Note: * Delayed branch instructions: JMP, JSR, BRA, BSR, RTS, RTE, BF/S, BT/S, BSRF, and BRAF.

6.9 Stack Status after Exception Handling Ends

Table 6.11 shows the stack status after exception handling ends.

Table 6.11 Stack Status after Exception Handling Ends

Type	Stack Status	Type	Stack Status
Address error		Interrupt	
Register bank error (overflow)		Register bank error (underflow)	
FPU exception		CPU operand-access MPU error	
Trap instruction		Slot illegal instruction	
General illegal instruction		Integer division exception	

6.10 Notes on Exceptional Handling

6.10.1 Value of Stack Pointer (SP)

The value of the stack pointer must always be a multiple of four. Otherwise, an address error will occur when the stack is accessed during exception handling.

6.10.2 Value of Vector Base Register (VBR)

The value of the vector base register must always be a multiple of four. Otherwise, an address error will occur when the stack is accessed during exception handling.

6.10.3 Address Errors Caused by Stacking of Address Error Exception Handling

When the stack pointer is not a multiple of four, an address error will occur during stacking of the exception handling (interrupts, etc.) and address error exception handling will start up as soon as the first exception handling is ended. Address errors will then also occur in the stacking for this address error exception handling. To ensure that address error exception handling does not go into an endless loop, no address errors are accepted at that point. This allows program control to be shifted to the address error exception service routine and enables error processing.

When an address error occurs during exception handling stacking, the stacking bus cycle (write) is executed. During stacking of the status register (SR) and program counter (PC), the SP is decremented by 4 for both, so the value of SP will not be a multiple of four after the stacking either. The address value output during stacking is the SP value, so the address where the error occurred is itself output. This means the write data stacked will be undefined.

6.10.4 Interrupt Control by Modifying Interrupt Mask Bits

Ensure that at least five instructions are placed between an interrupt-enabling instruction and interrupt-disabling instruction when an LDC or LDC.L instruction is used to enable or disable interrupts by manipulating the values of the interrupt mask bits (IMASK3 to IMASK0) of the status register (SR).

7. Resets

7.1 Introduction

The following four resets are implemented in the MCU: hardware, software, watchdog timer, and power-down mode wake-up.

Table 7.1 lists the Reset Specifications.

Figure 7.1 shows the Reset Circuit Block Diagram.

Table 7.1 Reset Specifications

Item		Function
Reset sources	Hardware reset	• A reset is performed when the RESET# pin changes from low to high. • Noise reduction via the analog noise filter
	Software reset	A reset is performed by writing H'FF to the SWRR register.
	Watchdog timer reset	A reset is performed in either of the following: • When the watchdog timer underflows • When a write is performed to the WDTRR register during a non-refresh period
	Power-down mode wake-up reset	An automatic reset is performed with the interrupt request source in power-down mode.
Reset source determination function		Check the type of the generated reset using the RSDR register.
Cold Start-Up/Warm Start-Up Function		By setting the CWSF bit in the RSDR register, this function is used to determinate cold start-up (reset process) when the power is turned on or warm start-up (reset process) when a reset signal is applied during operation.

Only the functions that their power has been shut off are initialized by a power-down mode wake-up reset. The functions that have been running or stopped are not initialized.

For the operation, stop, and power cut-off of each function, refer to section 5.5, Power Control.

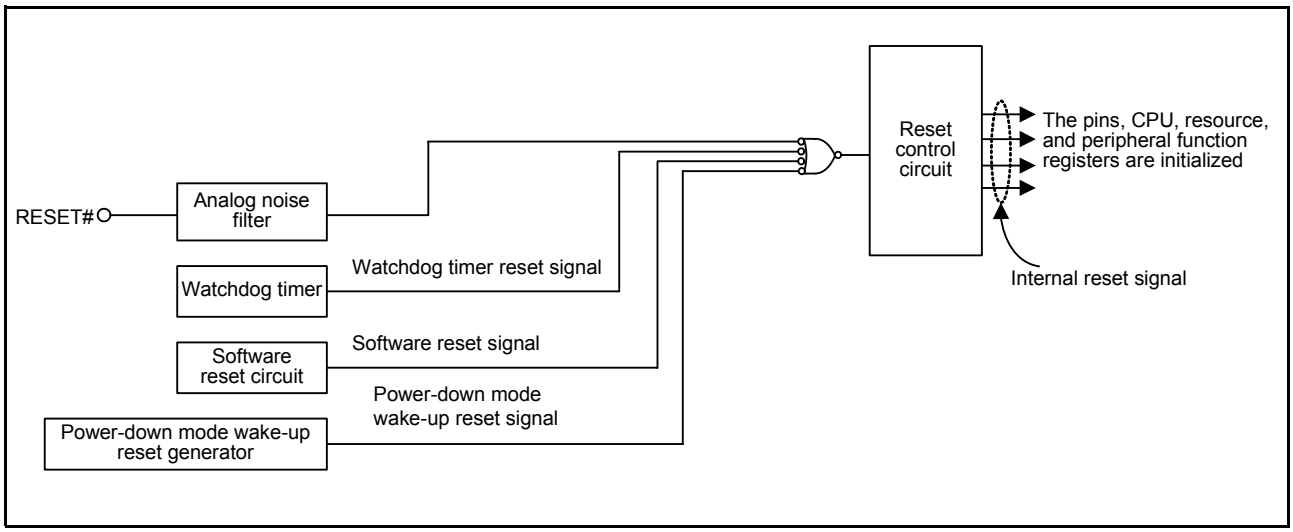


Figure 7.1 Reset Circuit Block Diagram

Table 7.2 lists the Reset I/O Pin.

Table 7.2 Reset I/O Pin

Pin Name	I/O	Description
RESET#	Input	Reset pin. When this pin is driven low, the MCU is reset.

7.2 Registers

Table 7.3 lists the Reset Circuit Registers.

Table 7.3 Reset Circuit Registers

Register Name	Symbol	After Reset	Address	Access Size
System protect register 2	SPR2	H'00	H'FF46 E263	8
Software reset register	SWRR	H'00	H'FF46 E203	8
Reset source determine register	RSDR	H'00	H'FF46 E20B	8

7.2.1 System Protect Register 2 (SPR2)

Address H'FF46 E263

	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	0	0	0	0	0	0	0	0

Bit	Description	R/W
b7 to b0	When written B'1111 0001: Protection unlocked Other than B'1111 0001: Protection locked When read Bit 0 (b0) 0: Protection locked 1: Protection unlocked Bit 7 to bit 1 (b7 to b1) are always read as 0.	R/W

The SPR2 register is used to set the protect function that protects registers SWRR, RSDR, SLCR1, PDWSSLi (i = 0, 2 to 5), PDWSSRi, PDWSESLj (j = 0 to 2), RSTSR0, and BURm (m = 0 to 31) from being rewritten easily. To change the values of these registers, perform the following procedure:

(1) Write H'F1 to the SPR2 register (writing to the registers enabled).
(2) Change the values of registers SWRR, RSDR, SLCR1, PDWSSLi, PDWSSRi, PDWSESLj, RSTSR0, and BURm.
(3) Write a value other than H'F1 to the SPR2 register (writing to the registers disabled).

7.2.2 Software Reset Register (SWRR)

Address H'FF46 E203

	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	0	0	0	0	0	0	0	0

Bit	Description	R/W
b7 to b0	The MCU is reset by writing H'FF. The read value is H'00.	R/W

This register is used to control the software reset. To set the SWPR register, use the SPR2 register to unlock the protection before writing.

Immediately after writing H'FF to SWRR, execute an instruction to read from SWRR and at least five NOP instructions.

7.2.3 Reset Source Determine Register (RSDR)

Address H'FF46 E20B

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	WDTRF	SWRF	HWRF	CWSF
After Reset	0	0	0	0	(Note 1)	(Note 1)	(Note 1)	(Note 1)

Bit	Symbol	Bit Name	Description	R/W
b7 to b4	—	Reserved	These bits are read as 0. The write value should always be 0.	R
b3	WDTRF	Watchdog Timer Reset Detect Flag	0: Not detected 1: Detected	R/W
b2	SWRF	Software Reset Detect Flag	0: Not detected 1: Detected	R/W
b1	HWRF	Hardware Reset Detect Flag	0: Not detected 1: Detected	R/W
b0	CWSF	Cold Start-Up/Warm Start-Up Determine Flag	0: Cold start-up 1: Warm start-up	R/W

Note 1. The reset value varies with the reset source.

This register is used to determine the reset source when the MCU performs a reset. This register is also used to determine whether cold start-up or warm start-up. To set the RSDR register, use the SPR2 register to unlock the protection before writing.

WDTRF Bit

The WDTRF flag becomes 1 (detected) when a watchdog timer reset is performed. This flag becomes 0 (not detected) when any other reset is performed. This flag can be set to 0 by a program. Writing a 1 has no effect.

SWRF Bit

The SWRF flag becomes 1 (detected) when a software reset is performed. This flag becomes 0 (not detected) when any other reset is performed. This flag can be set to 0 by a program. Writing a 1 have no effect.

HWRF Bit

The HWRF flag becomes 1 (detected) when a hardware reset is performed. This flag becomes 0 (not detected) when any other reset is performed. This flag can be set to 0 by a program. Writing a 1 has no effect.

CWSF Bit

The CWSF flag becomes 0 when power is supplied to the LSI (cold start-up). This flag remains unchanged at any other reset. This flag is set to 1 (warm start-up) by writing a 1 with a program. Writing a 0 has no effect.
To determine whether power is supplied, set this bit to 1.

7.3 Reset Sources

7.3.1 Hardware Reset

A reset is triggered by the RESET# pin. If the supply voltage meets the recommended operating conditions, the MCU is reset by applying a low-level signal to the RESET# pin.

7.3.1.1 When Power Supply is Stable

(1) Apply a low-level signal to the RESET# pin.
(2) Wait for 100 µs or more with the RESET# signal held low.
(3) Apply a high-level signal to the RESET# pin.

7.3.1.2 At Power-On

(1) Apply a low-level signal to the RESET# pin.
(2) Wait for 10 ms or more from VCCmin to reset cancellation as the hold time.
(3) Apply a high-level signal to the RESET# pin.

7.3.2 Software Reset

A reset is triggered by the software reset circuit. By writing H'FF to the SWRR register, the MCU is reset.

7.3.3 Watchdog Timer Reset

A reset is triggered by the watchdog timer. When the watchdog timer underflows or when a write to the WDTRR register is performed during a non-refresh period the MCU is reset.

7.3.4 Power-Down Mode Wake-Up Reset

An automatic reset is performed with the interrupt request source to wake up from power-down mode 0 or power-down mode 1. The functions that their power has been shut off are initialized, and the clocks that their oscillation has been stopped are started. The functions that have been running or stopped are not initialized. For details on power-down mode wake-up reset, refer to section 5.5, Power Control.

7.4 Reset Sequence

When an internal reset signal goes to high level, the MCU is reset and the pins, peripheral function registers, and the CPU are initialized. When the internal reset signal changes from high to low, the CPU executes a program beginning with the address indicated by the reset vector. When a reset is performed while writing to the internal RAM, the written value is undefined. Figure 7.2 shows a timing of the reset sequence, and table 7.4 shows the CPU Register States after Reset.

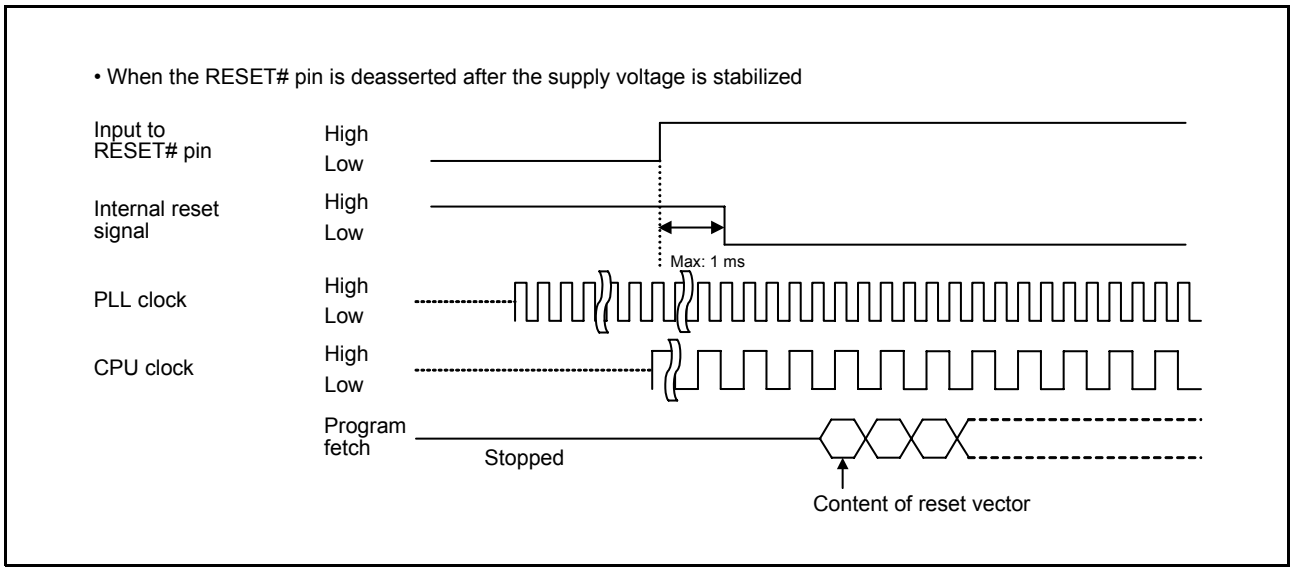


Figure 7.2 Reset Sequence

Table 7.4 CPU Register States after Reset

Classification		Register	Initial Value
CPU	General registers	R0 to R14	Undefined
		R15 (SP)	Value of the stack pointer (SP) in the vector address table
	Control registers	SR	Bits IMASK are 1111 (H'F), BO and CS are 0, reserved bits are 0, and others are undefined
		GBR, TBR	Undefined
		VBR	H'0000 0000
	System registers	MACH, MACL, PR	Undefined
		PC	Value of the program counter (PC) in the vector address table
RAM		—	Undefined after power-on

7.5 Reset Determination Function

This function is used to detect which reset source is used to reset the MCU. Whether a hardware reset, software reset, or watchdog timer reset has been performed can be checked by reading the RSDR register.

When a hardware reset is performed, the HWRF bit becomes 1 (detected). When a software reset is performed, the SWR bit becomes 1 (detected). When a watchdog timer occurs, the WDTRF bit becomes 1 (detected). These bits are individually set to 0 (not detected) when a reset other than the corresponding reset occurs. They can be set to 0 by a program.

7.6 Cold Start-Up/Warm Start-Up Determination Function

By setting the CWSF bit in the RSDR register, this function is used to determine whether cold start-up (reset process) when the power is turned on or warm start-up (reset process) when a reset signal is applied during operation.

The CWSF bit becomes 0 (cold start-up) when power is supplied to the LSI. This bit does not become 0 at a hardware reset, a software reset, watchdog timer reset, or power-down mode wake-up reset. This bit is set to 1 (warm start-up) by writing a 1 with a program. Writing 0 has no effect.

Figure 7.3 shows an Operation Example of Cold Start-Up/Warm Start-Up Determination Function.

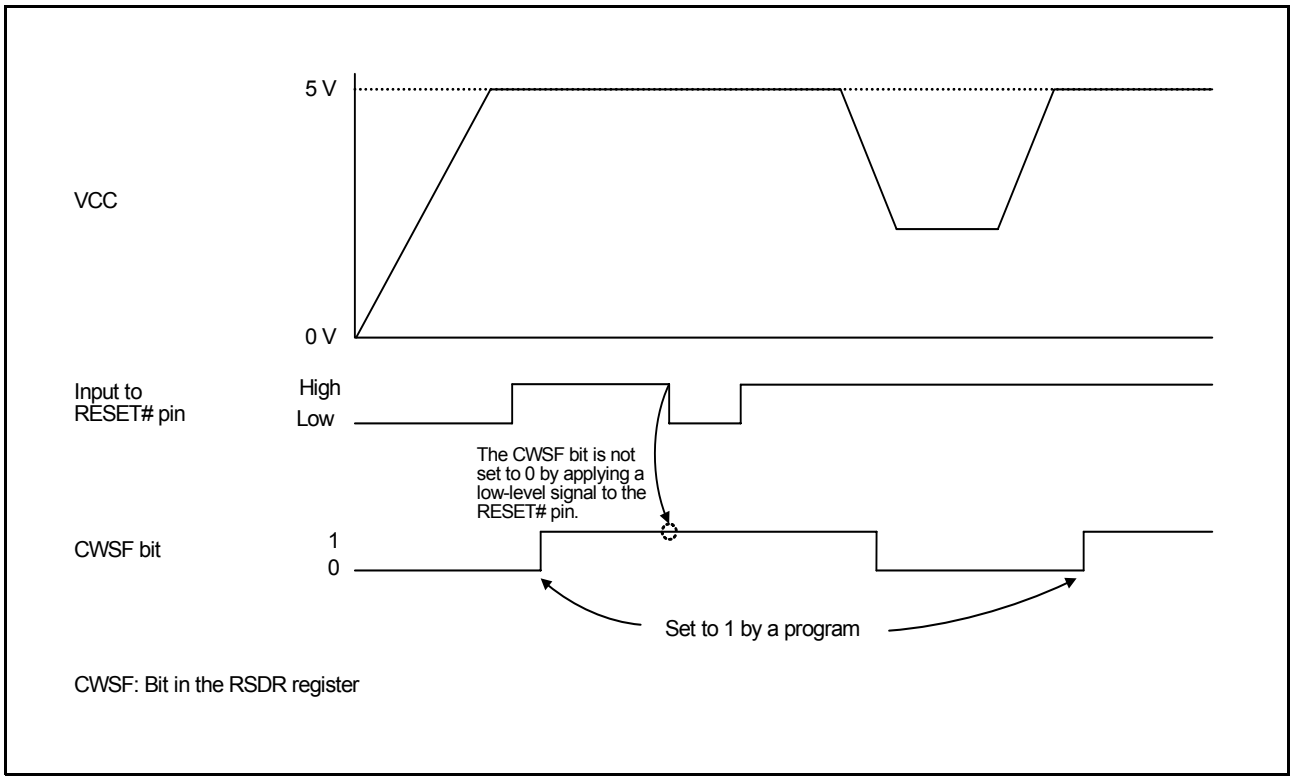


Figure 7.3 Operation Example of Cold Start-Up/Warm Start-Up Determination Function

8. Interrupt Controller (INTC)

8.1 Introduction

The interrupt controller (INTC) ascertains the priority of interrupt sources and controls interrupt requests to the CPU. The priority of each interrupt can be set in the INTC registers, and thus interrupts can be processed according to the set priority.

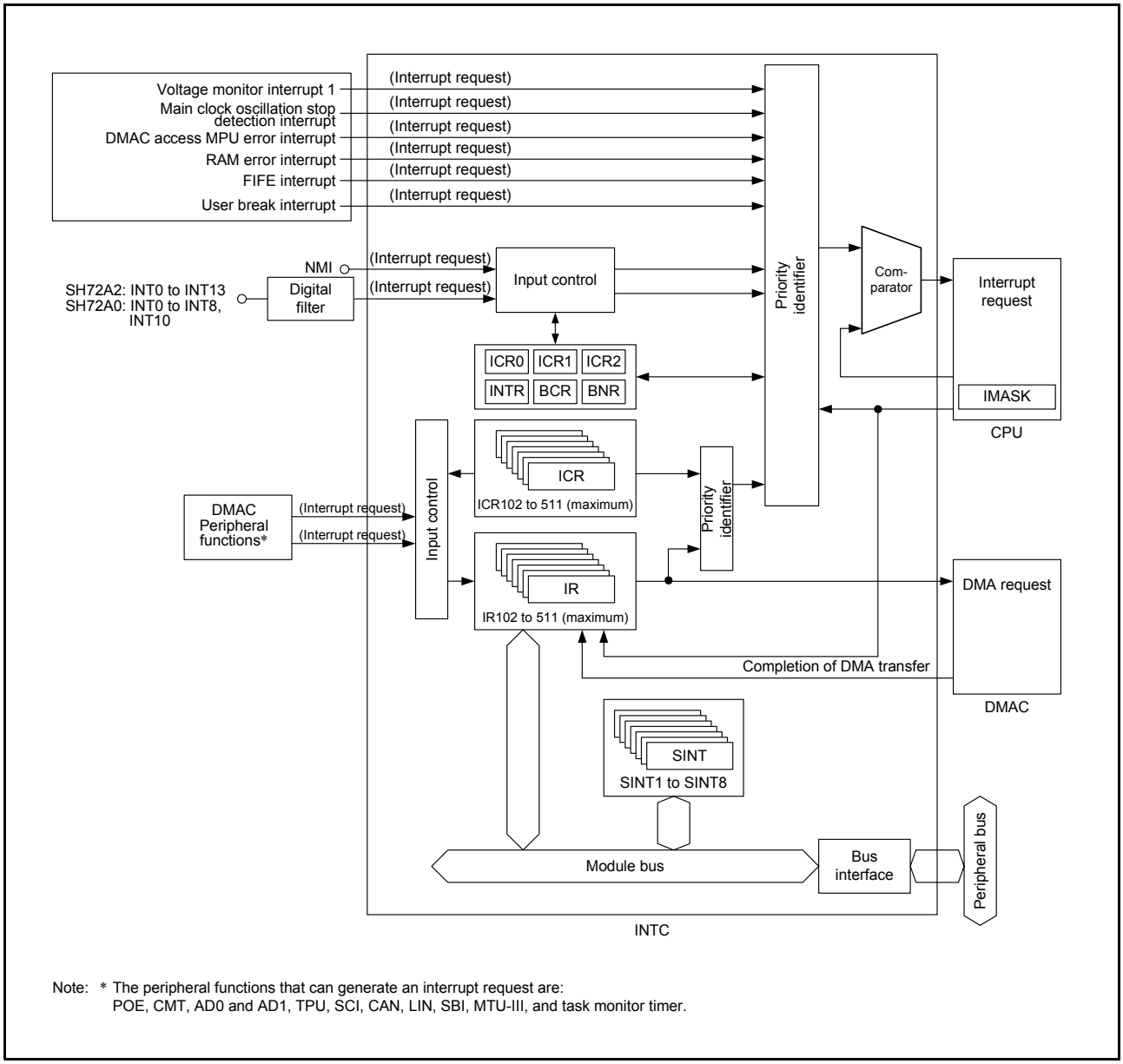
8.1.1 Specifications

The following shows the INTC Specifications.

Table 8.1 INTC Specifications

Item	Description
Interrupt priority	By setting the four interrupt priority level registers and the interrupt control registers for peripheral functions, the priorities of INT interrupts and peripheral function interrupts can be selected from 16 levels for request sources.
NMI detection	Falling edge or rising edge selectable
INT detection	Low level, falling edge, rising edge, or both edges selectable
Register banks	This LSI has register banks that enable register saving and restoring required in the interrupt processing to be performed at high speed.
Software interrupt (SINT)	By setting the software interrupt register, an interrupt with a given priority (8 to 1) can be generated from a program.

Figure 8.1 shows the INTC Block Diagram.



Note: * The peripheral functions that can generate an interrupt request are:
POE, CMT, AD0 and AD1, TPU, SCI, CAN, LIN, SBI, MTU-III, and task monitor timer.

Figure 8.1 INTC Block Diagram

8.2 Input/Output Pins

Table 8.2 lists the INTC Pins.

Table 8.2 INTC Pins

Pin Name	I/O	Description
NMI	Input	Input pin for non-maskable interrupt requests
SH72A2 group: INT0 to INT13 SH72A0 group: INT0 to INT8, INT10	Input	Input pins for maskable interrupt requests

8.3 Registers

Table 8.3 to Table 8.10 list the INTC Registers.

Table 8.3 INTC Registers (1)

Register Name	Symbol	After Reset	Address	Access Size
Interrupt control register 0	ICR0	*1	H'FFFD 9400	16, 32
Interrupt control register 1	ICR1	H'0000	H'FFFD 9402	16, 32
Interrupt control register 2	ICR2	H'0000	H'FFFD 9404	16
INT Interrupt request register	INTR	H'0000	H'FFFD 9408	16
Bank control register	BCR	H'0000	H'FFFD 940E	16
Bank number register	BNR	H'0000	H'FFFD 9410	16
Software interrupt register 1	SINT1	H'00	H'FFFD 9412	8
Software interrupt register 2	SINT2	H'00	H'FFFD 9413	8
Software interrupt register 3	SINT3	H'00	H'FFFD 9414	8
Software interrupt register 4	SINT4	H'00	H'FFFD 9415	8
Software interrupt register 5	SINT5	H'00	H'FFFD 9416	8
Software interrupt register 6	SINT6	H'00	H'FFFD 9417	8
Software interrupt register 7	SINT7	H'00	H'FFFD 9418	8
Software interrupt register 8	SINT8	H'00	H'FFFD 9419	8
Interrupt priority level register 1	IPR1	H'0000	H'FFFD 941A	16
Interrupt priority level register 2	IPR2	H'0000	H'FFFD 941C	16, 32
Interrupt priority level register 3	IPR3	H'0000	H'FFFD 941E	16, 32
Interrupt priority level register 4	IPR4	H'0000	H'FFFD 9420	16
Interrupt request register 102	IR102	H'00	H'FFFD 9800	8
Interrupt request register 103	IR103	H'00	H'FFFD 9801	8
Interrupt request register 104	IR104	H'00	H'FFFD 9802	8
Interrupt request register 105	IR105	H'00	H'FFFD 9803	8
Interrupt request register 106	IR106	H'00	H'FFFD 9804	8
Interrupt request register 107	IR107	H'00	H'FFFD 9805	8
Interrupt request register 108	IR108	H'00	H'FFFD 9806	8
Interrupt request register 109	IR109	H'00	H'FFFD 9807	8
Interrupt request register 134	IR134	H'00	H'FFFD 9820	8
Interrupt request register 135	IR135	H'00	H'FFFD 9821	8
Interrupt request register 136	IR136	H'00	H'FFFD 9822	8
Interrupt request register 142	IR142	H'00	H'FFFD 9828	8
Interrupt request register 143	IR143	H'00	H'FFFD 9829	8
Interrupt request register 144	IR144	H'00	H'FFFD 982A	8
Interrupt request register 145	IR145	H'00	H'FFFD 982B	8
Interrupt request register 146	IR146	H'00	H'FFFD 982C	8
Interrupt request register 147	IR147	H'00	H'FFFD 982D	8
Interrupt request register 148	IR148	H'00	H'FFFD 982E	8
Interrupt request register 149*2	IR149	H'00	H'FFFD 982F	8
Interrupt request register 150*2	IR150	H'00	H'FFFD 9830	8

Notes: The access cycle is two cycles in word access and four cycles in longword access.
1. H'8000 when the NMI pin is high and H'0000 when low.
2. Not used in the SH72A0 group.

Table 8.4 INTC Registers (2)

Register Name	Symbol	After Reset	Address	Access Size
Interrupt request register 151*	IR151	H'00	H'FFFD 9831	8
Interrupt request register 152*	IR152	H'00	H'FFFD 9832	8
Interrupt request register 153	IR153	H'00	H'FFFD 9833	8
Interrupt request register 154	IR154	H'00	H'FFFD 9834	8
Interrupt request register 155*	IR155	H'00	H'FFFD 9835	8
Interrupt request register 156*	IR156	H'00	H'FFFD 9836	8
Interrupt request register 165	IR165	H'00	H'FFFD 983F	8
Interrupt request register 166	IR166	H'00	H'FFFD 9840	8
Interrupt request register 173	IR173	H'00	H'FFFD 9847	8
Interrupt request register 174*	IR174	H'00	H'FFFD 9848	8
Interrupt request register 175*	IR175	H'00	H'FFFD 9849	8
Interrupt request register 176*	IR176	H'00	H'FFFD 984A	8
Interrupt request register 177	IR177	H'00	H'FFFD 984B	8
Interrupt request register 178*	IR178	H'00	H'FFFD 984C	8
Interrupt request register 179	IR179	H'00	H'FFFD 984D	8
Interrupt request register 180	IR180	H'00	H'FFFD 984E	8
Interrupt request register 191*	IR191	H'00	H'FFFD 9859	8
Interrupt request register 192*	IR192	H'00	H'FFFD 985A	8
Interrupt request register 193*	IR193	H'00	H'FFFD 985B	8
Interrupt request register 194*	IR194	H'00	H'FFFD 985C	8
Interrupt request register 195*	IR195	H'00	H'FFFD 985D	8
Interrupt request register 196*	IR196	H'00	H'FFFD 985E	8
Interrupt request register 214	IR214	H'00	H'FFFD 9870	8
Interrupt request register 215	IR215	H'00	H'FFFD 9871	8
Interrupt request register 221	IR221	H'00	H'FFFD 9877	8
Interrupt request register 222	IR222	H'00	H'FFFD 9878	8
Interrupt request register 223	IR223	H'00	H'FFFD 9879	8
Interrupt request register 224	IR224	H'00	H'FFFD 987A	8
Interrupt request register 225	IR225	H'00	H'FFFD 987B	8
Interrupt request register 226	IR226	H'00	H'FFFD 987C	8
Interrupt request register 227	IR227	H'00	H'FFFD 987D	8
Interrupt request register 228	IR228	H'00	H'FFFD 987E	8
Interrupt request register 229	IR229	H'00	H'FFFD 987F	8
Interrupt request register 230	IR230	H'00	H'FFFD 9880	8
Interrupt request register 231	IR231	H'00	H'FFFD 9881	8
Interrupt request register 232	IR232	H'00	H'FFFD 9882	8
Interrupt request register 233	IR233	H'00	H'FFFD 9883	8
Interrupt request register 234	IR234	H'00	H'FFFD 9884	8
Interrupt request register 235	IR235	H'00	H'FFFD 9885	8
Interrupt request register 236	IR236	H'00	H'FFFD 9886	8
Interrupt request register 237	IR237	H'00	H'FFFD 9887	8
Interrupt request register 238	IR238	H'00	H'FFFD 9888	8

Notes: The access cycle is two cycles in word access and four cycles in longword access.
* Not used in the SH72A0 group.

Table 8.5 INTC Registers (3)

Register Name	Symbol	After Reset	Address	Access Size
Interrupt request register 239	IR239	H'00	H'FFFD 9889	8
Interrupt request register 240	IR240	H'00	H'FFFD 988A	8
Interrupt request register 297	IR297	H'00	H'FFFD 98C3	8
Interrupt request register 298	IR298	H'00	H'FFFD 98C4	8
Interrupt request register 299	IR299	H'00	H'FFFD 98C5	8
Interrupt request register 300	IR300	H'00	H'FFFD 98C6	8
Interrupt request register 312	IR312	H'00	H'FFFD 98D2	8
Interrupt request register 313	IR313	H'00	H'FFFD 98D3	8
Interrupt request register 314	IR314	H'00	H'FFFD 98D4	8
Interrupt request register 315	IR315	H'00	H'FFFD 98D5	8
Interrupt request register 316	IR316	H'00	H'FFFD 98D6	8
Interrupt request register 317	IR317	H'00	H'FFFD 98D7	8
Interrupt request register 318	IR318	H'00	H'FFFD 98D8	8
Interrupt request register 319	IR319	H'00	H'FFFD 98D9	8
Interrupt request register 320	IR320	H'00	H'FFFD 98DA	8
Interrupt request register 321	IR321	H'00	H'FFFD 98DB	8
Interrupt request register 322	IR322	H'00	H'FFFD 98DC	8
Interrupt request register 323	IR323	H'00	H'FFFD 98DD	8
Interrupt request register 324	IR324	H'00	H'FFFD 98DE	8
Interrupt request register 325	IR325	H'00	H'FFFD 98DF	8
Interrupt request register 326	IR326	H'00	H'FFFD 98E0	8
Interrupt request register 327	IR327	H'00	H'FFFD 98E1	8
Interrupt request register 328	IR328	H'00	H'FFFD 98E2	8
Interrupt request register 329	IR329	H'00	H'FFFD 98E3	8
Interrupt request register 330	IR330	H'00	H'FFFD 98E4	8
Interrupt request register 331	IR331	H'00	H'FFFD 98E5	8
Interrupt request register 332	IR332	H'00	H'FFFD 98E6	8
Interrupt request register 333	IR333	H'00	H'FFFD 98E7	8
Interrupt request register 334	IR334	H'00	H'FFFD 98E8	8
Interrupt request register 335	IR335	H'00	H'FFFD 98E9	8
Interrupt request register 336	IR336	H'00	H'FFFD 98EA	8
Interrupt request register 337	IR337	H'00	H'FFFD 98EB	8
Interrupt request register 338	IR338	H'00	H'FFFD 98EC	8
Interrupt request register 339	IR339	H'00	H'FFFD 98ED	8
Interrupt request register 366	IR366	H'00	H'FFFD 9908	8
Interrupt request register 367*	IR367	H'00	H'FFFD 9909	8
Interrupt request register 378	IR378	H'00	H'FFFD 9914	8
Interrupt request register 379*	IR379	H'00	H'FFFD 9915	8
Interrupt request register 403	IR403	H'00	H'FFFD 992D	8
Interrupt request register 404	IR404	H'00	H'FFFD 992E	8
Interrupt request register 405	IR405	H'00	H'FFFD 992F	8
Interrupt request register 406	IR406	H'00	H'FFFD 9930	8

Notes: The access cycle is two cycles in word access and four cycles in longword access.
* Not used in the SH72A0 group.

Table 8.6 INTC Registers (4)

Register Name	Symbol	After Reset	Address	Access Size
Interrupt request register 407	IR407	H'00	H'FFFD 9931	8
Interrupt request register 408	IR408	H'00	H'FFFD 9932	8
Interrupt request register 409	IR409	H'00	H'FFFD 9933	8
Interrupt request register 410	IR410	H'00	H'FFFD 9934	8
Interrupt request register 411	IR411	H'00	H'FFFD 9935	8
Interrupt request register 412	IR412	H'00	H'FFFD 9936	8
Interrupt request register 413	IR413	H'00	H'FFFD 9937	8
Interrupt request register 414	IR414	H'00	H'FFFD 9938	8
Interrupt request register 415	IR415	H'00	H'FFFD 9939	8
Interrupt request register 416	IR416	H'00	H'FFFD 993A	8
Interrupt request register 417	IR417	H'00	H'FFFD 993B	8
Interrupt request register 418	IR418	H'00	H'FFFD 993C	8
Interrupt request register 427	IR427	H'00	H'FFFD 9945	8
Interrupt request register 428	IR428	H'00	H'FFFD 9946	8
Interrupt request register 429	IR429	H'00	H'FFFD 9947	8
Interrupt request register 430	IR430	H'00	H'FFFD 9948	8
Interrupt request register 431	IR431	H'00	H'FFFD 9949	8
Interrupt request register 432	IR432	H'00	H'FFFD 994A	8
Interrupt request register 433	IR433	H'00	H'FFFD 994B	8
Interrupt request register 434	IR434	H'00	H'FFFD 994C	8
Interrupt request register 435	IR435	H'00	H'FFFD 994D	8
Interrupt request register 436	IR436	H'00	H'FFFD 994E	8
Interrupt request register 437	IR437	H'00	H'FFFD 994F	8
Interrupt request register 438	IR438	H'00	H'FFFD 9950	8
Interrupt request register 439	IR439	H'00	H'FFFD 9951	8
Interrupt request register 440	IR440	H'00	H'FFFD 9952	8
Interrupt request register 441	IR441	H'00	H'FFFD 9953	8
Interrupt request register 442	IR442	H'00	H'FFFD 9954	8
Interrupt request register 443	IR443	H'00	H'FFFD 9955	8
Interrupt request register 444	IR444	H'00	H'FFFD 9956	8
Interrupt request register 445	IR445	H'00	H'FFFD 9957	8
Interrupt request register 446	IR446	H'00	H'FFFD 9958	8
Interrupt request register 447	IR447	H'00	H'FFFD 9959	8
Interrupt request register 448	IR448	H'00	H'FFFD 995A	8
Interrupt request register 449	IR449	H'00	H'FFFD 995B	8
Interrupt request register 450	IR450	H'00	H'FFFD 995C	8
Interrupt request register 451	IR451	H'00	H'FFFD 995D	8
Interrupt request register 452	IR452	H'00	H'FFFD 995E	8
Interrupt request register 453	IR453	H'00	H'FFFD 995F	8
Interrupt request register 454	IR454	H'00	H'FFFD 9960	8
Interrupt request register 455	IR455	H'00	H'FFFD 9961	8
Interrupt request register 456	IR456	H'00	H'FFFD 9962	8
Interrupt request register 457	IR457	H'00	H'FFFD 9963	8

Note: The access cycle is two cycles in word access and four cycles in longword access.

Table 8.7 INTC Registers (5)

Register Name	Symbol	After Reset	Address	Access Size
Interrupt request register 458	IR458	H'00	H'FFFD 9964	8
Interrupt request register 459	IR459	H'00	H'FFFD 9965	8
Interrupt request register 460	IR460	H'00	H'FFFD 9966	8
Interrupt request register 461	IR461	H'00	H'FFFD 9967	8
Interrupt request register 462	IR462	H'00	H'FFFD 9968	8
Interrupt request register 463	IR463	H'00	H'FFFD 9969	8
Interrupt request register 464	IR464	H'00	H'FFFD 996A	8
Interrupt request register 477	IR477	H'00	H'FFFD 9977	8
Interrupt control register 102	ICR102	Undefined	H'FFFD 999A	16
Interrupt control register 103	ICR103	Undefined	H'FFFD 999C	16
Interrupt control register 104	ICR104	Undefined	H'FFFD 999E	16
Interrupt control register 105	ICR105	Undefined	H'FFFD 99A0	16
Interrupt control register 106	ICR106	Undefined	H'FFFD 99A2	16
Interrupt control register 107	ICR107	Undefined	H'FFFD 99A4	16
Interrupt control register 108	ICR108	Undefined	H'FFFD 99A6	16
Interrupt control register 109	ICR109	Undefined	H'FFFD 99A8	16
Interrupt control register 134	ICR134	Undefined	H'FFFD 99DA	16
Interrupt control register 135	ICR135	Undefined	H'FFFD 99DC	16
Interrupt control register 136	ICR136	Undefined	H'FFFD 99DE	16
Interrupt control register 142	ICR142	Undefined	H'FFFD 99EA	16
Interrupt control register 143	ICR143	Undefined	H'FFFD 99EC	16
Interrupt control register 144	ICR144	Undefined	H'FFFD 99EE	16
Interrupt control register 145	ICR145	Undefined	H'FFFD 99F0	16
Interrupt control register 146	ICR146	Undefined	H'FFFD 99F2	16
Interrupt control register 147	ICR147	Undefined	H'FFFD 99F4	16
Interrupt control register 148	ICR148	Undefined	H'FFFD 99F6	16
Interrupt control register 149*	ICR149	Undefined	H'FFFD 99F8	16
Interrupt control register 150*	ICR150	Undefined	H'FFFD 99FA	16
Interrupt control register 151*	ICR151	Undefined	H'FFFD 99FC	16
Interrupt control register 152*	ICR152	Undefined	H'FFFD 99FE	16
Interrupt control register 153	ICR153	Undefined	H'FFFD 9A00	16
Interrupt control register 154	ICR154	Undefined	H'FFFD 9A02	16
Interrupt control register 155*	ICR155	Undefined	H'FFFD 9A04	16
Interrupt control register 156*	ICR156	Undefined	H'FFFD 9A06	16
Interrupt control register 165	ICR165	Undefined	H'FFFD 9A18	16
Interrupt control register 166	ICR166	Undefined	H'FFFD 9A1A	16
Interrupt control register 173	ICR173	Undefined	H'FFFD 9A28	16
Interrupt control register 174*	ICR174	Undefined	H'FFFD 9A2A	16
Interrupt control register 175*	ICR175	Undefined	H'FFFD 9A2C	16
Interrupt control register 176*	ICR176	Undefined	H'FFFD 9A2E	16
Interrupt control register 177	ICR177	Undefined	H'FFFD 9A30	16
Interrupt control register 178*	ICR178	Undefined	H'FFFD 9A32	16

Notes: The access cycle is two cycles in word access and four cycles in longword access.
* Not used in the SH72A0 group.

Table 8.8 INTC Registers (6)

Register Name	Symbol	After Reset	Address	Access Size
Interrupt control register 179	ICR179	Undefined	H'FFFD 9A34	16
Interrupt control register 180	ICR180	Undefined	H'FFFD 9A36	16
Interrupt control register 191*	ICR191	Undefined	H'FFFD 9A4C	16
Interrupt control register 192*	ICR192	Undefined	H'FFFD 9A4E	16
Interrupt control register 193*	ICR193	Undefined	H'FFFD 9A50	16
Interrupt control register 194*	ICR194	Undefined	H'FFFD 9A52	16
Interrupt control register 195*	ICR195	Undefined	H'FFFD 9A54	16
Interrupt control register 196*	ICR196	Undefined	H'FFFD 9A56	16
Interrupt control register 214	ICR214	Undefined	H'FFFD 9A7A	16
Interrupt control register 215	ICR215	Undefined	H'FFFD 9A7C	16
Interrupt control register 221	ICR221	Undefined	H'FFFD 9A88	16
Interrupt control register 222	ICR222	Undefined	H'FFFD 9A8A	16
Interrupt control register 223	ICR223	Undefined	H'FFFD 9A8C	16
Interrupt control register 224	ICR224	Undefined	H'FFFD 9A8E	16
Interrupt control register 225	ICR225	Undefined	H'FFFD 9A90	16
Interrupt control register 226	ICR226	Undefined	H'FFFD 9A92	16
Interrupt control register 227	ICR227	Undefined	H'FFFD 9A94	16
Interrupt control register 228	ICR228	Undefined	H'FFFD 9A96	16
Interrupt control register 229	ICR229	Undefined	H'FFFD 9A98	16
Interrupt control register 230	ICR230	Undefined	H'FFFD 9A9A	16
Interrupt control register 231	ICR231	Undefined	H'FFFD 9A9C	16
Interrupt control register 232	ICR232	Undefined	H'FFFD 9A9E	16
Interrupt control register 233	ICR233	Undefined	H'FFFD 9AA0	16
Interrupt control register 234	ICR234	Undefined	H'FFFD 9AA2	16
Interrupt control register 235	ICR235	Undefined	H'FFFD 9AA4	16
Interrupt control register 236	ICR236	Undefined	H'FFFD 9AA6	16
Interrupt control register 237	ICR237	Undefined	H'FFFD 9AA8	16
Interrupt control register 238	ICR238	Undefined	H'FFFD 9AAA	16
Interrupt control register 239	ICR239	Undefined	H'FFFD 9AAC	16
Interrupt control register 240	ICR240	Undefined	H'FFFD 9AAE	16
Interrupt control register 297	ICR297	Undefined	H'FFFD 9B20	16
Interrupt control register 298	ICR298	Undefined	H'FFFD 9B22	16
Interrupt control register 299	ICR299	Undefined	H'FFFD 9B24	16
Interrupt control register 300	ICR300	Undefined	H'FFFD 9B26	16
Interrupt control register 312	ICR312	Undefined	H'FFFD 9B3E	16
Interrupt control register 313	ICR313	Undefined	H'FFFD 9B40	16
Interrupt control register 314	ICR314	Undefined	H'FFFD 9B42	16
Interrupt control register 315	ICR315	Undefined	H'FFFD 9B44	16
Interrupt control register 316	ICR316	Undefined	H'FFFD 9B46	16
Interrupt control register 317	ICR317	Undefined	H'FFFD 9B48	16
Interrupt control register 318	ICR318	Undefined	H'FFFD 9B4A	16
Interrupt control register 319	ICR319	Undefined	H'FFFD 9B4C	16

Notes: The access cycle is two cycles in word access and four cycles in longword access.
* Not used in the SH72A0 group.

Table 8.9 INTC Registers (7)

Register Name	Symbol	After Reset	Address	Access Size
Interrupt control register 320	ICR320	Undefined	H'FFFD 9B4E	16
Interrupt control register 321	ICR321	Undefined	H'FFFD 9B50	16
Interrupt control register 322	ICR322	Undefined	H'FFFD 9B52	16
Interrupt control register 323	ICR323	Undefined	H'FFFD 9B54	16
Interrupt control register 324	ICR324	Undefined	H'FFFD 9B56	16
Interrupt control register 325	ICR325	Undefined	H'FFFD 9B58	16
Interrupt control register 326	ICR326	Undefined	H'FFFD 9B5A	16
Interrupt control register 327	ICR327	Undefined	H'FFFD 9B5C	16
Interrupt control register 328	ICR328	Undefined	H'FFFD 9B5E	16
Interrupt control register 329	ICR329	Undefined	H'FFFD 9B60	16
Interrupt control register 330	ICR330	Undefined	H'FFFD 9B62	16
Interrupt control register 331	ICR331	Undefined	H'FFFD 9B64	16
Interrupt control register 332	ICR332	Undefined	H'FFFD 9B66	16
Interrupt control register 333	ICR333	Undefined	H'FFFD 9B68	16
Interrupt control register 334	ICR334	Undefined	H'FFFD 9B6A	16
Interrupt control register 335	ICR335	Undefined	H'FFFD 9B6C	16
Interrupt control register 336	ICR336	Undefined	H'FFFD 9B6E	16
Interrupt control register 337	ICR337	Undefined	H'FFFD 9B70	16
Interrupt control register 338	ICR338	Undefined	H'FFFD 9B72	16
Interrupt control register 339	ICR339	Undefined	H'FFFD 9B74	16
Interrupt control register 366	ICR366	Undefined	H'FFFD 9BAA	16
Interrupt control register 367*	ICR367	Undefined	H'FFFD 9BAC	16
Interrupt control register 378	ICR378	Undefined	H'FFFD 9BC2	16
Interrupt control register 379*	ICR379	Undefined	H'FFFD 9BC4	16
Interrupt control register 403	ICR403	Undefined	H'FFFD 9BF4	16
Interrupt control register 404	ICR404	Undefined	H'FFFD 9BF6	16
Interrupt control register 405	ICR405	Undefined	H'FFFD 9BF8	16
Interrupt control register 406	ICR406	Undefined	H'FFFD 9BFA	16
Interrupt control register 407	ICR407	Undefined	H'FFFD 9BFC	16
Interrupt control register 408	ICR408	Undefined	H'FFFD 9BFE	16
Interrupt control register 409	ICR409	Undefined	H'FFFD 9C00	16
Interrupt control register 410	ICR410	Undefined	H'FFFD 9C02	16
Interrupt control register 411	ICR411	Undefined	H'FFFD 9C04	16
Interrupt control register 412	ICR412	Undefined	H'FFFD 9C06	16
Interrupt control register 413	ICR413	Undefined	H'FFFD 9C08	16
Interrupt control register 414	ICR414	Undefined	H'FFFD 9C0A	16
Interrupt control register 415	ICR415	Undefined	H'FFFD 9C0C	16
Interrupt control register 416	ICR416	Undefined	H'FFFD 9C0E	16
Interrupt control register 417	ICR417	Undefined	H'FFFD 9C10	16
Interrupt control register 418	ICR418	Undefined	H'FFFD 9C12	16
Interrupt control register 427	ICR427	Undefined	H'FFFD 9C24	16
Interrupt control register 428	ICR428	Undefined	H'FFFD 9C26	16

Notes: The access cycle is two cycles in word access and four cycles in longword access.
* Not used in the SH72A0 group.

Table 8.10 INTC Registers (8)

Register Name	Symbol	After Reset	Address	Access Size
Interrupt control register 429	ICR429	Undefined	H'FFFD 9C28	16
Interrupt control register 430	ICR430	Undefined	H'FFFD 9C2A	16
Interrupt control register 431	ICR431	Undefined	H'FFFD 9C2C	16
Interrupt control register 432	ICR432	Undefined	H'FFFD 9C2E	16
Interrupt control register 433	ICR433	Undefined	H'FFFD 9C30	16
Interrupt control register 434	ICR434	Undefined	H'FFFD 9C32	16
Interrupt control register 435	ICR435	Undefined	H'FFFD 9C34	16
Interrupt control register 436	ICR436	Undefined	H'FFFD 9C36	16
Interrupt control register 437	ICR437	Undefined	H'FFFD 9C38	16
Interrupt control register 438	ICR438	Undefined	H'FFFD 9C3A	16
Interrupt control register 439	ICR439	Undefined	H'FFFD 9C3C	16
Interrupt control register 440	ICR440	Undefined	H'FFFD 9C3E	16
Interrupt control register 441	ICR441	Undefined	H'FFFD 9C40	16
Interrupt control register 442	ICR442	Undefined	H'FFFD 9C42	16
Interrupt control register 443	ICR443	Undefined	H'FFFD 9C44	16
Interrupt control register 444	ICR444	Undefined	H'FFFD 9C46	16
Interrupt control register 445	ICR445	Undefined	H'FFFD 9C48	16
Interrupt control register 446	ICR446	Undefined	H'FFFD 9C4A	16
Interrupt control register 447	ICR447	Undefined	H'FFFD 9C4C	16
Interrupt control register 448	ICR448	Undefined	H'FFFD 9C4E	16
Interrupt control register 449	ICR449	Undefined	H'FFFD 9C50	16
Interrupt control register 450	ICR450	Undefined	H'FFFD 9C52	16
Interrupt control register 451	ICR451	Undefined	H'FFFD 9C54	16
Interrupt control register 452	ICR452	Undefined	H'FFFD 9C56	16
Interrupt control register 453	ICR453	Undefined	H'FFFD 9C58	16
Interrupt control register 454	ICR454	Undefined	H'FFFD 9C5A	16
Interrupt control register 455	ICR455	Undefined	H'FFFD 9C5C	16
Interrupt control register 456	ICR456	Undefined	H'FFFD 9C5E	16
Interrupt control register 457	ICR457	Undefined	H'FFFD 9C60	16
Interrupt control register 458	ICR458	Undefined	H'FFFD 9C62	16
Interrupt control register 459	ICR459	Undefined	H'FFFD 9C64	16
Interrupt control register 460	ICR460	Undefined	H'FFFD 9C66	16
Interrupt control register 461	ICR461	Undefined	H'FFFD 9C68	16
Interrupt control register 462	ICR462	Undefined	H'FFFD 9C6A	16
Interrupt control register 463	ICR463	Undefined	H'FFFD 9C6C	16
Interrupt control register 464	ICR464	Undefined	H'FFFD 9C6E	16
Interrupt control register 477	ICR477	Undefined	H'FFFD 9C88	16

Note: The access cycle is two cycles in word access and four cycles in longword access.

8.3.1 Interrupt Priority Level Register i (IPRi) (i = 1 to 4)

Address IPR1: H'FFFD 941A

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	INT0[3:0]				INT1[3:0]				INT2[3:0]				INT3[3:0]			
After Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Address IPR2: H'FFFD 941C

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	INT4[3:0]				INT5[3:0]				INT6[3:0]				INT7[3:0]			
After Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Address IPR3: H'FFFD 941E

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	INT8[3:0]				INT9[3:0]*				INT10[3:0]				INT11[3:0]*			
After Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Address IPR4: H'FFFD 9420

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	INT12[3:0]*				INT13[3:0]*				Reserved							
After Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Note: * These bits are reserved in the SH72A0 group.

Registers IPR1 to IPR4 are 16-bit readable/writable registers in which priority levels from 0 to 15 are set for INT interrupts.

The priority corresponding to each interrupt can be set by setting a value from H'0 (0000) to H'F (1111) to each 4 bit, bits 15 to 12, bits 11 to 8, bits 7 to 4, and bits 3 to 0. If the value is set to H'0, a level of the interrupt priority is set to 0 (lowest). If the value is set to H'F, a level of the interrupt priority is set to 15 (highest). When the priority level is set to 0, the corresponding interrupt is in the same state as disabled.

The reserved bits are read as 0. The write value should be 0.

8.3.2 Interrupt Control Register 0 (ICR0)

Address H'FFFD 9400

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	NMIL	—	—	—	—	—	—	NMIE	—	—	—	—	—	—	—	—
After Reset*		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15	NMIL	NMI Input Level Bit	This bit sets the level of the signal input at the NMI pin. The NMI pin level can be obtained by reading this bit. This bit cannot be modified. 0: Low level is input to NMI pin 1: High level is input to NMI pin	R
b14 to b9	—	Reserved	These bits are read as 0. The write value should be 0.	R
b8	NMIE	NMI Edge Select Bit	This bits selects whether the falling or rising edge of the interrupt request signal on the NMI pin is detected. 0: Interrupt request is detected on falling edge of NMI input 1: Interrupt request is detected on rising edge of NMI input	R/W
b7 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

Note: * 1 when the NMI pin is high, and 0 when the NMI pin is low.

Interrupt control register 0 (ICR0) is a 16-bit register that sets the input signal detection mode for the external interrupt input pin NMI, and indicates the input level at the NMI pin.

8.3.3 Interrupt Control Register 1 (ICR1)

Address H'FFFD 9402

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	INT7S[1:0]		INT6S[1:0]		INT5S[1:0]		INT4S[1:0]		INT3S[1:0]		INT2S[1:0]		INT1S[1:0]		INT0S[1:0]	
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, b14	INT7S[1:0]	INT7 Sense Select Bits	These bits select whether interrupt signals corresponding to pins INT7 to INT0 are detected on a low level, falling edge, rising edge, or both edges. b15b14: b13b12: b11b10: b9 b8 : b7 b6 : b5 b4 : b3 b2 : b1 b0 : 0 0 : Interrupt request is detected on low level of INTn input 0 1 : Interrupt request is detected on falling edge of INTn input 1 0 : Interrupt request is detected on rising edge of INTn input 1 1 : Interrupt request is detected on both edges of INTn input	R/W
b13, b12	INT6S[1:0]	INT6 Sense Select Bits		R/W
b11, b10	INT5S[1:0]	INT5 Sense Select Bits		R/W
b9, b8	INT4S[1:0]	INT4 Sense Select Bits		R/W
b7, b6	INT3S[1:0]	INT3 Sense Select Bits		R/W
b5, b4	INT2S[1:0]	INT2 Sense Select Bits		R/W
b3, b2	INT1S[1:0]	INT1 Sense Select Bits		R/W
b1, b0	INT0S[1:0]	INT0 Sense Select Bits		R/W

Note: n = 7 to 0

Interrupt control register 1 (ICR1) is a 16-bit register that specifies the detection mode for external interrupt input pins INT7 to INT0 individually: low level, falling edge, rising edge, or both edges.

8.3.4 Interrupt Control Register 2 (ICR2)

Address H'FFFD 9404

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	INT13S [1:0]*		INT12S [1:0]*		INT11S [1:0]*		INT10S [1:0]		INT9S [1:0]*		INT8S [1:0]	
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 b12	—	Reserved	These bits are read as 0. The write value should be 0.	R
b11, b10	INT13S [1:0]*	INT13 Sense Select Bits	These bits select whether interrupt signals corresponding to pins INT13 to INT8 are detected on a low level, falling edge, rising edge, or both edges. b11b10: b9 b8 : b7 b6 : b5 b4 : b3 b2 : b1 b0 : 0 0 : Interrupt request is detected on low level of INTn input 0 1 : Interrupt request is detected on falling edge of INTn input 1 0 : Interrupt request is detected on rising edge of INTn input 1 1 : Interrupt request is detected on both edges of INTn input	R/W
b9, b8	INT12S [1:0]*	INT12 Sense Select Bits		R/W
b7, b6	INT11S [1:0]*	INT11 Sense Select Bits		R/W
b5, b4	INT10S [1:0]	INT10 Sense Select Bits		R/W
b3, b2	INT9S [1:0]*	INT9 Sense Select Bits		R/W
b1, b0	INT8S [1:0]	INT8 Sense Select Bits		R/W

Notes: n = 13 to 8
* These bits are reserved in the SH72A0 group.

Interrupt control register 2 (ICR2) is a 16-bit register that specifies the detection mode for external interrupt input pins INT13 to INT8 individually: low level, falling edge, rising edge, or both edges.

8.3.5 INT Interrupt Request Register (INTR)

Address H'FFFD 9408

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	INT13F*	INT12F*	INT11F*	INT10F	INT9F*	INT8F	INT7F	INT6F	INT5F	INT4F	INT3F	INT2F	INT1F	INT0F
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, b14	—	Reserved	These bits are read as 0. The write value should be 0.	R
b13	INT13F*	INT13 Interrupt Request Bit	Edge detection (INTnS bits in the ICR1 or ICR2 register = 01, 10, 11) 0: INTn interrupt request not detected 1: INTn interrupt detected [Conditions to become 0] • When writing a 0 after reading as 1 • When the INTn interrupt is accepted [Condition to become 1] • When an edge corresponding to the INTn pin is input Level detection (INTnS bits in the ICR1 or ICR2 register = 00) 0: INTn interrupt request not detected 1: INTn interrupt detected [Condition to become 0] • INTn input is high [Condition to become 1] • INTn input is low Writing is invalid	R/W
b12	INT12F*	INT12 Interrupt Request Bit		R/W
b11	INT11F*	INT11 Interrupt Request Bit		R/W
b10	INT10F	INT10 Interrupt Request Bit		R/W
b9	INT9F*	INT9 Interrupt Request Bit		R/W
b8	INT8F	INT8 Interrupt Request Bit		R/W
b7	INT7F	INT7 Interrupt Request Bit		R/W
b6	INT6F	INT6 Interrupt Request Bit		R/W
b5	INT5F	INT5 Interrupt Request Bit		R/W
b4	INT4F	INT4 Interrupt Request Bit		R/W
b3	INT3F	INT3 Interrupt Request Bit		R/W
b2	INT2F	INT2 Interrupt Request Bit		R/W
b1	INT1F	INT1 Interrupt Request Bit		R/W
b0	INT0F	INT0 Interrupt Request Bit		R/W

Notes: When a pin corresponding to INTn is assigned to any function other than the INTn interrupt function, the corresponding bit may be set to 1.
When reading this register, use a program to perform masking or other processing to any bit which not used for the INTn interrupt so that the bit will be ignored.
n = 13 to 0
* This bit is reserved in the SH72A0 group.

The interrupt request register (INTR) is a 16-bit register that indicates interrupt requests from the INT pins.
If edge detection is set for the INT15 to INT0 interrupts, writing 0 to the INT13 to INT0 bits after reading them as 1 cancels the retained interrupts.

8.3.6 Bank Control Register (BCR)

Address H'FFFD 940E

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	E15	E14	E13	E12	E11	E10	E9	E8	E7	E6	E5	E4	E3	E2	E1	—
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	E15	Interrupt Priority Level 15 Register Bank Use Enable Bit	0: Use of register banks is disabled 1: Use of register banks is enabled	R/W
b14	E14	Interrupt Priority Level 14 Register Bank Use Enable Bit		R/W
b13	E13	Interrupt Priority Level 13 Register Bank Use Enable Bit		R/W
b12	E12	Interrupt Priority Level 12 Register Bank Use Enable Bit		R/W
b11	E11	Interrupt Priority Level 11 Register Bank Use Enable Bit		R/W
b10	E10	Interrupt Priority Level 10 Register Bank Use Enable Bit		R/W
b9	E9	Interrupt Priority Level 9 Register Bank Use Enable Bit		R/W
b8	E8	Interrupt Priority Level 8 Register Bank Use Enable Bit		R/W
b7	E7	Interrupt Priority Level 7 Register Bank Use Enable Bit		R/W
b6	E6	Interrupt Priority Level 6 Register Bank Use Enable Bit		R/W
b5	E5	Interrupt Priority Level 5 Register Bank Use Enable Bit		R/W
b4	E4	Interrupt Priority Level 4 Register Bank Use Enable Bit		R/W
b3	E3	Interrupt Priority Level 3 Register Bank Use Enable Bit		R/W
b2	E2	Interrupt Priority Level 2 Register Bank Use Enable Bit		R/W
b1	E1	Interrupt Priority Level 1 Register Bank Use Enable Bit		R/W
b0	—	Reserved	This bit is read as 0. The write value should be 0.	R

The bank control register (BCR) is a 16-bit register that enables or disables use of register banks for each interrupt priority level. The BCR is register is enabled when the BE bits in the BNR register are set to 11.

8.3.7 Bank Number Register (BNR)

Address H'FFFD 9410

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	BE[1:0]		BOVE	—	—	—	—	—	—	—	—	—	BN[3:0]			
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, b14	BE[1:0]	Register Bank Enable Bits	b15b14 0 0 : Use of register banks is disabled for all interrupts. The setting of the BCR register is ignored. 0 1 : Use of register banks is enabled for all interrupts except NMI and user break. The setting of the BCR register is ignored. 1 0 : Do not set. 1 1 : Use of register banks is controlled by the setting of the BCR register.	R/W
b13	BOVE	Register Bank Overflow Enable Bit	0: Generation of register bank overflow exception is disabled 1: Generation of register bank overflow exception is enabled	R/W
b12 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R
b3 to b0	BN[3:0]	Bank Number Bits	These bits indicate the bank number to which saving is performed next. When an interrupt using register banks is received, saving is performed to the register bank indicated by these bits, and BN is incremented by 1. After BN is decremented by 1 due to execution of a RESBANK (restore from register bank) instruction, restoring from the register bank is performed.	R

The bank number register (BNR) is a 16-bit register that enables or disables use of register banks and register bank overflow exception. The BNR register also indicates the bank number to which saving is performed next through the BN bits.

8.3.8 Software Interrupt Register i (SINTi) (i = 1 to 8)

Address SINT1: H'FFFD 9412, SINT2: H'FFFD 9413, SINT3: H'FFFD 9414, SINT4: H'FFFD 9415, SINT5: H'FFFD 9416, SINT6: H'FFFD 9417, SINT7: H'FFFD 9418, SINT8: H'FFFD 9419

	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	0	0	0	0	0	0	0	0

Bit	Description	R/W
b7 to b0	[Read operation] The counter value of SINTi is read. [Write operation] Writing H'01: The counter is incremented by 1. *1 Writing H'00: The counter is decremented by 1. *2	R/W

Notes: 1. When the counter is incremented by 1 while its value is H'FF, the counter is not incremented and the value remains as H'FF.
2. When the counter is decremented by 1 while its value is H'00, the counter is not incremented and the value remains as H'00.

Software interrupt registers (SINT8 to SINT1) are 8-bit registers that control software interrupts 8 to 1 (SINT8 to SINT1) and count the number of requests. The counter value is incremented by 1 by writing H'01 to these registers. The counter value is decremented by 1 by writing H'00 to these registers. When the counter value of these registers is 1 or greater, software interrupts 8 to 1 (SINT8 to SINT1) are generated. When these registers are read, the current counter value is read. Do not write a value other than H'00 and H'01 to these registers.

8.3.9 Interrupt Request Register i (IRi) (i = 102 to 109, 134 to 136, 142 to 156, 165 to 166, 173 to 180, 191 to 196, 214 to 215, 221 to 240, 297 to 300, 312 to 339, 366 to 367, 378 to 379, 403 to 418, 427 to 464, 477)

Address

IR102 to IR109: H'FFFD 9800 to H'FFFD 9807
IR134 to IR136: H'FFFD 9820 to H'FFFD 9822
IR142 to IR156: H'FFFD 9828 to H'FFFD 9836
IR165 to IR166: H'FFFD 983F to H'FFFD 9840
IR173 to IR180: H'FFFD 9847 to H'FFFD 984E
IR191 to IR196: H'FFFD 9859 to H'FFFD 985E
IR214 to IR215: H'FFFD 9870 to H'FFFD 9871
IR221 to IR240: H'FFFD 9877 to H'FFFD 988A
IR297 to IR300: H'FFFD 98C3 to H'FFFD 98C6
IR312 to IR339: H'FFFD 98D2 to H'FFFD 98ED
IR366 to IR367: H'FFFD 9908 to H'FFFD 9909
IR378 to IR379: H'FFFD 9914 to H'FFFD 9915
IR403 to IR418: H'FFFD 992D to H'FFFD 993C
IR427 to IR464: H'FFFD 9945 to H'FFFD 996A
IR477: H'FFFD 9977

	b7	b6	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	IR
After Reset	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R
b0	IR	Interrupt Request Bit	For edge type* 0: Interrupt request not detected 1: Interrupt request detected [Conditions to become 0] • When writing 0 • When the interrupt is accepted [Condition to become 1] • Interrupt request is generated For level type* 0: Interrupt request not detected 1: Interrupt request detected [Condition to become 0] • Cleared to 0 automatically when the interrupt request from the peripheral function of interrupt request source is cleard. [Condition to become 1] • When an interrupt request is generated Writing is invalid	R/W

Note: * Edge type and level type are available as the ICU input source types. For the level type, the IR bit cannot be cleared to 0 by software. Refer to Table 8.11 to Table 8.16 Interrupt Sources and Interrupt Vectors, for details on the ICU input source type for each interrupt source.

Interrupt request registers 102 to 511 (IR102 to IR511) are 8-bit registers that indicate the status of interrupt requests generated from the peripheral functions.

8.3.10 Interrupt Control Register i (ICRi) (i = 102 to 109, 134 to 136, 142 to 156, 165 to 166, 173 to 180, 191 to 196, 214 to 215, 221 to 240, 297 to 300, 312 to 339, 366 to 367, 378 to 379, 403 to 418, 427 to 464, 477)

Address

ICR102 to ICR109: H'FFFD 999A to H'FFFD 99A8
ICR134 to ICR136: H'FFFD 99DA to H'FFFD 99DE
ICR142 to ICR156: H'FFFD 99EA to H'FFFD 9A06
ICR165 to ICR166: H'FFFD 9A18 to H'FFFD 9A1A
ICR173 to ICR180: H'FFFD 9A28 to H'FFFD 9A36
ICR191 to ICR196: H'FFFD 9A4C to H'FFFD 9A56
ICR214 to ICR215: H'FFFD 9A7A to H'FFFD 9A7C
ICR221 to ICR240: H'FFFD 9A88 to H'FFFD 9AAE
ICR297 to ICR300: H'FFFD 9B20 to H'FFFD 9B26
ICR312 to ICR339: H'FFFD 9B3E to H'FFFD 9B74
ICR366 to ICR367: H'FFFD 9BAA to H'FFFD 9BAC
ICR378 to ICR379: H'FFFD 9BC2 to H'FFFD 9BC4
ICR403 to ICR418: H'FFFD 9BF4 to H'FFFD 9C12
ICR427 to ICR464: H'FFFD 9C24 to H'FFFD 9C6E
ICR477: H'FFFD 9C88

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	INTEN	—	—	—	—	—	—	—	—	—	—	—	IPR[3:0]			
After Reset	1	0	0	0	x	x	x	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15	INTEN	Interrupt Enable Bit	This bit enables or disables the interrupt request input. 0: Interrupt disabled 1: Interrupt enabled	R/W
b14 to b12	—	Reserved	These bits are read as 0. The write value should be 0.	R
b11 to b9	—	Reserved	The read value is undefined. The write value should be 0.	R
b8 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R
b3 to b0	IPR[3:0]	Interrupt Priority Level Bits	These bits set the interrupt priority (levels 0 to 15) for the peripheral function interrupts. The priority corresponding to each interrupt can be set by setting a value from H'0 (0000) to H'F (1111). If the value is set to H'0, a level of the interrupt priority is set to 0 (lowest). If the value is set to H'F, a level of the interrupt priority is set to 15 (highest).	R/W

The ICRi register is a 16-bit register that enables or disables the interrupt request input and sets interrupt priority levels.

8.4 Interrupt Sources

There are nine types of interrupt sources: voltage monitor, NMI, user break, DMAC access MPU error, RAM error, FIF error, software, INT, and peripheral function interrupts. Each interrupt has a priority level (0 to 16). A level of 0 corresponds to the lowest and a level of 16 corresponds to the highest. When the level is set to 0, the interrupt is masked at all times.

8.4.1 Voltage Monitor Interrupt

The voltage monitor interrupt has a priority level of 16 with two sources of voltage monitor interrupt 1 and main clock oscillation stop detection interrupt. This is received at all times.
Although the priority level of the voltage monitor interrupt is set to 16, the mask bits (I3 to I0) in the status register (SR) is set to 15 in the voltage monitor interrupt exception handler.

(1) Voltage monitor interrupt 1
One of these interrupts occurs when the voltage monitor function detects changes in the voltage input to the VCC pin. For details on the voltage monitor function, refer to section 9, Voltage Monitor Function.
(2) Main clock oscillation stop detection interrupt
Occurs when main clock stop detection function detects the oscillation stop of the main clock. For details on the main clock oscillation stop detection function, refer to section 5, Clocks.

8.4.2 NMI Interrupt

The NMI interrupt has a priority level of 16 and is received at all times. The edge of the NMI signal is detected as an NMI interrupt and the NMI edge select bit (NMIE) in interrupt control register 0 (ICR0) selects whether the rising edge or falling edge is detected.
Although the priority level of the NMI interrupt is 16, the interrupt mask level bits (I3 to I0) in the status register (SR) are set to level 15 in the NMI interrupt exception handler.

8.4.3 User Break Interrupt

A user break interrupt occurs when a break condition set in the user break controller (UBC) is satisfied and has a priority level of 15. A user break interrupt request is detected on the edge and held until the interrupt is received. Bits I3 to I0 in SR is set to level 15 in the user break interrupt exception handler.

8.4.4 DMAC Access MPU Error Interrupts

An interrupt request is generated when DMAC access violates areas set up for protection by the MPU. For details, see section 29, Memory Protection Unit (MPU).

8.4.5 RAM Error Interrupts

For details on RAM error interrupt sources, see section 27, RAM Control.

8.4.6 FIFE Interrupts

For details on FIFE interrupt sources, see section 24, ROM.

8.4.7 Software Interrupts

A software interrupt request can be generated by writing to a software interrupt register (SINT8 to SINT1).

8.4.8 INT Interrupts

An INT interrupt is generated by inputting to pins INT13 to INT0. For INT7 to INT0, the low level, falling edges, rising edge, or both edge of the INT signals is detected and the edge to be detected can be selected individually for each pin via the setting of the INT sense select bits (INTn0S to INTn1S) (n = 7 to 0) in interrupt control register 1 (ICR1). For INT13 to INT8, the low level, falling edge, rising edge, or both edges of the INT signals is detected and the edge to be detected can be selected individually for each pin via the setting of the INT sense select bits (INTn0S to INTn1S) (n = 13 to 8) in interrupt control register 2 (ICR2).
Each input circuit from INT0 to INT13 incorporates a digital filter. The INTFCKD bits in the INTDFCR1 register select the sampling clock. Levels being input on INT0 to INT13 are sampled on each cycle of the sampling clock, and a new level is conveyed internally when a sequence of three sampled levels matches.
Figure 8.2 shows the Configuration of the Digital Filters for INT0 to INT13.

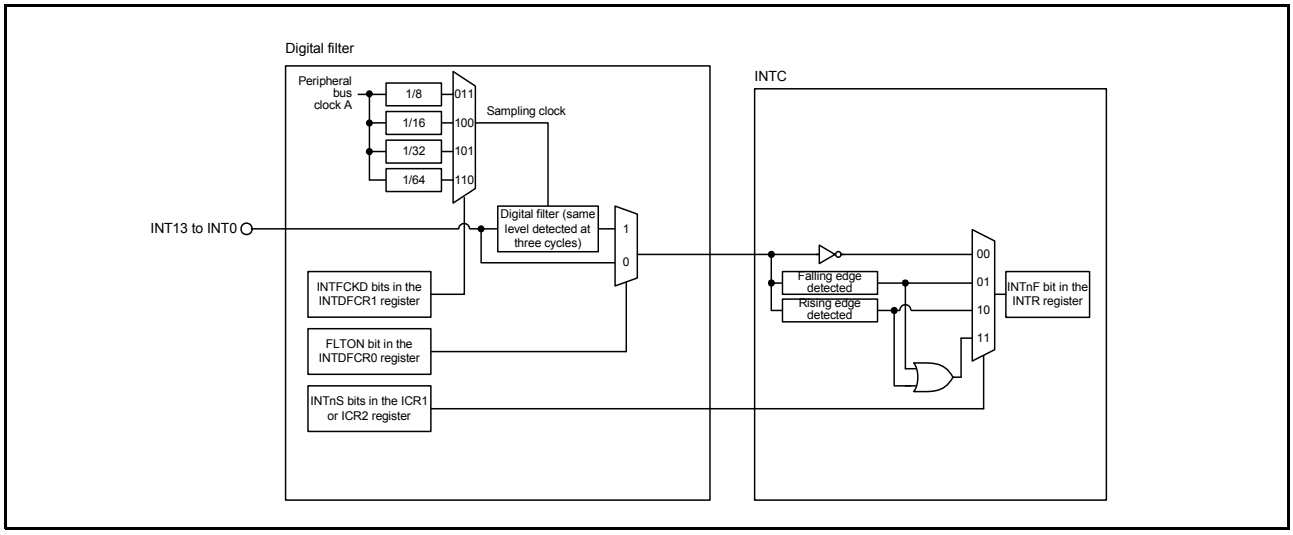


Figure 8.2 Configuration of the Digital Filters for INT0 to INT13

The priority level can be set individually in a range from 0 to 15 for each pin via interrupt priority registers 1 and 4 (IPR1 to IPR4).

When using the low-level sensing for INT13 to INT0, an interrupt is requested to the INTC while the INT pins are driven low. When the INT pins are driven high, the interrupt stops to be requested. Whether or not an INT interrupt has occurred can be checked by reading the INT interrupt request bits (INT13F to INT0F) in the INT interrupt request register (INTR).

When using an edge sensing for INT13 to INT0, an interrupt is requested to the INTC when an interrupt request is detected due to changes in pins INT13 to INT0.

The interrupt request is held until the interrupt is received. Whether or not an interrupt has requested can be checked by reading bits INT13F to INT0F in the INTR register. The request can be cleared by writing 0 to these bits after reading them as 1.

In the INT interrupt exception handler, bits I3 to I0 in SR are set to the priority level of the received INT interrupt.

8.4.9 Peripheral Function Interrupts

The following on-chip peripheral modules can generate on-chip peripheral module interrupts.

- DMAC interrupt
- Peripheral function interrupt

Since each source is assigned to a unique interrupt vector, the source does not need to be identified in the interrupt exception handler. A priority level in a range from 0 to 15 can be set for each module via the IPR bits in interrupt priority registers 102 to 511 (ICR102 to ICR511). In the exception handler for the on-chip peripheral module interrupt, bits I3 to I0 in SR is set to the priority level of the received on-chip peripheral module interrupt.
For details on the assignment of interrupt sources, refer to section 8.4.10, Interrupt Exception Handling Vector Table and Priority.

8.4.10 Interrupt Exception Handling Vector Table and Priority

Table 8.11 to Table 8.16 lists interrupt sources, their vector numbers, vector table address offsets, and priority levels. Each interrupt source is assigned to a unique vector number and a unique vector table address offset. Vector table addresses are calculated from the vector numbers and vector table address offsets. In the interrupt exception handler, the start address is fetched from the vector table pointed to by the vector table address. For details on calculation of the vector table address, refer to Table 6.4 Calculation of Exception Handling Vector Table Addresses in section 6, Exception Handling.

The priorities of INT interrupts and peripheral function interrupts can be set freely between 0 and 15 for each pin or peripheral function via interrupt priority registers 1 to 4 (IPR1 to IPR4) and the IPR bits in interrupt control registers 102 to 511 (ICR102 to ICR511). The priorities of INT interrupts and peripheral function interrupts are set to priority level 0 by a reset.
If the same priority level is assigned to two or more interrupt sources and interrupts from those sources occur simultaneously, they are processed according to the default priority levels shown in “Default Priority” in Table 8.11 to Table 8.16.

Table 8.11 Interrupt Sources and Vectors (1)

Interrupt Source		Interrupt Vector		Interrupt Priority (Initial Value)	Corresponding IPR (Bit)		Corre-spond-ing IR	ICU Input Source Type	Default Priority
		Vector No.	Vector Table Address Offset						
Excep-tions and reserved	Voltage monitor interrupt 1 or main clock oscillation stop detection interrupt (vector 7 shared)	7	H'0000 001C to H'0000 001F	16	—		—	Edge	High
	NMI interrupt	11	H'0000 002C to H'0000 002F	16	—		—		
	User break interrupt	12	H'0000 0030 to H'0000 0033	15	—		—	Level	
INT	INT0 pin interrupt	64	H'0000 0100 to H'0000 0103	0 to 15 (0)	IPR1	(15 to 12)	—	Edge/ Level	
	INT1 pin interrupt	65	H'0000 0104 to H'0000 0107	0 to 15 (0)		(11 to 8)	—		
	INT2 pin interrupt	66	H'0000 0108 to H'0000 010B	0 to 15 (0)		(7 to 4)	—		
	INT3 pin interrupt	67	H'0000 010C to H'0000 010F	0 to 15 (0)		(3 to 0)	—		
	INT4 pin interrupt	68	H'0000 0110 to H'0000 0113	0 to 15 (0)	IPR2	(15 to 12)	—		
	INT5 pin interrupt	69	H'0000 0114 to H'0000 0117	0 to 15 (0)		(11 to 8)	—		
	INT6 pin interrupt	70	H'0000 0118 to H'0000 011B	0 to 15 (0)		(7 to 4)	—		
	INT7 pin interrupt	71	H'0000 011C to H'0000 011F	0 to 15 (0)		(3 to 0)	—		
	INT8 pin interrupt	72	H'0000 0120 to H'0000 0123	0 to 15 (0)	IPR3	(15 to 12)	—		
	INT9 pin interrupt*	73	H'0000 0124 to H'0000 0127	0 to 15 (0)		(11 to 8)	—		
	INT10 pin interrupt	74	H'0000 0128 to H'0000 012B	0 to 15 (0)		(7 to 4)	—		
	INT11 pin interrupt*	75	H'0000 012C to H'0000 012F	0 to 15 (0)		(3 to 0)	—		
	INT12 pin interrupt*	76	H'0000 0130 to H'0000 0133	0 to 15 (0)	IPR4	(15 to 12)	—		
	INT13 pin interrupt*	77	H'0000 0134 to H'0000 0137	0 to 15 (0)		(11 to 8)	—		
RAM	RAM error interrupt	88	H'0000 0160 to H'0000 0163	15	—		—	Level	
(Reserved)		89 to 90	—					—	
ROM	FIF error interrupt	91	H'0000 016C to H'0000 016F	15	—		—	Level	
MPU	DMAC access MPU error interrupt	92	H'0000 0170 to H'0000 0173	15	—		—	Level	
(Reserved)		93	—					—	
SINT	SINT8 interrupt	94	H'0000 0178 to H'0000 017B	8	—		—	—	
	SINT7 interrupt	95	H'0000 017C to H'0000 017F	7	—		—		
	SINT6 interrupt	96	H'0000 0180 to H'0000 0183	6	—		—		
	SINT5 interrupt	97	H'0000 0184 to H'0000 0187	5	—		—		
	SINT4 interrupt	98	H'0000 0188 to H'0000 018B	4	—		—	—	
	SINT3 interrupt	99	H'0000 018C to H'0000 018F	3	—		—		Low

Note: * Not used in the SH72A0 group.

Table 8.12 Interrupt Sources and Vectors (2)

Interrupt Source		Interrupt Vector		Interrupt Priority (Initial Value)	Corresponding IPR (Bit)		Corre-spond-ing IR	ICU Input Source Type	Default Priority
		Vector No.	Vector Table Address Offset						
SINT	SINT2 interrupt	100	H'0000 0190 to H'0000 0193	2	—		—	—	High
	SINT1 interrupt	101	H'0000 0194 to H'0000 0197	1	—		—		
DMAC	DMA0 transfer complete interrupt	102	H'0000 0198 to H'0000 019B	0 to 15 (0)	ICR102	(3 to 0)	IR102	Edge	
	DMA1 transfer complete interrupt	103	H'0000 019C to H'0000 019F	0 to 15 (0)	ICR103	(3 to 0)	IR103		
	DMA2 transfer complete interrupt	104	H'0000 01A0 to H'0000 01A3	0 to 15 (0)	ICR104	(3 to 0)	IR104		
	DMA3 transfer complete interrupt	105	H'0000 01A4 to H'0000 01A7	0 to 15 (0)	ICR105	(3 to 0)	IR105		
	DMA4 transfer complete interrupt	106	H'0000 01A8 to H'0000 01AB	0 to 15 (0)	ICR106	(3 to 0)	IR106		
	DMA5 transfer complete interrupt	107	H'0000 01AC to H'0000 01AF	0 to 15 (0)	ICR107	(3 to 0)	IR107		
	DMA6 transfer complete interrupt	108	H'0000 01B0 to H'0000 01B3	0 to 15 (0)	ICR108	(3 to 0)	IR108		
	DMA7 transfer complete interrupt	109	H'0000 01B4 to H'0000 01B7	0 to 15 (0)	ICR109	(3 to 0)	IR109		
(Reserved)		110 to 133	—					—	
POE	POE0 interrupt	134	H'0000 0218 to H'0000 021B	0 to 15 (0)	ICR134	(3 to 0)	IR134	Edge	
	POE1 interrupt	135	H'0000 021C to H'0000 021F	0 to 15 (0)	ICR135	(3 to 0)	IR135		
	POE2 interrupt	136	H'0000 0220 to H'0000 0223	0 to 15 (0)	ICR136	(3 to 0)	IR136		
(Reserved)		137 to 141	—					—	
CMT	CMT0 interrupt (CMI0)	142	H'0000 0238 to H'0000 023B	0 to 15 (0)	ICR142	(3 to 0)	IR142	Edge	
	CMT1 interrupt (CMI1)	143	H'0000 023C to H'0000 023F	0 to 15 (0)	ICR143	(3 to 0)	IR143		
	CMT2 interrupt (CMI2)	144	H'0000 0240 to H'0000 0243	0 to 15 (0)	ICR144	(3 to 0)	IR144		
	CMT3 interrupt (CMI3)	145	H'0000 0244 to H'0000 0247	0 to 15 (0)	ICR145	(3 to 0)	IR145		
	CMT4 interrupt (CMI4)	146	H'0000 0248 to H'0000 024B	0 to 15 (0)	ICR146	(3 to 0)	IR146		
	CMT5 interrupt (CMI5)	147	H'0000 024C to H'0000 024F	0 to 15 (0)	ICR147	(3 to 0)	IR147		
AD1	AD1 scan conversion end interrupt	148	H'0000 0250 to H'0000 0253	0 to 15 (0)	ICR148	(3 to 0)	IR148	Edge	
	AD1IN0 interrupt conversion end interrupt*	149	H'0000 0254 to H'0000 0257	0 to 15 (0)	ICR149	(3 to 0)	IR149		
	AD1IN1 interrupt conversion end interrupt*	150	H'0000 0258 to H'0000 025B	0 to 15 (0)	ICR150	(3 to 0)	IR150		
	AD1IN2 interrupt conversion end interrupt*	151	H'0000 025C to H'0000 025F	0 to 15 (0)	ICR151	(3 to 0)	IR151		
	AD1IN3 interrupt conversion end interrupt*	152	H'0000 0260 to H'0000 0263	0 to 15 (0)	ICR152	(3 to 0)	IR152		
	AD1IN4 interrupt conversion end interrupt	153	H'0000 0264 to H'0000 0267	0 to 15 (0)	ICR153	(3 to 0)	IR153		
	AD1IN5 interrupt conversion end interrupt	154	H'0000 0268 to H'0000 026B	0 to 15 (0)	ICR154	(3 to 0)	IR154		
	AD1IN6 interrupt conversion end interrupt*	155	H'0000 026C to H'0000 026F	0 to 15 (0)	ICR155	(3 to 0)	IR155		
	AD1IN7 interrupt conversion end interrupt*	156	H'0000 0270 to H'0000 0273	0 to 15 (0)	ICR156	(3 to 0)	IR156		
(Reserved)		157 to 164	—					—	
AD1	AD1IN16 interrupt conversion end interrupt	165	H'0000 0294 to H'0000 0297	0 to 15 (0)	ICR165	(3 to 0)	IR165	Edge	
	AD1IN17 interrupt conversion end interrupt	166	H'0000 0298 to H'0000 029B	0 to 15 (0)	ICR166	(3 to 0)	IR166		
(Reserved)		167 to 172	—					—	
AD1	AD1IN24 interrupt conversion end interrupt	173	H'0000 02B4 to H'0000 02B7	0 to 15 (0)	ICR173	(3 to 0)	IR173	Edge	
	AD1IN25 interrupt conversion end interrupt*	174	H'0000 02B8 to H'0000 02BB	0 to 15 (0)	ICR174	(3 to 0)	IR174		
	AD1IN26 interrupt conversion end interrupt*	175	H'0000 02BC to H'0000 02BF	0 to 15 (0)	ICR175	(3 to 0)	IR175		Low

Note: * Not used in the SH72A0 group.

Table 8.13 Interrupt Sources and Vectors (3)

Interrupt Source		Interrupt Vector		Interrupt Priority (Initial Value)	Corresponding IPR (Bit)		Corre-spond-ing IR	ICU Input Source Type	Default Priority
		Vector No.	Vector Table Address Offset						
AD1	AD1IN27 interrupt conversion end interrupt*	176	H'0000 02C0 to H'0000 02C3	0 to 15 (0)	ICR176	(3 to 0)	IR176	Edge	High
	AD1IN28 interrupt conversion end interrupt	177	H'0000 02C4 to H'0000 02C7	0 to 15 (0)	ICR177	(3 to 0)	IR177		
	AD1IN29 interrupt conversion end interrupt*	178	H'0000 02C8 to H'0000 02CB	0 to 15 (0)	ICR178	(3 to 0)	IR178		
	AD1IN30 interrupt conversion end interrupt	179	H'0000 02CC to H'0000 02CF	0 to 15 (0)	ICR179	(3 to 0)	IR179		
	AD1IN31 interrupt conversion end interrupt	180	H'0000 02D0 to H'0000 02D3	0 to 15 (0)	ICR180	(3 to 0)	IR180		
(Reserved)		181 to 190	—					—	
AD1	AD1IN42 interrupt conversion end interrupt*	191	H'0000 02FC to H'0000 02FF	0 to 15 (0)	ICR191	(3 to 0)	IR191	Edge	
	AD1IN43 interrupt conversion end interrupt*	192	H'0000 0300 to H'0000 0303	0 to 15 (0)	ICR192	(3 to 0)	IR192		
	AD1IN44 interrupt conversion end interrupt*	193	H'0000 0304 to H'0000 0307	0 to 15 (0)	ICR193	(3 to 0)	IR193		
	AD1IN45 interrupt conversion end interrupt*	194	H'0000 0308 to H'0000 030B	0 to 15 (0)	ICR194	(3 to 0)	IR194		
	AD1IN46 interrupt conversion end interrupt*	195	H'0000 030C to H'0000 030F	0 to 15 (0)	ICR195	(3 to 0)	IR195		
	AD1IN47 interrupt conversion end interrupt*	196	H'0000 0310 to H'0000 0313	0 to 15 (0)	ICR196	(3 to 0)	IR196		
(Reserved)		197 to 213	—					—	
AD0	AD0 scan conversion end interrupt	214	H'0000 0358 to H'0000 035B	0 to 15 (0)	ICR214	(3 to 0)	IR214	Edge	
	AD0 parity error interrupt	215	H'0000 035C to H'0000 035F	0 to 15 (0)	ICR215	(3 to 0)	IR215		
(Reserved)		216 to 220	—					—	
TPU	TP1GR0 interrupt	221	H'0000 0374 to H'0000 0377	0 to 15 (0)	ICR221	(3 to 0)	IR221	Edge	
	TP1GR1 interrupt	222	H'0000 0378 to H'0000 037B	0 to 15 (0)	ICR222	(3 to 0)	IR222		
	TP1GR2 interrupt	223	H'0000 037C to H'0000 037F	0 to 15 (0)	ICR223	(3 to 0)	IR223		
	TP1GR3 interrupt	224	H'0000 0380 to H'0000 0383	0 to 15 (0)	ICR224	(3 to 0)	IR224		
	TPU1 counter overflow interrupt	225	H'0000 0384 to H'0000 0387	0 to 15 (0)	ICR225	(3 to 0)	IR225		
	TP2GR0 interrupt	226	H'0000 0388 to H'0000 038B	0 to 15 (0)	ICR226	(3 to 0)	IR226		
	TP2GR1 interrupt	227	H'0000 038C to H'0000 038F	0 to 15 (0)	ICR227	(3 to 0)	IR227		
	TP2GR2 interrupt	228	H'0000 0390 to H'0000 0393	0 to 15 (0)	ICR228	(3 to 0)	IR228		
	TP2GR3 interrupt	229	H'0000 0394 to H'0000 0397	0 to 15 (0)	ICR229	(3 to 0)	IR229		
	TPU2 counter overflow interrupt	230	H'0000 0398 to H'0000 039B	0 to 15 (0)	ICR230	(3 to 0)	IR230		
	TP3GR0 interrupt	231	H'0000 039C to H'0000 039F	0 to 15 (0)	ICR231	(3 to 0)	IR231		
	TP3GR1 interrupt	232	H'0000 03A0 to H'0000 03A3	0 to 15 (0)	ICR232	(3 to 0)	IR232		
	TP3GR2 interrupt	233	H'0000 03A4 to H'0000 03A7	0 to 15 (0)	ICR233	(3 to 0)	IR233		
	TP3GR3 interrupt	234	H'0000 03A8 to H'0000 03AB	0 to 15 (0)	ICR234	(3 to 0)	IR234		
	TPU3 counter overflow interrupt	235	H'0000 03AC to H'0000 03AF	0 to 15 (0)	ICR235	(3 to 0)	IR235		
	TP4GR0 interrupt	236	H'0000 03B0 to H'0000 03B3	0 to 15 (0)	ICR236	(3 to 0)	IR236		
	TP4GR1 interrupt	237	H'0000 03B4 to H'0000 03B7	0 to 15 (0)	ICR237	(3 to 0)	IR237		
	TP4GR2 interrupt	238	H'0000 03B8 to H'0000 03BB	0 to 15 (0)	ICR238	(3 to 0)	IR238		
	TP4GR3 interrupt	239	H'0000 03BC to H'0000 03BF	0 to 15 (0)	ICR239	(3 to 0)	IR239		
	TPU4 counter overflow interrupt	240	H'0000 03C0 to H'0000 03C3	0 to 15 (0)	ICR240	(3 to 0)	IR240		
(Reserved)		241 to 296	—					—	
TPU	TPU1 counter reset interrupt	297	H'0000 04A4 to H'0000 04A7	0 to 15 (0)	ICR297	(3 to 0)	IR297	Edge	Low

Note: * Not used in the SH72A0 group.

Table 8.14 Interrupt Sources and Vectors (4)

Interrupt Source		Interrupt Vector		Interrupt Priority (Initial Value)	Corresponding IPR (Bit)		Corre-spond-ing IR	ICU Input Source Type	Default Priority
		Vector No.	Vector Table Address Offset						
TPU	TPU2 counter reset interrupt	298	H'0000 04A8 to H'0000 04AB	0 to 15 (0)	ICR298	(3 to 0)	IR298	Edge	High
	TPU3 counter reset interrupt	299	H'0000 04AC to H'0000 04AF	0 to 15 (0)	ICR299	(3 to 0)	IR299		
	TPU4 counter reset interrupt	300	H'0000 04B0 to H'0000 04B3	0 to 15 (0)	ICR300	(3 to 0)	IR300		
(Reserved)		301 to 311	—					—	
SCI	SCI0 receive error interrupt	312	H'0000 04E0 to H'0000 04E3	0 to 15 (0)	ICR312	(3 to 0)	IR312	Level	
	SCI0 receive buffer full interrupt	313	H'0000 04E4 to H'0000 04E7	0 to 15 (0)	ICR313	(3 to 0)	IR313	Edge	
	SCI0 transmit buffer empty interrupt	314	H'0000 04E8 to H'0000 04EB	0 to 15 (0)	ICR314	(3 to 0)	IR314		
	SCI0 transmit end interrupt	315	H'0000 04EC to H'0000 04EF	0 to 15 (0)	ICR315	(3 to 0)	IR315	Level	
	SCI1 receive error interrupt	316	H'0000 04F0 to H'0000 04F3	0 to 15 (0)	ICR316	(3 to 0)	IR316		
	SCI1 receive buffer full interrupt	317	H'0000 04F4 to H'0000 04F7	0 to 15 (0)	ICR317	(3 to 0)	IR317	Edge	
	SCI1 transmit buffer empty interrupt	318	H'0000 04F8 to H'0000 04FB	0 to 15 (0)	ICR318	(3 to 0)	IR318		
	SCI1 transmit end interrupt	319	H'0000 04FC to H'0000 04FF	0 to 15 (0)	ICR319	(3 to 0)	IR319	Level	
	SCI2 receive error interrupt	320	H'0000 0500 to H'0000 0503	0 to 15 (0)	ICR320	(3 to 0)	IR320		
	SCI2 receive buffer full interrupt	321	H'0000 0504 to H'0000 0507	0 to 15 (0)	ICR321	(3 to 0)	IR321	Edge	
	SCI2 transmit buffer empty interrupt	322	H'0000 0508 to H'0000 050B	0 to 15 (0)	ICR322	(3 to 0)	IR322		
	SCI2 transmit end interrupt	323	H'0000 050C to H'0000 050F	0 to 15 (0)	ICR323	(3 to 0)	IR323	Level	
	SCI3 receive error interrupt	324	H'0000 0510 to H'0000 0513	0 to 15 (0)	ICR324	(3 to 0)	IR324		
	SCI3 receive buffer full interrupt	325	H'0000 0514 to H'0000 0517	0 to 15 (0)	ICR325	(3 to 0)	IR325	Edge	
	SCI3 transmit buffer empty interrupt	326	H'0000 0518 to H'0000 051B	0 to 15 (0)	ICR326	(3 to 0)	IR326		
	SCI3 transmit end interrupt	327	H'0000 051C to H'0000 051F	0 to 15 (0)	ICR327	(3 to 0)	IR327	Level	
CAN	CAN0 reception interrupt	328	H'0000 0520 to H'0000 0523	0 to 15 (0)	ICR328	(3 to 0)	IR328	Edge	
	CAN0 transmission interrupt	329	H'0000 0524 to H'0000 0527	0 to 15 (0)	ICR329	(3 to 0)	IR329		
	CAN0 reception FIFO interrupt	330	H'0000 0528 to H'0000 052B	0 to 15 (0)	ICR330	(3 to 0)	IR330		
	CAN0 transmission FIFO interrupt	331	H'0000 052C to H'0000 052F	0 to 15 (0)	ICR331	(3 to 0)	IR331		
	CAN0 error interrupt	332	H'0000 0530 to H'0000 0533	0 to 15 (0)	ICR332	(3 to 0)	IR332		
	CAN0 wake-up interrupt	333	H'0000 0534 to H'0000 0537	0 to 15 (0)	ICR333	(3 to 0)	IR333		
	CAN1 reception interrupt	334	H'0000 0538 to H'0000 053B	0 to 15 (0)	ICR334	(3 to 0)	IR334		
	CAN1 transmission interrupt	335	H'0000 053C to H'0000 053F	0 to 15 (0)	ICR335	(3 to 0)	IR335		
	CAN1 reception FIFO interrupt	336	H'0000 0540 to H'0000 0543	0 to 15 (0)	ICR336	(3 to 0)	IR336		
	CAN1 transmission FIFO interrupt	337	H'0000 0544 to H'0000 0547	0 to 15 (0)	ICR337	(3 to 0)	IR337		
	CAN1 error interrupt	338	H'0000 0548 to H'0000 054B	0 to 15 (0)	ICR338	(3 to 0)	IR338		
	CAN1 wake-up interrupt	339	H'0000 054C to H'0000 054F	0 to 15 (0)	ICR339	(3 to 0)	IR339		
(Reserved)		340 to 365	—					—	
LIN	LIN2 interrupt	366	H'0000 05B8 to H'0000 05BB	0 to 15 (0)	ICR366	(3 to 0)	IR366	Level	
	LIN3 interrupt*	367	H'0000 05BC to H'0000 05BF	0 to 15 (0)	ICR367	(3 to 0)	IR367		
(Reserved)		368 to 377	—					—	
LIN	LIN2 Low detection interrupt	378	H'0000 05E8 to H'0000 05EB	0 to 15 (0)	ICR378	(3 to 0)	IR378	Edge	
	LIN3 Low detection interrupt*	379	H'0000 05EC to H'0000 05EF	0 to 15 (0)	ICR379	(3 to 0)	IR379		
(Reserved)		380 to 402	—					—	Low

Note: * Not used in the SH72A0 group.

Table 8.15 Interrupt Sources and Vectors (5)

Interrupt Source		Interrupt Vector		Interrupt Priority (Initial Value)	Corresponding IPR (Bit)		Corre-spond-ing IR	ICU Input Source Type	Default Priority
		Vector No.	Vector Table Address Offset						
SBI	SBI0 receive interrupt	403	H'0000 064C to H'0000 064F	0 to 15 (0)	ICR403	(3 to 0)	IR403	Edge	High
	SBI1 receive interrupt	404	H'0000 0650 to H'0000 0653	0 to 15 (0)	ICR404	(3 to 0)	IR404		
	SBI2 receive interrupt	405	H'0000 0654 to H'0000 0657	0 to 15 (0)	ICR405	(3 to 0)	IR405		
	SBI3 receive interrupt*	406	H'0000 0658 to H'0000 065B	0 to 15 (0)	ICR406	(3 to 0)	IR406		
	SBI0 transmit interrupt	407	H'0000 065C to H'0000 065F	0 to 15 (0)	ICR407	(3 to 0)	IR407		
	SBI1 transmit interrupt	408	H'0000 0660 to H'0000 0663	0 to 15 (0)	ICR408	(3 to 0)	IR408		
	SBI2 transmit interrupt	409	H'0000 0664 to H'0000 0667	0 to 15 (0)	ICR409	(3 to 0)	IR409		
	SBI3 transmit interrupt*	410	H'0000 0668 to H'0000 066B	0 to 15 (0)	ICR410	(3 to 0)	IR410		
	SBI0 idle interrupt	411	H'0000 066C to H'0000 066F	0 to 15 (0)	ICR411	(3 to 0)	IR411	Level	
	SBI1 idle interrupt	412	H'0000 0670 to H'0000 0673	0 to 15 (0)	ICR412	(3 to 0)	IR412		
	SBI2 idle interrupt	413	H'0000 0674 to H'0000 0677	0 to 15 (0)	ICR413	(3 to 0)	IR413		
	SBI3 idle interrupt*	414	H'0000 0678 to H'0000 067B	0 to 15 (0)	ICR414	(3 to 0)	IR414		
	SBI0 error interrupt	415	H'0000 067C to H'0000 067F	0 to 15 (0)	ICR415	(3 to 0)	IR415	Edge	
	SBI1 error interrupt	416	H'0000 0680 to H'0000 0683	0 to 15 (0)	ICR416	(3 to 0)	IR416		
	SBI2 error interrupt	417	H'0000 0684 to H'0000 0687	0 to 15 (0)	ICR417	(3 to 0)	IR417		
	SBI3 error interrupt*	418	H'0000 0688 to H'0000 068B	0 to 15 (0)	ICR418	(3 to 0)	IR418		
(Reserved)		419 to 426	—					—	
MTU-III	MT0GRA interrupt	427	H'0000 06AC to H'0000 06AF	0 to 15 (0)	ICR427	(3 to 0)	IR427	Level	
	MT0GRB interrupt	428	H'0000 06B0 to H'0000 06B3	0 to 15 (0)	ICR428	(3 to 0)	IR428		
	MT0GRC interrupt	429	H'0000 06B4 to H'0000 06B7	0 to 15 (0)	ICR429	(3 to 0)	IR429		
	MT0GRD interrupt	430	H'0000 06B8 to H'0000 06BB	0 to 15 (0)	ICR430	(3 to 0)	IR430		
	MT0 counter overflow interrupt	431	H'0000 06BC to H'0000 06BF	0 to 15 (0)	ICR431	(3 to 0)	IR431		
	MT0GRE interrupt	432	H'0000 06C0 to H'0000 06C3	0 to 15 (0)	ICR432	(3 to 0)	IR432		
	MT0GRF interrupt	433	H'0000 06C4 to H'0000 06C7	0 to 15 (0)	ICR433	(3 to 0)	IR433		
	MT1GRA interrupt	434	H'0000 06C8 to H'0000 06CB	0 to 15 (0)	ICR434	(3 to 0)	IR434		
	MT1GRB interrupt	435	H'0000 06CC to H'0000 06CF	0 to 15 (0)	ICR435	(3 to 0)	IR435		
	MT1 counter overflow interrupt	436	H'0000 06D0 to H'0000 06D3	0 to 15 (0)	ICR436	(3 to 0)	IR436		
	MT1 counter underflow interrupt	437	H'0000 06D4 to H'0000 06D7	0 to 15 (0)	ICR437	(3 to 0)	IR437		
	MT2GRA interrupt	438	H'0000 06D8 to H'0000 06DB	0 to 15 (0)	ICR438	(3 to 0)	IR438		
	MT2GRB interrupt	439	H'0000 06DC to H'0000 06DF	0 to 15 (0)	ICR439	(3 to 0)	IR439		
	MT2 counter overflow interrupt	440	H'0000 06E0 to H'0000 06E3	0 to 15 (0)	ICR440	(3 to 0)	IR440		
	MT2 counter underflow interrupt	441	H'0000 06E4 to H'0000 06E7	0 to 15 (0)	ICR441	(3 to 0)	IR441		
	MT3GRA interrupt	442	H'0000 06E8 to H'0000 06EB	0 to 15 (0)	ICR442	(3 to 0)	IR442		
	MT3GRB interrupt	443	H'0000 06EC to H'0000 06EF	0 to 15 (0)	ICR443	(3 to 0)	IR443		
	MT3GRC interrupt	444	H'0000 06F0 to H'0000 06F3	0 to 15 (0)	ICR444	(3 to 0)	IR444		
	MT3GRD interrupt	445	H'0000 06F4 to H'0000 06F7	0 to 15 (0)	ICR445	(3 to 0)	IR445		
	MT3 counter overflow interrupt	446	H'0000 06F8 to H'0000 06FB	0 to 15 (0)	ICR446	(3 to 0)	IR446		
	MT4GRA interrupt	447	H'0000 06FC to H'0000 06FF	0 to 15 (0)	ICR447	(3 to 0)	IR447		
	MT4GRB interrupt	448	H'0000 0700 to H'0000 0703	0 to 15 (0)	ICR448	(3 to 0)	IR448		
	MT4GRC interrupt	449	H'0000 0704 to H'0000 0707	0 to 15 (0)	ICR449	(3 to 0)	IR449		
	MT4GRD interrupt	450	H'0000 0708 to H'0000 070B	0 to 15 (0)	ICR450	(3 to 0)	IR450		
	MT4 counter overflow/underflow interrupt	451	H'0000 070C to H'0000 070F	0 to 15 (0)	ICR451	(3 to 0)	IR451		
	MT5GRU interrupt	452	H'0000 0710 to H'0000 0713	0 to 15 (0)	ICR452	(3 to 0)	IR452		
	MT5GRV interrupt	453	H'0000 0714 to H'0000 0717	0 to 15 (0)	ICR453	(3 to 0)	IR453		Low

Note: * Not used in the SH72A0 group.

Table 8.16 Interrupt Sources and Vectors (6)

Interrupt Source		Interrupt Vector		Interrupt Priority (Initial Value)	Corresponding IPR (Bit)		Corre-spond-ing IR	ICU Input Source Type	Default Priority
		Vector No.	Vector Table Address Offset						
MTU-III	MT5GRW interrupt	454	H'0000 0718 to H'0000 071B	0 to 15 (0)	ICR454	(3 to 0)	IR454	Level	High
	MT6GRA interrupt	455	H'0000 071C to H'0000 071F	0 to 15 (0)	ICR455	(3 to 0)	IR455		
	MT6GRB interrupt	456	H'0000 0720 to H'0000 0723	0 to 15 (0)	ICR456	(3 to 0)	IR456		
	MT6GRC interrupt	457	H'0000 0724 to H'0000 0727	0 to 15 (0)	ICR457	(3 to 0)	IR457		
	MT6GRD interrupt	458	H'0000 0728 to H'0000 072B	0 to 15 (0)	ICR458	(3 to 0)	IR458		
	MT6 counter overflow interrupt	459	H'0000 072C to H'0000 072F	0 to 15 (0)	ICR459	(3 to 0)	IR459		
	MT7GRA interrupt	460	H'0000 0730 to H'0000 0733	0 to 15 (0)	ICR460	(3 to 0)	IR460		
	MT7GRB interrupt	461	H'0000 0734 to H'0000 0737	0 to 15 (0)	ICR461	(3 to 0)	IR461		
	MT7GRC interrupt	462	H'0000 0738 to H'0000 073B	0 to 15 (0)	ICR462	(3 to 0)	IR462		
	MT7GRD interrupt	463	H'0000 073C to H'0000 073F	0 to 15 (0)	ICR463	(3 to 0)	IR463		
	MT7 counter overflow/underflow interrupt	464	H'0000 0740 to H'0000 0743	0 to 15 (0)	ICR464	(3 to 0)	IR464		
(Reserved)		465 to 476	—					—	
Task monitor timer	Task monitor timer interrupt	477	H'0000 0774 to H'0000 0777	0 to 15 (0)	ICR477	(3 to 0)	IR477	Edge	
(Reserved)		478 to 511	—					—	Low

8.4.11 Interrupt Operation Sequence

The sequence of interrupt operations is described below. Figure 8.3 shows the Interrupt Operation Flow.

(1) The interrupt request sources requests an interrupt to the interrupt controller.
(2) The interrupt controller selects the highest-priority interrupt from the received interrupts according to the priority levels set in interrupt priority registers 1 to 4 (IPR1 to IPR4) and the IPR bits in the interrupt control registers 102 to 511 (ICR102 to ICR511). Remaining interrupts are ignored *. If multiple interrupts with the same IPR priority level occur, the interrupt with the highest priority is selected according to “Default Priority” shown in Table 8.11 to Table 8.16.
(3) The priority level of the interrupt selected in the interrupt controller is compared with the interrupt level mask bits (I3 to I0) in the status register (SR) of the CPU. If the interrupt request priority level is equal to or less than the level set in bits I3 to I0, the interrupt request is ignored. If the interrupt request priority level is higher than the level in bits I3 to I0, the interrupt controller receives the interrupt and requests an interrupt to the CPU.
(4) The CPU detects the interrupt requested from the interrupt controller when the CPU decodes the instruction to be executed. Instead of executing the decoded instruction, the CPU starts interrupt exception handling (Figure 8.5).
(5) The start address of the interrupt exception handler corresponding to the received interrupt is fetched from the exception handling vector table.
(6) The contents of the status register (SR) are saved to the stack, and the priority level of the received interrupt is copied to bits I3 to I0 in SR.
(7) The contents of the program counter (PC) are saved to the stack.
(8) The CPU branches to the fetched start address of the interrupt exception handler and starts executing the program. This branch is not a delayed branch.

Note: * Interrupt requests that are designated as edge-sensing are held pending until the interrupt requests are received. INT interrupts, however, can be cancelled by accessing the INT interrupt request register (INTR). For details, section 8.4.8, INT Interrupts. Interrupts held pending due to edge-sensing are cleared by a reset.

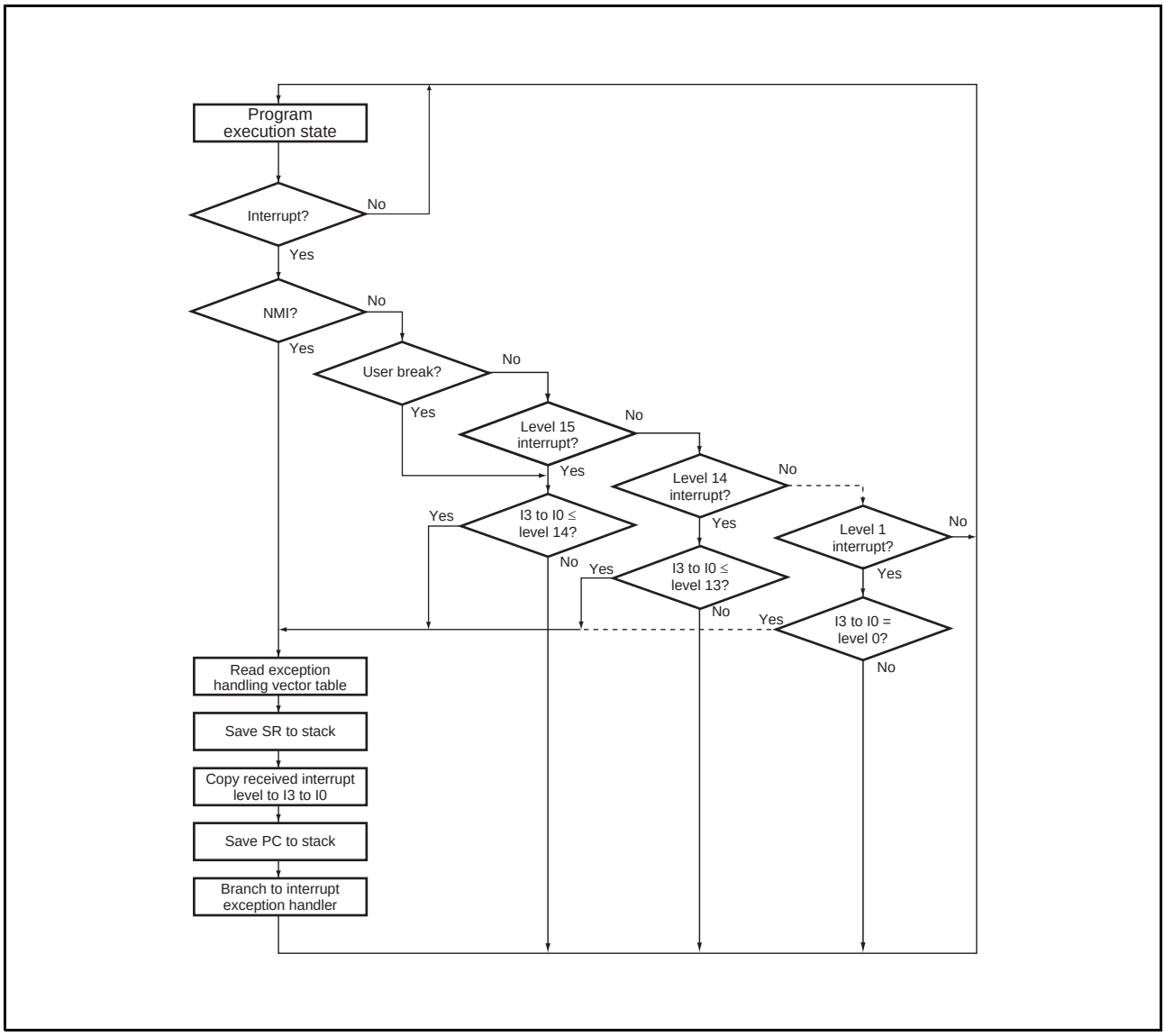
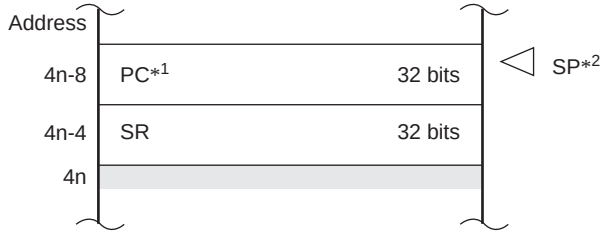


Figure 8.3 Interrupt Operation Flow

8.4.12 Stack after Interrupt Exception Handling

Figure 8.4 shows the Stack after Interrupt Exception Handling.



Notes: 1. PC: Start address of the next instruction (instruction at return address) after the executed instruction
2. Always make sure that SP is a multiple of 4.

Figure 8.4 Stack after Interrupt Exception Handling

8.4.13 Detection Control of Interrupt Sources

The INTC has interrupt control registers 102 to 511 (ICR102 to ICR511) to control detection of peripheral function interrupts. The following shows the setting procedure:

(1) Clear the IPR bits in the interrupt control register (ICR) corresponding to the interrupt source.
(2) Clear the INTEN bit in the interrupt control register (ICR) corresponding to the interrupt source.
(3) Check that the corresponding interrupt control request register (IR) has been cleared.
(4) Set the IPR bits in the interrupt control register (ICR) corresponding to the interrupt source.
(5) Set the INTEN bit in the interrupt control register (ICR) corresponding to the interrupt source.
(6) Read access: Since the CPU is performing pipeline operations, read the interrupt control register (ICR) to ensure that the above settings have been completed.

8.5 Interrupt Response Time

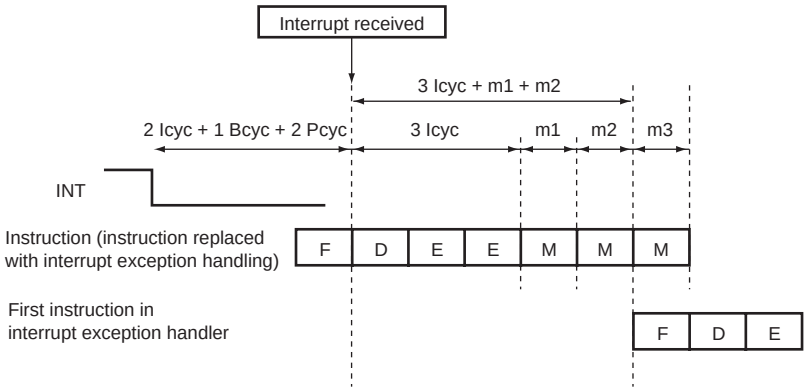
Table 8.17 lists the Interrupt Response Time, which is the time from the occurrence of an interrupt request until the interrupt exception handling starts and fetching of the first instruction in the exception handler begins. The interrupt processing operations differ in the cases when banking is disabled, when banking is enabled without register bank overflow, and when banking is enabled with register bank overflow.

Figure 8.5 and Figure 8.6 show examples of pipeline operation when banking is disabled. Figure 8.7 and Figure 8.8 show examples of pipeline operation when banking is enabled without register bank overflow. Figure 8.9 and Figure 8.10 show examples of pipeline operation when banking is enabled with register bank overflow.

Table 8.17 Interrupt Response Time

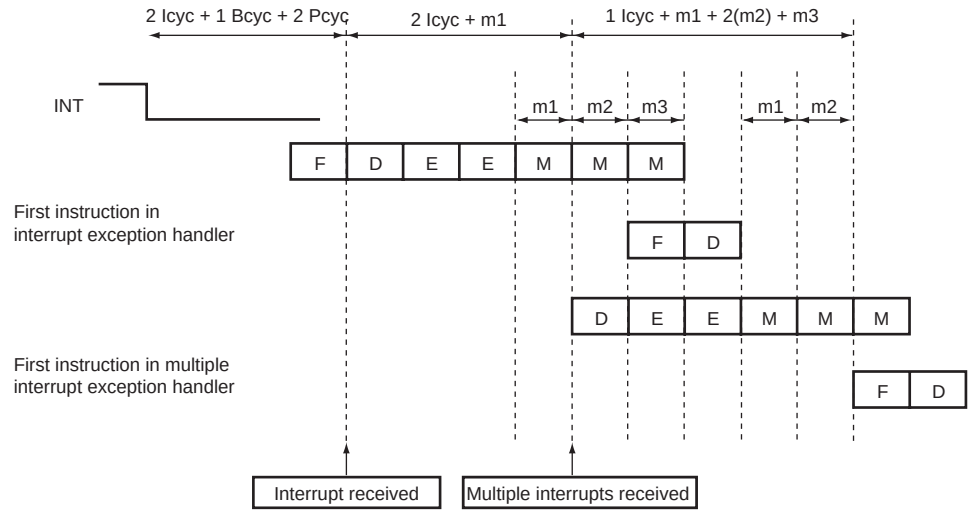
Item			Number of Cycles				Remarks
			NMI	User Break	INT	Peripheral Module	
Time from occurrence of interrupt request until interrupt controller identifies priority, compares it with mask bits in SR, and sends interrupt request signal to CPU			2 Icyc + 3 Pcyc	3 Icyc	2 Icyc + 1 Bcyc + 2 Pcyc	2 Icyc + 3 Bcyc	For INT interrupts, the time delayed by setting a digital filter is not included.
Time from input of interrupt request signal to CPU until sequence currently being executed is completed, interrupt exception handling starts, and first instruction in interrupt exception handler is fetched	No register bank	Min.	3 Icyc + m1 + m2				Min. is when the interrupt wait time is zero. Max. is when a higher-priority interrupt request has occurred during interrupt exception handling.
		Max.	4 Icyc + 2(m1 + m2) + m3				
	Register bank without register bank overflow	Min.	—		3 Icyc + m1 + m2		Min. is when the interrupt wait time is zero. Max. is when an interrupt request has occurred during execution of the RESBANK instruction.
		Max.	—		12 Icyc + m1 + m2		
	Register bank with register bank overflow	Min.	—		3 Icyc + m1 + m2		Min. is when the interrupt wait time is zero. Max. is when an interrupt request has occurred during execution of the RESBANK instruction.
		Max.	—		3 Icyc + m1 + m2 + 19(m4)		
Interrupt response time	No register banking	Min.	5 Icyc + 3 Pcyc + m1 + m2	6 Icyc + m1 + m2	5 Icyc + 1 Bcyc+ 2 Pcyc + m1 + m2	5 Icyc + 3 Bcyc + m1 + m2	
		Max.	6 Icyc + 3 Pcyc + 2(m1+m2) + m3	7 Icyc + 2(m1 + m2) + m3	6 Icyc + 1 Bcyc + 2 Pcyc + 2(m1+m2) + m3	6 Icyc + 3 Bcyc + 2(m1+m2) + m3	
	Register banking without register bank overflow	Min.	—	—	5 Icyc + 1 Bcyc + 2 Pcyc + m1 + m2	5 Icyc + 3 Bcyc + m1 + m2	
		Max.	—	—	14 Icyc + 1 Bcyc + 2 Pcyc + m1 + m2	14 Icyc + 3 Bcyc + m1 + m2	
	Register banking with register bank overflow	Min.	—	—	5 Icyc + 1 Bcyc + 2 Pcyc + m1 + m2	5 Icyc + 3 Bcyc + m1 + m2	
		Max.	—	—	5 Icyc + 1 Bcyc + 2 Pcyc + m1 + m2 + 19(m4)	5 Icyc + 3 Bcyc + m1 + m2 + 19(m4)	

Notes: • m1 to m4 denotes the number of states needed for the following memory accesses.
m1: Vector address read (longword read)
m2: SR saving (longword write)
m3: PC saving (longword write)
m4: Restoring banked registers (R0 to R14, GBR, MACH, MACL, and PR) from the stack.
• Icyc, Bcyc, and Pcyc indicate one cycle of the following clocks.
Icyc: CPU clock f(CPU)
Bcyc: Bus clock f(BUS)
Pcyc: Peripheral bus clock A f(PBA)



[Legend]
m1: Vector address read
m2: Saving of SR (stack)
m3: Saving of PC (stack)
F: Instruction fetch. Instruction is fetched from memory in which program is stored.
D: Instruction decoding. Fetched instruction is decoded.
E: Instruction execution. Data operation or address calculation is performed in accordance with the result of decoding.
M: Memory access. Memory data is accessed

Figure 8.5 Example of Pipeline Operation when IRQ Interrupt is Received (No Register Banking)



[Legend]
m1: Vector address read
m2: Saving of SR (stack)
m3: Saving of PC (stack)

Figure 8.6 Example of Pipeline Operation for Multiple Interrupts (No Register Banking)

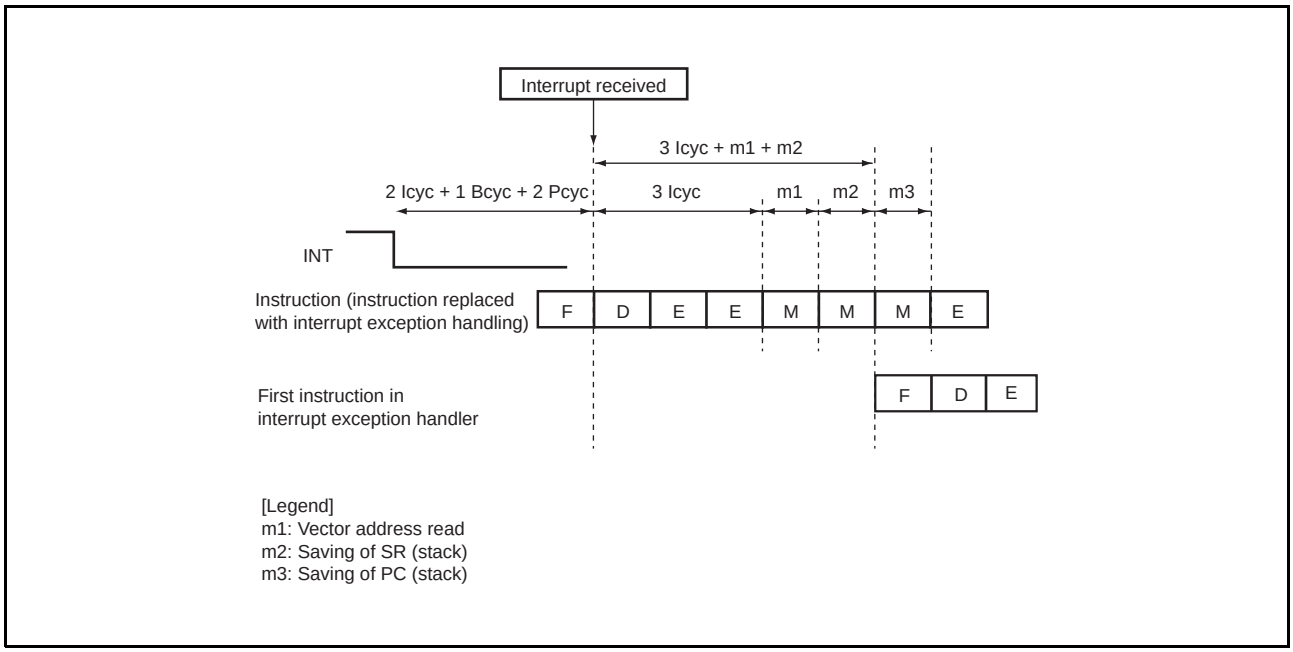


Figure 8.7 Example of Pipeline Operation when INT Interrupt is Received (Register Banking without Register Bank Overflow)

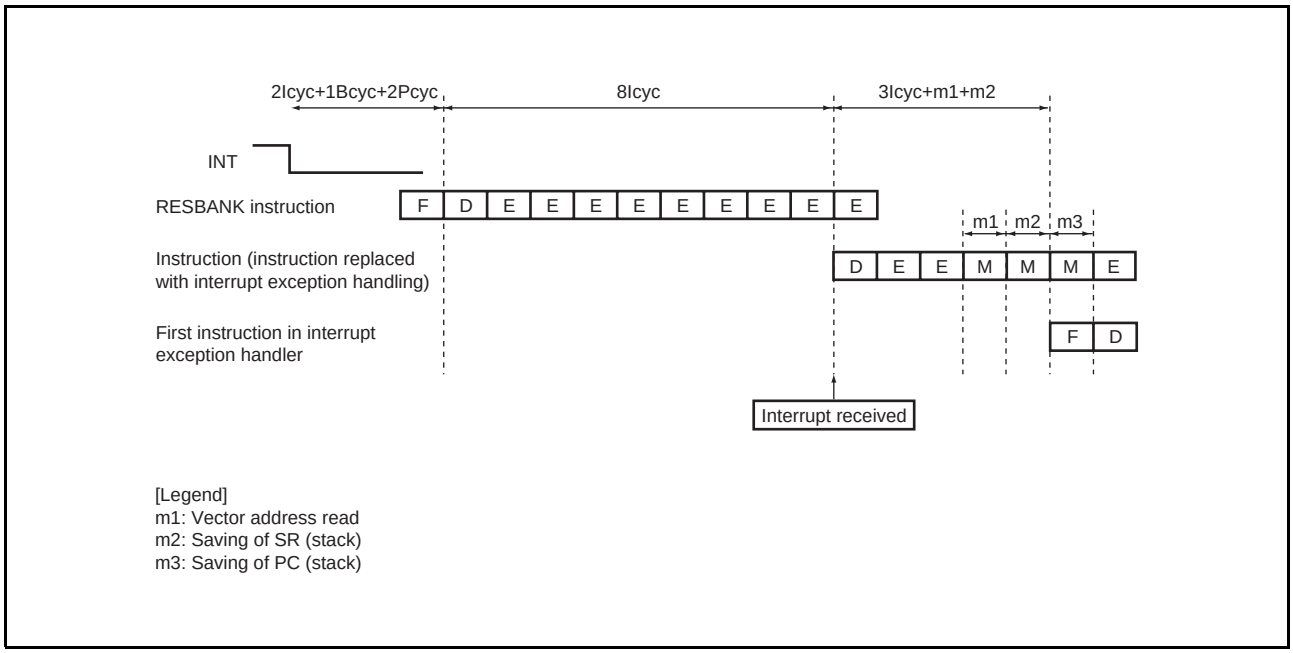


Figure 8.8 Example of Pipeline Operation when Interrupt is Received during RESBANK Instruction Execution (Register Banking without Register Bank Overflow)

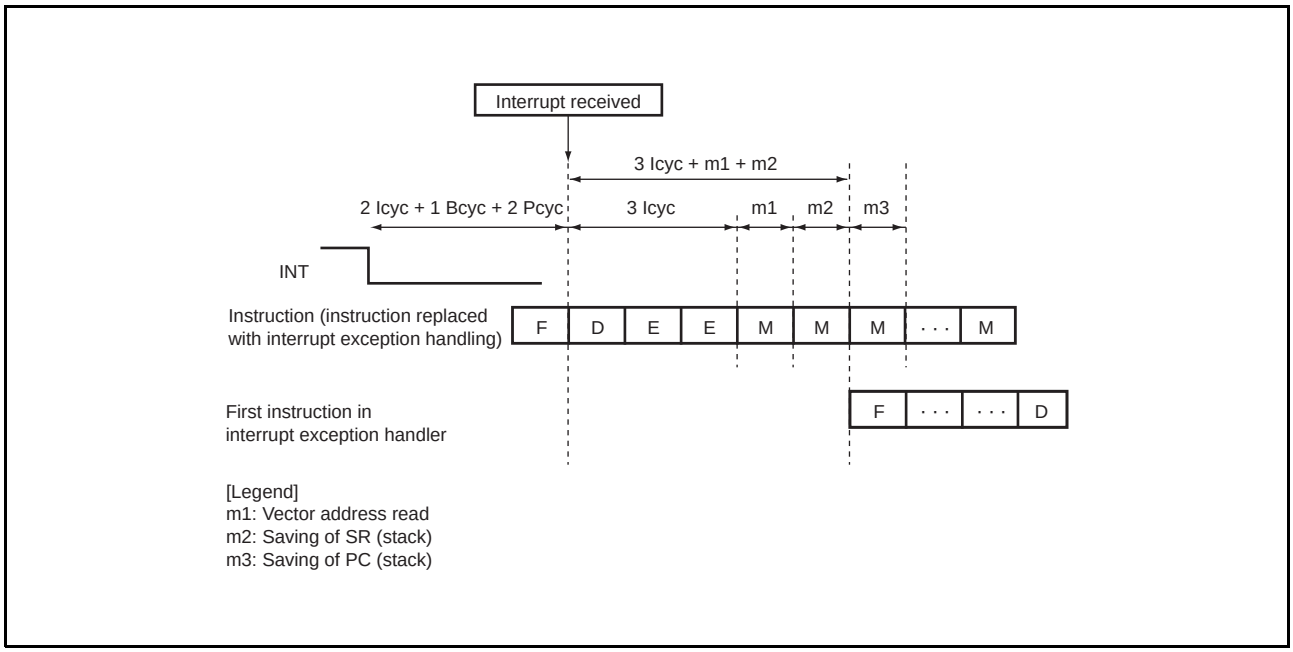


Figure 8.9 Example of Pipeline Operation when IRQ Interrupt is Received (Register Banking with Register Bank Overflow)

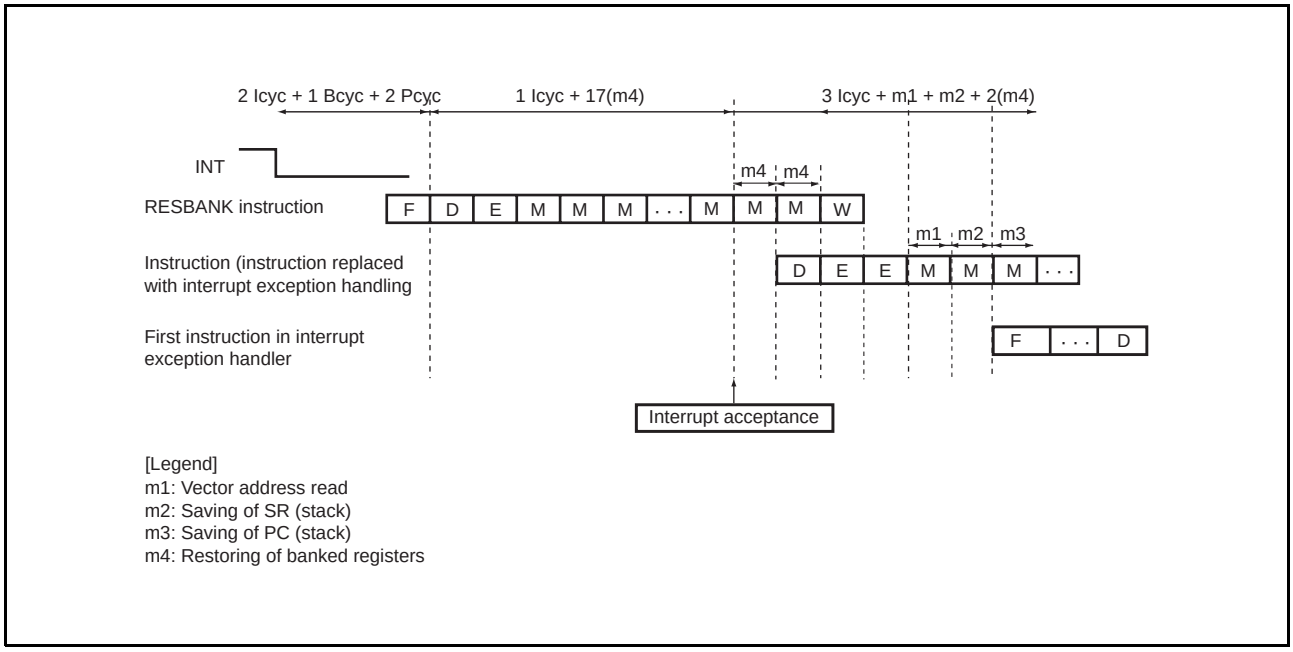


Figure 8.10 Example of Pipeline Operation when Interrupt is Received during RESBANK Instruction Execution (Register Banking with Register Bank Overflow)

8.6 Register Banks

This LSI has fifteen register banks used to save and restore registers for the interrupt processing at high speed. Figure 8.11 is the Register Bank Configuration.

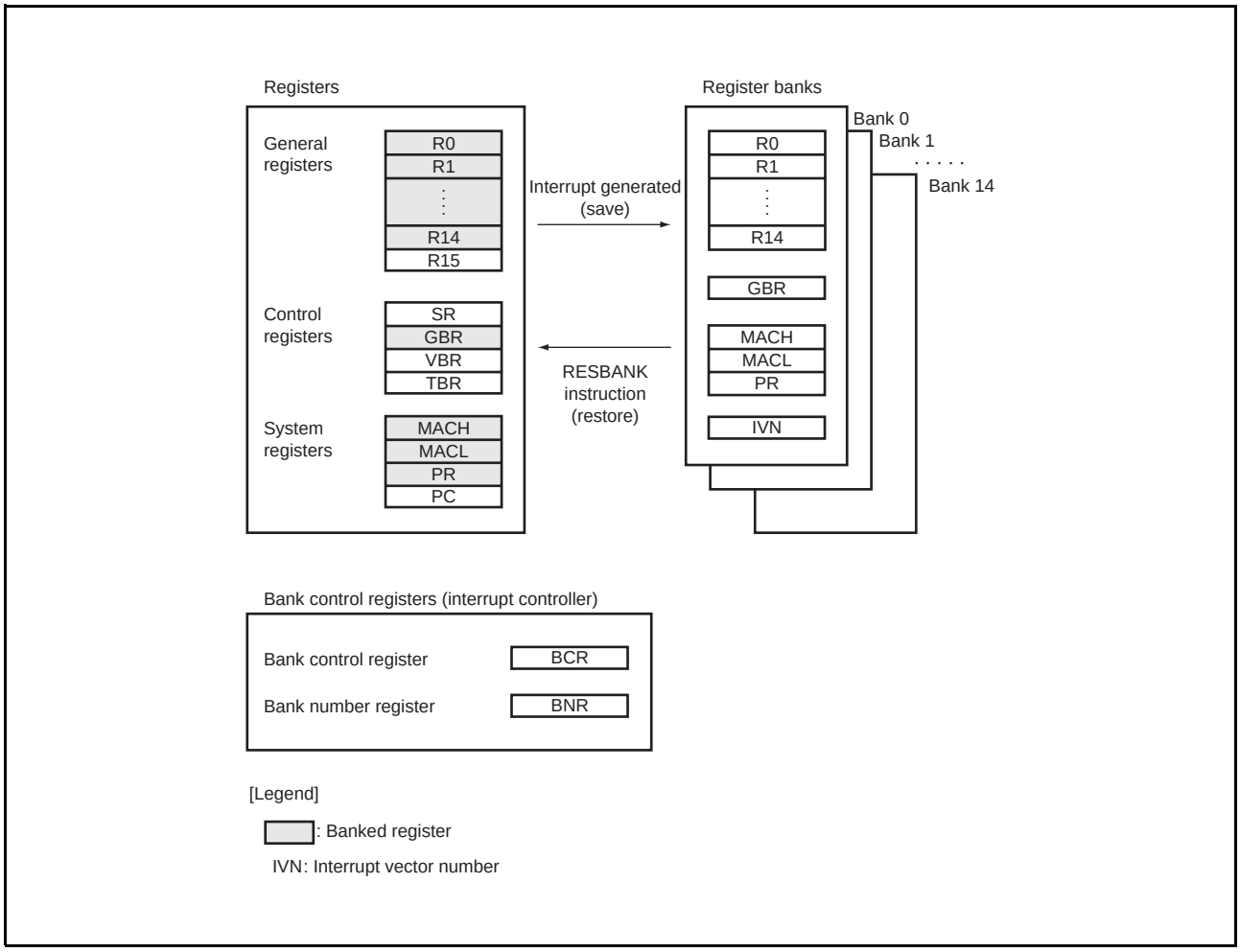


Figure 8.11 Register Bank Configuration

8.6.1 Register Banks and Bank Control Registers

(1) Contents to be Banked

The contents of the general registers (R0 to R14), global base register (GBR), multiply and accumulate registers (MACH and MACL), and procedure register (PR), and the vector table address number are banked.

(2) Register Banks

This LSI has fifteen register banks, bank 0 to bank 14. Register banks are stacked in first-in last-out (FILO) sequence. Saving takes place in the order of the bank number from 0 to 14 and restoring takes place in the reverse order from the last bank saved.

8.6.2 Bank Saving and Restoring Operations

(1) Saving to Bank

Figure 8.12 shows Bank Saving Operation. The following operations are performed when an interrupt for which usage of the register banks is enabled via the BCR register is received by the CPU.

(a) Assume that the bank number bit (BN) in the bank number register (BNR) before the interrupt is generated is i.

(b) The contents of registers R0 to R14, GBR, MACH, MACL, and PR, and the vector table address offset (IVN) of the received interrupt are saved in bank i.

(c) The BN bit is incremented by 1.

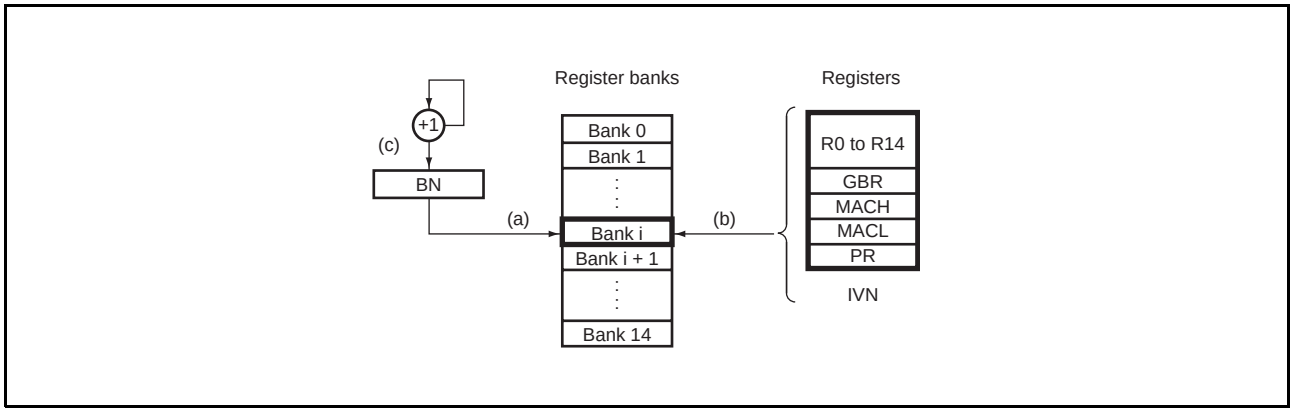


Figure 8.12 Bank Saving Operation

Figure 8.13 shows the timing for saving registers to a register bank. Saving registers to a register bank takes place between the start of interrupt exception handling and the start of fetching the first instruction in the interrupt exception handler.

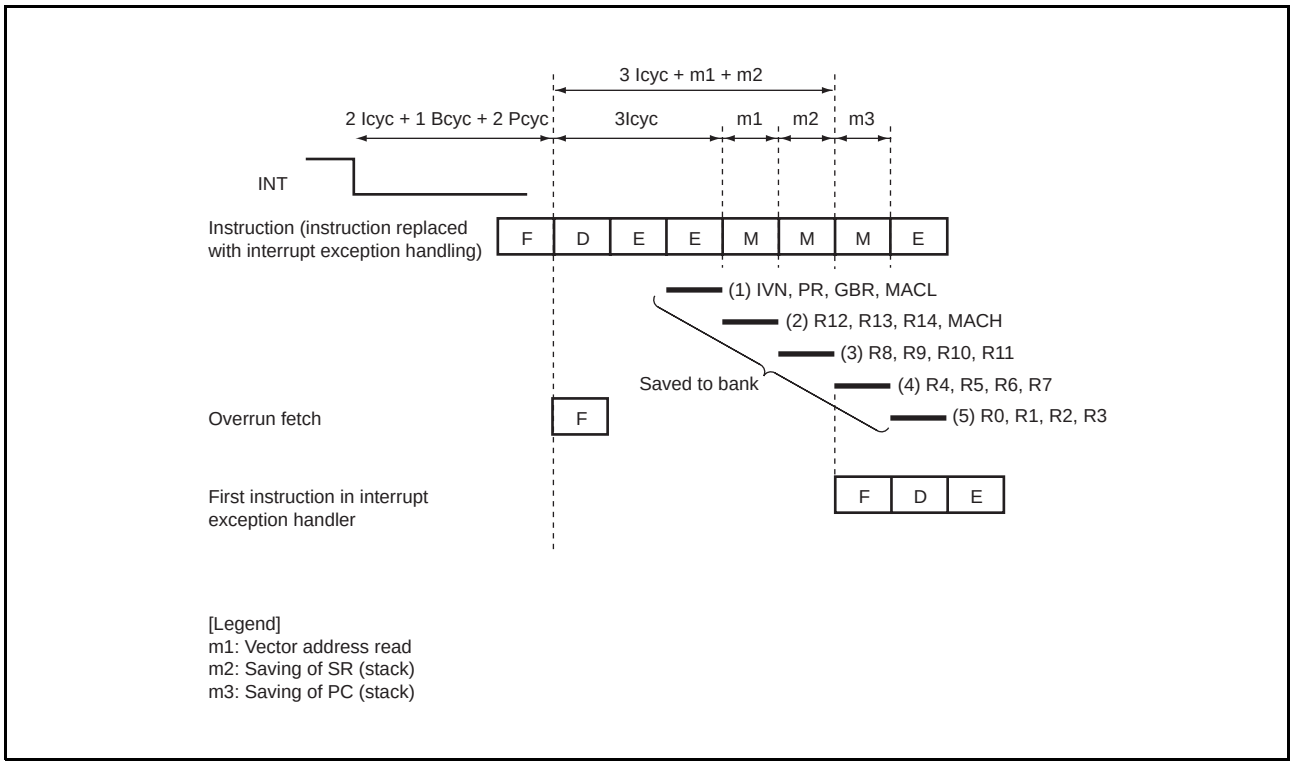


Figure 8.13 Bank Save Timing

(2) Restoring from Bank

The RESBANK (restoring from register bank) instruction is used to restore data saved in a register bank. After restoring data from the register banks with the RESBANK instruction at the end of the interrupt exception handler, execute the RTE instruction to return from the interrupt exception handler.

8.6.3 Saving and Restoring Operations after Saving Registers to All Banks

Assume that all register banks have been used for saving registers when an interrupt for which usage of the register banks is enabled is received by the CPU. When the BOVE bit in the bank number register (BNR) is cleared to 0, registers are automatically saved to the stack area instead of saving to a register bank. When the BOVE bit in the BNR register is set to 1, a register bank overflow exception occurs and registers are not saved to the stack area.
Saving and restoring operations when using the stack are shown below.

(1) Saving to Stack
(a) The status register (SR) and program counter (PC) are saved to the stack during interrupt exception handling.
(b) The contents of the banked registers (R0 to R14, GBR, MACH, MACL, and PR) are saved to the stack. The registers are saved to the stack in the order of MACL, MACH, GBR, PR, R14, R13, …, R1, and R0.
(c) The BO (register bank overflow) bit in the SR register is set to 1.
(d) The bank number BN in the bank number register (BNR) remains set to a maximum value of 15.

(2) Restoring from Stack
Operations when the RESBANK (restoring from register bank) instruction is executed with the BO (register bank overflow) bit in the SR register set to 1 are shown below.
(a) The contents of the banked registers (R0 to R14, GBR, MACH, MACL, and PR) are restored from the stack. The registers are restored from the stack in the order of R0, R1, …, R13, R14, PR, GBR, MACH, and MACL.
(b) The bank number BN in the bank number register (BNR) remains set to a maximum value of 15.

8.6.4 Register Bank Exception

There are two register bank exceptions (register bank errors): a register bank overflow and a register bank underflow.

(1) Register Bank Overflow
Assume that all register banks have been used for saving registers when an interrupt for which usage of the register banks is enabled is received by the CPU. When the BOVE bit in BNR is set to 1, a register bank overflow exception occurs. In this case, the bank number BN in the bank number register (BNR) remains set to the bank number of 15 and registers are not saved to the stack area.

(2) Register Bank Underflow
This exception occurs if the RESBANK (restoring from register bank) instruction is executed when no data has been saved to the register banks. In this case, the contents of R0 to R14, GBR, MACH, MACL, and PR do not change. In addition, the bank number BN in the bank number register (BNR) remains set to 0.

8.6.5 Register Bank Error Exception Handling

When a register bank error occurs, register bank error exception handling starts. When this happens, the CPU operation is shown below.

(1) The start address of the exception handler for the register bank error is fetched from the exception handling vector table.
(2) The status register (SR) is saved to the stack.
(3) The program counter (PC) is saved to the stack. At this time, the start address of the instruction to be executed after the last executed instruction is in PC and is saved when a register bank overflow occurs. The start address of the executed RESBANK instruction is in PC and is saved when a register bank underflow occurs.
(4) Program execution starts from the exception handler start address.

8.7 Notes on INTC

8.7.1 Timing to Clear Interrupt Source

Clear the interrupt source flag to 0 in the interrupt handler. As shown in Table 8.17 “Interrupt Response Time”, the time described as “Time from occurrence of interrupt request until interrupt controller identifies priority, compares it with mask bits in SR, and sends interrupt request signal to CPU” is required until the interrupt is cleared in the CPU after clearing the interrupt source flag to 0. To ensure that the interrupt request that should have been cleared is not received again erroneously, read the interrupt source flag after clearing it. After that, execute the RTE instruction.

8.7.2 Notes on INT Interrupt Request Register

To use a pin corresponding to INTn for any function other than the INTn interrupt function, set the corresponding interrupt priority level to 0 (interrupt disabled).

A bit corresponding to any function other than the INTn function may be set to 1 in the INT interrupt request register. When reading this register, use a program to perform masking or other processing to any bit which not used for the INTn interrupt so that the bit will be ignored.

8.7.3 Points for Caution Regarding Interrupt Control Registers 1 and 2 (ICR1 and ICR2) and INT Input Digital Filter Control Registers 0 and 1 (INTDFCR0 and INTDFCR1)

Disable the corresponding INTn interrupt (setting its priority level to 0) before changing the values of ICR1, ICR2, INTDFCR0, or INTDFCR1.

Follow the procedure below before re-enabling the corresponding INTn interrupt after changing the values of ICR1, ICR2, INTDFCR0, or INTDFCR1.

1. From among ICR1, ICR2, INTDFCR0, and INTDFCR1, read the last register to have been a destination for writing.
2. If the INTnFLTON bit in INTDFCR0 is 1 or the value of the INTFCKD bits in INTDFCR1 has been changed, wait for three cycles of the clock selected for the digital filter.
3. If edge sensing has been selected by the given bit in ICR1 or ICR2, write 0 to the given bit to clear the flag after reading the INT interrupt request register (INTR).

Note: n = 13 to 0

9. Voltage Monitor Function

9.1 Introduction

The voltage monitor function consists of voltage monitor function 1 which is used for generating an event by detecting changes in the supply voltage (VCC). The event to be generated is voltage monitor interrupt 1.

Table 9.1 lists the Voltage Monitor Function Specifications.
Figure 9.1 shows a Block Diagram of Voltage Monitor Function.
Figure 9.2 shows a Block Diagram of Voltage Monitor Interrupt 1.

Table 9.1 Voltage Monitor Function Specifications

Function		Voltage Monitor Circuit 1
VCC monitor	Voltage to be monitored	Vdet1: 3 levels among 3.9, 4.15, and 4.3 V
	Voltage monitor	Available
Reset	Reset function	Not available
	Reset condition	—
	Operation at power-on	—
Interrupt	Interrupt function	Available
	Interrupt request generation conditions	When the VCC voltage passes through the Vdet1 voltage
	Operation at power-on	Stopped (can be operated by software)
Others	Digital filter	Available

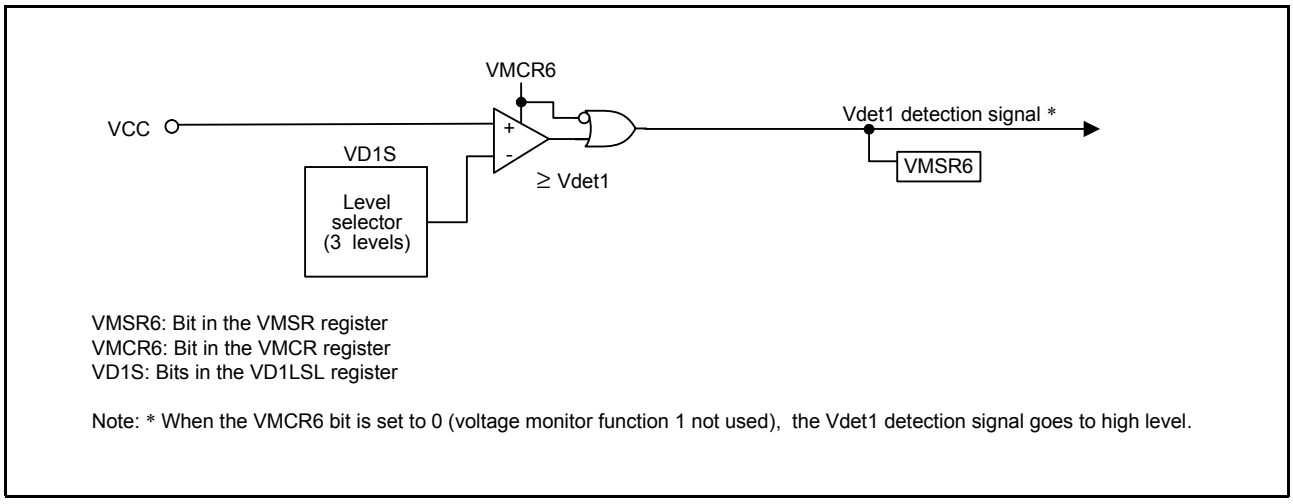
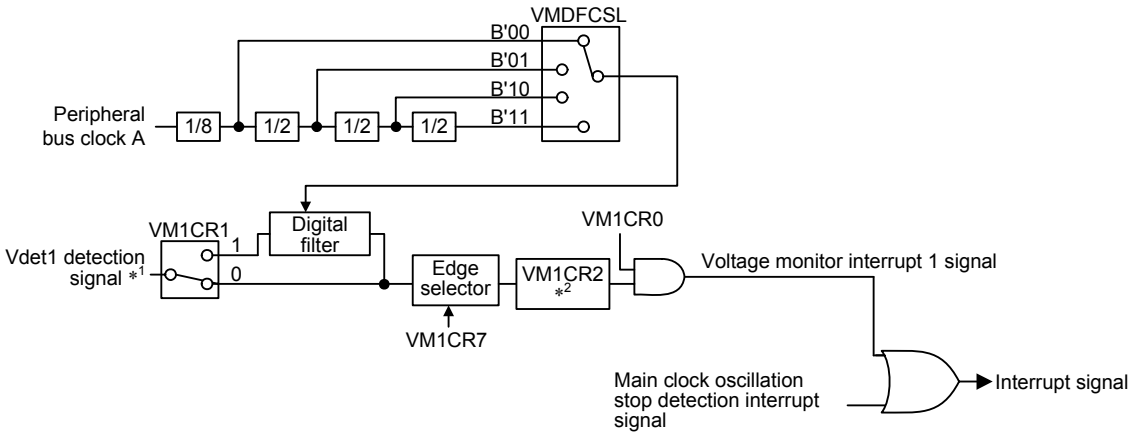


Figure 9.1 Block Diagram of Voltage Monitor Function



VM1CR0, VM1CR1, VM1CR2, VM1CR7: Bits in the VM1CR register
VMDFCSL: Bits in the VMDFCSL register

Notes: 1. When the VMCR6 bit in the VMCR register is set to 0 (voltage monitor function 1 not used), the Vdet1 detection signal goes to high level.
2. When the VMCR6 bit is set to 0, the VM1CR2 bit becomes 0.

Figure 9.2 Block Diagram of Voltage Monitor Interrupt 1

9.2 Registers

Table 9.2 lists the Registers of Voltage Monitor Function.

Table 9.2 Registers of Voltage Monitor Function

Register Name	Symbol	After Reset	Address	Access Size
System protect register 0	SPR0	H'00	H'FF46 E063	8
Vdet1 level select register	VD1LSL	H'07	H'FF46 E017	8
Voltage monitor status register	VMSR	H'C0	H'FF46 E403	8
Voltage monitor control register	VMCR	H'20	H'FF46 E00F	8
Voltage monitor circuit 1 control register	VM1CR	H'00	H'FF46 E407	8
Digital filter clock select register for voltage monitor circuits	VMDFCSL	H'00	H'FF46 E40F	8

9.2.1 System Protect Register 0 (SPR0)

Address H'FF46 E063

	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	0	0	0	0	0	0	0	0

Bit	Description	R/W
b7 to b0	When written B'1111 0001: Protection unlocked Other than B'1111 0001: Protection locked When read Bit 0 (b0) 0: Protection locked 1: Protection unlocked Bit 7 to bit 1 (b7 to b1) are always read as 0.	R/W

The SPR0 register is used to set the protect function that protects registers LOCR, SLCR0, VMCR, VD1LSL, RAMECC, RAMACYC, and AUDEN from being rewritten easily. To change the values of these registers, perform the following procedure:

(1) Write H'F1 to the SPR0 register (writing to the registers enabled).
(2) Change the values of registers LOCR, SLCR0, VMCR, VD1LSL, RAMECC, RAMACYC, and AUDEN.
(3) Write a value other than H'F1 to the SPR0 register (writing to the registers disabled).

9.2.2 Vdet1 Level Select Register (VD1LSL)

Address H'FF46 E017

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	VD1S[2:0]		
After Reset	0	0	0	0	0	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R
b2 to b0	VD1S[2:0]	Vdet1 Voltage Level Select Bits (typical voltage when the voltage falls)	b2 b1 b0 1 0 1 : 4.15 V (Vdet1_5) 1 1 0 : 3.90 V (Vdet1_6) 1 1 1 : 4.30 V (Vdet1_7) Others: Setting prohibited.	R/W

To set the VD1LSL register, use system protect register 0 (SPR0) to unlock the protection function before writing to the VD1LSL register.

VD1S Bits

These bits are used to set a voltage level of Vdet1.

9.2.3 Voltage Monitor Status Register (VMSR)

Address H'FF46 E403

	b7	b6	b5	b4	b3	b2	b1	b0
	—	VMSR6	—	—	—	—	—	—
After Reset	1	1	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	—	Reserved	This bit is read as 1. The write value should be 1.	R
b6	VMSR6	Voltage Change Monitor Flag 1	0: VCC < Vdet1 1: VCC ≥ Vdet1	R
b5 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

VMSR6 Bit

This flag indicates the monitor result of the voltage monitored in voltage monitor function 1.

When the VMCR6 bit in the VMCR register is set to 1 (voltage monitor function 1 used), if the monitored voltage level goes below Vdet1, the VMSR6 bit becomes 0. If the monitored voltage level reaches Vdet1 or above, or when the VMCR6 bit is set to 0 (voltage monitor function 1 not used), the VMSR6 bit becomes 1.

9.2.4 Voltage Monitor Control Register (VMCR)

Address H'FF46 E00F

	b7	b6	b5	b4	b3	b2	b1	b0
	—	VMCR6	—	—	—	—	—	—
After Reset	0	0	1	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R
b6	VMCR6	Voltage Monitor Function 1 Enable Bit	0: Voltage monitor function 1 not used 1: Voltage monitor function 1 used	R/W
b5	—	Reserved	This bit is read as 1. The write value should be 1.	R
b4 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

To set the VMCR register, use system protect register 0 (SPR0) to disable the protect function before writing to the VMCR register.

VMCR6 Bit

The VMCR6 bit is used to voltage monitor function 1. After the VMCR6 bit is set to 1 from 0, allow td(E-A) to elapse before voltage monitor circuit 1 starts operation.

9.2.5 Voltage Monitor Circuit 1 Control Register (VM1CR)

Address H'FF46 E407

	b7	b6	b5	b4	b3	b2	b1	b0
	VM1CR7	—	—	—	—	VM1CR2	VM1CR1	VM1CR0
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	VM1CR7	Voltage Monitor Interrupt 1 Generation Condition Select Bit	0: Voltage falls (VCC < Vdet1) 1: Voltage rises (VCC > Vdet1)	R/W
b6 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R
b2	VM1CR2	Voltage Change Detect Flag	0: Not detected 1: Detected	R/W
b1	VM1CR1	Digital Filter Use Bit	0: Digital filter not used 1: Digital filter used	R/W
b0	VM1CR0	Voltage Monitor Interrupt 1 Enable Bit	0: Voltage monitor interrupt 1 not used 1: Voltage monitor interrupt 1 used	R/W

Rewriting the VM1CR register may change the VM1CR2 bit to 1. Set the VM1CR2 bit to 0 after rewriting this register.

VM1CR7 Bit

This bit is used to select a generation condition of voltage monitor interrupt 1.

VM1CR2 Bit

This flag indicates whether the voltage to be monitored has passed through Vdet1. This bit is enabled when the VMCR6 bit in the VMCR register is set to 1 and the VM1CR0 bit in the VM1CR register is set to 1.

The VM1CR2 bit becomes 1 when the voltage has passed through Vdet1 under the condition selected with the VM1CR7 bit.

The VM1CR2 bit is set to 0 by writing 0 by a program. Writing 1 does not have any effect.

VM1CR1 Bit

This bit is used to control the digital filter of voltage monitor circuit 1. To use a voltage monitor interrupt 1 request for wake-up from standby mode, do no use the digital filter.

VM1CR0 Bit

This bit is used to enable voltage monitor interrupt 1. When the VMCR6 bit in the VMCR register is set to 0 (voltage monitor function 1 not used), set the VM1CR0 bit to 0.

9.2.6 Digital Filter Clock Select Register for Voltage Monitor Circuits (VMDFCSL)

Address H'FF46 E40F

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	VMDFCSL [1:0]	
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b1, b0	VMDFCSL [1:0]	Sampling Clock Select Bits	b1 b0 0 0 : Peripheral bus clock A divided by 8 0 1 : Peripheral bus clock A divided by 16 1 0 : Peripheral bus clock A divided by 32 1 1 : Peripheral bus clock A divided by 64	R/W

When rewriting the VMDFCSL register, the VM1CR2 bit in the VM1CR register may change to 1. Set the VM1CR2 bit to 0 after rewriting the VMDFCSL register.

VMDFCSL Bits

These bits are used to select a sampling clock of the digital filter for voltage monitor circuit 1.

9.3 Operations

9.3.1 Voltage Monitor Function 1

A voltage monitor interrupt 1 request can be generated when the input voltage to the VCC pin passes through Vdet1 (V) by falling or rising. To use voltage monitor function 1, set the VMCR6 bit in the VMCR register to 1. A voltage level of Vdet1 can be selected among three levels using the VD1LSL register.

Figure 9.3 shows an Operation Example of Voltage Monitor Function 1.

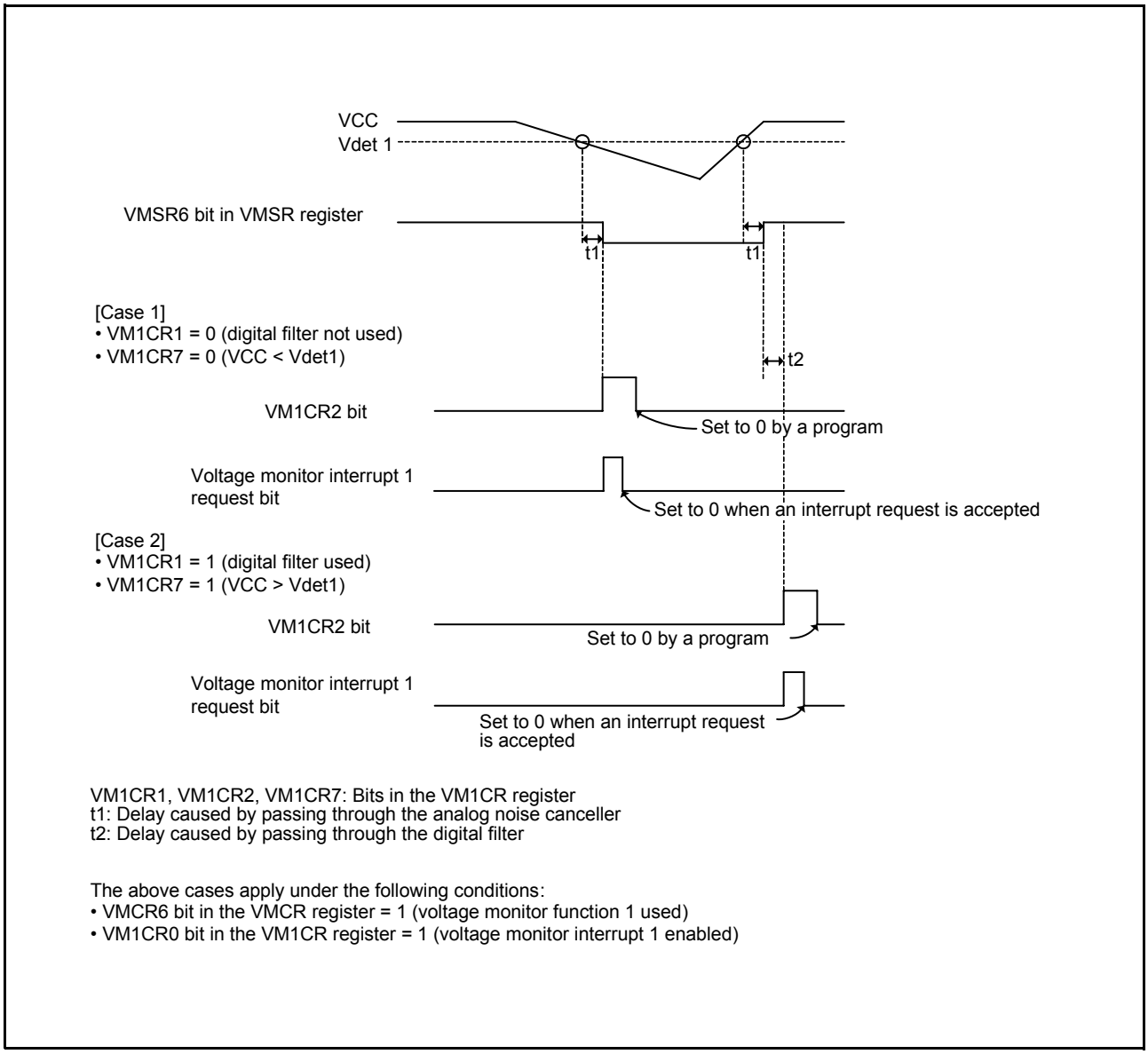


Figure 9.3 Operation Example of Voltage Monitor Function 1

9.4 Interrupts

9.4.1 Voltage Monitor Interrupt 1

Voltage monitor interrupt 1 can be used by setting the VM1CR0 bit in the VM1CR register to 1. A generation condition of voltage monitor interrupt 1 can be selected by the VM1CR7 bit in the VM1CR register.
When the VM1CR2 bit changes from 0 to 1, a voltage monitor interrupt 1 request is generated. This bit does not automatically become 0 even if the interrupt is accepted. Set the bit to 0 by a program.
Whether the input voltage has passed through Vdet1 by falling or rising can be checked by reading the VMSR6 bit in the VMSR register.

A voltage monitor interrupt 1 request can be used for wake-up from CPU sleep mode or standby mode using the interrupt, and as a power-down mode wake-up reset from power-down mode. For power-down mode wake-up reset, dedicated registers are used instead of registers VMSR, VM1CR, and VMDFCSL, and the MCU will be in the reset state after wake-up. For details, refer to section 5.5, Power Control in section 5, Clocks.
Voltage monitor interrupt 1 shares a vector with main clock oscillation stop detection interrupt. When using voltage monitor interrupt 1 and this interrupt at the same time, read the VM1CR2 bit in the interrupt routine to confirm voltage monitor interrupt 1 has been generated.

9.4.2 Digital Filter Function

A digital filer for noise reduction is available for voltage monitor interrupt 1. The digital filter is used to sample the Vdet1 detection signal and cancel pulses which do not meet three clocks of the sampling clock. To use the digital filter, set the VM1CR bit in the VM1CR register to 1 (digital filter used). When changing the VM1CR bit, make sure interrupts are disabled. A sampling clock can be selected with the VMDFCSL bits in the VMDFCSL register. When the sampling clock stops, the digital filter is disabled.

Figure 9.4 shows an Operation Example of Digital Filter.

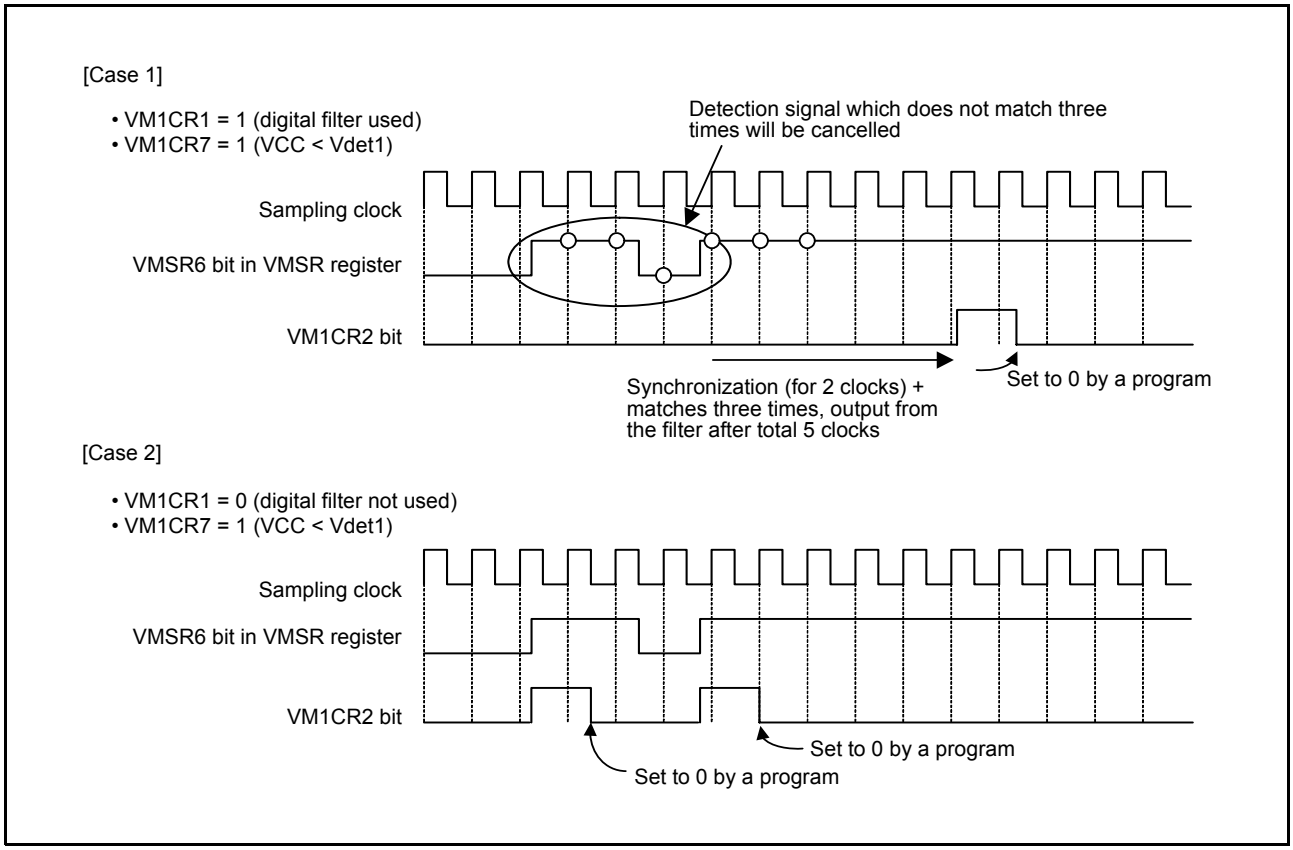


Figure 9.4 Operation Example of Digital Filter

9.5 Notes on Voltage Monitor Function

The voltage change detect flag (the VM1CR2 bit) must be cleared in an interrupt handler.
If this flag is read immediately after clearing, it may not have been cleared. Reading the flag is not necessary to clear it. However, execute an RTE instruction after waiting until the voltage change detect flag (the VM1CR2 bit) is set to 0 in loop or other processing to ensure it has been cleared.

10. Watchdog Timer

10.1 Introduction

The watchdog timer is used to restore processing to normal if a program runs out of control. The watchdog timer contains a 14-bit free-run counter, and the MCU is reset when this counter underflows. To prevent it from underflowing, perform a refresh operation (a write to the WDTRR register) during a refresh period. A write during a non-refresh period also resets the MCU.

Table 10.1 and Table 10.2 list the Watchdog Timer Specifications.
Figure 10.1 shows the Watchdog Timer Block Diagram.

Table 10.1 Watchdog Timer Specifications (1)

Item	Function
Count sources	• Peripheral bus clock A divided by 16 • Peripheral bus clock A divided by 128 • WDT count source (approx. 125 kHz)
Count operation	Decrement using the 14-bit free-run counter
Count start condition	Count is started by a refresh operation
Count stop	Count cannot be stopped after it started (except when count is stopped after a reset)
Reset generation conditions	• When the watchdog timer underflows • When a write to the WDTRR register is performed during a non-refresh period
Timer value read	The count value of the 14-bit free-run counter can be read by reading the WDTR register.

Table 10.2 Watchdog Timer Specifications (2)

Item		Setting Values of CSS Bits in WDTCR0 Register					
		B'00		B'01		B'11	
Count sources		Peripheral bus clock A divided by 16		Peripheral bus clock A divided by 128		WDT count source (output from the low-speed on-chip oscillator)	
Underflow period *1							
		Number of cycles	Time *2	Number of cycles	Time *2	Number of cycles	Time *3
	TOPS bits = B'00	16384	Approx. 0.4 ms	131072	Approx. 3.3 ms	1024	Approx. 8.2 ms
	TOPS bits = B'01	65536	Approx. 1.6 ms	524288	Approx. 13.1 ms	4096	Approx. 32.8 ms
	TOPS bits = B'10	131072	Approx. 3.3 ms	1048576	Approx. 26.2 ms	8192	Approx. 65.5 ms
	TOPS bits = B'11	262144	Approx. 6.6 ms	2097152	Approx. 52.4 ms	16384	Approx. 131.1 ms
Operation at underflow		The MCU is reset.					

Notes: 1. An error of one cycle of the count source for the free-run counter will occur.
2. f(PBA) = at 40-MHz operation
3. f(WDT) = at 125-kHz operation

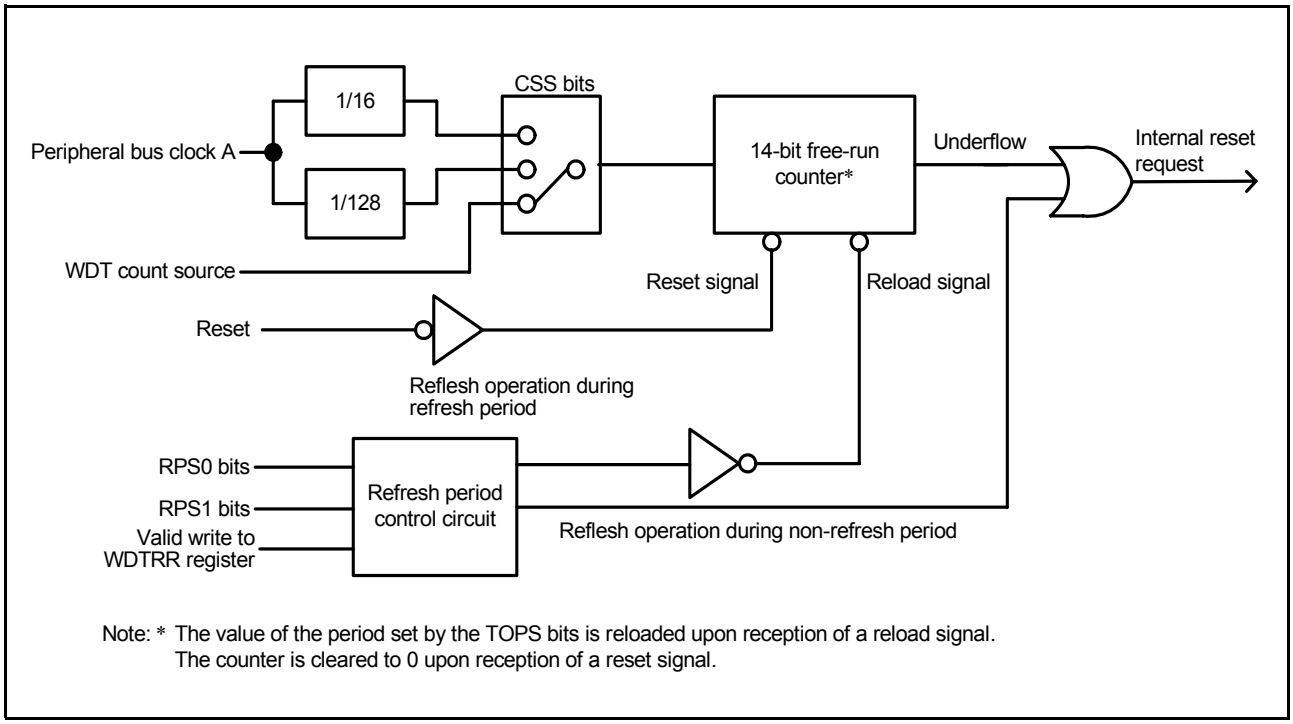


Figure 10.1 Watchdog Timer Block Diagram

10.2 Registers

Table 10.3 lists the Watchdog Timer Registers.

Table 10.3 Watchdog Timer Registers

Register Name	Symbol	After Reset	Address	Access Size
WDT refresh register	WDTRR	—	H'FF46 8003	8
WDT control register 0	WDTCR0	H'DE	H'FF46 8002	8
WDT register	WDTR	H'0000	H'FF46 8000	16
WDT control register 1	WDTCR1	H'03	H'FF46 8013	8

10.2.1 WDT Refresh Register (WDTRR)

Address H'FF46 8003

	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	—	—	—	—	—	—	—	—

Bit	Function	R/W
b7 to b0	The watchdog timer is refreshed by writing H'00 and then H'FF to this register continuously. Writing a value other than the above is disabled.	W

The WDTRR register is a write-only register that is used to refresh the watchdog timer.

The value set by the TOPS bits in the WDTCR0 register is reloaded to the 14-bit free-run counter by writing H'00 and then H'FF to this register during a refresh period.

The MCU is reset when the free-run counter underflows or by performing a refresh operation in a non-refresh period.

10.2.2 WDT Control Register 0 (WDTCR0)

Address H'FF46 8002

	b7	b6	b5	b4	b3	b2	b1	b0
	CSS[1:0]		—	RPS0[1:0]		TOPS[1:0]		—
After Reset	1	1	0	1	1	1	1	0

Bit	Symbol	Bit Name	Description	R/W
b7, b6	CSS[1:0]	Count Source Select Bits	b7 b6 0 0 : Peripheral bus clock A divided by 16 0 1 : Peripheral bus clock A divided by 128 1 0 : Setting prohibited 1 1 : WDT count source (output from the low-speed on-chip oscillator)	R/W
b5	—	Reserved	This bit is read as 0. The write value should be 0.	R
b4, b3	RPS0[1:0]	Refresh Period Select Bits	b4 b3 0 0 : 25% 0 1 : 50% 1 0 : 75% 1 1 : 100%	R/W
b2, b1	TOPS[1:0]	Underflow Period Select Bits	b2 b1 0 0 : 1024 cycles (initial value: H'03FF) 0 1 : 4096 cycles (initial value: H'0FFF) 1 0 : 8192 cycles (initial value: H'1FFF) 1 1 : 16384 cycles (initial value: H'3FFF)	R/W
b0	—	Reserved	This bit is read as 0. The write value should be 0.	R

The WDTCR0 register can be written only once after a reset.

CSS Bits

These bits are used to select a count source for the watchdog timer.

RPS0 Bits

These bits are used to set a refresh period for the watchdog timer.

Table 10.4 Watchdog Timer Refresh Period

Time-Out Period	100%	75%	50%	25%
H'3FFF (initial value)	H'3FFF or less	H'2FFF or less	H'1FFF or less	H'0FFF or less
H'1FFF (initial value)	H'1FFF or less	H'17FF or less	H'0FFF or less	H'07FF or less
H'0FFF (initial value)	H'0FFF or less	H'0BFF or less	H'07FF or less	H'03FF or less
H'03FF (initial value)	H'03FF or less	H'02FF or less	H'01FF or less	H'00FF or less

TOPS Bits

These bits are used to set an underflow period of the watchdog timer.

10.2.3 WDT Register (WDTR)

Address H'FF46 8000

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—														
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, b14	—	Reserved	The read value is undefined.	R
b13 to b0	The watchdog timer value is read.			R

This register is used to read the counter value of the watchdog timer.

10.2.4 WDT Control Register 1 (WDTCR1)

Address H'FF46 8013

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	RPS1[1:0]	
After Reset	0	0	0	0	0	0	1	1

Bit	Symbol	Bit Name	Description	R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b1, b0	RPS1[1:0]	Non-Refresh Period Select Bits	b1 b0 0 0 : 75% 0 1 : 50% 1 0 : 25% 1 1 : No non-refresh period	R/W

Write to the WDTCR1 register after a reset, and then write to the WDTCR0 register.

RPS1 Bits

These bits are used to select a non-refresh period. A non-refresh period is set after selected from the refresh periods set by the PRS0 bits.

10.3 Operations

A period to accept a refresh operation for the watchdog timer is set by the combination the RPS0 bits in the WDTCR0 register and the RPS1 bits in the WDTCR1 register. Figure 10.2 shows the Period to Accept Refresh Operation for Watchdog Timer.

Assuming that the time-out period is 100%, a refresh operation executed during an acceptable period is accepted.

A reset operation executed during a non-acceptable period is considered as an incorrect write, and the MCU is reset.

Period to accept a refresh operation set by the combination of bits RPS0 and RPS1

		RPS1 Bits			
		00	01	10	11
RPS0 bits	00	Setting prohibited	Setting prohibited	Setting prohibited	(10)
	01	Setting prohibited	Setting prohibited	(9)	(8)
	10	Setting prohibited	(7)	(6)	(5)
	11	(4)	(3)	(2)	(1)

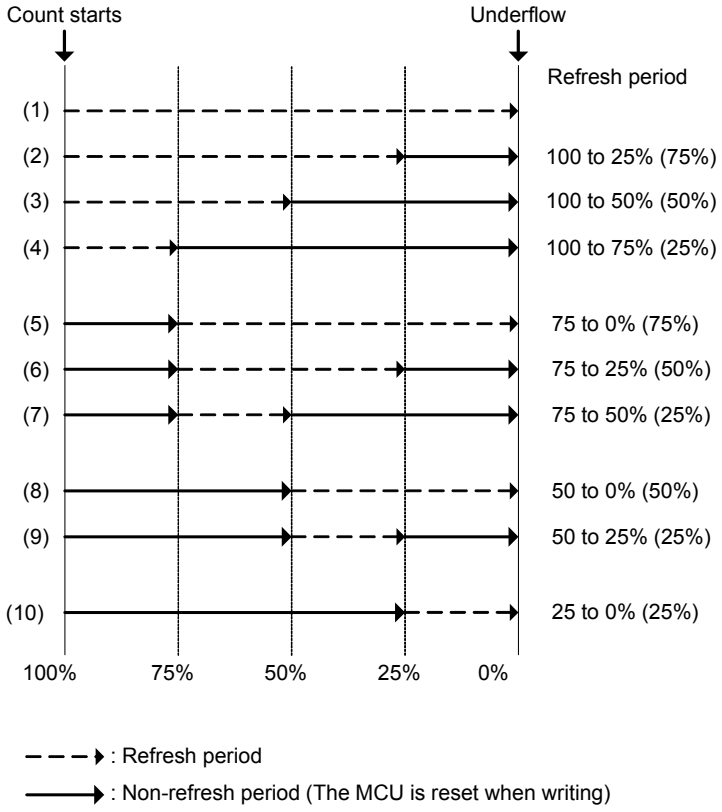


Figure 10.2 Period to Accept Refresh Operation for Watchdog Timer

10.4 Usage Notes on Watchdog Timer

Make the following settings for the watchdog timer before the transition to standby mode.

1. Set the CSS bits in the WDTCR0 register to B'11.
2. Set the TOPS bits in the WDTCR0 register to B'11.
3. Refresh the watchdog timer.

Restore the previous values to the watchdog timer after wake-up from standby mode.

11. Task Monitor Timer

11.1 Introduction

The 16-bit task monitor timer decrements the value set in the timer, and generates an interrupt request when it underflows. The timer automatically stops when it underflows.

Table 11.1 lists the Task Monitor Timer Specifications.
Figure 11.1 shows the Task Monitor Timer Block Diagram.

Table 11.1 Task Monitor Timer Specifications

Item	Specification
Number of channels	1 channel
Count sources	• Peripheral bus clock A divided by 2 • Peripheral bus clock A divided by 16 • Peripheral bus clock A divided by 64 • Peripheral bus clock A divided by 256
Count operations	• Decrement • When the timer underflows, count stops (by writing to the TMTLEN register, the contents of the reload register are reloaded to the counter and count restarts).
Counter period	(n + 1)/fj n: Value set in the TMTR register (H'0000 to H'FFFF) fj: Frequency of the count source
Count start conditions	When the following two conditions are met (order is random): • The TEN bit in the TMTEN register is set to 1 (count enabled) • The counter value is other than H'0000
Count stop conditions	• The TEN bit in the TMTEN register is set to 0 (count stopped) • When the timer underflows (the counter reaches H'0000)
Interrupt request generation timing	When the timer underflows
Read from timer	The counter value is read by reading the TMTR register.
Write to timer	When a value is written to the TMTR register, the value is written to the reload register at the same time. Then the value of the reload register is written to the counter by writing to the TMTLEN register.

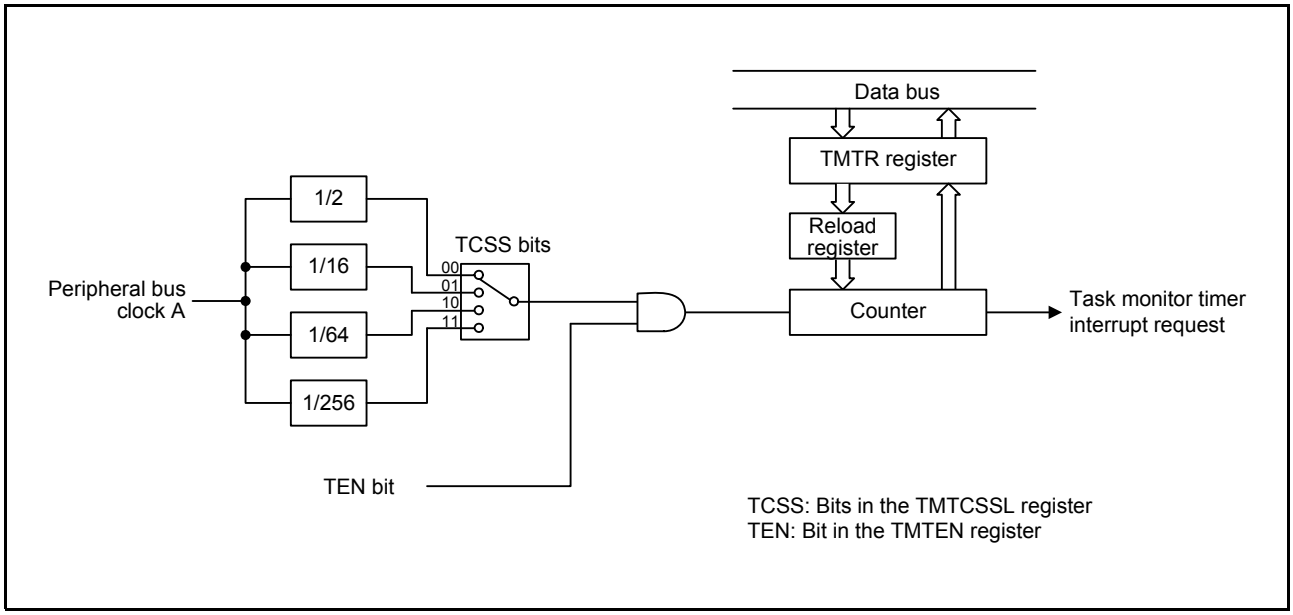


Figure 11.1 Task Monitor Timer Block Diagram

11.2 Registers

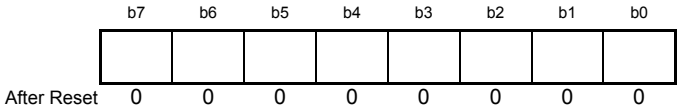
Table 11.2 lists the Task Monitor Timer Registers.

Table 11.2 Task Monitor Timer Registers

Register Name	Symbol	After Reset	Address	Access Size
Task monitor timer protect register	TMTPR	H'00	H'FF46 C00F	8
Task monitor timer load enable register	TMTLEN	—	H'FF46 C007	8
Task monitor timer count source select register	TMTCSSL	H'00	H'FF46 C006	8
Task monitor timer register	TMTR	H'0000	H'FF46 C004	16
Task monitor timer enable register	TMTEN	H'00	H'FF46 C00B	8

11.2.1 Task Monitor Timer Protect Register (TMTPR)

Address H'FF46 C00F



Bit	Description	R/W
b7 to b0	When written B'1111 0001: Protection unlocked A value other than B'1111 0001: Protection locked When read Bit 0 (b0) 0: Protection locked 1: Protection unlocked Bit 7 to bit 1 (b7 to b1) are always read as 0.	R/W

The TMTPR register is used to set the protect function that protects registers TMTLEN, TMTCSSL, TMTR, and TMTEN from being rewritten easily. To change the values of registers TMTLEN, TMTCSSL, TMTR, and TMTEN, perform the following procedure:

(1) Write H'F1 to the TMTPR register (writing to the registers enabled).
(2) Change the values of registers TMTLEN, TMTCSSL, TMTR, and TMTEN.
(3) Write a value other than H'F1 to the TMTPR register (writing to the registers disabled).

11.2.2 Task Monitor Timer Load Enable Register (TMTLEN)

Address H'FF46 C007

	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	—	—	—	—	—	—	—	—

Bit	Description	R/W
b7 to b0	The value written to the TMTR register (the value of the reload register) is written to the counter.	W

TMTLEN is a write-only register.
When an arbitrary value (H'00 to H'FF) is written to the TMTLEN register, the value written to the TMTR register (the value of the reload register) is written to the counter. To write to the TMTLEN register, use the TMTPR register to unlock the protection beforehand.

11.2.3 Task Monitor Timer Count Source Select Register (TMTCSSL)

Address H'FF46 C006

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	TCSS[1:0]	
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should always be 0.	R
b1, b0	TCSS[1:0]	Task Monitor Timer Count Source Select Bits	b1 b0 0 0 : Peripheral bus clock A divided by 2 0 1 : Peripheral bus clock A divided by 16 1 0 : Peripheral bus clock A divided by 64 1 1 : Peripheral bus clock A divided by 256	R/W

To change the value of the TMTCSSL register, use the TMTPR register to unlock the protection beforehand.

TCSS Bits

These bits are used to select a count source for the task monitor timer.

11.2.4 Task Monitor Timer Register (TMTR)

Address H'FF46 C004

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Description	R/W
b15 to b0	When written The value is written to the reload register without changes. (Then the value of the reload register is written to the counter by writing to the TMTLEN register.) If the setting value is n, the counter period is (n + 1)/fj. fj: Frequency of the count source Range of the setting value (n): H'0000 to H'FFFF When read The counter value can be read.	R/W

Set the TMTR register in 16-bit units.

To change the value of the TMTR register, use the TMTPR register to unlock the protection beforehand.

11.2.5 Task Monitor Timer Enable Register (TMTEN)

Address H'FF46 C00B

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	TEN
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should always be 0.	R
b0	TEN	Task Monitor Timer Enable Bit	0: Count stopped 1: Count enabled	R/W

To change the value of the TMTEN register, use the TMTPR register to unlock the protection beforehand.

TEN Bit

This bit is used to enable or stop the task monitor timer.

11.3 Operation

Figure 11.2 shows an Operation Example of Task Monitor Timer.

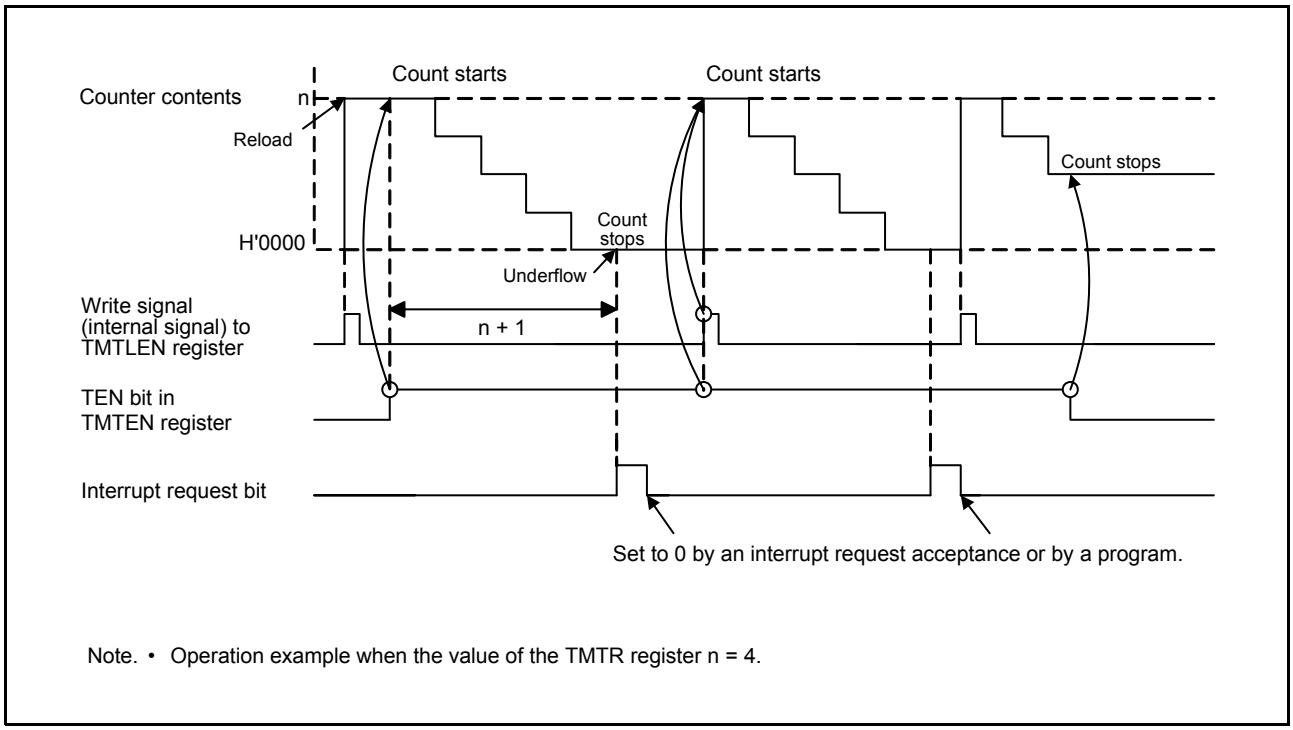


Figure 11.2 Operation Example of Task Monitor Timer

11.4 Notes on Task Monitor Timer

The timer is stopped after a reset. After setting a count source by the TMTCSSL register, set the TEN bit in the TMTEN register to 1 (count enabled).

To change the count source even after a reset, set the TEN bit to 0 (count stopped) before making the change.

12. DMAC

12.1 Introduction

The DMAC (direct memory access controller) is a module that handles data transfer without using the CPU, and consists of 8 channels. When a transfer request is generated, the DMAC transfers data of the source address to the destination address. The number of bytes of data set in the DMiCNT counter (i = 0 to 7) is transferred per DMA transfer.

Table 12.1 lists the DMAC Specifications. Figure 12.1 shows the DMAC Block Diagram.

Table 12.1 DMAC Specifications

Item		Description	
Number of channels		8 channels	
Transfer memory space		4 Gbytes (areas other than reserved areas in addresses H'0000 0000 to H'FFFF FFFF)	
Maximum number of transfers		64 Mbytes	
DMA request sources		• Software trigger • Interrupt requests from peripheral functions	
Channel priority level		DMA0 > DMA1 > ... DMA6 > DMA7 (DMA0 is the highest)	
Transfer data	One data	8, 16, and 32 bits	
	One operand	1, 2, 4, 8, 16, 32, 64, and 128 data units	
Transfer modes	Cycle-stealing transfer mode	The bus mastership is returned between the read and write accesses of transferring one data.	
	Pipelined transfer mode	Transfer is continuously performed until a single operand transfer is completed.	
Transfer methods	Operand transfer methods	Unit	• One operand is transferred per DMA request. • Channel arbitration is performed after a single operand transfer is completed. • A DMA request is required for each completion of a single operand transfer until DMA transfer is completed.
		Sequential	• One operand is repeatedly transferred per DMA request until DMA transfer is completed. • Channel arbitration is performed each time a single operand transfer is completed. • Only the first DMA request is required.
	Non-stop transfer method	• Transfer is continuously performed per DMA request until DMA transfer is completed. • Channel arbitration is not performed until DMA transfer is completed. • Only the first DMA request is required.	
DMA transfer start conditions		DMA transfer starts when all of the following conditions are met: • The DEN bit in the DMiCR1 register (i = 0 to 7) is set to 1 (DMA transfer enabled) • The DMST bit in the DMSTR register is set to 1 (DMAC operates) • A DMA request for DMAi is generated and the execution right is obtained through channel arbitration.	
DMA transfer complete condition		When the DMiCNT counter reaches H'0000 000 (DMA transfer completed)	
Interrupt request generation timing		When the DMiCNT counter reaches H'0000 000 (DMA transfer completed)	
Single-data transfer time		• Cycle-stealing transfer mode: Minimum of 3 bus clocks • Pipelined transfer mode: Minimum of 1 bus clock	
Other function		• Reload function When DMA transfer is completed, the values of the reload registers for the source address, destination address, and transfer byte count are reloaded into the current registers.	

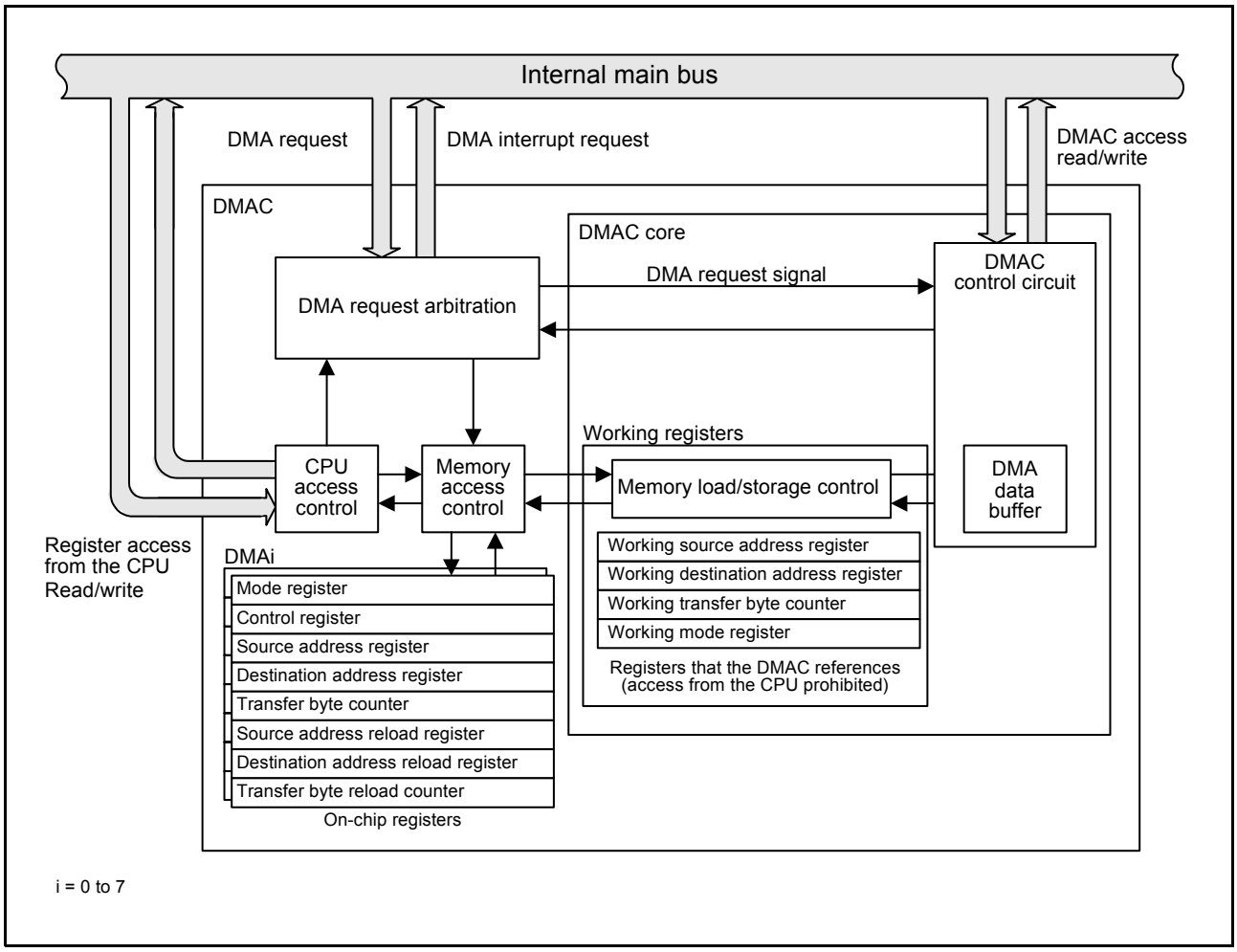


Figure 12.1 DMAC Block Diagram

12.2 Registers

Table 12.2 and Table 12.3 list the DMAC Registers.

Table 12.2 DMAC Registers (1)

Register Name	Symbol	After Reset	Address	Access Size
DMA0 mode register	DM0MD	Undefined	H'FF46 000C	32
DMA1 mode register	DM1MD	Undefined	H'FF46 001C	32
DMA2 mode register	DM2MD	Undefined	H'FF46 002C	32
DMA3 mode register	DM3MD	Undefined	H'FF46 003C	32
DMA4 mode register	DM4MD	Undefined	H'FF46 004C	32
DMA5 mode register	DM5MD	Undefined	H'FF46 005C	32
DMA6 mode register	DM6MD	Undefined	H'FF46 006C	32
DMA7 mode register	DM7MD	Undefined	H'FF46 007C	32
DMA0 control register 0	DM0CR0	H'0000 0000	H'FF46 0400	32
DMA1 control register 0	DM1CR0	H'0000 0000	H'FF46 0408	32
DMA2 control register 0	DM2CR0	H'0000 0000	H'FF46 0410	32
DMA3 control register 0	DM3CR0	H'0000 0000	H'FF46 0418	32
DMA4 control register 0	DM4CR0	H'0000 0000	H'FF46 0420	32
DMA5 control register 0	DM5CR0	H'0000 0000	H'FF46 0428	32
DMA6 control register 0	DM6CR0	H'0000 0000	H'FF46 0430	32
DMA7 control register 0	DM7CR0	H'0000 0000	H'FF46 0438	32
DMA0 control register 1	DM0CR1	H'0000 0000	H'FF46 0404	32
DMA1 control register 1	DM1CR1	H'0000 0000	H'FF46 040C	32
DMA2 control register 1	DM2CR1	H'0000 0000	H'FF46 0414	32
DMA3 control register 1	DM3CR1	H'0000 0000	H'FF46 041C	32
DMA4 control register 1	DM4CR1	H'0000 0000	H'FF46 0424	32
DMA5 control register 1	DM5CR1	H'0000 0000	H'FF46 042C	32
DMA6 control register 1	DM6CR1	H'0000 0000	H'FF46 0434	32
DMA7 control register 1	DM7CR1	H'0000 0000	H'FF46 043C	32
DMA0 source address register	DM0SA	Undefined	H'FF46 0000	32
DMA1 source address register	DM1SA	Undefined	H'FF46 0010	32
DMA2 source address register	DM2SA	Undefined	H'FF46 0020	32
DMA3 source address register	DM3SA	Undefined	H'FF46 0030	32
DMA4 source address register	DM4SA	Undefined	H'FF46 0040	32
DMA5 source address register	DM5SA	Undefined	H'FF46 0050	32
DMA6 source address register	DM6SA	Undefined	H'FF46 0060	32
DMA7 source address register	DM7SA	Undefined	H'FF46 0070	32
DMA0 destination address register	DM0DA	Undefined	H'FF46 0004	32
DMA1 destination address register	DM1DA	Undefined	H'FF46 0014	32
DMA2 destination address register	DM2DA	Undefined	H'FF46 0024	32
DMA3 destination address register	DM3DA	Undefined	H'FF46 0034	32
DMA4 destination address register	DM4DA	Undefined	H'FF46 0044	32
DMA5 destination address register	DM5DA	Undefined	H'FF46 0054	32
DMA6 destination address register	DM6DA	Undefined	H'FF46 0064	32
DMA7 destination address register	DM7DA	Undefined	H'FF46 0074	32

Table 12.3 DMAC Registers (2)

Register Name	Symbol	After Reset	Address	Access Size
DMA0 transfer byte counter	DM0CNT	Undefined	H'FF46 0008	32
DMA1 transfer byte counter	DM1CNT	Undefined	H'FF46 0018	32
DMA2 transfer byte counter	DM2CNT	Undefined	H'FF46 0028	32
DMA3 transfer byte counter	DM3CNT	Undefined	H'FF46 0038	32
DMA4 transfer byte counter	DM4CNT	Undefined	H'FF46 0048	32
DMA5 transfer byte counter	DM5CNT	Undefined	H'FF46 0058	32
DMA6 transfer byte counter	DM6CNT	Undefined	H'FF46 0068	32
DMA7 transfer byte counter	DM7CNT	Undefined	H'FF46 0078	32
DMA0 source address reload register	DM0SAR	Undefined	H'FF46 0200	32
DMA1 source address reload register	DM1SAR	Undefined	H'FF46 0210	32
DMA2 source address reload register	DM2SAR	Undefined	H'FF46 0220	32
DMA3 source address reload register	DM3SAR	Undefined	H'FF46 0230	32
DMA4 source address reload register	DM4SAR	Undefined	H'FF46 0240	32
DMA5 source address reload register	DM5SAR	Undefined	H'FF46 0250	32
DMA6 source address reload register	DM6SAR	Undefined	H'FF46 0260	32
DMA7 source address reload register	DM7SAR	Undefined	H'FF46 0270	32
DMA0 destination address reload register	DM0DAR	Undefined	H'FF46 0204	32
DMA1 destination address reload register	DM1DAR	Undefined	H'FF46 0214	32
DMA2 destination address reload register	DM2DAR	Undefined	H'FF46 0224	32
DMA3 destination address reload register	DM3DAR	Undefined	H'FF46 0234	32
DMA4 destination address reload register	DM4DAR	Undefined	H'FF46 0244	32
DMA5 destination address reload register	DM5DAR	Undefined	H'FF46 0254	32
DMA6 destination address reload register	DM6DAR	Undefined	H'FF46 0264	32
DMA7 destination address reload register	DM7DAR	Undefined	H'FF46 0274	32
DMA0 transfer byte counter reload register	DM0CNTR	Undefined	H'FF46 0208	32
DMA1 transfer byte counter reload register	DM1CNTR	Undefined	H'FF46 0218	32
DMA2 transfer byte counter reload register	DM2CNTR	Undefined	H'FF46 0228	32
DMA3 transfer byte counter reload register	DM3CNTR	Undefined	H'FF46 0238	32
DMA4 transfer byte counter reload register	DM4CNTR	Undefined	H'FF46 0248	32
DMA5 transfer byte counter reload register	DM5CNTR	Undefined	H'FF46 0258	32
DMA6 transfer byte counter reload register	DM6CNTR	Undefined	H'FF46 0268	32
DMA7 transfer byte counter reload register	DM7CNTR	Undefined	H'FF46 0278	32
DMA interrupt control register	DMICR	H'0000	H'FF46 0508	16
DMA start register	DMSTR	H'00	H'FF46 0501	8
DMA transfer status register	DMSR	H'0000	H'FF46 0518	16
DMA transfer completion status register	DMCSR	H'0000	H'FF46 0514	16

12.2.1 DMAi Mode Register (DMiMD) (i = 0 to 7)

Address DM0MD: H'FF46 000C, DM1MD: H'FF46 001C, DM2MD: H'FF46 002C, DM3MD: H'FF46 003C, DM4MD: H'FF46 004C, DM5MD: H'FF46 005C, DM6MD: H'FF46 006C, DM7MD: H'FF46 007C

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	OPSEL [3:0]				—	—	—	—	—	SZSEL [2:0]		
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
	—	SAD [2:0]			—	DAD [2:0]			—	—	—	—	—	—	—	—
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
b31 to b28	—	Reserved	The read value is undefined. The write value should be 0.	R
b27 to b24	OPSEL [3:0]	Operand Transfer Data Count Select Bits	b27b26b25b24 0 0 0 0 : 1 data unit 0 0 0 1 : 2 data units 0 0 1 0 : 4 data units 0 0 1 1 : 8 data units 0 1 0 0 : 16 data units 0 1 0 1 : 32 data units 0 1 1 0 : 64 data units 0 1 1 1 : 128 data units Do not set values other than the above.	R/W
b23 to b19	—	Reserved	The read value is undefined. The write value should be 0.	R
b18 to b16	SZSEL [2:0]	Transfer Data Size Select Bits	b18b17b16 0 0 0 : 8 bits 0 0 1 : 16 bits 0 1 0 : 32 bits Do not set values other than the above.	R/W
b15	—	Reserved	The read value is undefined. The write value should be 0.	R
b14 to b12	SAD [2:0]	Source Address Direction Control Bits	b14b13b12 0 0 0 : Fixed 0 0 1 : Increment 0 1 0 : Decrement 0 1 1 : Rotation Do not set values other than the above.	R/W
b11	—	Reserved	The read value is undefined. The write value should be 0.	R
b10 to b8	DAD [2:0]	Destination Address Direction Control Bits	b10 b9 b8 0 0 0 : Fixed 0 0 1 : Increment 0 1 0 : Decrement 0 1 1 : Rotation Do not set values other than the above.	R/W
b7 to b0	—	Reserved	The read value is undefined. The write value should be 0.	R

The DMiMD register is used to set the calculation content of a transfer source address or destination address and the size of transfer data. Write data to the DMiMD register when the DMAC is stopped, or when DMA transfer is disabled and data is not being transferred on the corresponding channel.

Access the DMiMD register in 32-bit units.

OPSEL Bits

These bits are used to set the number of data units transferred in a single operand. When using the operand transfer method, the number of data units set by the OPSEL bits is transferred continuously as one operand. When using the non-stop transfer method, the setting of the OPSEL bits becomes invalid and the number of bytes of data set in the DMiCNT counter is transferred continuously.

SZSEL Bits

These bits are used to set the size of transfer data.

Bits SAD and DAD

These bits are used to set the calculation content of an address which is being DMA transferred.
When rotation is selected, the address is incremented. If a single operand transfer is completed, the address is set to the value when DMA transfer is started.
The calculation content of the address will be as follows according to the transfer size.

Table 12.4 Address Calculation Content According to Transfer Size

SZSEL Bits	Bits SAD and DAD			
	B'000 (Fixed)	B'001 (Increment)	B'010 (Decrement)	B'011 (Rotation)
B'000 (8 bits)	±0	+1	-1	+1
B'001 (16 bits)	±0	+2	-2	+2
B'010 (32 bits)	±0	+4	-4	+4

12.2.2 DMAi Control Register 0 (DMiCR0) (i = 0 to 7)

Address DM0CR0: H'FF46 0400, DM1CR0: H'FF46 0408, DM2CR0: H'FF46 0410, DM3CR0: H'FF46 0418, DM4CR0: H'FF46 0420, DM5CR0: H'FF46 0428, DM6CR0: H'FF46 0430, DM7CR0: H'FF46 0438

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	MDSEL [1:0]		—	—	DSEL [1:0]		—	—	—	—	—	—	STRG [1:0]	
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
	—	—	—	—	—	BRLOD	SRLOD	DRLOD	DCTG [7:0]							
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, b30	—	Reserved	These bits are read as 0. The write value should be 0.	R
b29, b28	MDSEL [1:0]	Transfer Mode Select Bits	b29b28 0 0 : Cycle-stealing transfer mode 0 1 : Pipelined transfer mode Do not set values other than the above.	R/W
b27, b26	—	Reserved	These bits are read as 0. The write value should be 0.	R
b25, b24	DSEL [1:0]	Transfer Method Select Bits	b25b24 0 0 : Unit operand transfer 0 1 : Sequential operand transfer 1 0 : Do not set. 1 1 : Non-stop transfer	R/W
b23 to b18	—	Reserved	These bits are read as 0. The write value should be 0.	R
b17, b16	STRG [1:0]	Input Sense Mode Select Bits	b17b16 0 0 : Software trigger 0 1 : Do not set. 1 0 : Edge sense 1 1 : Level sense	R/W
b15 to b11	—	Reserved	These bits are read as 0. The write value should be 0.	R
b10	BRLOD	Transfer Byte Count Reload Function Select Bit	0: Transfer byte count reload function not used 1: Transfer byte count reload function used	R/W
b9	SRLOD	Source Address Reload Function Select Bit	0: Source address reload function not used 1: Source address reload function used	R/W
b8	DRLOD	Destination Address Reload Function Select Bit	0: Destination address reload function not used 1: Destination address reload function used	R/W
b7 to b0	DCTG [7:0]	DMA Request Source Select Bits	Used to set DMA request sources (see Table 12.6).	R/W

MDSEL Bits

These bits are used to set a transfer mode. To set the MDSEL bits, write a value when the DMAC is stopped, or when DMA transfer is disabled and data is not being transferred on the corresponding channel.

DSEL Bits

These bits are used to set a transfer method. To set the DSEL bits, write a value when the DMAC is stopped, or when DMA transfer is disabled and data is not being transferred on the corresponding channel.

STRG Bits

These bits are used to set the input sense mode for the DMA request signals (internal signals) input from the request source set by the DCTG bits to the DMAC. Table 12.5 shows the combinations of the STRG and DCTG bits. Set the STRG bits in accord with the settings of the DCTG bits.

Table 12.5 Combinations of the STRG and DCTG Bits

DMA Request Source No.	DCTG Bits	DMA Request Source	STRG Bits	Input Sense Mode
0	0000 0000	Software trigger	00	Software trigger
1 ⋮ 169	0000 0001 ⋮ 1010 1001	CMT0 interrupt request (CMI0) ⋮ SBI3 transmit interrupt request	10 ⋮ 10	Edge sense ⋮ Edge sense
172 ⋮ 198	1010 1100 ⋮ 1100 0110	MT0GRA interrupt request ⋮ MT7GRD interrupt request	11 ⋮ 11	Level sense ⋮ Level sense

Bits DRLOD, SRLOD, and BRLOD

These bits are used to control the destination address, source address, and transfer byte count reload functions. If setting these bits to 1, the contents of the respective reload registers are reloaded to the respective current registers when the DMA transfer is completed. When not using the reload function, set the ECLR bit in the DMiCR1 register to 1 so that the DEN bit should be set to 0.

DCTG Bits

These bits are used to set DMA request sources. To set the DCTG bits, write a value when the DMAC is stopped, or when DMA transfer is disabled and data is not being transferred on the corresponding channel. If the DCTG bits are set, make sure DMAC operation and DMA transfer are enabled after setting the DREQ bit in the DMiCR1 register to 0.

Table 12.6 lists the DCTG Bit Setting.

Table 12.6 DCTG Bit Setting

DMA Request Source No.	DCTG Bits	DMA Request Source
0	0000 0000	Software trigger
1	0000 0001	CMT0 interrupt request (CMI0)
2	0000 0010	CMT1 interrupt request (CMI1)
3	0000 0011	CMT2 interrupt request (CMI2)
4	0000 0100	CMT3 interrupt request (CMI3)
5	0000 0101	CMT4 interrupt request (CMI4)
6	0000 0110	CMT5 interrupt request (CMI5)
7	0000 0111	AD1 scan conversion end interrupt request
8	0000 1000	AD1IN0 interrupt conversion end interrupt request*
9	0000 1001	AD1IN1 interrupt conversion end interrupt request*
10	0000 1010	AD1IN2 interrupt conversion end interrupt request*
11	0000 1011	AD1IN3 interrupt conversion end interrupt request*
12	0000 1100	AD1IN4 interrupt conversion end interrupt request
13	0000 1101	AD1IN5 interrupt conversion end interrupt request
14	0000 1110	AD1IN6 interrupt conversion end interrupt request*
15	0000 1111	AD1IN7 interrupt conversion end interrupt request*
16 to 23	0001 0000 to 0001 0111	(Reserved) Do not set.
24	0001 1000	AD1IN16 interrupt conversion end interrupt request
25	0001 1001	AD1IN17 interrupt conversion end interrupt request
26 to 31	0001 1010 to 0001 1111	(Reserved) Do not set.
32	0010 0000	AD1IN24 interrupt conversion end interrupt request
33	0010 0001	AD1IN25 interrupt conversion end interrupt request*
34	0010 0010	AD1IN26 interrupt conversion end interrupt request*
35	0010 0011	AD1IN27 interrupt conversion end interrupt request*
36	0010 0100	AD1IN28 interrupt conversion end interrupt request
37	0010 0101	AD1IN29 interrupt conversion end interrupt request*
38	0010 0110	AD1IN30 interrupt conversion end interrupt request
39	0010 0111	AD1IN31 interrupt conversion end interrupt request
40 to 49	00010 1000 to 0011 0001	(Reserved) Do not set.
50	0011 0010	AD1IN42 interrupt conversion end interrupt request*
51	0011 0011	AD1IN43 interrupt conversion end interrupt request*
52	0011 0100	AD1IN44 interrupt conversion end interrupt request*
53	0011 0101	AD1IN45 interrupt conversion end interrupt request*
54	0011 0110	AD1IN46 interrupt conversion end interrupt request*
55	0011 0111	AD1IN47 interrupt conversion end interrupt request*
56 to 72	0011 1000 to 0100 1000	(Reserved) Do not set.
73	0100 1001	AD0 scan conversion end interrupt request
74 to 77	0100 1010 to 0100 1101	(Reserved) Do not set.
78	0100 1110	TP1GR0 interrupt request
79	0100 1111	TP1GR1 interrupt request
80	0101 0000	TP1GR2 interrupt request
81	0101 0001	TP1GR3 interrupt request
82	0101 0010	TP2GR0 interrupt request
83	0101 0011	TP2GR1 interrupt request
84	0101 0100	TP2GR2 interrupt request
85	0101 0101	TP2GR3 interrupt request
86	0101 0110	TP3GR0 interrupt request
87	0101 0111	TP3GR1 interrupt request
88	0101 1000	TP3GR2 interrupt request
89	0101 1001	TP3GR3 interrupt request
90	0101 1010	TP4GR0 interrupt request
91	0101 1011	TP4GR1 interrupt request
92	0101 1100	TP4GR2 interrupt request
93	0101 1101	TP4GR3 interrupt request
94 to 138	0101 1110 to 1000 1010	(Reserved) Do not set.
139	1000 1011	TPU1 counter reset interrupt request
140	1000 1100	TPU2 counter reset interrupt request
141	1000 1101	TPU3 counter reset interrupt request
142	1000 1110	TPU4 counter reset interrupt request
143 to 153	1000 1111 to 1001 1001	(Reserved) Do not set.
154	1001 1010	SCI0 receive buffer full interrupt request
155	1001 1011	SCI0 transmit buffer empty interrupt request
156	1001 1100	SCI1 receive buffer full interrupt request
157	1001 1101	SCI1 transmit buffer empty interrupt request
158	1001 1110	SCI2 receive buffer full interrupt request
159	1001 1111	SCI2 transmit buffer empty interrupt request
160	1010 0000	SCI3 receive buffer full interrupt request
161	1010 0001	SCI3 transmit buffer empty interrupt request
162	1010 0010	SBI0 receive interrupt request
163	1010 0011	SBI1 receive interrupt request
164	1010 0100	SBI2 receive interrupt request
165	1010 0101	SBI3 receive interrupt request*
166	1010 0110	SBI0 transmit interrupt request
167	1010 0111	SBI1 transmit interrupt request
168	1010 1000	SBI2 transmit interrupt request
169	1010 1001	SBI3 transmit interrupt request*
170, 171	1010 1010, 1010 1011	(Reserved) Do not set.
172	1010 1100	MT0GRA interrupt request
173	1010 1101	MT0GRB interrupt request
174	1010 1110	MT0GRC interrupt request
175	1010 1111	MT0GRD interrupt request
176	1011 0000	MT1GRA interrupt request
177	1011 0001	MT1GRB interrupt request
178	1011 0010	MT2GRA interrupt request
179	1011 0011	MT2GRB interrupt request
180	1011 0100	MT3GRA interrupt request
181	1011 0101	MT3GRB interrupt request
182	1011 0110	MT3GRC interrupt request
183	1011 0111	MT3GRD interrupt request
184	1011 1000	MT4GRA interrupt request

DMA Request Source No.	DCTG Bits	DMA Request Source
185	1011 1001	MT4GRB interrupt request
186	1011 1010	MT4GRC interrupt request
187	1011 1011	MT4GRD interrupt request
188	1011 1100	MT5GRU interrupt request
189	1011 1101	MT5GRV interrupt request
190	1011 1110	MT5GRW interrupt request
191	1011 1111	MT6GRA interrupt request
192	1100 0000	MT6GRB interrupt request
193	1100 0001	MT6GRC interrupt request
194	1100 0010	MT6GRD interrupt request
195	1100 0011	MT7GRA interrupt request
196	1100 0100	MT7GRB interrupt request
197	1100 0101	MT7GRC interrupt request
198	1100 0110	MT7GRD interrupt request
199 to 255	1100 0111 to 1111 1111	(Reserved) Do not set.

Note: * Not used in the SH72A0 group.

12.2.3 DMAi Control Register 1 (DMiCR1) (i = 0 to 7)

Address DM0CR1: H'FF46 0404, DM1CR1: H'FF46 040C, DM2CR1: H'FF46 0414, DM3CR1: H'FF46 041C, DM4CR1: H'FF46 0424, DM5CR1: H'FF46 042C, DM6CR1: H'FF46 0434, DM7CR1: H'FF46 043C

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	DEN	—	—	—	—	—	—	—	DREQ
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
	—	—	—	—	—	—	—	ECLR	—	—	—	—	—	—	—	DSCLR
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 b25	—	Reserved	These bits are read as 0. The write value should be 0.	R
b24	DEN	DMA Transfer Enable Bit	0: DMA transfer disabled 1: DMA transfer enabled	R/W
b23 to b17	—	Reserved	These bits are read as 0. The write value should be 0.	R
b16	DREQ	DMA Request Bit	0: DMA request is not present 1: DMA request is present	R/W
b15 to b9	—	Reserved	These bits are read as 0. The write value should be 0.	R
b8	ECLR	DMA Transfer Enable Clear Bit	0: The DEN bit is not set to 0 when DMA transfer is completed 1: The DEN bit is set to 0 when DMA transfer is completed	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R
b0	DSCLR	DMAC Internal State Initialize Bit	Writing 1 initializes the DMAC internal state. Writing 0 is invalid. The read value is 0.	R/W

DEN Bit

This bit is used to enable DMA transfer. When the ECLR bit is 1, the DEN bit is automatically set to 0 when DMA transfer is completed.

When the DEN bit is set to 0 during a transfer with the operand transfer method, the channel suspends DMA transfer after the current single operand transfer is completed. If the DEN bit is then set to 1, DMA transfer is restarted. During non-stop transfer, DMA transfer is not suspended even if the DEN bit is set to 0, and transfer continues until DMA transfer is completed.

DREQ Bit

This bit indicates whether a DMA request is currently present. The DREQ bit varies with whether a DMA request is currently present even while the DMAC is stopped or DMA transfer is disabled. When the software trigger is set as the DMA request source, writing 1 to the DREQ bit by a program generates a DMA request. If the DMA request source is other than the software trigger, do not write 1 to the DREQ bit by a program. When writing 0 to the DREQ bit, write the value when the DMAC is stopped, or when the DMA transfer is disabled and data is not being transferred on the corresponding channel. When writing 1 to the DREQ bit, the value can be written regardless of the DMA transfer state.

The DREQ bit is set (DREQ = 1) or cleared (DREQ = 0) depending on the DMA request source as follows.

(1) If the DMA request source is a software trigger
(The DCTG bits are 0000 0000, and the STRG bits are 00.)
[Condition to become 1]
• The program writes 1.
[Conditions to become 0]
• The program writes 0.
• The DMA request is accepted, and data transfer starts.

(2) If the DMA request source is edge-type detection for a peripheral function
(The DCTG bits are 0000 0001 to 1010 1001, and the STRG bits are 10.)
[Condition to become 1]
• A DMA request is generated in a peripheral function.
[Conditions to become 0]
• The program writes 0.
• The DMA request is accepted, and data transfer starts.

(3) If the DMA request source is level-type detection for a peripheral function
(The DCTG bits are 1010 1100 to 1100 0110, and the STRG bits are 11.)
[Condition to become 1]
• A DMA request is generated in a peripheral function.
[Condition to become 0]
• A DMA request generated in a peripheral function disappears.

ECLR Bit

This bit is used to control the DEN bit when DMA transfer is completed. Setting the ECLR bit to 1 sets the DEN bit to 0 when DMA transfer is completed, and the subsequent DMA transfer on the channel is not performed. When not using the reload function, set the ECLR bit to 1 so that the DEN bit should be set to 0. To set the ECLR bit, write a value when the DMAC is stopped, or when DMA transfer is disabled and data is not being transferred on the corresponding channel.

DSCLR Bit

This bit is used to initialize the DMAC internal state. By setting the DSCLR bit to 1 after suspending DMA transfer, the remaining DMA transfer is aborted and the transfer status in the internal DMAC is initialized. Note that each register is not initialized at this time. As the written value of 1 is not retained, the read value is always 0. Writing 0 is invalid. To set the DSCLR bit, write data when the DMAC is stopped, or when DMA transfer is disabled and data is not being transferred on the corresponding channel.

12.2.4 DMAi Source Address Register (DMiSA) (i = 0 to 7)

Address: DM0SA: H'FF46 0000, DM1SA: H'FF46 0010, DM2SA: H'FF46 0020, DM3SA: H'FF46 0030, DM4SA: H'FF46 0040, DM5SA: H'FF46 0050, DM6SA: H'FF46 0060, DM7SA: H'FF46 0070

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
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
After Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Bit	Description	R/W
b31 to b0	Sets the start address of the transfer source. H'0000 0000 to H'FFFF FFFF (4 Gbytes)	R/W

The DMiSA register is used to set the start address of the transfer source. To set the DMiSA register, write data when the DMAC is stopped, or when the DMA transfer is disabled and data is not being transferred on the corresponding channel. Access the DMiSA register in 32-bit units.

Set the DMiSA register to an address of a multiple of 2 when the bit length is 16 bits, and an address of a multiple of 4 when the bit length is 32 bits, so b31 to b0 should be set to A31 to A0.

The value set in the DMiSA register is transferred to the working register in the DMAC core when DMA transfer is started, and the value of the working register is returned when a single operand transfer or DMA transfer is completed. However, if the SAD bits in the DMiMD register are set to B'011 (rotation), the DMiSA register value does not change to the working register value, and remains the value set when DMA transfer is started. If the SRLOD bit in the DMiCR0 register is set to 1 (source address reload function used), the value of the DMiSAR register is reloaded when DMA transfer is completed.

12.2.5 DMAi Destination Address Register (DMiDA) (i = 0 to 7)

Address: DM0DA: H'FF46 0004, DM1DA: H'FF46 0014, DM2DA: H'FF46 0024, DM3DA: H'FF46 0034, DM4DA: H'FF46 0044, DM5DA: H'FF46 0054, DM6DA: H'FF46 0064, DM7DA: H'FF46 0074

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
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
After Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Bit	Description	R/W
b31 to b0	Sets the start address of the transfer destination. H'0000 0000 to H'FFFF FFFF (4 Gbytes)	R/W

The DMiDA register is used to set the start address of the transfer destination. To set the DMiDA register, write data when the DMAC is stopped, or when the DMA transfer is disabled and data is not being transferred on the corresponding channel. Access the DMiDA register in 32-bit units.

Set the DMiDA register to an address of a multiple of 2 when the bit length is 16 bits, and an address of a multiple of 4 when the bit length is 32 bits, so b31 to b0 should be set to A31 to A0.

The value set in the DMiDA register is transferred to the working register in the DMAC core when DMA transfer is started, and the value of the working register is returned when a single operand transfer or DMA transfer is completed. However, if the DAD bits in the DMiMD register are set to B'011 (rotation), the DMiDA register value does not change to the working register value, and remains the value set when DMA transfer is started. If the DRLOD bit in the DMiCR0 register is set to 1 (destination address reload function used), the value of DMiDAR register is reloaded when DMA transfer is completed.

12.2.6 DMAi Transfer Byte Counter (DMiCNT) (i = 0 to 7)

Address: DM0CNT: H'FF46 0008, DM1CNT: H'FF46 0018, DM2CNT: H'FF46 0028, DM3CNT: H'FF46 0038, DM4CNT: H'FF46 0048, DM5CNT: H'FF46 0058, DM6CNT: H'FF46 0068, DM7CNT: H'FF46 0078

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—										
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
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
b31 to b26	—	Reserved	These bits are read as 0. The write value should be 0.	R
b25 to b0	Sets the number of DMA transfer bytes H'0000 000 to H'3FFF FFF			R/W

The DMiCNT counter is used to set the number of bytes for DMA transfer. To set the DMiCNT counter, write data when the DMAC is stopped, or when the DMA transfer is disabled and data is not being transferred on the corresponding channel. Access the DMiCNT counter in 32-bit units.

Set the DMiCNT counter to a multiple of 2 when the bit length is 16 bits, and a multiple of 4 when the bit length is 32 bits. If H'0000 000 is set, the number of transfer bytes will be set to 64 Mbytes.

The value set in the DMiCNT counter is transferred to the working register in the DMAC core when DMA transfer starts. The value of the working register is decremented by the number of transferred bytes (by 1 when the data length is 8 bits, 2 when 16 bits, and 4 when 32 bits) per single data transfer. When the value reaches H'0000 000, DMA transfer is completed. The value of the working register is returned to the DMiCNT counter when the transfer channel is switched or DMA transfer is completed. However, if the BRLOD bit in the DMiCR0 register is set to 1 (transfer byte count reload function used), the value of the DMiCNTR register is reloaded when DMA transfer is completed.

12.2.7 DMAi Source Address Reload Register (DMiSAR) (i = 0 to 7)

Address: DM0SAR: H'FF46 0200, DM1SAR: H'FF46 0210, DM2SAR: H'FF46 0220, DM3SAR: H'FF46 0230,
DM4SAR: H'FF46 0240, DM5SAR: H'FF46 0250, DM6SAR: H'FF46 0260, DM7SAR: H'FF46 0270

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
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
After Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Bit	Description	R/W
b31 to b0	Sets an address to be reloaded to the DMiSA register. H'0000 0000 to H'FFFF FFFF (4 Gbytes)	R/W

The DMiSAR register is used to set an address to be reloaded to the DMiSA register. Access the DMiSAR register in 32-bit units.

When the SRLOD bit in the DMiCR0 register is set to 1, the value of the DMiSAR register is reloaded to the DMiSA register when DMA transfer is completed. Set the DMiSAR register to an address of a multiple of 2 when the bit length is 16 bits, and an address of a multiple of 4 when the bit length is 32 bits, so b31 to b0 should be set to A31 to A0.

12.2.8 DMAi Destination Address Reload Register (DMiDAR) (i = 0 to 7)

Address: DM0DAR: H'FF46 0204, DM1DAR: H'FF46 0214, DM2DAR: H'FF46 0224, DM3DAR: H'FF46 0234,
DM4DAR: H'FF46 0244, DM5DAR: H'FF46 0254, DM6DAR: H'FF46 0264, DM7DAR: H'FF46 0274

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
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
After Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Bit	Description	R/W
b31 to b0	Sets an address to be reloaded to the DMiDA register. H'0000 0000 to H'FFFF FFFF (4 Gbytes)	R/W

The DMiDAR register is used to set an address to be reloaded to the DMiDA register. Access the DMiDAR register in 32-bit units.

When the DRLOD bit in the DMiCR0 register is set to 1, the value of the DMiDAR register is reloaded to the DMiDA register when DMA transfer is completed. Set the DMiDAR register to an address of a multiple of 2 when the bit length is 16 bits, and an address of a multiple of 4 when the bit length is 32 bits, so b31 to b0 should be set to A31 to A0.

12.2.9 DMAi Transfer Byte Counter Reload Register (DMiCNTR) (i = 0 to 7)

Address: DM0CNTR: H'FF46 0208, DM1CNTR: H'FF46 0218, DM2CNTR: H'FF46 0228, DM3CNTR: H'FF46 0238, DM4CNTR: H'FF46 0248, DM5CNTR: H'FF46 0258, DM6CNTR: H'FF46 0268, DM7CNTR: H'FF46 0278

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—										
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
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
b31 to b26	—	Reserved	These bits are read as 0. The write value should be 0.	R
b25 to b0	Sets the number of DMA transfer bytes to be reloaded to the DMiCNT counter. H'0000 000 to H'3FFF FFF			R/W

The DMiCNTR register is used to set the number of DMA transfer bytes to be reloaded to the DMiCNT counter. Access the DMiCNTR register in 32-bit units.

When the BRLOD bit in the DMiCR0 register is set to 1, the value of the DMiCNTR register is reloaded to the DMiCNT counter when DMA transfer is completed. Set the DMiCNTR register to a multiple of 2 when the bit length is 16 bits, and a multiple of 4 when the bit length is 32 bits. If H'0000 000 is set, the number of transfer bytes will be set to 64 Mbytes.

12.2.10 DMA Interrupt Control Register (DMICR)

Address H'FF46 0508

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	DINTM0	DINTM1	DINTM2	DINTM3	DINTM4	DINTM5	DINTM6	DINTM7	—	—	—	—	—	—	—	—
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	DINTM0	DMA0 Interrupt Enable Bit	0: Interrupt disabled 1: Interrupt enabled	R/W
b14	DINTM1	DMA1 Interrupt Enable Bit		R/W
b13	DINTM2	DMA2 Interrupt Enable Bit		R/W
b12	DINTM3	DMA3 Interrupt Enable Bit		R/W
b11	DINTM4	DMA4 Interrupt Enable Bit		R/W
b10	DINTM5	DMA5 Interrupt Enable Bit		R/W
b9	DINTM6	DMA6 Interrupt Enable Bit		R/W
b8	DINTM7	DMA7 Interrupt Enable Bit		R/W
b7 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

DINTMi Bit

When the DINTMi bit is set to 1, a DMAi interrupt request is generated when the DMA transfer on DMAi is completed. When this bit is set to 0, no interrupt request is generated.

12.2.11 DMA Start Register (DMSTR)

Address H'FF46 0501

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	DMST
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R
b0	DMST	DMAC Module Start Bit	0: DMAC stops 1: DMAC operates	R/W

DMST Bit

When the DMST bit is set to 1, the DMAC module starts.

When the DMST bit is set to 0 during a transfer with the operand transfer method, all channels suspend DMA transfer after the current single operand transfer is completed. If the DMST bit is then set to 1, DMA transfer is restarted.

During non-stop transfer, DMA transfer is not suspended even if the DMST bit is set to 0, and transfer continues until DMA transfer is completed.

12.2.12 DMA Transfer Status Register (DMSR)

Address H'FF46 0518

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	DASTS0	DASTS1	DASTS2	DASTS3	DASTS4	DASTS5	DASTS6	DASTS7	—	—	—	—	—	—	—	—
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	DASTS0	DMA0 Arbitration Status Flag	When read 0: Data transfer is not in progress 1: Data transfer is in progress (single operand transfer or non-stop transfer is in progress) When written 0: Invalid 1: The DASTSi flag (i = 0 to 7) is set to 0.	R/W
b14	DASTS1	DMA1 Arbitration Status Flag		R/W
b13	DASTS2	DMA2 Arbitration Status Flag		R/W
b12	DASTS3	DMA3 Arbitration Status Flag		R/W
b11	DASTS4	DMA4 Arbitration Status Flag		R/W
b10	DASTS5	DMA5 Arbitration Status Flag		R/W
b9	DASTS6	DMA6 Arbitration Status Flag		R/W
b8	DASTS7	DMA7 Arbitration Status Flag		R/W
b7 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

DASTSi Bit

When DMAi data transfer (single operand transfer or non-stop transfer) is started, the DASTSi flag becomes 1. When data transfer is completed, this flag becomes 0.

The DASTSi flag is set to 0 by writing 1 by a program. The written value of 1 is not retained at this time. Writing 0 is invalid.

12.2.13 DMA Transfer Completion Status Register (DMCSR)

Address H'FF46 0514

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	DEDET0	DEDET1	DEDET2	DEDET3	DEDET4	DEDET5	DEDET6	DEDET7	—	—	—	—	—	—	—	—
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	DEDET0	DMA0 DMA Transfer Completion Detect Flag	When read 0: Not detected 1: Detected When written 0: Invalid 1: The DEDETi flag (i = 0 to 7) is set to 0.	R/W
b14	DEDET1	DMA1 DMA Transfer Completion Detect Flag		R/W
b13	DEDET2	DMA2 DMA Transfer Completion Detect Flag		R/W
b12	DEDET3	DMA3 DMA Transfer Completion Detect Flag		R/W
b11	DEDET4	DMA4 DMA Transfer Completion Detect Flag		R/W
b10	DEDET5	DMA5 DMA Transfer Completion Detect Flag		R/W
b9	DEDET6	DMA6 DMA Transfer Completion Detect Flag		R/W
b8	DEDET7	DMA7 DMA Transfer Completion Detect Flag		R/W
b7 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

DEDETi Bit

If DMA transfer on DMAi is completed, the DEDETi flag becomes 1. Once this flag has become 1, it does not automatically become 0. To set the DEDETi flag to 0, write 1 by a program. The written value of 1 is not retained at this time. Writing 0 is invalid.

To use a DMA interrupt, write 1 to the DEDETi flag of the channel where an interrupt request occurred in the interrupt routine.

12.3 Operations

12.3.1 Transfer Modes

Two transfer modes with different bus access methods are available. The internal main bus consists of the following. If targets are different, the bus access from the master can be operated simultaneously in parallel.

- Internal main bus master: CPU or DMAC source, or DMAC destination
- Internal main bus targets: On-chip SRAM, on-chip flash ROM, and peripheral functions

12.3.1.1 Cycle-Stealing Transfer Mode

Transfer is performed leaving at least one cycle between the read and write accesses of one data. During this interval, the CPU can access the target that the DMAC currently accesses.

12.3.1.2 Pipelined Transfer Mode

Bus accesses are consecutively performed. The CPU cannot access the target that the DMAC currently accesses until the single operand transfer is completed. Pipelined transfer through the same target is not possible.

Figure 12.2 shows an Example of Bus Mastership Alternation between DMAC and CPU in Transfer Modes.

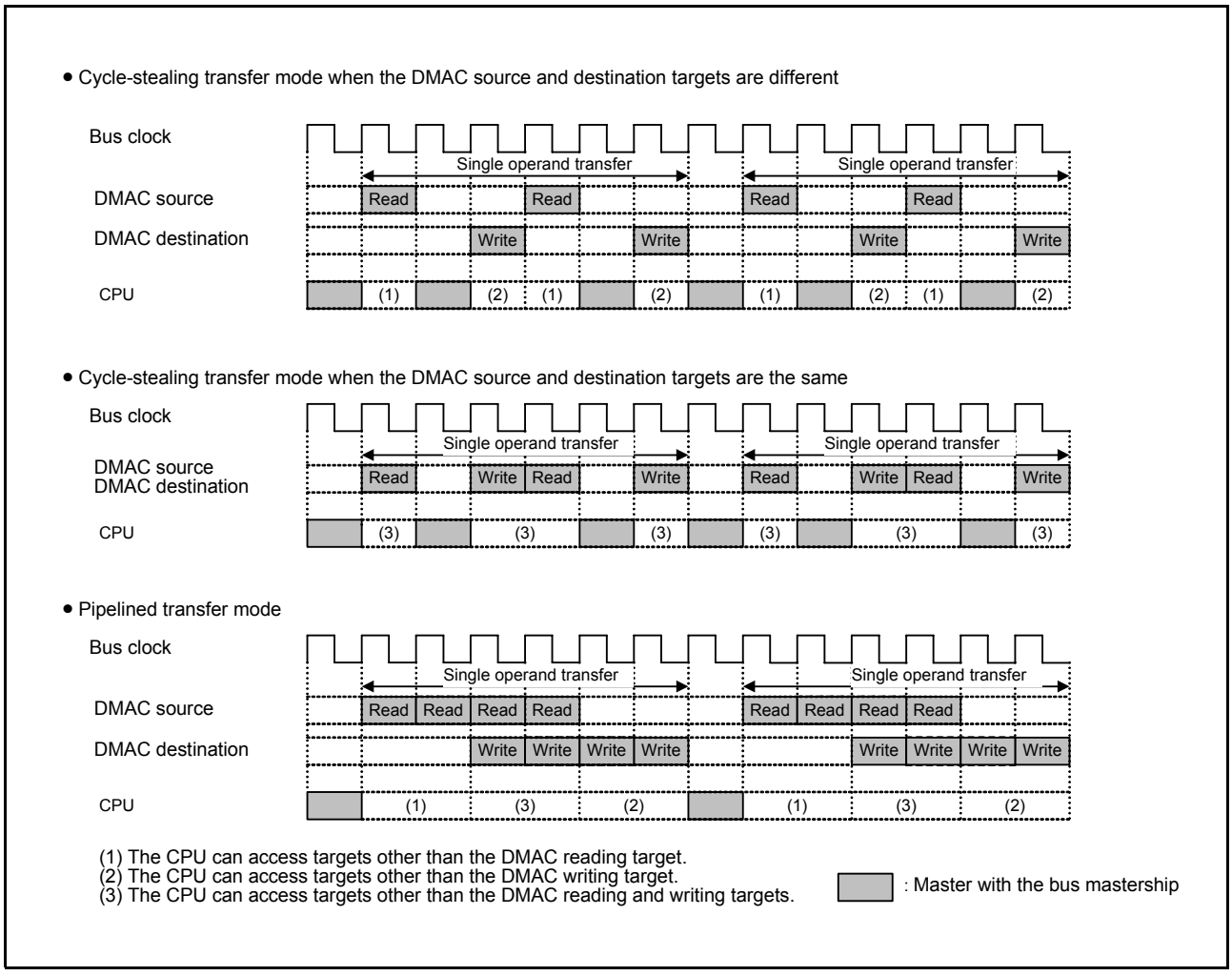


Figure 12.2 Example of Bus Mastership Alternation between DMAC and CPU in Transfer Modes

12.3.2 Transfer Methods

Operand transfer and non-stop transfer are available as transfer methods, and unit operand transfer and sequential operand transfer are available as operand transfer methods. In unit operand transfer, only one operand is transferred per DMA request. In sequential operand transfer, one operand is repeatedly transferred per DMA request until DMA transfer is completed. In non-stop transfer, data is continuously transferred per DMA request until DMA transfer is completed.

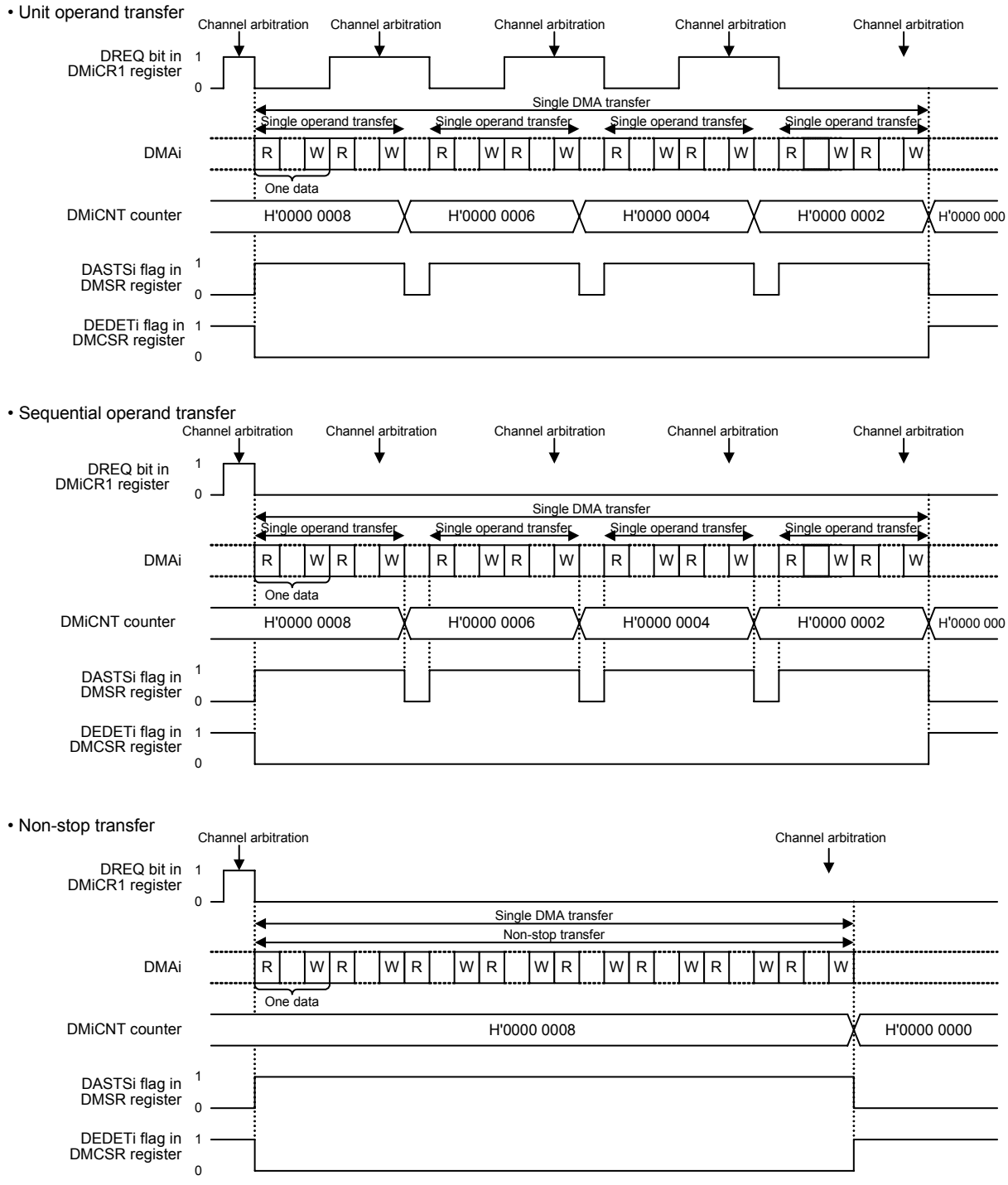
Regardless of transfer methods, a single DMA transfer is completed when the number of bytes of data set in the DMiCNT counter is transferred and the DMiCNT counter reaches H'0000 000.
Table 12.7 lists the Transfer Methods.

Table 12.7 Transfer Methods

DSEL Bits in DMiCR0 Register (i = 0 to 7)	Transfer Method	Number of Transfer Bytes per DMA Request
B'00 (Unit operand transfer)	• Data of one operand is transferred after DMA transfer is started. • One operand is transferred per DMA request until DMA transfer is completed. • Channel arbitration is performed after a single operand transfer is completed. • A DMA request is required for each completion of a single operand transfer until DMA transfer is completed.	The number of data units of one operand × the number of bytes corresponding to the bit length
B'01 (Sequential operand transfer)	• Data of one operand is transferred after DMA transfer is started. • One operand is repeatedly transferred per DMA request until DMA transfer is completed. • Channel arbitration is performed each time a single operand transfer is completed. • Only the first DMA request is required.	The number of bytes of data set in the DMiCNT counter
B'11 (Non-stop transfer)	• Data is continuously transferred after DMA transfer is started. • Transfer is continuously performed per DMA request until DMA transfer is completed. • Channel arbitration is not performed until DMA transfer is completed. • Only the first DMA request is required.	The number of bytes of data set in the DMiCNT counter

When the operand transfer method is used, the request is executed if there is a DMA request from a higher-priority channel through the channel arbitration when single operand transfer is completed. If there is no DMA request, the next operand is transferred continuously. In unit operand transfer, however, the next operand is not transferred if there is no DMA request.

When the non-stop transfer method is used, DMA requests from a higher-priority channel are not accepted during DMA transfer because data is continuously transferred when DMA transfer is started until it is completed.
Figure 12.3 shows Transfer Examples According to Transfer Methods.



The above applies under the following conditions:
- In the DMiMD register: The SZSEL bits = B'000 (8 bits) and the OPSEL bits = B'0001 (2 data units).
- In the DMiCR0 register:
  The BRLOD bit = 0 (transfer byte count reload function not used)
  The MDSEL bits = B'00 (cycle-stealing transfer mode)
- DMiCNT counter = H'0000 0008 (8 bytes)
- No DMA request other than for one DMAi channel

Figure 12.3 Transfer Examples According to Transfer Methods

12.3.3 DMAC Activation

Figure 12.4 shows the Register Setting Procedure.

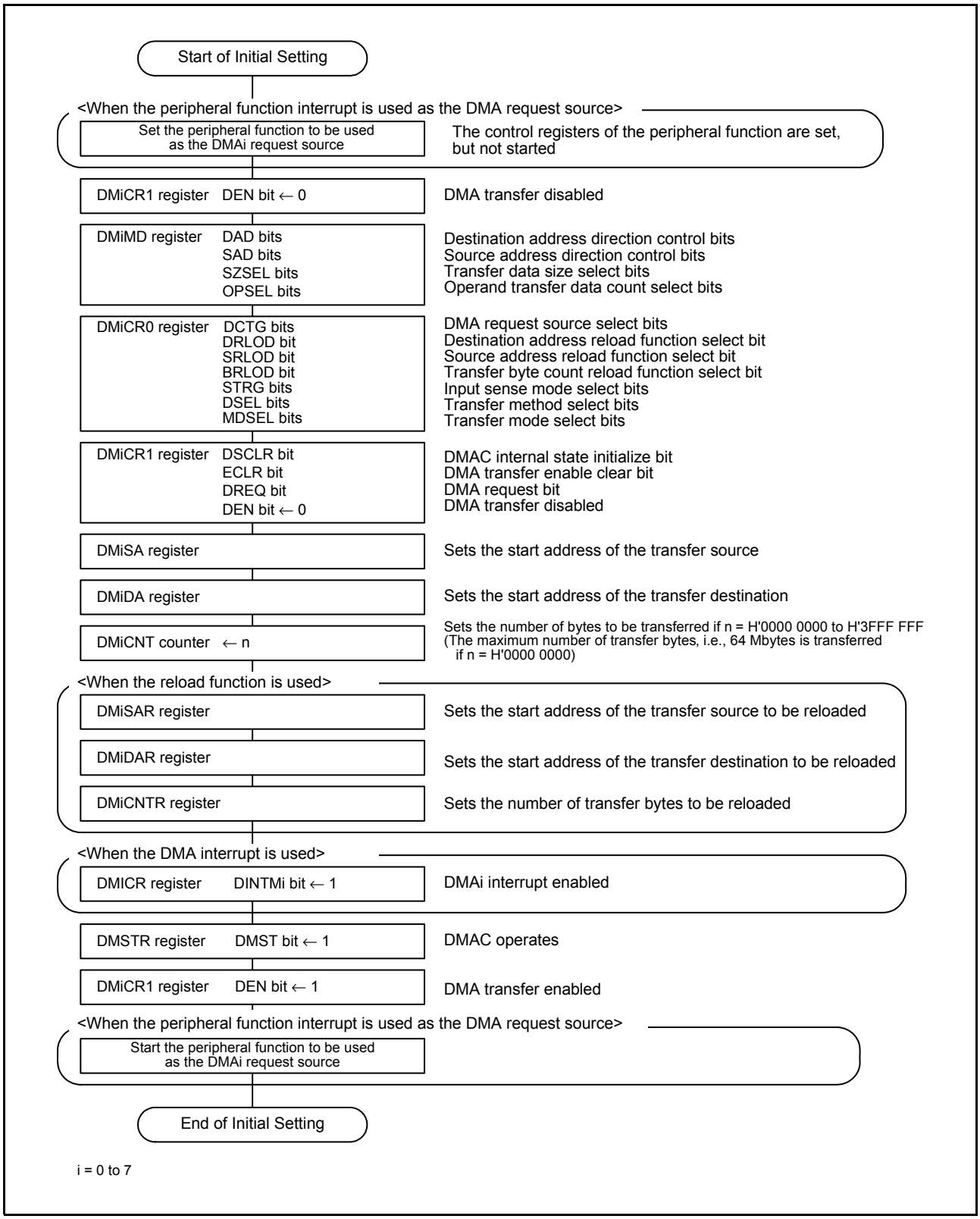


Figure 12.4 Register Setting Procedure

12.3.4 Starting DMA Transfer

When the DEN bit in the DMiCR1 register (i = 0 to 7) is set to 1 (DMA transfer enabled) and the DMST bit in the DMSTR register is set to 1 (DMAC operates), DMA transfer on DMAi is enabled. If multiple DMA transfer requests are generated, channel arbitration is performed, the DMA request corresponding to the highest-priority channel is accepted, and DMA transfer on the channel is started. When DMA transfer is started, the DASTSi flag in the DMSR register becomes 1 (data transfer is in progress).

12.3.5 Completing DMA Transfer

When the DMiCNT counter reaches H'0000 0000, DMA transfer on DMAi is completed, and the following processes are performed:

- The DEDETi flag in the DMCSR register becomes 1 (DMA transfer completion detected)
- If the DINTMi bit in the DMICR register is 1 (interrupt enabled), a DMAi interrupt request is generated.
- If the ECLR bit in the DMiCR1 register is 1, the DEN bit is set to 0 (DMA transfer disabled), and DMA transfer on DMAi is not subsequently performed.
- When the reload function is used, the value of the reload register is reloaded to the current register.

12.3.6 Suspending, Restarting, and Stopping DMA Transfer

During a DMA transfer using the operand transfer method, DMA transfer on all channels is suspended when the DMST bit is set to 0 (DMAC stops).
When the DEN bit is set to 0 (DMA transfer disabled), DMA transfer on the corresponding channel is suspended. If data is being transferred at this time, DMA transfer is suspended after the current single operand transfer is completed.

During non-stop transfer, DMA transfer is not suspended even if the DMST or DEN bit is set to 0, and transfer continues until DMA transfer is completed.
On the suspended channel, DMA transfer can be restarted by setting the DMST or DEN bit to 1.
By writing 1 to the DSCLR bit in the DMiCR1 register while each channel is suspended, DMA transfer is stopped and the DMAC internal state is initialized. However, only the transfer status of the DMAC internal circuit is initialized; each register is not initialized.

12.3.7 DMA Request Sources

DMA request sources are selected from software triggers and peripheral function interrupts.

12.3.7.1 Software Trigger

When a software trigger is selected as the DMA request source, a DMA request is generated by writing 1 (DMA request is present) to the DREQ bit in the DMiCR1 register (i = 0 to 7) by a program. The DREQ bit can be set to 1 regardless of the DMA transfer state. When setting the DREQ bit to 0 (DMA request is not present), write the value only when the DMAC is stopped, or when DMA transfer is disabled and data is not being transferred on the corresponding channel.

12.3.7.2 Peripheral Function Interrupts

When an interrupt request from the peripheral function is selected as the DMA transfer source, a DMA request is generated if an interrupt request is generated from the selected peripheral function. However, the DMAC is not affected by the IMASK, IPR, or interrupt control register, so DMA transfer is performed even if acceptance of interrupt requests is disabled. The IR bit in the interrupt request register is set to 1 if a DMA request is generated by an interrupt request from the peripheral function, but the IR bit is not set to 0 even if DMA transfer is performed.

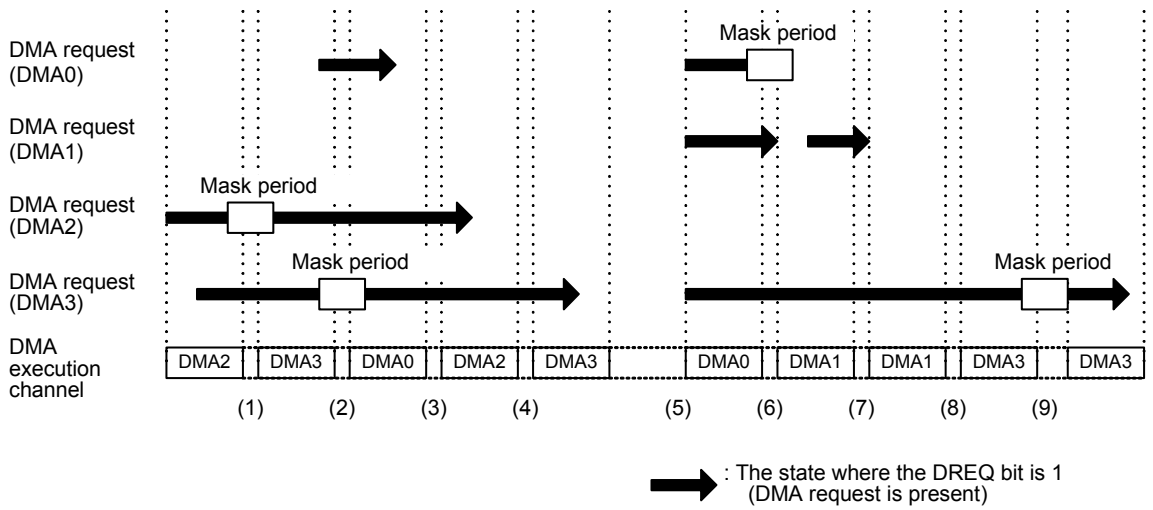
12.3.8 Channel Arbitration

If there are multiple DMA requests, the DMAC determines the priorities of the channels where DMA requests occurred. The priorities of channels are fixed in the following order:

DMA0 > DMA1 > ... > DMA6 > DMA7

If a DMA request occurs during data transfer, channel arbitration is started when write access to the last data is started. If a DMA request with higher channel priority occurs during data transfer, therefore, transfer on the higher-priority channel is started after the data transfer is completed.

Figure 12.5 shows an Example of Channel Arbitration for Multiple DMA Requests.



(1) Transfer on DMA3 is started because DMA requests are assumed to be non-existent in DMA2 because DMA2 is during the mask period.
(2) Transfer on DMA0 is started because this channel has the highest channel priority.
(3) Transfer on DMA2 is started because this channel has the highest channel priority at this point.
(4) Transfer on DMA3 is started because there are no other requests at this point.
(5) If DMA requests for DMA0, DMA1, and DMA3 occur at the same time, transfer on DMA0 is started because DMA0 has the highest priority.
(6) When transfer on DMA0 ends, transfer on DMA1 is started because DMA1 has the second highest priority.
(7) If another DMA request for DMA1 occurs during DMA transfer on DMA1, DMA transfer on DMA1 is started again after the current DMA transfer on DMA1 ends. If an edge sense is set, no mask period exists.
(8) When DMA transfer on DMA1 ends, DMA transfer on DMA3 is started because there is no other transfer request.
(9) When DMA3 is during the mask period, DMA transfer is not started because there is no other transfer request. DMA transfer on DMA3 is started after the mask period ends.

Figure 12.5 Example of Channel Arbitration for Multiple DMA Requests

12.3.9 Reload Function

The reload function reloads the values of the reload registers (DMiSAR, DMiDAR, and DMiCNTR) (i = 0 to 7) to the current registers (DMiSA, DMiDA, and DMiCNT) when DMA transfer is completed. This function can be used for the source address, destination address, or transfer byte count.

The reload function provides continuous transfer to dispersed areas. This enables to continuously transfer several transfer blocks in different transfer areas and with a different number of bytes through the same channel. Writing values to the reload registers before transfer is completed enables to prepare the next transfer without affecting the current register during DMA transfer.

When using the reload function, set data both in the reload register and current register. The reload register must be set before start of the last data transfer (completion of DMA transfer). If the reload register is set after start of the last operand transfer, this setting may not been reflected when data is reloaded after DMA transfer is completed. When not using the reload function, set the ECLR bit in the DMiCR1 register to 1 to clear the DEN bit.

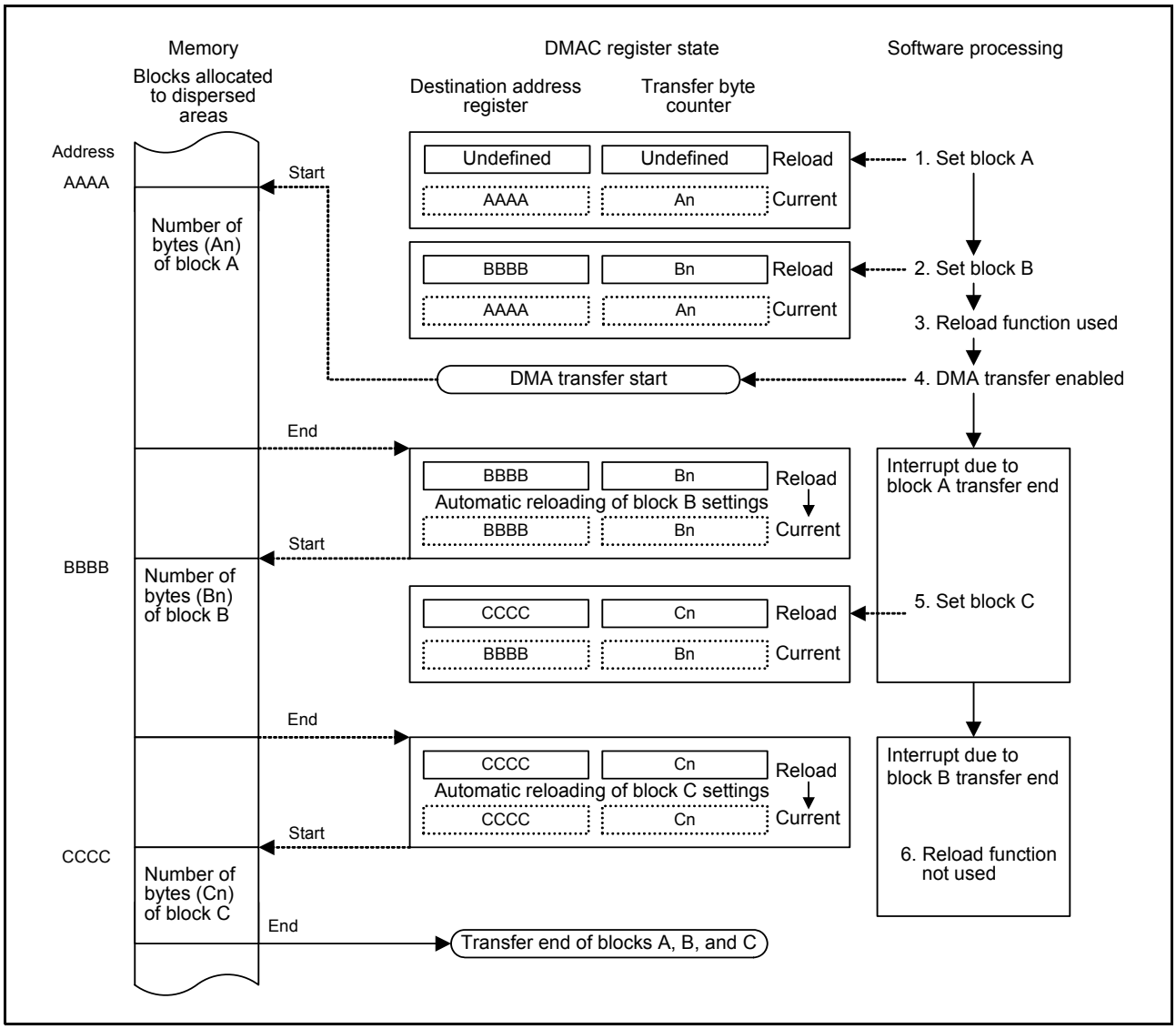


Figure 12.6 Example of Transfer by Using Reload Function

12.3.10 Data Relocation

The bus width of the data bus is 32 bits. Access is thus performed in 4-byte units. If the bit length of transfer data is set to 8 bits or 16 bits, data is relocated at the 4n boundary according to the physical data width and addresses.
Figure 12.7 shows an Example of Data Relocation.

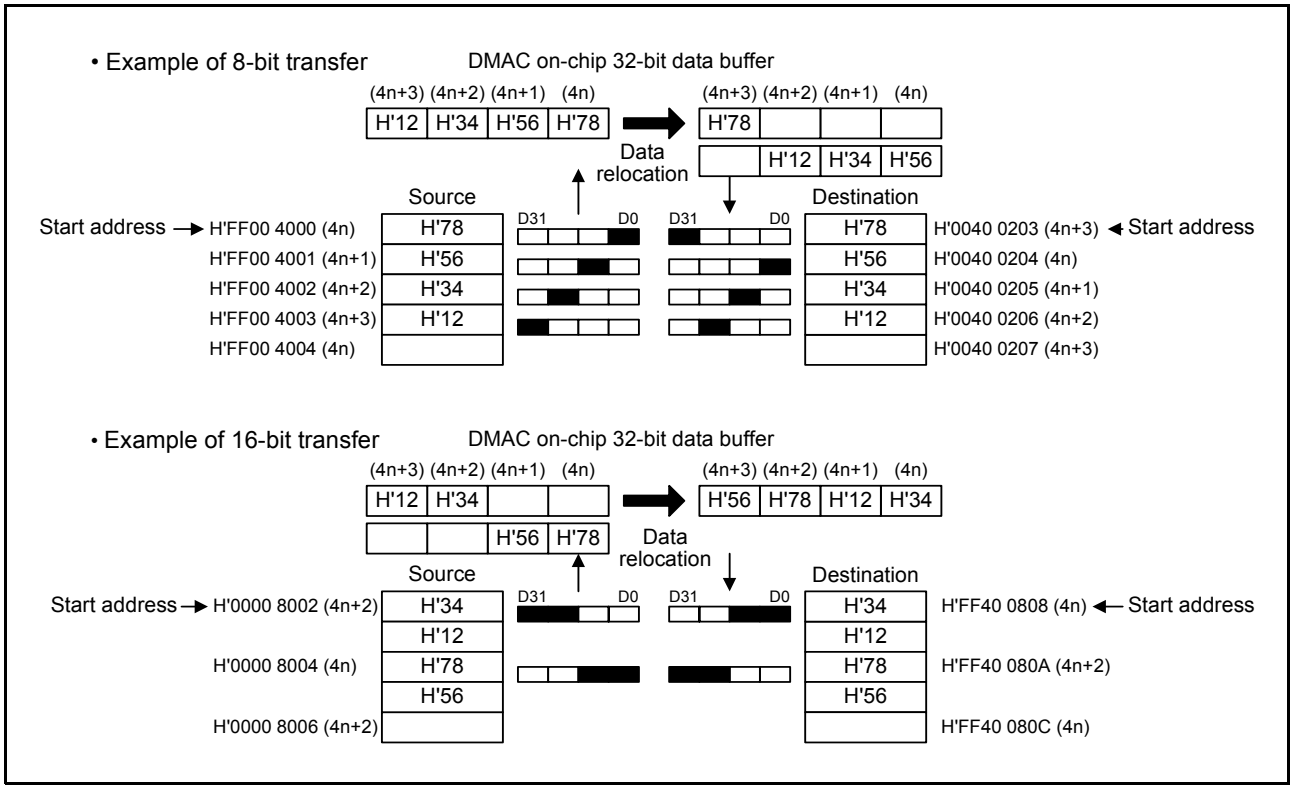


Figure 12.7 Example of Data Relocation

12.3.11 Rotation

When rotation is selected by the SAD or DAD bits in the DMiMD register (i = 0 to 7), the address is incremented during data transfer. If data transfer is completed, the value set when DMA transfer is started is returned to the address register. Figure 12.8 shows an Example of Transfer Using Rotation.

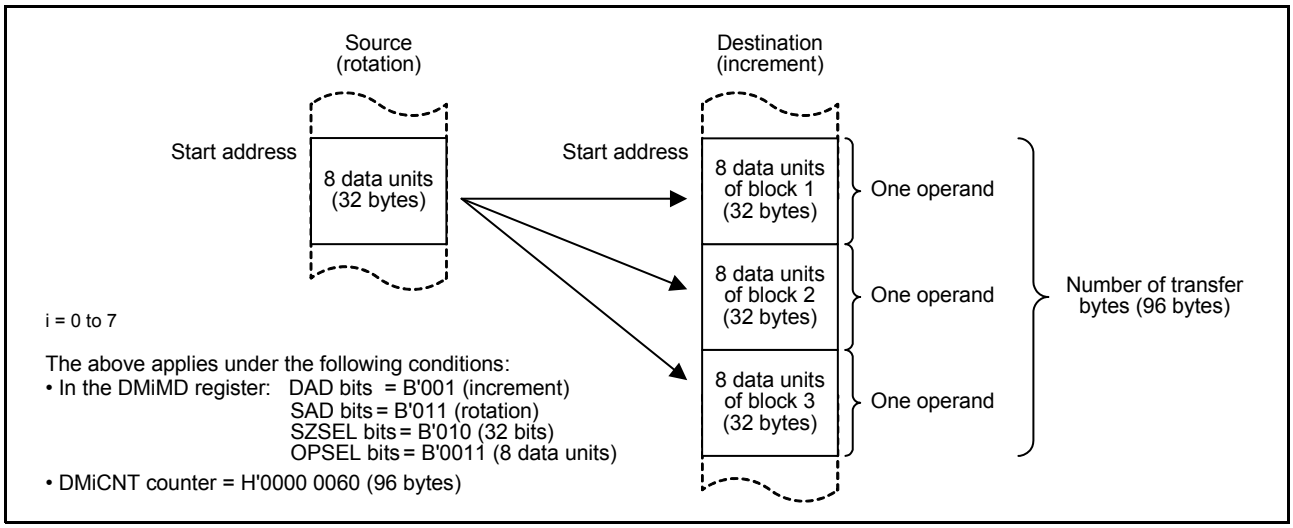


Figure 12.8 Example of Transfer Using Rotation

12.4 Interrupts

When the DINTMi bit in the DMICR register (i = 0 to 7) is set to 1 (interrupt enabled), a DMAi interrupt request is generated if DMA transfer on DMAi is completed. To use a DMA interrupt, write 1 to the DEDETi flag in the DMCSR register of the channel where an interrupt request occurred in the interrupt routine. This sets the DEDETi flag to 0.

12.5 Notes on DMAC

12.5.1 Register Setting

- To set the following bits or registers, make settings when DMAC is stopped (the DMST bit in the DMSTR register is 0), or when DMA transfer is disabled (the DEN bit in the DMiCR1 register is 0) and data is not being transferred (the DASTSi flag in the DMSR register is 0) on the corresponding channel.
  - Registers DMiMD, DMiCR1, DMiSA, DMiDA, and DMiCNT
  - Bits DCTG, STRG, DSEL, and MDSEL in the DMiCR0 register
  - The DSCLR bit in the DMiCR1 register
  - Write 0 to the DREQ bit in the DMiCR1 register
    (When writing 1 to the DREQ bit, note that the value can be written regardless of the DMA transfer state).
- Access the following registers in 32-bit units.
  Registers DMiMD, DMiSA, DMiDA, DMiCNT, DMiSAR, DMiDAR, and DMiCNTR
- Write to the ECLR bit in the DMiCR1 register when the DASTSi flag is 0. When not using the reload function, set the ECLR bit to 1 to clear the DEN bit.
- When the DCTG or STRG bits are set, make sure bits DMST and DEN are set to 1 after setting the DREQ bit of the specified channel to 0.
- The DREQ bit varies with whether a DMA request is currently present regardless of the settings of bits DMST and DEN. If the DMA request source is other than the software trigger, do not write 1 to the DREQ bit by a program.
- For addresses and the number of bytes, set the aligned values to the registers according to the bit length.
  Table 12.8 lists the Alignment and Setting Values of Lower 2 Bits According to Bit Length.

Table 12.8 Alignment and Setting Values of Lower 2 Bits According to Bit Length

SZSEL Bits in DMiMD Register	Alignment	Address Register		Byte Counter	
		b1	b0	b1	b0
B'000 (8 bits)	Integral multiple	—	—	—	—
B'001 (16 bits)	Multiple of 2	—	0	—	0
B'010 (32 bits)	Multiple of 4	0	0	0	0

[Legend] —: Can be set to 0 or 1

13. I/O Ports

13.1 Introduction

I/O ports consist of 80 pins (SH72A2 Group) and 46 pins (SH72A0 Group) , designated as A to G, J to L, and N. Each port can be used as input or output by setting the respective port direction registers (PN00 to PN05 are input-only ports). Each port also functions as a multiplexed pin shared with on-chip peripheral module signals. Each multiplexed pin function is selected by setting the corresponding function select register in each port.

Other port functions including input threshold value switching function are also incorporated.

Table 13.1. lists the I/O Port Specifications. Figures 13.1 and 13.2 are circuit diagrams of I/O ports.

Table 13.1 I/O Port Specifications

Item	Description	
Number of ports	SH72A2 Group (total 80 pins) Port A : PA00 to PA05 (6 pins) Port B : PB12 to PB15 (4 pins) Port C : PC08 to PC12 (5 pins) Port D : PD00 to PD10 (11 pins) Port E : PE00 to PE07 (8 pins) Port F : PF00 to PF02, PF07 (4 pins) Port G : PG00 to PG03, PG10 to PG13 (8 pins) Port J : PJ00 to PJ11 (12 pins) Port K : PK00, PK01, PK08 to PK15 (10 pins) Port L : PL10 to PL15 (6 pins) Port N : PN00 to PN05 (6 pins) (input only)	SH72A0 Group (total 46 pins) Port A : PA00, PA01, PA03 to PA05 (5 pins) Port C : PC08, PC09 (2 pins) Port D : PD00 to PD07 (8 pins) Port E : PE00 to PE07 (8 pins) Port G : PG00 to PG03, PG10 to PG12 (7 pins) Port J : PJ04 to PJ05, PJ08 to PJ09 (4 pins) Port K : PK00, PK01, PK08, PK12, PK14, PK15 (6 pins) Port N : PN00 to PN05 (6 pins) (input only)
Port function	Input or output can be set in port units using the port direction register (other than port N).	
Internal pull-up resistors	Internal pull-up resistors can be set*2 in port-pin units.*1	
Input threshold value switching function	The input threshold value can be specified to three voltage levels (0.5 VCC, 0.70 VCC, or TTL level) in port-pin units.*1 To use the pins as input pins for the peripheral functions, this function should also be set. The voltage for ports K and L is AVCC1. The voltage for port N is AVCC0.	

Notes: 1. This is for the applicable pins from among pins Pn00 to Pn07 and Pn08 to Pn15 (n: A, B, C, D, E, F, G, J, K, L, N)
2. Internal pull-up resistor settings are invalid for pins in use as analog inputs.

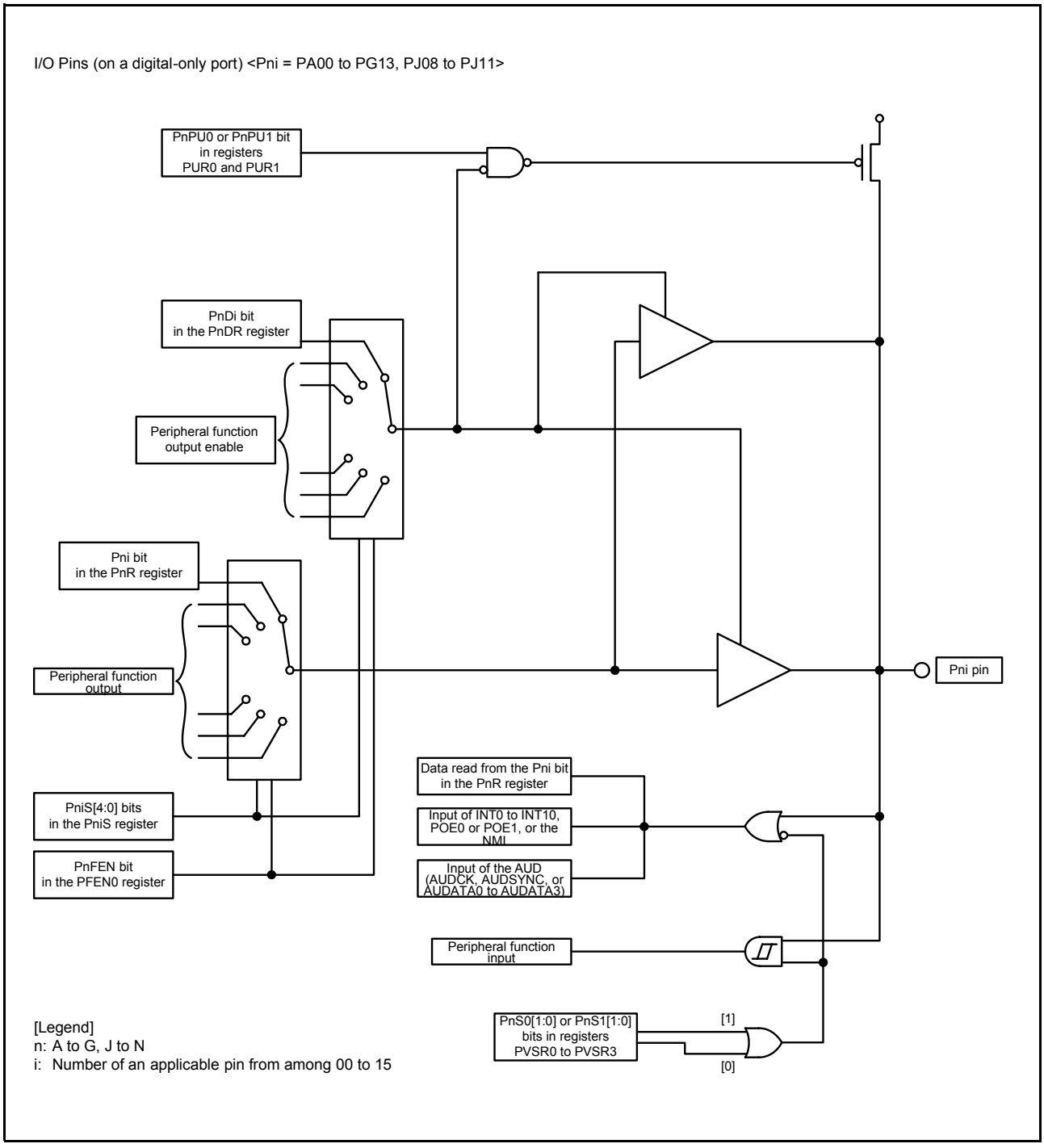


Figure 13.1 Circuit Diagram of I/O Ports (1)

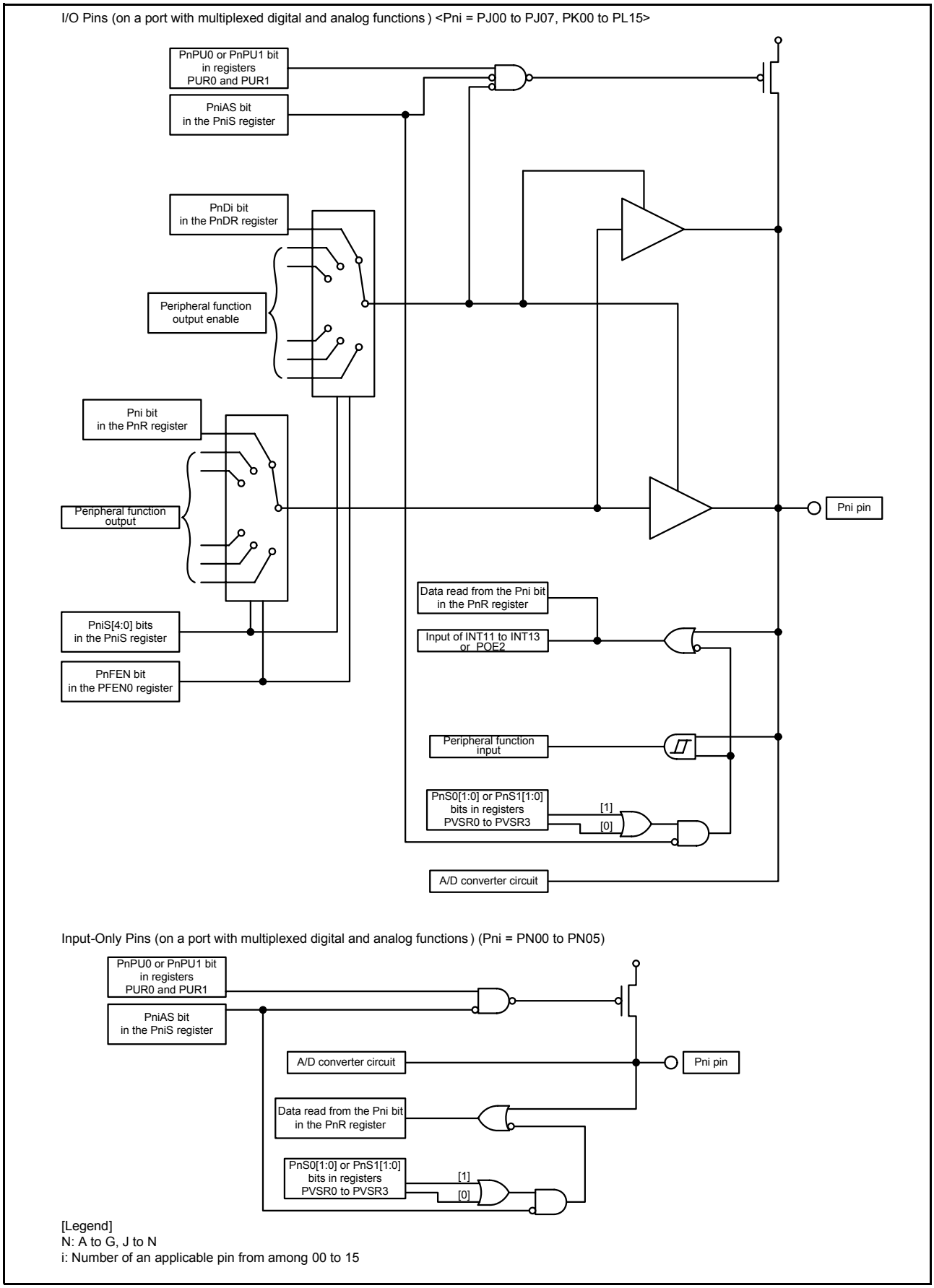


Figure 13.2 Circuit Diagram of I/O Ports (2)

Table 13.2 to Table 13.12 list the multiplexed pin functions used for each port.

Table 13.2 Multiplexed Pin Functions for Port A

Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)	Function 6 (Related Module)	Function 7 (Related Module)	Function 8 (Related Module)	Function 9 (Related Module)	Function 10 (Related Module)	Function 11 (Related Module)	Function 12 (Related Module)	Function 13 (Related Module)
PA00 (Port)	INT2 (INTC)	—	—	—	CTX0 (CAN)	—	—	—	—	—	—	—
PA01 (Port)	INT1 (INTC)	—	—	—	CRX0 (CAN)	—	—	—	—	—	—	—
PA02 * (Port)	—	—	—	—	—	—	—	—	—	—	—	—
PA03 (Port)	—	—	—	—	CTX1 (CAN)	—	—	—	—	—	—	—
PA04 (Port)	INT0 (INTC)	—	—	—	CRX1 (CAN)	—	—	—	—	—	—	—
PA05 (Port)	NMI/ POE0# (INTC)	—	—	—	—	—	—	—	—	—	—	—

Note: * The PA02 pin is not available in the SH72A0 Group.

Table 13.3 Multiplexed Pin Functions for Port B

Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)	Function 6 (Related Module)	Function 7 (Related Module)	Function 8 (Related Module)	Function 9 (Related Module)	Function 10 (Related Module)	Function 11 (Related Module)	Function 12 (Related Module)	Function 13 (Related Module)
PB12 * (Port)	—	—	—	—	—	—	—	—	—	—	—	—
PB13 * (Port)	—	—	—	SCK3 (SCI)	—	—	—	—	—	—	—	—
PB14 * (Port)	—	TP04C (TPU)	—	RXD3 (SCI)	—	LRX2 (LIN)	—	—	—	—	—	—
PB15 * (Port)	—	TP04D (TPU)	—	TXD3 (SCI)	—	LTX2 (LIN)	—	—	—	—	ADTRG#	—

Note: * The port B pins are not available in the SH72A0 Group.

Table 13.4 Multiplexed Pin Functions for Port C

Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)	Function 6 (Related Module)	Function 7 (Related Module)	Function 8 (Related Module)	Function 9 (Related Module)	Function 10 (Related Module)	Function 11 (Related Module)	Function 12 (Related Module)	Function 13 (Related Module)
PC08 (Port)	—	TP04A (TPU)	TIOC1B/ TCLKA (MTU-III)	RXD3 (SCI)	—	—	—	—	—	—	—	—
PC09 (Port)	—	TP04B (TPU)	TCLKB (MTU-III)	TXD3 (SCI)	—	—	—	—	—	—	—	—
PC10 * (Port)	—	—	TIC5W (MTU-III)	SCK3 (SCI)	—	—	—	MOSI3 (SBI)	—	—	—	—
PC11 * (Port)	—	—	TIC5V (MTU-III)	—	—	—	—	RSPCK3 (SBI)	—	—	—	—
PC12 * (Port)	—	—	TIC5U (MTU-III)	—	—	—	—	MISO3 (SBI)	—	—	—	—

Note: * Pins PC10, PC11, and PC12 are not available in the SH72A0 Group.

Table 13.5 Multiplexed Pin Functions for Port D

Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)	Function 6 (Related Module)	Function 7 (Related Module)	Function 8 (Related Module)	Function 9 (Related Module)	Function 10 (Related Module)	Function 11 (Related Module)	Function 12 (Related Module)	Function 13 (Related Module)
PD00 *1 (Port)	INT7 (INTC)	TP03A (TPU)	TIOC3B (MTU-III)	—	—	—	—	SSL21 (SBI)	—	—	—	—
PD01 (Port)	INT6 (INTC)	TP03B (TPU)	TIOC3D (MTU-III)	TXD2 (SCI)	—	—	—	SSL20 (SBI)	—	—	—	—
PD02 (Port)	INT5 (INTC)	TP03C (TPU)	TIOC4A (MTU-III)	RXD2 (SCI)	—	—	—	MOSI2 (SBI)	—	—	—	—
PD03 (Port)	—	TP03D (TPU)	TIOC4C (MTU-III)	SCK2 (SCI)	—	—	—	RSPCK2 (SBI)	—	—	—	—
PD04 (Port)	—	—	TIOC4B (MTU-III)	—	—	LTX2 (LIN)	—	MISO2 (SBI)	—	—	—	—
PD05 (Port)	—	—	TIOC4D (MTU-III)	—	—	LRX2 (LIN)	—	SSL22 (SBI)	—	—	—	—
PD06 (Port)	INT4 (INTC)	—	TIOC3A (MTU-III)	TXD2 (SCI)	—	—	—	SSL23 (SBI)	—	—	—	—
PD07 (Port)	INT3 (INTC)	—	TIOC3C (MTU-III)	RXD2 (SCI)	—	—	—	SSL30 (SBI) *3	—	—	—	UBCTRG# (Debug)
PD08 *2 (Port)	—	—	TIOC2A (MTU-III)	—	—	—	—	SSL31 (SBI)	—	—	—	—
PD09 *2 (Port)	—	—	TIOC2B (MTU-III)	—	—	—	—	SSL32 (SBI)	—	—	—	—
PD10 *2 (Port)	—	—	TIOC1A (MTU-III)	—	—	—	—	SSL33 (SBI)	—	—	—	—

Notes: 1. PD00 is shared with CLKOUT. Setting the COE bit in the COCR register to 1 enables this port pin to be used as the CLKOUT function.
2. Pins PD08, PD09, and PD10 are not available in the SH72A0 Group.
3. The SSL30 (SBI) function is not available in the SH72A0 Group.

Table 13.6 Multiplexed Pin Functions for Port E

Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)	Function 6 (Related Module)	Function 7 (Related Module)	Function 8 (Related Module)	Function 9 (Related Module)	Function 10 (Related Module)	Function 11 (Related Module)	Function 12 (Related Module)	Function 13 (Related Module)
PE00 (Port)	INT8 (INTC)	TP02C (TPU)	—	—	—	—	—	—	—	—	—	—
PE01 (Port)	—	—	—	RXD0 (SCI)	—	—	—	—	—	—	—	—
PE02 (Port)	—	—	—	TXD0 (SCI)	—	—	—	—	—	—	—	—
PE03 (Port)	—	TP02D (TPU)	—	SCK0 (SCI)	—	—	—	—	—	—	—	—
PE04 (Port)	—	—	—	—	—	—	—	—	—	—	—	—
PE05 (Port)	—	—	—	SCK1 (SCI)	—	—	—	—	—	—	—	—
PE06 (Port)	—	—	—	RXD1 (SCI)	—	—	—	—	—	—	—	—
PE07 (Port)	—	—	—	TXD1 (SCI)	—	—	—	—	—	—	—	—

Table 13.7 Multiplexed Pin Functions for Port F

Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)	Function 6 (Related Module)	Function 7 (Related Module)	Function 8 (Related Module)	Function 9 (Related Module)	Function 10 (Related Module)	Function 11 (Related Module)	Function 12 (Related Module)	Function 13 (Related Module)
PF00 * (Port)	—	—	—	SCK1 (SCI)	—	—	—	SSL13 (SBI)	—	—	—	—
PF01 * (Port)	—	—	—	—	—	LRX3 (LIN)	—	SSL12 (SBI)	—	—	—	—
PF02 * (Port)	—	TP02A (TPU)	—	RXD1 (SCI)	—	LTX3 (LIN)	—	SSL11 (SBI)	—	—	—	—
PF07 * (Port)	INT9 (INTC)	TP02B (TPU)	—	TXD1 (SCI)	—	—	—	—	—	—	—	—

Note: * The port F pins are not available in the SH72A0 Group.

Table 13.8 Multiplexed Pin Functions for Port G

Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)	Function 6 (Related Module)	Function 7 (Related Module)	Function 8 (Related Module)	Function 9 (Related Module)	Function 10 (Related Module)	Function 11 (Related Module)	Function 12 (Related Module)	Function 13 (Related Module)
PG00 (Port)	—	—	TIOC7A (MTU-III)	—	—	—	—	MISO0 (SBI)	—	—	—	—
PG01 (Port)	—	—	TIOC7C (MTU-III)	—	—	—	—	SSL00 (SBI)	—	—	—	—
PG02 (Port)	INT10 (INTC)	—	TIOC7B (MTU-III)	—	—	—	—	SSL01 (SBI)	—	—	—	—
PG03 (Port)	—	—	TIOC7D (MTU-III)	—	—	—	—	SSL02 (SBI)	—	—	—	—
PG10 (Port)	—	TP01A (TPU)	TIOC0A (MTU-III)	—	—	—	—	RSPCK1 (SBI)	—	—	—	—
PG11 (Port)	—	TP01B (TPU)	TIOC0B (MTU-III)	—	—	—	—	MOSI1 (SBI)	—	—	—	—
PG12 (Port)	—	TP01C (TPU)	TIOC0C (MTU-III)	—	—	—	—	MISO1 (SBI)	—	—	—	—
PG13 * (Port)	—	TP01D (TPU)	TIOC0D (MTU-III)	—	—	—	—	SSL10 (SBI)	—	—	—	—

Note: * The PG13 pin is not available in the SH72A0 Group.

Table 13.9 Multiplexed Pin Functions for Port J

Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)	Function 6 (Related Module)	Function 7 (Related Module)	Function 8 (Related Module)	Function 9 (Related Module)	Function 10 (Related Module)	Function 11 (Related Module)	Function 12 (Related Module)	Function 13 (Related Module)
PJ00 * (Port)	—	TP02A (TPU)	—	—	—	—	—	—	—	—	AN1IN00 (A/D)	—
PJ01 * (Port)	—	TP02B (TPU)	—	—	—	—	—	—	—	—	AN1IN01 (A/D)	—
PJ02 * (Port)	—	TP02C (TPU)	—	—	—	—	—	—	—	—	AN1IN02 (A/D)	—
PJ03 * (Port)	—	TP02D (TPU)	—	—	—	—	—	—	—	—	AN1IN03 (A/D)	—
PJ04 (Port)	—	TP03A (TPU)	—	—	—	—	—	—	—	—	AN1IN04 (A/D)	—
PJ05 (Port)	POE1# (INTC)	TP03B (TPU)	—	—	—	—	—	—	—	—	AN1IN05 (A/D)	—
PJ06 * (Port)	—	—	TCLKC (MTU-III)	—	—	—	—	—	—	—	AN1IN06 (A/D)	—
PJ07 * (Port)	INT11 (INTC)	—	TCLKD (MTU-III)	—	—	—	—	SSL03 (SBI)	—	—	AN1IN07 (A/D)	—
PJ08 (Port)	—	—	TIOC6B (MTU-III)	—	—	—	—	MOSI0 (SBI)	—	—	—	—
PJ09 (Port)	—	—	TIOC6D (MTU-III)	—	—	—	—	RSPCK0 (SBI)	—	—	—	—
PJ10 * (Port)	—	TP03C (TPU)	TIOC6A (MTU-III)	—	—	—	—	—	—	—	—	—
PJ11 * (Port)	—	TP03D (TPU)	TIOC6C (MTU-III)	—	—	—	—	—	—	—	—	—

Note: * Pins PJ00, PJ01, PJ02, PJ03, PJ06, PJ07, PJ10, and PJ11 are not available in the SH72A0 Group.

Table 13.10 Multiplexed Pin Functions for Port K

Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)	Function 6 (Related Module)	Function 7 (Related Module)	Function 8 (Related Module)	Function 9 (Related Module)	Function 10 (Related Module)	Function 11 (Related Module)	Function 12 (Related Module)	Function 13 (Related Module)
PK00 (Port)	POE2# (INTC)	—	—	—	—	—	—	—	—	—	AN1IN16 (A/D)	—
PK01 (Port)	—	—	—	—	—	—	—	—	—	—	AN1IN17 (A/D)	—
PK08 (Port)	—	—	—	—	—	—	—	—	—	—	AN1IN24 (A/D)	—
PK09 * (Port)	—	—	—	—	—	—	—	—	—	—	AN1IN25 (A/D)	—
PK10 * (Port)	INT13 (INTC)	—	—	—	—	—	—	—	—	—	AN1IN26 (A/D)	—
PK11 * (Port)	INT12 (INTC)	—	—	—	—	—	—	—	—	—	AN1IN27 (A/D)	—
PK12 (Port)	—	—	—	—	—	—	—	—	—	—	AN1IN28 (A/D)	—
PK13 * (Port)	—	—	—	—	—	—	—	—	—	—	AN1IN29 (A/D)	—
PK14 (Port)	—	—	—	—	—	—	—	—	—	—	AN1IN30 (A/D)	—
PK15 (Port)	—	—	—	—	—	—	—	—	—	—	AN1IN31 (A/D)	—

Note: *Pins PK09, PK10, PK11, and PK13 are not available in the SH72A0 Group.

Table 13.11 Multiplexed Pin Functions for Port L

Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)	Function 6 (Related Module)	Function 7 (Related Module)	Function 8 (Related Module)	Function 9 (Related Module)	Function 10 (Related Module)	Function 11 (Related Module)	Function 12 (Related Module)	Function 13 (Related Module)
PL10 * (Port)	—	—	—	—	—	—	—	—	—	—	AN1IN42 (A/D)	—
PL11 * (Port)	—	—	—	—	—	—	—	—	—	—	AN1IN43 (A/D)	—
PL12 * (Port)	—	TP04A (TPU)	—	—	—	—	—	—	—	—	AN1IN44 (A/D)	—
PL13 * (Port)	—	TP04B (TPU)	—	—	—	—	—	—	—	—	AN1IN45 (A/D)	—
PL14 * (Port)	—	TP04C (TPU)	—	—	—	—	—	—	—	—	AN1IN46 (A/D)	—
PL15 * (Port)	—	TP04D (TPU)	—	—	—	—	—	—	—	—	AN1IN47 (A/D)	—

Note: * The port L pins are not available in the SH72A0 Group.

Table 13.12 Multiplexed Pin Functions for Port N

Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)	Function 4 (Related Module)	Function 5 (Related Module)	Function 6 (Related Module)	Function 7 (Related Module)	Function 8 (Related Module)	Function 9 (Related Module)	Function 10 (Related Module)	Function 11 (Related Module)	Function 12 (Related Module)	Function 13 (Related Module)
PN00 (Port)	—	—	—	—	—	—	—	—	—	—	AN0IN00 (A/D)	—
PN01 (Port)	—	—	—	—	—	—	—	—	—	—	AN0IN01 (A/D)	—
PN02 (Port)	—	—	—	—	—	—	—	—	—	—	AN0IN02 (A/D)	—
PN03 (Port)	—	—	—	—	—	—	—	—	—	—	AN0IN03 (A/D)	—
PN04 (Port)	—	—	—	—	—	—	—	—	—	—	AN0IN04 (A/D)	—
PN05 (Port)	—	—	—	—	—	—	—	—	—	—	AN0IN05 (A/D)	—

13.2 Registers

Table 13.13 to table 13.15 list the I/O port registers.

Table 13.13 I/O Port Registers (1)

Register Name	Symbol	After Reset	Address	Access Size
Port protect register	PPR	H'00	H'FF46 4180	8
Port A register	PAR	Undefined	H'FF46 4000	8, 16, 32
Port B register*	PBR	Undefined	H'FF46 4002	8, 16, 32
Port C register	PCR	Undefined	H'FF46 4004	8, 16, 32
Port D register	PDR	Undefined	H'FF46 4006	8, 16, 32
Port E register	PER	Undefined	H'FF46 4008	8, 16, 32
Port F register*	PFR	Undefined	H'FF46 400A	8, 16, 32
Port G register	PGR	Undefined	H'FF46 400C	8, 16, 32
Port J register	PJR	Undefined	H'FF46 4012	8, 16, 32
Port K register	PKR	Undefined	H'FF46 4014	8, 16, 32
Port L register*	PLR	Undefined	H'FF46 4016	8, 16, 32
Port N register	PNR	Undefined	H'FF46 401A	8, 16, 32
Port A direction Register	PADR	H'0000	H'FF46 4020	8, 16, 32
Port B direction Register*	PBDR	H'0000	H'FF46 4022	8, 16, 32
Port C direction Register	PCDR	H'0000	H'FF46 4024	8, 16, 32
Port D direction Register	PDDR	H'0000	H'FF46 4026	8, 16, 32
Port E direction Register	PEDR	H'0000	H'FF46 4028	8, 16, 32
Port F direction Register*	PFDR	H'0000	H'FF46 402A	8, 16, 32
Port G direction Register	PGDR	H'0000	H'FF46 402C	8, 16, 32
Port J direction Register	PJDR	H'0000	H'FF46 4032	8, 16, 32
Port K direction Register	PKDR	H'0000	H'FF46 4034	8, 16, 32
Port L direction Register*	PLDR	H'0000	H'FF46 4036	8, 16, 32
Pull-up control register 0	PUR0	H'0000	H'FF46 4040	8, 16, 32
Pull-up control register 1	PUR1	H'0000	H'FF46 4042	8, 16, 32
Input threshold value select register 0	PVSR0	H'0000	H'FF46 4160	8, 16, 32
Input threshold value select register 1	PVSR1	H'0000	H'FF46 4162	8, 16, 32
Input threshold value select register 2	PVSR2	H'0000	H'FF46 4164	8, 16, 32
Input threshold value select register 3	PVSR3	H'0000	H'FF46 4166	8, 16, 32
Port function select register 0	PFS0	H'0000	H'FF46 4140	8, 16, 32
Port function select register 1	PFS1	H'0000	H'FF46 4142	8, 16, 32
Port function select register 2	PFS2	H'0000	H'FF46 4144	8, 16, 32
Port function select register 3	PFS3	H'0000	H'FF46 4146	8, 16, 32
Port A00 function select register	PA00S	H'00	H'FF46 4060	8, 16, 32
Port A01 function select register	PA01S	H'00	H'FF46 4061	8, 16, 32
Port A02 function select register*	PA02S	H'00	H'FF46 4062	8, 16, 32
Port A03 function select register	PA03S	H'00	H'FF46 4063	8, 16, 32
Port A04 function select register	PA04S	H'00	H'FF46 4064	8, 16, 32
Port A05 function select register	PA05S	H'00	H'FF46 4065	8, 16, 32
Port B12 function select register*	PB12S	H'00	H'FF46 4074	8, 16, 32
Port B13 function select register*	PB13S	H'00	H'FF46 4075	8, 16, 32

Note: * This register is not usable in the SH72A0 Group.

Table 13.14 I/O Port Registers (2)

Register Name	Symbol	After Reset	Address	Access Size
Port B14 function select register*	PB14S	H'00	H'FF46 4076	8, 16, 32
Port B15 function select register*	PB15S	H'00	H'FF46 4077	8, 16, 32
Port C08 function select register	PC08S	H'00	H'FF46 4080	8, 16, 32
Port C09 function select register	PC09S	H'00	H'FF46 4081	8, 16, 32
Port C10 function select register*	PC10S	H'00	H'FF46 4082	8, 16, 32
Port C11 function select register*	PC11S	H'00	H'FF46 4083	8, 16, 32
Port C12 function select register*	PC12S	H'00	H'FF46 4084	8, 16, 32
Port D00 function select register	PD00S	H'00	H'FF46 4088	8, 16, 32
Port D01 function select register	PD01S	H'00	H'FF46 4089	8, 16, 32
Port D02 function select register	PD02S	H'00	H'FF46 408A	8, 16, 32
Port D03 function select register	PD03S	H'00	H'FF46 408B	8, 16, 32
Port D04 function select register	PD04S	H'00	H'FF46 408C	8, 16, 32
Port D05 function select register	PD05S	H'00	H'FF46 408D	8, 16, 32
Port D06 function select register	PD06S	H'00	H'FF46 408E	8, 16, 32
Port D07 function select register	PD07S	H'00	H'FF46 408F	8, 16, 32
Port D08 function select register*	PD08S	H'00	H'FF46 4090	8, 16, 32
Port D09 function select register*	PD09S	H'00	H'FF46 4091	8, 16, 32
Port D10 function select register*	PD10S	H'00	H'FF46 4092	8, 16, 32
Port E00 function select register	PE00S	H'00	H'FF46 4098	8, 16, 32
Port E01 function select register	PE01S	H'00	H'FF46 4099	8, 16, 32
Port E02 function select register	PE02S	H'00	H'FF46 409A	8, 16, 32
Port E03 function select register	PE03S	H'00	H'FF46 409B	8, 16, 32
Port E04 function select register	PE04S	H'00	H'FF46 409C	8, 16, 32
Port E05 function select register	PE05S	H'00	H'FF46 409D	8, 16, 32
Port E06 function select register	PE06S	H'00	H'FF46 409E	8, 16, 32
Port E07 function select register	PE07S	H'00	H'FF46 409F	8, 16, 32
Port F00 function select register*	PF00S	H'00	H'FF46 40A8	8, 16, 32
Port F01 function select register*	PF01S	H'00	H'FF46 40A9	8, 16, 32
Port F02 function select register*	PF02S	H'00	H'FF46 40AA	8, 16, 32
Port F07 function select register*	PF07S	H'00	H'FF46 40AF	8, 16, 32
Port G00 function select register	PG00S	H'00	H'FF46 40B0	8, 16, 32
Port G01 function select register	PG01S	H'00	H'FF46 40B1	8, 16, 32
Port G02 function select register	PG02S	H'00	H'FF46 40B2	8, 16, 32
Port G03 function select register	PG03S	H'00	H'FF46 40B3	8, 16, 32
Port G10 function select register	PG10S	H'00	H'FF46 40BA	8, 16, 32
Port G11 function select register	PG11S	H'00	H'FF46 40BB	8, 16, 32
Port G12 function select register	PG12S	H'00	H'FF46 40BC	8, 16, 32
Port G13 function select register*	PG13S	H'00	H'FF46 40BD	8, 16, 32
Port J00 function select register*	PJ00S	H'00	H'FF46 40C8	8, 16, 32
Port J01 function select register*	PJ01S	H'00	H'FF46 40C9	8, 16, 32
Port J02 function select register*	PJ02S	H'00	H'FF46 40CA	8, 16, 32
Port J03 function select register*	PJ03S	H'00	H'FF46 40CB	8, 16, 32
Port J04 function select register	PJ04S	H'00	H'FF46 40CC	8, 16, 32

Note: * This register is not usable in the SH72A0 Group.

Table 13.15 I/O Port Registers (3)

Register Name	Symbol	After Reset	Address	Access Size
Port J05 function select register	PJ05S	H'00	H'FF46 40CD	8, 16, 32
Port J06 function select register*	PJ06S	H'00	H'FF46 40CE	8, 16, 32
Port J07 function select register*	PJ07S	H'00	H'FF46 40CF	8, 16, 32
Port J08 function select register	PJ08S	H'00	H'FF46 40D0	8, 16, 32
Port J09 function select register	PJ09S	H'00	H'FF46 40D1	8, 16, 32
Port J10 function select register*	PJ10S	H'00	H'FF46 40D2	8, 16, 32
Port J11 function select register*	PJ11S	H'00	H'FF46 40D3	8, 16, 32
Port K00 function select register	PK00S	H'00	H'FF46 40D8	8, 16, 32
Port K01 function select register	PK01S	H'00	H'FF46 40D9	8, 16, 32
Port K08 function select register	PK08S	H'00	H'FF46 40E0	8, 16, 32
Port K09 function select register*	PK09S	H'00	H'FF46 40E1	8, 16, 32
Port K10 function select register*	PK10S	H'00	H'FF46 40E2	8, 16, 32
Port K11 function select register*	PK11S	H'00	H'FF46 40E3	8, 16, 32
Port K12 function select register	PK12S	H'00	H'FF46 40E4	8, 16, 32
Port K13 function select register*	PK13S	H'00	H'FF46 40E5	8, 16, 32
Port K14 function select register	PK14S	H'00	H'FF46 40E6	8, 16, 32
Port K15 function select register	PK15S	H'00	H'FF46 40E7	8, 16, 32
Port L10 function select register*	PL10S	H'00	H'FF46 40F2	8, 16, 32
Port L11 function select register*	PL11S	H'00	H'FF46 40F3	8, 16, 32
Port L12 function select register*	PL12S	H'00	H'FF46 40F4	8, 16, 32
Port L13 function select register*	PL13S	H'00	H'FF46 40F5	8, 16, 32
Port L14 function select register*	PL14S	H'00	H'FF46 40F6	8, 16, 32
Port L15 function select register*	PL15S	H'00	H'FF46 40F7	8, 16, 32
Port N00 function select register	PN00S	H'00	H'FF46 4108	8, 16, 32
Port N01 function select register	PN01S	H'00	H'FF46 4109	8, 16, 32
Port N02 function select register	PN02S	H'00	H'FF46 410A	8, 16, 32
Port N03 function select register	PN03S	H'00	H'FF46 410B	8, 16, 32
Port N04 function select register	PN04S	H'00	H'FF46 410C	8, 16, 32
Port N05 function select register	PN05S	H'00	H'FF46 410D	8, 16, 32
Port function enable register 0	PFEN0	H'0000	H'FF46 4120	8, 16
POE0 control register	POE0CR	H'0000	H'FF46 41A0	8, 16, 32
POE1 control register	POE1CR	H'0000	H'FF46 41A2	8, 16, 32
POE2 control register	POE2CR	H'0000	H'FF46 41A4	8, 16, 32
POE monitor register	POEM	H'00	H'FF46 41A6	8, 16, 32

Note: * This register is not usable in the SH72A0 Group.

13.2.1 Port Protect Register (PPR)

Address H'FF46 4180

	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	0	0	0	0	0	0	0	0

Bit	Description	R/W
b7 to 0	This register is used to control the protection for registers PADR to PGDR, PJDR to PLDR, PVSR0 to PVSR3, PFS0 to PFS3, PFEN0, POE0CR to POE2CR, and POEM. H'AA: Write enabled Other than H'AA: Write disabled	R/W

The port protect register (PPR) is used to control whether to enable or disable writing to the port-related registers. The registers protected by the PPR register are as follows:

- Port i direction register (i = A to G and J to L)
- Input threshold value select registers 0 to 3
- Port function select registers 0 to 3
- Port function enable register 0
- POEi control register (i = 0 to 2)
- POE monitor register

To change a value in these registers, perform the following:

(1) Write H'AA in the PPR register (to enable writing to each register).
(2) Change a value in registers PADR to PGDR, PJDR to PLDR, PVSR0 to PVSR3, PFS0 to PFS3, PFEN0, POE0CR to POE2CR, or POEM.
(3) Write a value other than H'AA in the PPR register (to prohibit writing to each register).

13.2.2 Port i Register (PiR) (i = A to G, J to L, N)

Address PAR: H'FF46 4000

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	PA05	PA04	PA03	PA02 *	PA01	PA00
After Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Address PBR: H'FF46 4002

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PB15 *	PB14 *	PB13 *	PB12 *	—	—	—	—	—	—	—	—	—	—	—	—
After Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Address PCR: H'FF46 4004

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	PC12 *	PC11 *	PC10 *	PC09	PC08	—	—	—	—	—	—	—	—
After Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Address PDR: H'FF46 4006

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	PD10 *	PD09 *	PD08 *	PD07	PD06	PD05	PD04	PD03	PD02	PD01	PD00
After Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Address PER: H'FF46 4008

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	PE07	PE06	PE05	PE04	PE03	PE02	PE01	PE00
After Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Address PFR: H'FF46 400A

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	PF07 *	—	—	—	—	PF02 *	PF01 *	PF00 *
After Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Address PGR: H'FF46 400C

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	PG13 *	PG12	PG11	PG10	—	—	—	—	—	—	PG03	PG02	PG01	PG00
After Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Address PJR: H'FF46 4012

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	PJ11 *	PJ10 *	PJ09	PJ08	PJ07 *	PJ06 *	PJ05	PJ04	PJ03 *	PJ02 *	PJ01 *	PJ00 *
After Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Address PKR: H'FF46 4014

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PK15	PK14	PK13 *	PK12	PK11 *	PK10 *	PK09 *	PK08	—	—	—	—	—	—	PK01	PK00
After Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Address PLR: H'FF46 4016

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PL15 *	PL14 *	PL13 *	PL12 *	PL11 *	PL10 *	—	—	—	—	—	—	—	—	—	—
After Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Address PNR: H'FF46 401A

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	PN05	PN04	PN03	PN02	PN01	PN00
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
b15	Pi15	Port i15	When the corresponding direction bit is set to 0 (input): The written value is written to the register, but not output to the pin. When these bits are read, the corresponding pin state can be read. 0: The pin state is low 1: The pin state is high When the corresponding direction bit is set to 1 (output): The written level is output to the pin. 0: Low is output 1: High is output When these bits are read while the PRFS bit in the PFS0 register is 0, the setting value of the port register is read. When these bits are read while the PRFS bit in the PFS0 register is 1, the input level of the pin is directly read.	R/W
b14	Pi14	Port i14		R/W
b13	Pi13	Port i13		R/W
b12	Pi12	Port i12		R/W
b11	Pi11	Port i11		R/W
b10	Pi10	Port i10		R/W
b9	Pi09	Port i09		R/W
b8	Pi08	Port i08		R/W
b7	Pi07	Port i07		R/W
b6	Pi06	Port i06		R/W
b5	Pi05	Port i05		R/W
b4	Pi04	Port i04		R/W
b3	Pi03	Port i03		R/W
b2	Pi02	Port i02		R/W
b1	Pi01	Port i01		R/W
b0	Pi00	Port i00		R/W

Notes: • The following port bits are reserved and nothing is assigned. The read value is undefined, and the write value should be 0.
Port A: b15 to b6
Port B: b11 to b0
Port C: b15 to b13, b7 to b0
Port D: b15 to b11
Port E: b15 to b8
Port F: b15 to b8, b6 to b3
Port G: b15, b14, b9 to b4
Port J: b15 to b12
Port K: b7 to b2

Port L: b9 to b0
Port N: b15 to b6

- As PN00 to PN05 are input-only ports, only the state of the pin is read when the port PN00 to PN05 bits are read. The write value should be 0.
- This register is for the reading of pins that have a fixed state. When variable signals are applied to the pins, employ processing such as reading the register multiple times as required.

* The following port bits are reserved and nothing is assigned in the SH72A0 Group.
The read value is undefined, and the write value should be 0.
Port A: b2
Port B: b15 to b12
Port C: b12 to b10
Port D: b10 to b8
Port F: b7, b2 to b0
Port G: b13
Port J: b11, b10, b7, b6, b3 to b0
Port K: b13, b11 to b9
Port L: b15 to b10

13.2.3 Port i Direction Register (PiDR) (i = A to G and J to L)

Address PADR: H'FF46 4020

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	PAD05	PAD04	PAD03	PAD02 *	PAD01	PAD00
After Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Address PBDR: H'FF46 4022

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PBD15 *	PBD14 *	PBD13 *	PBD12 *	—	—	—	—	—	—	—	—	—	—	—	—
After Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Address PCDR: H'FF46 4024

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	PCD12 *	PCD11 *	PCD10 *	PCD09	PCD08	—	—	—	—	—	—	—	—
After Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Address PDDR: H'FF46 4026

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	PDD10 *	PDD09 *	PDD08 *	PDD07	PDD06	PDD05	PDD04	PDD03	PDD02	PDD01	PDD00
After Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Address PEDR: H'FF46 4028

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	PED07	PED06	PED05	PED04	PED03	PED02	PED01	PED00
After Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Address PFDR: H'FF46 402A

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	PFD07 *	—	—	—	—	PFD02 *	PFD01 *	PFD00 *
After Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Address PGDR: H'FF46 402C

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	PGD13 *	PGD12	PGD11	PGD10	—	—	—	—	—	—	PGD03	PGD02	PGD01	PGD00
After Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Address PJDR: H'FF46 4032

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	PJD11 *	PJD10 *	PJD09	PJD08	PJD07 *	PJD06 *	PJD05	PJD04	PJD03 *	PJD02 *	PJD01 *	PJD00 *
After Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Address PKDR: H'FF46 4034

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PKD15	PKD14	PKD13 *	PKD12	PKD11 *	PKD10 *	PKD09 *	PKD08	—	—	—	—	—	—	PKD01	PKD00
After Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Address PLDR: H'FF46 4036

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PLD15 *	PLD14 *	PLD13 *	PLD12 *	PLD11 *	PLD10 *	—	—	—	—	—	—	—	—	—	—
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	PiD15	Port i15 Direction	These bits select the input or output direction of the corresponding pin. 0: Input 1: Output	R/W
b14	PiD14	Port i14 Direction		R/W
b13	PiD13	Port i13 Direction		R/W
b12	PiD12	Port i12 Direction		R/W
b11	PiD11	Port i11 Direction		R/W
b10	PiD10	Port i10 Direction		R/W
b9	PiD09	Port i09 Direction		R/W
b8	PiD08	Port i08 Direction		R/W
b7	PiD07	Port i07 Direction		R/W
b6	PiD06	Port i06 Direction		R/W
b5	PiD05	Port i05 Direction		R/W
b4	PiD04	Port i04 Direction		R/W
b3	PiD03	Port i03 Direction		R/W
b2	PiD02	Port i02 Direction		R/W
b1	PiD01	Port i01 Direction		R/W
b0	PiD00	Port i00 Direction		R/W

Notes: • Set the PPR register to H'AA (write enabled) before rewriting this register.
• The following port bits are reserved and nothing is assigned. The read value is 0, and the write value should be 0.
Port A: b15 to b6
Port B: b11 to b0
Port C: b15 to b13, b7 to b0
Port D: b15 to b11
Port E: b15 to b8
Port F: b15 to b8, b6 to b3
Port G: b15, b14, b9 to b4
Port J: b15 to b12
Port K: b7 to b2
Port L: b9 to b0
• As PN00 to PN05 are input-only ports, there is no direction register.
* The following port bits are reserved and nothing is assigned in the SH72A0 Group.
These bits are read as undefined, and the write value should be 0.
Port A: b2
Port B: b15 to b12
Port C: b12 to b10

Port D: b10 to b8
Port F: b7, b2 to b0
Port G: b13
Port J: b11, b10, b7, b6, b3 to b0
Port K: b13, b11 to b9
Port L: b15 to b10

The port i direction register (PiDR) is used to select the input or output direction of the corresponding pin. This register is protected by the port protect register (PPR) from being written inadvertently.

13.2.4 Pull-Up Control Register 0 (PUR0)

Address H'FF46 4040

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	PGPU1	PGPU0	—	PFPU0	—	PEPU0	PDPU1	PDPU0	PCPU1	—	PBPU1	—	—	PAPU0
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, b14	—	Reserved	These bits are read as 0. The write value should be 0.	R
b13	PGPU1	PG10 to PG13 Pull-Up Control	0: PG10 to PG13 not pulled up 1: PG10 to PG13 pulled up *1	R/W
b12	PGPU0	PG00 to PG03 Pull-Up Control	0: PG00 to PG03 not pulled up 1: PG00 to PG03 pulled up	R/W
b11	—	Reserved	This bit is read as 0. The write value should be 0.	R
b10	PFPU0	PF00 to PF02, PF07 Pull-Up Control	0: PF00 to PF02, PF07 not pulled up 1: PF00 to PF02, PF07 pulled up *2	R/W
b9	—	Reserved	This bit is read as 0. The write value should be 0.	R
b8	PEPU0	PE00 to PE07 Pull-Up Control	0: PE00 to PE07 not pulled up 1: PE00 to PE07 pulled up	R/W
b7	PDPU1	PD08 to PD10 Pull-Up Control	0: PD08 to PD10 not pulled up 1: PD08 to PD10 pulled up *3	R/W
b6	PDPU0	PD00 to PD07 Pull-Up Control	0: PD00 to PD07 not pulled up 1: PD00 to PD07 pulled up	R/W
b5	PCPU1	PC08 to PC12 Pull-Up Control	0: PC08 to PC12 not pulled up 1: PC08 to PC12 pulled up *4	R/W
b4	—	Reserved	This bit is read as 0. The write value should be 0.	R
b3	PBPU1	PB12 to PB15 Pull-Up Control	0: PB12 to PB15 not pulled up 1: PB12 to PB15 pulled up *5	R/W
b2, b1	—	Reserved	These bits are read as 0. The write value should be 0.	R
b0	PAPU0	PA00 to PA05 Pull-Up Control	0: PA00 to PA05 not pulled up 1: PA00 to PA05 pulled up *6	R/W

Notes: 1. The PG13 pin is not available in the SH72A0 Group.
2. Pins PF00 to PF02 and PF07 are not available in the SH72A0 Group.
3. Pins PD08 to PD10 are not available in the SH72A0 Group.
4. Pins PC10 to PC12 are not available in the SH72A0 Group.
5. Pins PB12 to PB15 are not available in the SH72A0 Group.
6. The PA02 pin is not available in the SH72A0 Group.

Pull-up control register 0 (PUR0) is used to select whether or not to pull up ports A to G for the corresponding applicable pins per port.

13.2.5 Pull-Up Control Register 1 (PUR1)

Address H'FF46 4042

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	PNPU0	—	—	PLPU1	—	PKPU1	PKPU0	PJPU1	PJPU0	—	—
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 b11	—	Reserved	These bits are read as 0. The write value should be 0.	R
b10	PNPU0	PN00 to PN05 Pull-Up Control	0: PN00 to PN05 not pulled up 1: PN00 to PN05 pulled up	R/W
b9, b8	—	Reserved	These bits are read as 0. The write value should be 0.	R
b7	PLPU1	PL10 to PL15 Pull-Up Control	0: PL10 to PL15 not pulled up 1: PL10 to PL15 pulled up *1	R/W
b6	—	Reserved	This bit is read as 0. The write value should be 0.	R
b5	PKPU1	PK08 to PK15 Pull-Up Control	0: PK08 to PK15 not pulled up 1: PK08 to PK15 pulled up *2	R/W
b4	PKPU0	PK00 and PK01 Pull-Up Control	0: PK00 and PK01 not pulled up 1: PK00 and PK01 pulled up	R/W
b3	PJPU1	PJ08 to PJ11 Pull-Up Control	0: PJ08 to PJ11 not pulled up 1: PJ08 to PJ11 pulled up *3	R/W
b2	PJPU0	PJ00 to PJ07 Pull-Up Control	0: PJ00 to PJ07 not pulled up 1: PJ00 to PJ07 pulled up *4	R/W
b1, b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

Notes: • For pins on which A/D functions are multiplexed, if the corresponding port function select register is set to select operation as an analog input pin, the pull-up settings for the pins become invalid. However, the values in the pull-up control register do not reflect this.
• For pins on which A/D functions are multiplexed, if the corresponding pull-up and pull-down settings are made simultaneously, this is not reflected in the pull-up settings.
Furthermore, the pull-down setting that is valid for a given pin while it is in use as an analog input pin becomes invalid if the pin is used in another role. However, the values in the pull-up control register and analog-port pull-down registers do not reflect this.
1. Pins PL10 to PL15 are not available in the SH72A0 Group.
2. Pins PK09 to PK11 and PK13 are not available in the SH72A0 Group.
3. Pins PJ10 and PJ11 are not available in the SH72A0 Group.
4. Pins PJ00 to PJ03, PJ06, and PJ07 are not available in the SH72A0 Group.

Pull-up control register 1 (PUR1) is used to select whether or not to pull up ports J to L, and N for the corresponding applicable pins per port.

13.2.6 Input Threshold Value Select Register 0 (PVSR0)

Address H'FF46 4160

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PDS1[1:0]		PDS0[1:0]		PCS1[1:0]		—	—	PBS1[1:0]		—	—	—	—	PAS0[1:0]	
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, b14	PDS1[1:0]	PD08 to PD10 Input Level Set	Input threshold value of pins PD08 to PD10 b15b14 0 0 : Input prohibited state 0 1 : 0.5 VCC 1 0 : 0.70 VCC 1 1 : TTL level *1	R/W
b13, b12	PDS0[1:0]	PD00 to PD07 Input Level Set	Input threshold values of pins PD00 to PD07 b13b12 0 0 : Input prohibited state 0 1 : 0.5 VCC 1 0 : 0.70 VCC 1 1 : TTL level	R/W
b11, b10	PCS1[1:0]	PC08 to PC12 Input Level Set	Input threshold values of pins PC08 to PC12 b11b10 0 0 : Input prohibited state 0 1 : 0.5 VCC 1 0 : 0.70 VCC 1 1 : TTL level *2	R/W
b9, b8	—	Reserved	These bits are read as 0. The write value should be 0.	R
b7, b6	PBS1[1:0]	PB12 to PB15 Input Level Set	Input threshold values of pins PB12 to PB15 b7 b6 0 0 : Input prohibited state 0 1 : 0.5 VCC 1 0 : 0.70 VCC 1 1 : TTL level *3	R/W
b5 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b1, b0	PAS0[1:0]	PA00 to PA05 Input Level Set	Input threshold values of pins PA00 to PA05 b1 b0 0 0 : Input prohibited state 0 1 : 0.5 VCC 1 0 : 0.70 VCC 1 1 : TTL level *4	R/W

Notes: To use the pins as input pins for the peripheral functions, this function should also be set.
Set the PPR register to H'AA (write enabled) before rewriting this register.
1. Pins PD08 to PD10 are not available in the SH72A0 Group.
2. Pins PC10 to PC12 are not available in the SH72A0 Group.
3. Pins PB12 to PB15 are not available in the SH72A0 Group.
4. The PA02 pin is not available in the SH72A0 Group.

Input threshold value select register 0 (PVSR0) is used to control the threshold values of ports A to D.
The input threshold value can be selected, for the corresponding applicable pins per port, from among 0.5 VCC, 0.70 VCC, and TTL level.
After a reset, the pins are in the input prohibited state (B'00), so it is necessary to set a value other than B'00 to use them as input pins.

The PVSR0 register is protected by the port protect register (PPR) from being written inadvertently.

13.2.7 Input Threshold Value Select Register 1 (PVSR1)

Address H'FF46 4162

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	PGS1[1:0]		PGS0[1:0]		—	—	PFS0[1:0]		—	—	PES0[1:0]	
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 b12	—	Reserved	These bits are read as 0. The write value should be 0.	R
b11, b10	PGS1[1:0]	PG10 to PG13 Input Level Set	Input threshold value of pins PG10 to PG13 b11b10 0 0 : Input prohibited state 0 1 : 0.5 VCC 1 0 : 0.70 VCC 1 1 : TTL level *1	R/W
b9, b8	PGS0[1:0]	PG00 to PG03 Input Level Set	Input threshold value of pins PG00 to PG03 b9 b8 0 0 : Input prohibited state 0 1 : 0.5 VCC 1 0 : 0.70 VCC 1 1 : TTL level	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R
b5, b4	PFS0[1:0]	PF00 to PF02, PF07 Input Level Set	Input threshold value of pins PF00 to PF02, PF07 b5 b4 0 0 : Input prohibited state 0 1 : 0.5 VCC 1 0 : 0.70 VCC 1 1 : TTL level *2	R/W
b3, b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b1, b0	PES0[1:0]	PE00 to PE07 Input Level Set	Input threshold value of pins PE00 to PE07 b1 b0 0 0 : Input prohibited state 0 1 : 0.5 VCC 1 0 : 0.70 VCC 1 1 : TTL level	R/W

Notes: To use the pins as input pins for the peripheral functions, this function should also be set.
Set the PPR register to H'AA (write enabled) before rewriting this register.
1. The PG13 pin is not available in the SH72A0 Group.
2. Pins PF00 to PF02 and PF07 are not available in the SH72A0 Group.

Input threshold value select register 1 (PVSR1) is used to control the threshold values of ports E to G.
The input threshold value can be selected, for the corresponding applicable pins per port, from among 0.5 VCC, 0.70 VCC, and TTL level.
After a reset, the pins are in the input prohibited state (B'00), so it is necessary to set a value other than B'00 to use them as input pins.

The PVSR1 register is protected by the port protect register (PPR) from being written inadvertently.

13.2.8 Input Threshold Value Select Register 2 (PVSR2)

Address H'FF46 4164

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	PLS1[1:0]		—	—	PKS1[1:0]		PKS0[1:0]		PJS1[1:0]		PJS0[1:0]	
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 b12	—	Reserved	These bits are read as 0. The write value should be 0.	R
b11, b10	PLS1[1:0]	PL10 to PL15 Input Level Set	Input threshold value of pins PL10 to PL15 b11b10 0 0 : Input prohibited state 0 1 : 0.5 AVCC1 1 0 : 0.70 AVCC1 1 1 : TTL level *1	R/W
b9, b8	—	Reserved	These bits are read as 0. The write value should be 0.	R
b7, b6	PKS1[1:0]	PK08 to PK15 Input Level Set	Input threshold value of pins PK08 to PK15 b7 b6 0 0 : Input prohibited state 0 1 : 0.5 AVCC1 1 0 : 0.70 AVCC1 1 1 : TTL level *2	R/W
b5, b4	PKS0[1:0]	PK00 and PK01 Input Level Set	Input threshold value of pins PK00 and PK01 b5 b4 0 0 : Input prohibited state 0 1 : 0.5 AVCC1 1 0 : 0.70 AVCC1 1 1 : TTL level	R/W
b3, b2	PJS1[1:0]	PJ08 to PJ11 Input Level Set	Input threshold value of pins PJ08 to PJ11 b3 b2 0 0 : Input prohibited state 0 1 : 0.5 VCC 1 0 : 0.70 VCC 1 1 : TTL level *3	R/W
b1, b0	PJS0[1:0]	PJ00 to PJ07 Input Level Set	Input threshold value of pins PJ00 to PJ07 b1 b0 0 0 : Input prohibited state 0 1 : 0.5 VCC 1 0 : 0.70 VCC 1 1 : TTL level *4	R/W

Notes: To use the pins as input pins for the peripheral functions, this function should also be set.
To use the pins as analog pins, there is no need to set this function.
Set the PPR register to H'AA (write enabled) before rewriting this register.
1. Pins PL10 to PL15 are not available in the SH72A0 Group.
2. Pins PK09 to PK11 and PK13 are not available in the SH72A0 Group.
3. Pins PJ10 and PJ11 are not available in the SH72A0 Group.
4. Pins PJ00 to PJ03, PJ06, and PJ07 are not available in the SH72A0 Group.

Input threshold value select register 2 (PVSR2) is used to control the threshold values of ports J to L.

The input threshold value can be selected, for the corresponding applicable pins per port, from among 0.5 VCC, 0.70 VCC, and TTL level (for ports K and L, 0.5 AVCC1, 0.70 AVCC1, and TTL level).

After a reset, the pins are in the input prohibited state (B'00), so it is necessary to set a value other than B'00 to use them as input pins.

The PVSR2 register is protected by the port protect register (PPR) from being written inadvertently.

13.2.9 Input Threshold Value Select Register 3 (PVSR3)

Address H'FF46 4166

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PNS0[1:0]	
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 b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b1, b0	PNS0[1:0]	PN00 to PN05 Input Level Set	Input threshold value of pins PN00 to PN05 b1 b0 0 0 : Input prohibited state 0 1 : 0.5 AVCC0 1 0 : 0.70 AVCC0 1 1 : TTL level	R/W

Notes: To use the pins as analog pins, there is no need to set this function.
Set the PPR register to H'AA (write enabled) before rewriting this register.

Input threshold value select register 3 (PVSR3) is used to control the threshold value of port N.
The input threshold value can be selected, for the corresponding applicable pins per port, from among 0.5 AVCC0, 0.70 AVCC0, and TTL level.
After a reset, the pins are in the input prohibited state (B'00), so it is necessary to set a value other than B'00 to use them as input pins.
The PVSR3 register is protected by the port protect register (PPR) from being written inadvertently.

13.2.10 Port Function Select Register 0 (PFS0)

Address H'FF46 4140

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PRFS	—	—	—	—	—	—	—	—	—	—	—	—	—	—	NMIS
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	PRFS	Port Register Read Function Select	0: When the port register is read while the corresponding port direction register is 1 (output), the value of the register is read. 1: When the port register is read while the corresponding port direction register is 1 (output), the state of the pin is directly read. *	R/W
b14 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R
b0	NMIS	NMI Valid	0: PA05 I/O pin (NMI invalid) 1: PA05 input-only pin (NMI valid)	R/W

Notes: Set the PPR register to H'AA (write enabled) before rewriting this register.
* When a pin state is to be read directly, make an input threshold setting other than “input prohibited” for the given pin.

Port function select register 0 (PFS0) is used to select the functions of pins PA05, NMI, and POE0#, and the function of reading port registers set as outputs.

To use NMI, set the PAS0 bit in the PVSR0 register to a value other than B'00 (input prohibited state) before setting the NMIS bit to 1.
The PFS0 register is protected by the port protect register (PPR) from being written inadvertently.

13.2.11 Port Function Select Register 1 (PFS1)

Address H'FF46 4142

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	TP04S	TP03S	TP02S	—	—
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 b5	—	Reserved	These bits are read as 0. The write value should be 0.	R
b4	TP04S	TP04 Input Pin Select	0: TP04A to TP04D input is set to pins PC08, PC09, PB14, and PB15 1: TP04A to TP04D input is set to pins PL12 to PL15 *1	R/W
b3	TP03S	TP03 Input Pin Select	0: TP03A to TP03D input is set to pins PD00 to PD03 1: TP03A to TP03D input is set to pins PJ04, PJ05, PJ10, and PJ11 *2	R/W
b2	TP02S	TP02 Input Pin Select	0: TP02A to TP02D input is set to pins PF02, PF07, PE00, and PE03 1: TP02A to TP02D input is set to pins PJ00 to PJ03 *3	R/W
b1, b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

Notes: Set the PPR register to H'AA (write enabled) before rewriting this register.
1. Pins PB14, PB15, and PL12 to PL15 are not available in the SH72A0 Group.
2. Pins PJ10 and PJ11 are not available in the SH72A0 Group.
3. Pins PF02, PF07, and PJ00 to PJ03 are not available in the SH72A0 Group.

Port function select register 1 (PFS1) is for selecting input pins for a peripheral function (the TPU) that is multiplexed at multiple sites.

The PFS1 register is protected by the port protect register (PPR) from being written inadvertently.

13.2.12 Port Function Select Register 2 (PFS2)

Address H'FF46 4144

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	SC3S	SC2S[1:0]		—	SC1S	—	—
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 b7	—	Reserved	These bits are read as 0. The write value should be 0.	R
b6	SC3S	SCI3 Input Pin Select	0: RXD3 and SCK3 inputs are set to pins PC08 and PC10 1: RXD3 and SCK3 inputs are set to pins PB14 and PB13 *1	R/W
b5, b4	SC2S[1:0]	SCI2 Input Pin Select	b5 b4 0 0 : RXD2 and SCK2 inputs are set to pins PD02 and PD03 0 1 : RXD2 and SCK2 inputs are set to pins PD07 and PD03 1 0 : Do not set 1 1 : Do not set	R/W
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R
b2	SC1S	SCI1 Input Pin Select	0: RXD1 and SCK1 inputs are set to pins PE06 and PE05 1: RXD1 and SCK1 inputs are set to pins PF02 and PF00*2	R/W
b1, b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

Notes: Set the PPR register to H'AA (write enabled) before rewriting this register.
1. Pins PC10, PB13, and PB14 are not available in the SH72A0 Group.
2. Pins PF00 and PF02 are not available in the SH72A0 Group.

Port function select register 2 (PFS2) is for selecting input pins for a peripheral function (the SCI) that is multiplexed at multiple sites.

The PFS2 register is protected by the port protect register (PPR) from being written inadvertently.

13.2.13 Port Function Select Register 3 (PFS3)

Address H'FF46 4146

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	LN2S	—	—
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 b3	—	Reserved	These bits are read as 0. The write value should be 0.	R
b2	LN2S	LRX2 Input Pin Select	0: LRX2 input is set to the PD05 pin 1: LRX2 input is set to the PB14 pin*	R/W
b1, b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

Notes: Set the PPR register to H'AA (write enabled) before rewriting this register.
* The PB14 pin is not available in the SH72A0 Group.

Port function select register 3 (PFS3) is for selecting an input pin for a peripheral function (the LIN) that is multiplexed at multiple sites.

The PFS3 register is protected by the port protect register (PPR) from being written inadvertently.

13.2.14 Port Ai Function Select Register (PAiS) (i = 00 to 05)

Address PA00S: H'FF46 4060, PA01S: H'FF46 4061, PA02S: H'FF46 4062, PA03S: H'FF46 4063, PA04S: H'FF46 4064, PA05S: H'FF46 4065

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	PAiS[4:0]				
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R
b4 to b0	PAiS[4:0]	PA Function Select	b4 b3 b2b1b0 0 0 0 0 0 : I/O port, or interrupt function *1 0 0 1 0 0 : CAN function Do not set other than above.	R/W

Notes: • These registers are enabled when the PAFEN bit in the port function enable register 0 (PFEN0) is set to 1. When the PAFEN bit is set to 0, port A functions as an I/O port.
• The PA02S register is not available in the SH72A0 Group.
1. To use as an interrupt function, set the corresponding bit to “input” by the PADR register.

The port Ai function select register is used to select the functions of the multiplexed pins of port A.

Register	PA Function Select Bits (PAiS)	
	“00000”	“00100”
PA00S	PA00, INT2	CTX0
PA01S	PA01, INT1	CRX0
PA02S	PA02	—
PA03S	PA03	CTX1
PA04S	PA04, INT0	CRX1
PA05S	PA05, NMI/POE0#	—

13.2.15 Port Bi Function Select Register (PBiS) (i = 12 to 15)

Address PB12S: H'FF46 4074, PB13S: H'FF46 4075, PB14S: H'FF46 4076, PB15S: H'FF46 4077

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	PBiS[4:0]				
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R
b4 to b0	PBiS[4:0]	PB Function Select	b4 b3 b2b1b0 0 0 0 0 0: I/O port, or ADTRG function 0 0 0 0 1 : TPU function 0 0 0 1 1 : SCI function 0 0 1 0 1 : LIN function Do not set other than above.	R/W

Notes: • These registers are enabled when the PBFEN bit in the port function enable register 0 (PFEN0) is set to 1. When the PBFEN bit is set to 0, port B functions as an I/O port.
• Registers PB12S to PB15S are not available in the SH72A0 Group.

The port Bi function select register is used to select the functions of the multiplexed pins of port B.

Register	PB Function Select Bits (PBiS)			
	“00000”	“00001”	“00011”	“00101”
PB12S	PB12	—	—	—
PB13S	PB13	—	SCK3	—
PB14S	PB14	TP04C	RXD3	LRX2
PB15S	PB15, ADTRG#	TP04D	TXD3	LTX2

13.2.16 Port Ci Function Select Register (PCiS) (i = 08 to 12)

Address PC08S: H'FF46 4080, PC09S: H'FF46 4081, PC10S: H'FF46 4082, PC11S: H'FF46 4083, PC12S: H'FF46 4084

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	PCiS[4:0]				
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R
b4 to b0	PCiS[4:0]	PC Function Select	b4 b3 b2b1b0 0 0 0 0 0 : I/O port 0 0 0 0 1 : TPU function 0 0 0 1 0 : MTU-III function (input) 1 0 0 1 0 : MTU-III function (output) 0 0 0 1 1 : SCI function 0 0 1 1 0 : SBI function Do not set other than above.	R/W

Notes: • These registers are enabled when the PCFEN bit in the port function enable register 0 (PFEN0) is set to 1. When the PCFEN bit is set to 0, port C functions as an I/O port.
• Registers PC10S to PC12S are not available in the SH72A0 Group.
• The SBI function is disabled in the SH72A0 Group.

The port Ci function select register is used to select the functions of the multiplexed pins of port C.

Register	PC Function Select Bits (PCiS)					
	“00000”	“00001”	“00010”	“10010”	“00011”	“00110”
PC08S	PC08	TP04A	TIOC1B/ TCLKA	TIOC1B	RXD3	—
PC09S	PC09	TP04B	TCLKB	—	TXD3	—
PC10S	PC10	—	TIC5W	—	SCK3	MOSI3
PC11S	PC11	—	TIC5V	—	—	RSPCK3
PC12S	PC12	—	TIC5U	—	—	MISO3

13.2.17 Port Di Function Select Register (PDiS) (i = 00 to 10)

Address: PD00S: H'FF46 4088, PD01S: H'FF46 4089, PD02S: H'FF46 408A, PD03S: H'FF46 408B, PD04S: H'FF46 408C, PD05S: H'FF46 408D, PD06S: H'FF46 408E, PD07S: H'FF46 408F, PD08S: H'FF46 4090, PD09S: H'FF46 4091, PD10S: H'FF46 4092

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	PDiS[4:0]				
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R
b4 to b0	PDiS[4:0]	PD Function Select	b4 b3 b2 b1b0 0 0 0 0 0 : I/O port, CLKOUT function*2, or interrupt function *1 0 0 0 0 1 : TPU function 0 0 0 1 0 : MTU-III function (input) 1 0 0 1 0 : MTU-III function (output) 0 0 0 1 1 : SCI function 0 0 1 0 1 : LIN function 0 0 1 1 0 : SBI function 0 1 0 1 0 : Debug function Do not set other than above.	R/W

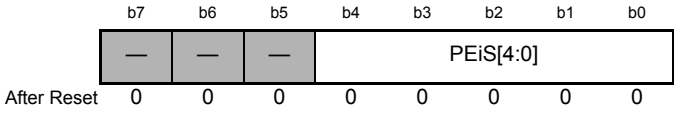
Notes: • These registers are enabled when the PDFEN bit in the port function enable register 0 (PFEN0) is set to 1. When the PDFEN bit is set to 0, port D functions as an I/O port.
• PD00 is shared with CLKOUT. Setting the COE bit in the COCR register to 1 enables the CLKOUT function (the setting of the COCR register has priority).
• Registers PD08S to PD10S are not available in the SH72A0 Group.
1. If the pin to be used is set as input, set this bit to 0.
2. This function is not available in the SH72A0 Group.

The port Di function select register is used to select the functions of the multiplexed pins of port D.

Register	PD Function Select Bits (PDiS)							
	“00000”	“00001”	“00010”	“10010”	“00011”	“00101”	“00110”	“01010”
PD00S	PD00, INT7, CLKOUT	TP03A	TIOC3B	TIOC3B	—	—	SSL21	—
PD01S	PD01, INT6	TP03B	TIOC3D	TIOC3D	TXD2	—	SSL20	—
PD02S	PD02, INT5	TP03C	TIOC4A	TIOC4A	RXD2	—	MOSI2	—
PD03S	PD03	TP03D	TIOC4C	TIOC4C	SCK2	—	RSPCK2	—
PD04S	PD04	—	TIOC4B	TIOC4B	—	LTX2	MISO2	—
PD05S	PD05	—	TIOC4D	TIOC4D	—	LRX2	SSL22	—
PD06S	PD06, INT4	—	TIOC3A	TIOC3A	TXD2	—	SSL23	—
PD07S	PD07, INT3	—	TIOC3C	TIOC3C	RXD2	—	SSL30	UBCTRG#
PD08S	PD08	—	TIOC2A	TIOC2A	—	—	SSL31	—
PD09S	PD09	—	TIOC2B	TIOC2B	—	—	SSL32	—
PD10S	PD10	—	TIOC1A	TIOC1A	—	—	SSL33	—

13.2.18 Port Ei Function Select Register (PEiS) (i = 00 to 07)

Address PE00S: H'FF46 4098, PE01S: H'FF46 4099, PE02S: H'FF46 409A, PE03S: H'FF46 409B, PE04S: H'FF46 409C, PE05S: H'FF46 409D, PE06S: H'FF46 409E, PE07S: H'FF46 409F



Bit	Symbol	Bit Name	Description	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R
b4 to b0	PEiS[4:0]	PE Function Select	b4 b3 b2 b1b0 0 0 0 0 0 : I/O port or interrupt function *1 0 0 0 0 1 : TPU function 0 0 0 1 1 : SCI function Do not set other than above.	R/W

Notes: These registers are enabled when the PEFEN bit in the port function enable register 0 (PFEN0) is set to 1. When the PEFEN bit is set to 0, port E functions as an I/O port.
1. To use as an interrupt function, set the corresponding bit to “input” by the PEDR register.

The port Ei function select register is used to select the functions of the multiplexed pins of port E.

Register	PE Function Select Bits (PEiS)		
	“00000”	“00001”	“00011”
PE00S	PE00, INT8	TP02C	—
PE01S	PE01	—	RXD0
PE02S	PE02	—	TXD0
PE03S	PE03	TP02D	SCK0
PE04S	PE04	—	—
PE05S	PE05	—	SCK1
PE06S	PE06	—	RXD1
PE07S	PE07	—	TXD1

Pins of port E have a further function as pins for use with the advanced user debugger (AUD-II). Operation as an input/output port is enabled when the AUDEN0 bit in the AUDEN register is 0 and operation as an AUD-II pin is forcibly enabled when the AUDEN0 bit in the AUDEN register is 1. The setting in this register is invalid if the AUDEN0 bit is 1.
For details on the AUD-II, refer to section 28, Advanced User Debugger-II (AUD-II).

13.2.19 Port Fi Function Select Register (PFiS) (i = 00 to 02, 07)

Address PF00S: H'FF46 40A8, PF01S: H'FF46 40A9, PF02S: H'FF46 40AA, PF07S: H'FF46 40AF

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	PFiS[4:0]				
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R
b4 to b0	PFiS[4:0]	PF Function Select	b4 b3 b2 b1b0 0 0 0 0 0 : I/O port, or interrupt function*1 0 0 0 0 1 : TPU function 0 0 0 1 1 : SCI function 0 0 1 0 1 : LIN function 0 0 1 1 0 : SBI function Do not set other than above.	R/W

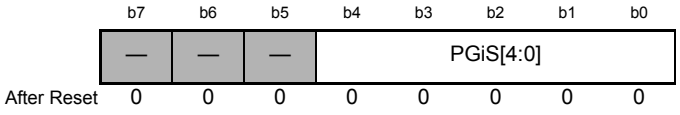
Notes: • These registers are enabled when the PFFEN bit in the port function enable register 0 (PFEN0) is set to 1. When the PFFEN bit is set to 0, port F functions as an I/O port.
• Registers PF00S to PF02S and PF07S are not available in the SH72A0 Group.
1. To use as an interrupt function, set the corresponding bit to “input” by the PFDR register.

The port Fi function select register is used to select the functions of the multiplexed pins of port F.

Register	PF Function Select Bits (PFiS)				
	“00000”	“00001”	“00011”	“00101”	“00110”
PF00S	PF00	—	SCK1	—	SSL13
PF01S	PF01	—	—	LRX3	SSL12
PF02S	PF02	TP02A	RXD1	LTX3	SSL11
PF07S	PF07, INT9	TP02B	TXD1	—	—

13.2.20 Port Gi Function Select Register (PGiS) (i = 00 to 03, 10 to 13)

Address PG00S: H'FF46 40B0, PG01S: H'FF46 40B1, PG02S: H'FF46 40B2, PG03S: H'FF46 40B3,
PG10S: H'FF46 40BA, PG11S: H'FF46 40BB, PG12S: H'FF46 40BC, PG13S: H'FF46 40BD



Bit	Symbol	Bit Name	Description	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R
b4 to b0	PGiS[4:0]	PG Function Select	b4 b3 b2 b1b0 0 0 0 0 0 : I/O port or interrupt function*1 0 0 0 0 1 : TPU function 0 0 0 1 0 : MTU-III function (input) 1 0 0 1 0 : MTU-III function (output) 0 0 1 1 0 : SBI function Do not set other than above.	R/W

Notes: • These registers are enabled when the PGFEN bit in the port function enable register 0 (PFEN0) is set to 1. When the PGFEN bit is set to 0, port G functions as an I/O port.
• The PG13S register is not available in the SH72A0 Group.
1. To use as an interrupt function, set the corresponding bit to “input” by the PGDR register.

The port Gi function select register is used to select the functions of the multiplexed pins of port G.

Register	PG Function Select Bits (PGiS)				
	“00000”	“00001”	“00010”	“10010”	“00110”
PG00S	PG00	—	TIOC7A	TIOC7A	MISO0
PG01S	PG01	—	TIOC7C	TIOC7C	SSL00
PG02S	PG02, INT10	—	TIOC7B	TIOC7B	SSL01
PG03S	PG03	—	TIOC7D	TIOC7D	SSL02
PG10S	PG10	TP01A	TIOC0A	TIOC0A	RSPCK1
PG11S	PG11	TP01B	TIOC0B	TIOC0B	MOSI1
PG12S	PG12	TP01C	TIOC0C	TIOC0C	MISO1
PG13S	PG13	TP01D	TIOC0D	TIOC0D	SSL10

13.2.21 Port Ji Function Select Register (PJiS) (i = 00 to 11)

Address: PJ00S: H'FF46 40C8, PJ01S: H'FF46 40C9, PJ02S: H'FF46 40CA, PJ03S: H'FF46 40CB, PJ04S: H'FF46 40CC, PJ05S: H'FF46 40CD, PJ06S: H'FF46 40CE, PJ07S: H'FF46 40CF, PJ08S: H'FF46 40D0, PJ09S: H'FF46 40D1, PJ10S: H'FF46 40D2, PJ11S: H'FF46 40D3

	b7	b6	b5	b4	b3	b2	b1	b0
	PJiAS	—	—	PJiS[4:0]				
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	PJiAS	PJ Analog Input Pin Select*1	0: Not used as an analog input pin 1: Used as an analog input pin*2	R/W
b6, b5	—	Reserved	These bits are read as 0. The write value should be 0.	R
b4 to b0	PJiS[4:0]	PJ Function Select	b4 b3 b2 b1b0 0 0 0 0 0 : I/O port, interrupt function, or A/D function *3 0 0 0 0 1 : TPU function 0 0 0 1 0 : MTU-III function (input) 1 0 0 1 0 : MTU-III function (output) 0 0 1 1 0 : SBI function Do not set other than above.	R/W

Notes: • These registers are enabled when the PJFEN bit in the port function enable register 0 (PFEN0) is set to 1. When the PJFEN bit is set to 0, port J functions as an I/O port.
• Registers PJ00S to PJ03S, PJ06S, PJ07S, PJ10S, and PJ11S are not available in the SH72A0 Group.
1. To use the A/D function, set this bit to 1. Otherwise, set it to 0.
2. When the PJPU0 bit in the pull-up control register 1 (PUR1) is set to “enabled” while pins are in use as analog input pins, the pull-up settings for the pins set as analog inputs become invalid.
3. To use as an interrupt function or A/D function, set the corresponding bit to “input” by the PJDR register.

The port Ji function select register is used to select the functions of the multiplexed pins of port J.

Register	PJ Function Select Bits (PJiS)				
	“00000”	“00001”	“00010”	“10010”	“00110”
PJ00S	PJ00, AN1IN00	TP02A	—	—	—
PJ01S	PJ01, AN1IN01	TP02B	—	—	—
PJ02S	PJ02, AN1IN02	TP02C	—	—	—
PJ03S	PJ03, AN1IN03	TP02D	—	—	—
PJ04S	PJ04, AN1IN04	TP03A	—	—	—
PJ05S	PJ05, POE1#, AN1IN05	TP03B	—	—	—
PJ06S	PJ06, AN1IN06	—	TCLKC	—	—
PJ07S	PJ07, INT11, AN1IN07	—	TCLKD	—	SSL03
PJ08S	PJ08	—	TIOC6B	TIOC6B	MOSI0
PJ09S	PJ09	—	TIOC6D	TIOC6D	RSPCK0
PJ10S	PJ10	TP03C	TIOC6A	TIOC6A	—
PJ11S	PJ11	TP03D	TIOC6C	TIOC6C	—

13.2.22 Port Ki Function Select Register (PKiS) (i = 00, 01, and 08 to 15)

Address: PK00S: H'FF46 40D8, PK01S: H'FF46 40D9, PK08S: H'FF46 40E0, PK09S: H'FF46 40E1, PK10S: H'FF46 40E2, PK11S: H'FF46 40E3, PK12S: H'FF46 40E4, PK13S: H'FF46 40E5, PK14S: H'FF46 40E6, PK15S: H'FF46 40E7

	b7	b6	b5	b4	b3	b2	b1	b0
	PKiAS	—	—	PKiS[4:0]				
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	PKiAS	PK Analog Input Pin Select*1	0: Not used as an analog input pin 1: Used as an analog input pin*2	R/W
b6, b5	—	Reserved	These bits are read as 0. The write value should be 0.	R
b4 to b0	PKiS[4:0]	PK Function Select	b4 b3 b2 b1b0 0 0 0 0 0 : I/O port, interrupt function, or A/D function*3 Do not set other than above.	R/W

Notes: • These registers are enabled when the PKFEN bit in the port function enable register 0 (PFEN0) is set to 1. When the PKFEN bit is set to 0, port K functions as an I/O port.
• Registers PK09S to PK11S, and PK13S are not available in the SH72A0 Group.
1. To use the A/D function, set this bit to 1. Otherwise, set it to 0.
2. When the PKPU0 or PKPU1 bit in the pull-up control register 1 (PUR1) is set to “enabled” while pins are in use as analog input pins, the pull-up settings for the pins set as analog inputs become invalid.
3. To use as an interrupt function or A/D function, set the corresponding bit to “input” by the PKDR register.

The port Ki function select register is used to select the functions of the multiplexed pins of port K.

Register	PK Function Select Bits (PKiS)
	“00000”
PK00S	PK00, POE2#, AN1IN16
PK01S	PK01, AN1IN17
PK08S	PK08, AN1IN24
PK09S	PK09, AN1IN25
PK10S	PK10, INT13, AN1IN26
PK11S	PK11, INT12, AN1IN27
PK12S	PK12, AN1IN28
PK13S	PK13, AN1IN29
PK14S	PK14, AN1IN30
PK15S	PK15, AN1IN31

13.2.23 Port Li Function Select Register (PLiS) (i = 10 to 15)

Address PL10S: H'FF46 40F2, PL11S: H'FF46 40F3, PL12S: H'FF46 40F4, PL13S: H'FF46 40F5, PL14S: H'FF46 40F6, PL15S: H'FF46 40F7

	b7	b6	b5	b4	b3	b2	b1	b0
	PLiAS	—	—	PLiS[4:0]				
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	PLiAS	PL Analog Input Pin Select*1	0: Not used as an analog input pin 1: Used as an analog input pin*2	R/W
b6, b5	—	Reserved	These bits are read as 0. The write value should be 0.	R
b4 to b0	PLiS[4:0]	PL Function Select	b4 b3 b2b1b0 0 0 0 0 0 : I/O port, or A/D function*3 0 0 0 0 1 : TPU function Do not set other than above.	R/W

Notes: • These registers are enabled when the PLFEN bit in the port function enable register 0 (PFEN0) is set to 1. When the PLFEN bit is set to 0, port L functions as an I/O port.
• Registers PL10S to PL15S are not available in the SH72A0 Group.
1. To use the A/D function, set this bit to 1. Otherwise, set it to 0.
2. When the PLPU1 bit in the pull-up control register 1 (PUR1) is set to “enabled” while pins are in use as analog input pins, the pull-up settings for the pins set as analog inputs become invalid.
3. To use as an A/D function, set the corresponding bit to “input” by the PLDR register.

The port Li function select register is used to select the functions of the multiplexed pins of port L.

Register	PL Function Select Bits (PLiS)	
	“00000”	“00001”
PL10S	PL10, AN1IN42	—
PL11S	PL11, AN1IN43	—
PL12S	PL12, AN1IN44	TP04A
PL13S	PL13, AN1IN45	TP04B
PL14S	PL14, AN1IN46	TP04C
PL15S	PL15, AN1IN47	TP04D

13.2.24 Port Ni Function Select Register (PNiS) (i = 00 to 05)

Address PN00S: H'FF46 4108, PN01S: H'FF46 4109, PN02S: H'FF46 410A,
PN03S: H'FF46 410B, PN04S: H'FF46 410C, PN05S: H'FF46 410D

	b7	b6	b5	b4	b3	b2	b1	b0
	PNiAS	—	—	PNiS[4:0]				
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	PNiAS	PN Analog Input Pin Select*1	0: Not used as an analog input pin 1: Used as an analog input pin*2	R/W
b6, b5	—	Reserved	These bits are read as 0. The write value should be 0.	R
b4 to b0	PNiS[4:0]	PN Function Select	b4 b3 b2 b1b0 0 0 0 0 0 : I/O port or A/D function Do not set other than above.	R/W

Notes: These registers are enabled when the PNFEN bit in the port function enable register 0 (PFEN0) is set to 1. When the PNFEN bit is set to 0, port N functions as an input port.
1. To use the A/D function, set this bit to 1. Otherwise, set it to 0.
2. When the PNPU0 bit in the pull-up control register 1 (PUR1) is set to “enabled” while pins are in use as analog input pins, the pull-up settings for the pins set as analog inputs become invalid.

The port Ni function select register is used to select the functions of the multiplexed pins of port N.

Register	PN Function Select Bits (PNiS)
	“00000”
PN00S	PN00, AN0IN00
PN01S	PN01, AN0IN01
PN02S	PN02, AN0IN02
PN03S	PN03, AN0IN03
PN04S	PN04, AN0IN04
PN05S	PN05, AN0IN05

13.2.25 Port Function Enable Register 0 (PFEN0)

Address H'FF46 4120

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	PNFEN	—	PLFEN	PKFEN	PJFEN	—	—	PGFEN	PFFEN	PEFEN	PDFEN	PCFEN	PBFEN	PAFEN
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, b14	—	Reserved	These bits are read as 0. The write value should be 0.	R
b13	PNFEN	PN Function Enable	0: PN00 to PN05 function as I/O ports 1: PN00 to PN05 functions selected by bits PN00S to PN05S are enabled	R/W
b12	—	Reserved	This bit is read as 0. The write value should be 0.	R
b11	PLFEN	PL Function Enable	0: PL10 to PL15 function as I/O ports 1: PL10 to PL15 functions selected by bits PL10S to PL15S are enabled *1	R/W
b10	PKFEN	PK Function Enable	0: PK00, PK01, and PK08 to PK15 function as I/O ports 1: PK00, PK01, and PK08 to PK15 functions selected by bits PK00S, PK01S, and PK08S to PK15S are enabled *2	R/W
b9	PJFEN	PJ Function Enable	0: PJ00 to PJ11 function as I/O ports 1: PJ00 to PJ11 functions selected by bits PJ00S to PJ11S are enabled *3	R/W
b8, b7	—	Reserved	These bits are read as 0. The write value should be 0.	R
b6	PGFEN	PG Function Enable	0: PG00 to PG03 and PG10 to PG13 function as I/O ports 1: PG00 to PG03 and PG10 to PG13 functions selected by bits PG00S to PG03S and PG10S to PG13S are enabled *4	R/W
b5	PFFEN	PF Function Enable	0: PF00 to PF02 and PF07 function as I/O ports 1: PF00 to PF02 and PF07 functions selected by bits PF00S to PF02S and PF07S are enabled *5	R/W
b4	PEFEN	PE Function Enable	0: PE00 to PE07 function as I/O ports 1: PE00 to PE07 functions selected by bits PE00S to PE07S are enabled	R/W
b3	PDFEN	PD Function Enable	0: PD00 to PD10 function as I/O ports 1: PD00 to PD10 functions selected by bits PD00S to PD10S are enabled *6	R/W
b2	PCFEN	PC Function Enable	0: PC08 to PC12 function as I/O ports 1: PC08 to PC12 functions selected by bits PC08S to PC12S are enabled *7	R/W
b1	PBFEN	PB Function Enable	0: PB12 to PB15 function as I/O ports 1: PB12 to PB15 functions selected by bits PB12S to PB15S are enabled *8	R/W
b0	PAFEN	PA Function Enable	0: PA00 to PA05 function as I/O ports 1: PA00 to PA05 functions selected by bits PA00S to PA05S are enabled *9	R/W

Notes: Set the PPR register to H'AA (write enabled) before rewriting this register.
1. Pins PL10 to PL15 are not available in the SH72A0 Group.
2. Pins PK09 to PK11 and PK13 are not available in the SH72A0 Group.
3. Pins PJ00 to PJ03, PJ06, PJ07, PJ10, and PJ11 are not available in the SH72A0 Group.
4. The PG13 pin is not available in the SH72A0 Group.
5. Pins PF00 to PF02 and PF07 are not available in the SH72A0 Group.

6. Pins PD08 to PD10 are not available in the SH72A0 Group.
7. Pins PC10 to PC12 are not available in the SH72A0 Group.
8. Pins PB12 to PB15 are not available in the SH72A0 Group.
9. The PA02 pin is not available in the SH72A0 Group.

Port function enable register 0 (PFEN0) is used to enable the functions selected by setting the port function select registers (PAiS to PGiS, PJiS to PLiS, and PNiS) (i = number of an applicable register from among 00 to 15).
The PFEN0 register is protected by the port protect register (PPR) from being written inadvertently.

13.2.26 POE0 Control Register (POE0CR)

Address H'FF46 41A0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	TIOC4DE	TIOC4CE	TIOC4BE	TIOC4AE	TIOC3DE	TIOC3BE	—	—	—	—	SMP[1:0]		—	POE0E
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, b14	—	Reserved	These bits are read as 0. The write value should be 0.	R
b13	TIOC4DE	TIOC4D Pin Control	0: TIOC4D pin is not changed by POE0 input 1: TIOC4D pin is set to the high-impedance state by POE0 input	R/W
b12	TIOC4CE	TIOC4C Pin Control	0: TIOC4C pin is not changed by POE0 input 1: TIOC4C pin is set to the high-impedance state by POE0 input	R/W
b11	TIOC4BE	TIOC4B Pin Control	0: TIOC4B pin is not changed by POE0 input 1: TIOC4B pin is set to the high-impedance state by POE0 input	R/W
b10	TIOC4AE	TIOC4A Pin Control	0: TIOC4A pin is not changed by POE0 input 1: TIOC4A pin is set to the high-impedance state by POE0 input	R/W
b9	TIOC3DE	TIOC3D Pin Control	0: TIOC3D pin is not changed by POE0 input 1: TIOC3D pin is set to the high-impedance state by POE0 input	R/W
b8	TIOC3BE	TIOC3B Pin Control	0: TIOC3B pin is not changed by POE0 input 1: TIOC3B pin is set to the high-impedance state by POE0 input	R/W
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R
b3, b2	SMP[1:0]	POE0 Pin Sampling Period Set	b3 b2 0 0 : No sampling period (falling edge) 0 1 : Low level is detected during 128 cycles of peripheral clock A 1 0 : Low level is detected during 256 cycles of peripheral clock A 1 1 : Low level is detected during 2048 cycles of peripheral clock A	R/W
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R
b0	POE0E	POE0 Function Enable	0: POE0 function disabled 1: POE0 function enabled	R/W

Note: Set the PPR register to H'AA (write enabled) before rewriting this register.

The POE0 control register (POE0CR) is used to set the MTU-III corresponding pin (TIOC3B, TIOC3D, TIOC4A, TIOC4B, TIOC4C, or TIOC4D) to the high-impedance state.
The POE0CR register is protected by the port protect register (PPR) from being written inadvertently.

13.2.27 POE1 Control Register (POE1CR)

Address H'FF46 41A2

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	TIOC7DE	TIOC7CE	TIOC7BE	TIOC7AE	TIOC6DE	TIOC6BE	—	—	—	—	SMP[1:0]		—	POE1E
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, b14	—	Reserved	These bits are read as 0. The write value should be 0.	R
b13	TIOC7DE	TIOC7D Pin Control	0: TIOC7D pin is not changed by POE1 input 1: TIOC7D pin is set to the high-impedance state by POE1 input	R/W
b12	TIOC7CE	TIOC7C Pin Control	0: TIOC7C pin is not changed by POE1 input 1: TIOC7C pin is set to the high-impedance state by POE1 input	R/W
b11	TIOC7BE	TIOC7B Pin Control	0: TIOC7B pin is not changed by POE1 input 1: TIOC7B pin is set to the high-impedance state by POE1 input	R/W
b10	TIOC7AE	TIOC7A Pin Control	0: TIOC7A pin is not changed by POE1 input 1: TIOC7A pin is set to the high-impedance state by POE1 input	R/W
b9	TIOC6DE	TIOC6D Pin Control	0: TIOC6D pin is not changed by POE1 input 1: TIOC6D pin is set to the high-impedance state by POE1 input	R/W
b8	TIOC6BE	TIOC6B Pin Control	0: TIOC6B pin is not changed by POE1 input 1: TIOC6B pin is set to the high-impedance state by POE1 input	R/W
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R
b3, b2	SMP[1:0]	POE1 Pin Sampling Period Set	b3 b2 0 0 : No sampling period (falling edge) 0 1 : Low level is detected during 128 cycles of peripheral clock A 1 0 : Low level is detected during 256 cycles of peripheral clock A 1 1 : Low level is detected during 2048 cycles of peripheral clock A	R/W
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R
b0	POE1E	POE1 Function Enable	0: POE1 function disabled 1: POE1 function enabled	R/W

Note: Set the PPR register to H'AA (write enabled) before rewriting this register.

The POE1 control register (POE1CR) is used to set the MTU-III corresponding pin (TIOC6B, TIOC6D, TIOC7A, TIOC7B, TIOC7C, or TIOC7D) to the high-impedance state.

The POE1CR register is protected by the port protect register (PPR) from being written inadvertently.

13.2.28 POE2 Control Register (POE2CR)

Address H'FF46 41A4

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	TIOC0DE	TIOC0CE	TIOC0BE	TIOC0AE	—	—	—	—	SMP[1:0]		—	POE2E
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 b12	—	Reserved	These bits are read as 0. The write value should be 0.	R
b11	TIOC0DE	TIOC0D Pin Function	0: TIOC0D pin is not changed by POE2 input 1: TIOC0D pin is set to the high-impedance state by POE2 input *	R/W
b10	TIOC0CE	TIOC0C Pin Function	0: TIOC0C pin is not changed by POE2 input 1: TIOC0C pin is set to the high-impedance state by POE2 input	R/W
b9	TIOC0BE	TIOC0B Pin Function	0: TIOC0B pin is not changed by POE2 input 1: TIOC0B pin is set to the high-impedance state by POE2 input	R/W
b8	TIOC0AE	TIOC0A Pin Function	0: TIOC0A pin is not changed by POE2 input 1: TIOC0A pin is set to the high-impedance state by POE2 input	R/W
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R
b3, b2	SMP[1:0]	POE2 Pin Sampling Period Set	b3 b2 0 0 : No sampling period (falling edge) 0 1 : Low level is detected during 128 cycles of peripheral clock A 1 0 : Low level is detected during 256 cycles of peripheral clock A 1 1 : Low level is detected during 2048 cycles of peripheral clock A	R/W
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R
b0	POE2E	POE2 Function Enable	0: POE2 function disabled 1: POE2 function enabled	R/W

Notes: Set the PPR register to H'AA (write enabled) before rewriting this register.
* The TIOC0DE bit is disabled in the SH72A0 Group.

The POE2 control register (POE2CR) is used to set the MTU-III corresponding pin (TIOC0A, TIOC0B, TIOC0C, or TIOC0D) to the high-impedance state.

The POE2CR register is protected by the port protect register (PPR) from being written inadvertently.

13.2.29 POE Monitor Register (POEM)

Address H'FF46 41A6

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	POE2M	POE1M	POE0M
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R
b2	POE2M	POE2 Monitor	0: No POE2 input 1: POE2 input	R/W
b1	POE1M	POE1 Monitor	0: No POE1 input 1: POE1 input	R/W
b0	POE0M	POE0 Monitor	0: No POE0 input 1: POE0 input	R/W

Note: Set the PPR register to H'AA (write enabled) before rewriting this register.

The POE monitor register (POEM) is used to monitor a POEi input which meets the condition set by the SMP bits in the POEiCR register.
The POEM register is protected by the port protect register (PPR) from being written inadvertently.

POEiM Bit

This bit is enabled when the corresponding POEiE bit in the POEiCR register is set to 1 (POEi function enabled).
This bit is set to 1 when there is a POEi input which meets the condition set by the SMP bits in the POEiCR register. At this time, the pin selected as high-impedance state by the POEiCR register goes to the high-impedance state.
To clear this bit to 0 from 1, write 0 to this bit or write 0 to the corresponding POEiE bit. After writing 0, read the POEiM bit to check that the bit has been cleared to 0. If 1 is read from the bit, write 0 to it again.
When 1 is written to this bit while the bit is 0, the corresponding pin goes to high impedance. However, this bit does not change (and the result of the POEi input is retained). No interrupt is generated.

13.3 POEi Interrupt

A POEi interrupt is generated when a valid signal input to the POEi# pin is detected. This interrupt is used to release the pin from the high-impedance state after the pin has become a high-impedance state through the POEi input.

[Interrupt Request Generation Conditions]

- When the SMP bits in the POEiCR register is set to 00 (falling edge setting) and the POEiE bit is set to 1 (POEi function enabled), if a falling edge is input to the POEi# pin while the POEiM bit in the POE monitor register is 0, an interrupt request is generated and the corresponding IR bit in the interrupt request register for the INTC is set to 1. At this time, the POEiM bit is set to 1.
- When the SMP bits in the POEiCR register are set to a value other than 00 and the POEiE bit is set to 1 (POEi function enabled), if a low-level signal is input to the POEi# pin during the period set by the SMP bits while the POEiM bit in the POE monitor register is 0, an interrupt request is generated and the corresponding IR bit in the interrupt request register for the INTC is set to 1. At this time, the POEiM bit is also set to 1.

[Interrupt Request Clearing Conditions]

The IR bit in the interrupt request register for the INTC is automatically cleared when the interrupt is accepted. However, the next interrupt request is not generated while the POEiM bit is 1, thus clear this bit to 0.
To clear the POEiM bit to 0 from 1, write 0 to this bit or write 0 to the POEiE bit. After writing 0, read the POEiM bit to check that the bit has been cleared to 0. If 1 is read from the bit, write 0 to it again.
When the POEiM bit is cleared to 0, the high-impedance state of the corresponding pin is released.

13.4 Selection of I/O Port Function

When the Pi function enable bit in the PFEN0 register is set to 0, the corresponding port functions as an I/O port.
If the function select bits (b4 to b0) in the port function select register for the corresponding port are set to 00000 when the Pi function enable bit in the PFEN0 register is set to 1, the corresponding port functions as an I/O port.

13.4.1 Input Ports

When an I/O port is selected by setting the PFEN0 register and the corresponding port function select register, and input is selected by the PiDR register, the corresponding port functions as an input port.
The level of the pin can be read by reading the port i register. When the port is used as a peripheral function, the level of the pin can also be read by reading the port i register.

13.4.2 Output Ports

When an I/O port is selected by setting the PFEN0 register and the corresponding port function select register, and output is selected by the PiDR register, the corresponding port functions as an output port.
The value written to the port i register is output. When the port i register is read while the PRFS bit in the PFS0 register is 0, the setting value of the port i register can be read. When the port i register is read while the PRFS bit in the PFS0 register is 1, the input level of the pin can be read.

13.5 Selection of Peripheral Functions

When the Pi function enable bit in the PFEN0 register is set to 1, a peripheral function is selected by setting the corresponding port function select register.

13.5.1 Selection of Input Pins Multiplexed at Multiple Sites

The input pins for the peripheral function that are multiplexed at multiple sites can be selected by setting registers PFS1 to PFS3.

13.5.2 Selection of Output Pins Multiplexed at Multiple Sites

The output pins for the peripheral function that are multiplexed at multiple sites are set to the output of the peripheral function, thus outputs from multiple pins are enabled.

13.5.3 Selection of Interrupt and A/D Converter Input Functions

To select the input function of NMI, INT0 to INT13, POE0# to POE2#, ADTRG#, AN0IN00 to AN0IN05, AN1IN00 to AN1IN07, AN1IN16, AN1IN17, AN1IN24 to AN1IN31, or AN1IN42 to AN1IN47*, set the corresponding bit in the PiDR register to 0 (input). For AN0IN00 to AN0IN05, AN1IN00 to AN1IN07, AN1IN16, AN1IN17, AN1IN24 to AN1IN31, and AN1IN42 to AN1IN47*, set the corresponding analog input pin select bit in the port function select register to 1.

Analog input pin select bits are used to prevent the power consumption of the input buffer from increasing due to the midpoint potential generated at analog input.

Note: * Pins INT9, INT11 to INT13, ADTRG#, AN1IN00 to AN1IN03, AN1IN06, AN1IN07, AN1IN25 to AN1IN27, AN1IN29, and AN1IN42 to AN1IN47 are not available in the SH72A0 Group.

13.5.4 Selection of CLKOUT Functions

To select the CLKOUT function, set the PD00S bits in the port D00 function select register (PD00S) to 00000, and set the COE bit in the COCR register to 1.

13.6 Other Functions

Regardless of the settings of ports or peripheral functions, the setting of internal pull-up resistors (for applicable pins per port), and the switching of input threshold values (for applicable pins per port).

Note: Internal pull-up resistor settings are invalid for pins in use as analog inputs.

13.7 Note on the I/O Ports

13.7.1 Setting the Port Pni Function Select Register (PniS) (n = A, B, C, D, E, F, G, J, K, L, N; i = number of an applicable IO port pin)

Registers for setting the functions of Pni pins apply control over whether each of the pins has an input or output function.

When a pin is set to operate as an input, the levels on the pin before and after the change not being the same may lead to the generation of an edge in the signal and in turn to unintended operations. Accordingly, follow the procedure below when setting up a pin to operate as an input pin.

1. Disable the input function (or operation) of the pin in the settings for the peripheral module handling the currently selected pin function.
2. Set the input threshold value register for the given pin to a value other than “input disabled”.
3. Set the function select register for the corresponding Pni to select the input function for the pin.
4. Enable the input function (or operation) in the settings of the peripheral module that works with the current pin function.

When an output function is to be set for the pin, the pin settings before and after the change being different may lead to the internal generation of an edge and in turn to unintended operations. Accordingly, follow the procedure below when setting up a pin to operate as an output pin.

1. Disable the output function (or operation) of the pin in the settings for the peripheral module handling the currently selected pin function.
2. Set the function select register for the corresponding Pni to select the output function for the pin.
3. Enable the output function (or operation) in the settings of the peripheral module that works with the current pin function.

14. Compare Match Timer (CMT)

14.1 Introduction

This LSI has three units (the unit 0, the unit 1, and the unit 2) of the compare match timer (CMT) that comprise two 16-bit timer channels, which means that it has six channels in total. The CMT has a 16-bit counter, and can generate interrupt requests at set intervals.

Table 14.1 lists CMT Specifications and Figure 14.1 shows CMT Block Diagram in Unit 0.

Table 14.1 CMT Specifications

Item	Description
Count source	The followings can be selected for each channel. • Peripheral bus clock A divided by 8 • Peripheral bus clock A divided by 32 • Peripheral bus clock A divided by 128 • Peripheral bus clock A divided by 512 • CMT count source origin (Only the CMT0 can select this source origin.)
Interrupt request generation timing	Compare match: the value of the CMiCNT counter matches the value of the CMiCMSE register.

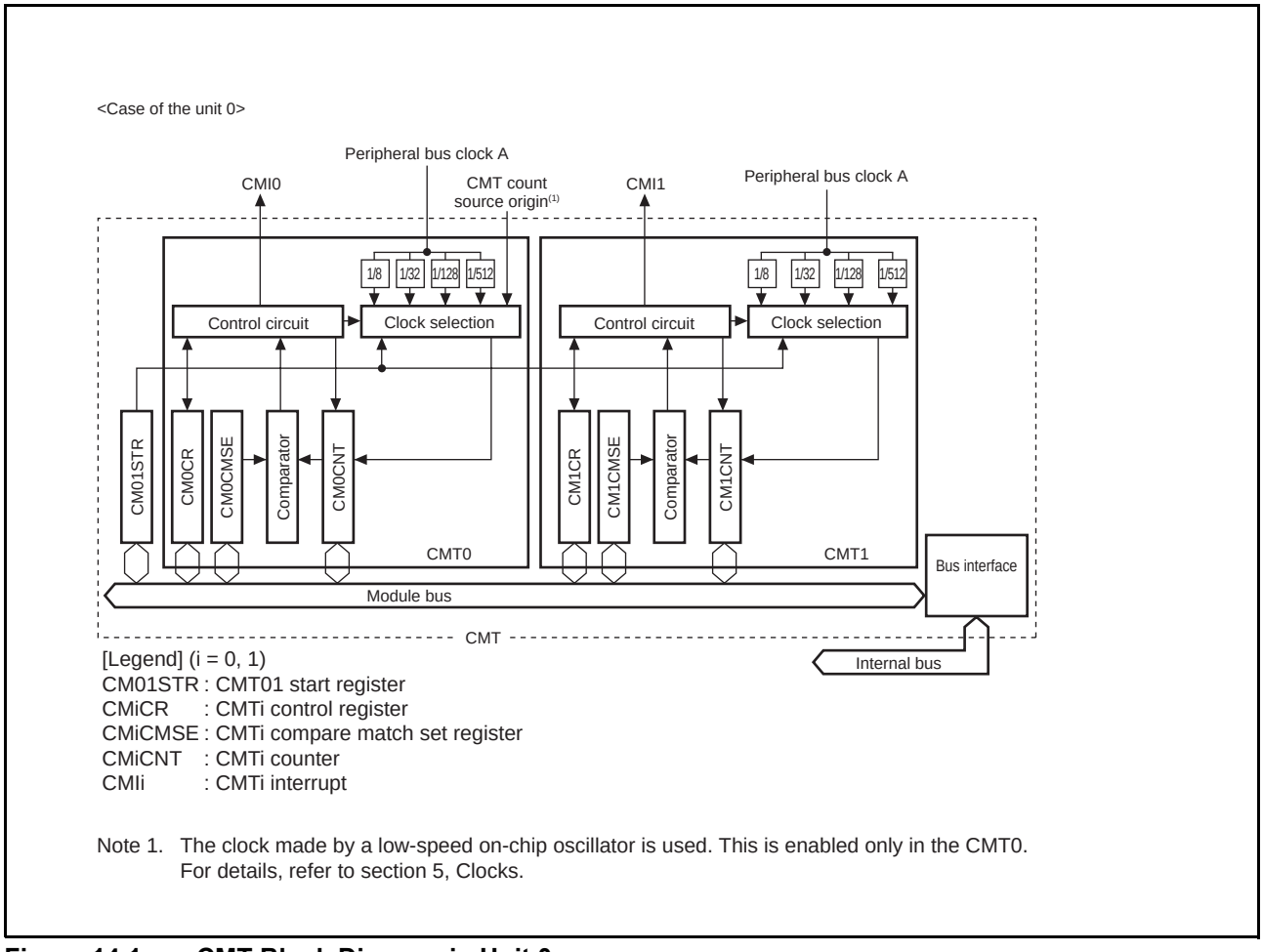


Figure 14.1 CMT Block Diagram in Unit 0

14.2 Registers

Table 14.2 lists Registers.

Table 14.2 Registers

Unit	Module	Register Name	Symbol	After Reset	Address	Access Size
Unit 0	Common	CMT01 start register	CM01STR	H'0000	H'FFFE C000	16
	CMT0	CMT0 control register	CM0CR	H'0000	H'FFFE C002	16
		CMT0 counter	CM0CNT	H'0000	H'FFFE C004	16
		CMT0 compare match set register	CM0CMSE	H'FFFF	H'FFFE C006	16
	CMT1	CMT1 control register	CM1CR	H'0000	H'FFFE C008	16
		CMT1 counter	CM1CNT	H'0000	H'FFFE C00A	16
		CMT1 compare match set register	CM1CMSE	H'FFFF	H'FFFE C00C	16
Unit 1	Common	CMT23 start register	CM23STR	H'0000	H'FFFE C010	16
	CMT2	CMT2 control register	CM2CR	H'0000	H'FFFE C012	16
		CMT2 counter	CM2CNT	H'0000	H'FFFE C014	16
		CMT2 compare match set register	CM2CMSE	H'FFFF	H'FFFE C016	16
	CMT3	CMT3 control register	CM3CR	H'0000	H'FFFE C018	16
		CMT3 counter	CM3CNT	H'0000	H'FFFE C01A	16
		CMT3 compare match set register	CM3CMSE	H'FFFF	H'FFFE C01C	16
Unit 2	Common	CMT45 start register	CM45STR	H'0000	H'FFFE C020	16
	CMT4	CMT4 control register	CM4CR	H'0000	H'FFFE C022	16
		CMT4 counter	CM4CNT	H'0000	H'FFFE C024	16
		CMT4 compare match set register	CM4CMSE	H'FFFF	H'FFFE C026	16
	CMT5	CMT5 control register	CM5CR	H'0000	H'FFFE C028	16
		CMT5 counter	CM5CNT	H'0000	H'FFFE C02A	16
		CMT5 compare match set register	CM5CMSE	H'FFFF	H'FFFE C02C	16

14.2.1 CMT01 Start Register (CM01STR)

Address H'FFFE C000

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STR1	STR0
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 b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b1	STR1	Count Start 1 Bit	0: Stops count operation of CM1CNT counter 1: Starts count operation of CM1CNT counter	R/W
b0	STR0	Count Start 0 Bit	0: Stops count operation of CM0CNT counter 1: Starts count operation of CM0CNT counter	R/W

14.2.2 CMT23 Start Register (CM23STR)

Address H'FFFE C010

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STR3	STR2
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 b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b1	STR3	Count Start 3 Bit	0: Stops count operation of CM3CNT counter 1: Starts count operation of CM3CNT counter	R/W
b0	STR2	Count Start 2 Bit	0: Stops count operation of CM2CNT counter 1: Starts count operation of CM2CNT counter	R/W

14.2.3 CMT45 Start Register (CM45STR)

Address H'FFFE C020

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STR5	STR4
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 b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b1	STR5	Count Start 5 Bit	0: Stops count operation of CM5CNT counter 1: Starts count operation of CM5CNT counter	R/W
b0	STR4	Count Start 4 Bit	0: Stops count operation of CM4CNT counter 1: Starts count operation of CM4CNT counter	R/W

14.2.4 CMTi Control Register (CMiCR) (i = 0 to 5)

Address CM0CR: H'FFFE C002, CM1CR: H'FFFE C008, CM2CR: H'FFFE C012, CM3CR: H'FFFE C018, CM4CR: H'FFFE C022, CM5CR: H'FFFE C028

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	CMIE	—	—	—	LOCO	CKS[1:0]	
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 b8	—	Reserved	These bits are read as 0. The write value should be 0.	R
b7	—	Reserved	This bit is read as an undefined value. The write value should be 1.	R
b6	CMIE	CMTi Interrupt Enable Bit	0: CMTi interrupt (CMIi) is disabled. 1: CMTi interrupt (CMIi) is enabled.	R/W
b5 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R
b2	LOCO	LOCO Select Bit*	0: Disabled 1: CMT count source origin is selected.	R/W
b1, b0	CKS[1:0]	Clock Select Bits	b1 b0 0 0 : Peripheral bus clock A divided by 8 0 1 : Peripheral bus clock A divided by 32 1 0 : Peripheral bus clock A divided by 128 1 1 : Peripheral bus clock A divided by 512	R/W

Note: * This bit is enabled only in the CMT0. In the CMT1 to the CMT5, this bit is a reserved bit. (This bit is read as 0. The write value should be 0.)

CMIE Bit

This bit selects whether to enable/disable generate the CMTi interrupts (CMIi) when the value in the CMiCNT matches the value of the CMiCMSE.

LOCO Bit

This bit is enabled only in the CMT0.

When this bit is 1, the clock origin of the CMT count source made by a low-speed on-chip oscillator is selected as the count source. On the setting for the clock origin of the CMT count source, refer to section 5, Clocks.

When this bit is 0, the count source set in the CKS bit is selected as the count source.

CKS Bit

This bit selects the count source.

14.2.5 CMTi Counter (CMiCNT) (i = 0 to 5)

Address: CM0CNT: H'FFFE C004, CM1CNT: H'FFFE C00A, CM2CNT: H'FFFE C014, CM3CNT: H'FFFE C01A, CM4CNT: H'FFFE C024, CM5CNT: H'FFFE C02A

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Description	R/W
b15 to b0	Up counter for generating interrupt requests	R/W

When the STRi bit in registers CM01STR, CM23STR, or CM45STR is set to 1, the CMiCNT counter starts the count up operation.
When the value in the CMiCNT counter matches the value of the CMiCMSE register, the CMiCNT counter is cleared to H'0000 and then the CMTi interrupt request (CMIi) is generated. (i = 0 to 5)

14.2.6 CMTi Compare Match Set Register (CMiCMSE) (i = 0 to 5)

Address: CM0CMSE: H'FFFE C006, CM1CMSE: H'FFFE C00C, CM2CMSE: H'FFFE C016, CM3CMSE: H'FFFE C01C, CM4CMSE: H'FFFE C026, CM5CMSE: H'FFFE C02C

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit	Description	R/W
b15 to b0	Compare match interval with CMiCNT counter is set. (i= 0 to 5)	R/W

14.3 Operations

14.3.1 Interval Count Operation

When a count source is selected with the CKS bit in the CMiCR register and the STRi bit in registers CM01STR, CM23STR, and CM45STR is set to 1, the CMiCNT counter starts counting up. (Only the CMT0 can select the CMT count source origin with the LOSC bit.)

When the value in the CMiCNT counter matches the value of the CMiCMSE register, the CMiCNT counter is cleared to H'0000 and then the CMTi interrupt request (CMIi) is generated. Then the CMiCNT counter starts counting up again from H'0000.

Figure 14.2 shows the CMiCNT counter operation. (i = 0 to 5)

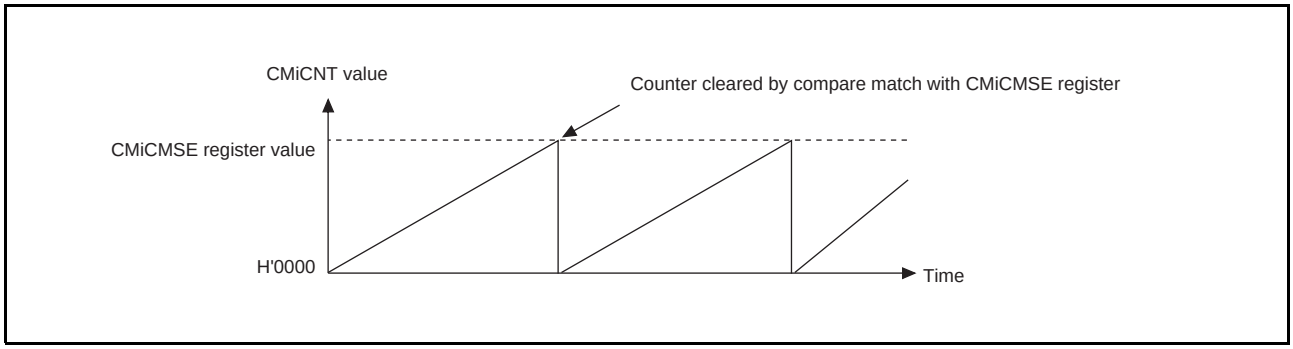


Figure 14.2 Counter Operation

14.3.2 CMiCNT Counter Count Timing

Four count sources obtained by dividing the peripheral bus clock A can be selected with the CKS bit in the CMiCR register. (Only the CMT0 can select the CMT count source origin with the LOSC bit.)

Figure 14.3 shows the CMiCNT counter count timing. (i = 0 to 5)

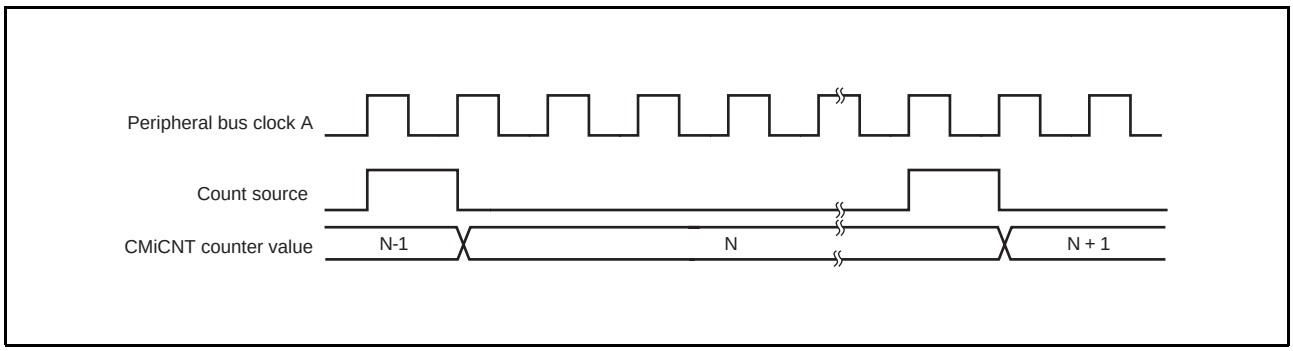


Figure 14.3 Count Timing

14.4 Interrupts

14.4.1 Interrupt Sources

The CMT has channels and each of them to which a different vector address is allocated has the CMTi interrupt (CMIi). When the CMTi interrupt is generated, the corresponding interrupt request is output. When the interrupt request is used to activate a CPU interrupt, the priority of channels can be changed by the interrupt controller settings. For details, see section 8, Interrupt Controller (INTC).

Table 14.3 CMT Interrupt Sources

Name	Interrupt Source	Interrupt Flag	DMAC Activation
CMT0 interrupt (CMI0)	Compare match between CM0CNT and CM0CMSE	IR142	Possible
CMT1 interrupt (CMI1)	Compare match between CM1CNT and CM1CMSE	IR143	Possible
CMT2 interrupt (CMI2)	Compare match between CM2CNT and CM2CMSE	IR144	Possible
CMT3 interrupt (CMI3)	Compare match between CM3CNT and CM3CMSE	IR145	Possible
CMT4 interrupt (CMI4)	Compare match between CM4CNT and CM4CMSE	IR146	Possible
CMT5 interrupt (CMI5)	Compare match between CM5CNT and CM5CMSE	IR147	Possible

14.4.2 CMTi Interrupt Generation Timing

When the value in the CMiCNT counter matches the value of the CMiCMSE register, the CMTi interrupt (CMIi) is generated. The compare match signal is generated in the last state in which the values are matched (at the time the CMiCNT counter is updated with the matched count value). That is, after the value in the CMiCNT counter matches the value of the CMiCMSE register, the compare match signal (an internal signal) is not generated until the next count source of the CMiCNT counter is input. Figure 14.4 shows Timing of CMTi Interrupt Request Output. (i = 0 to 5)

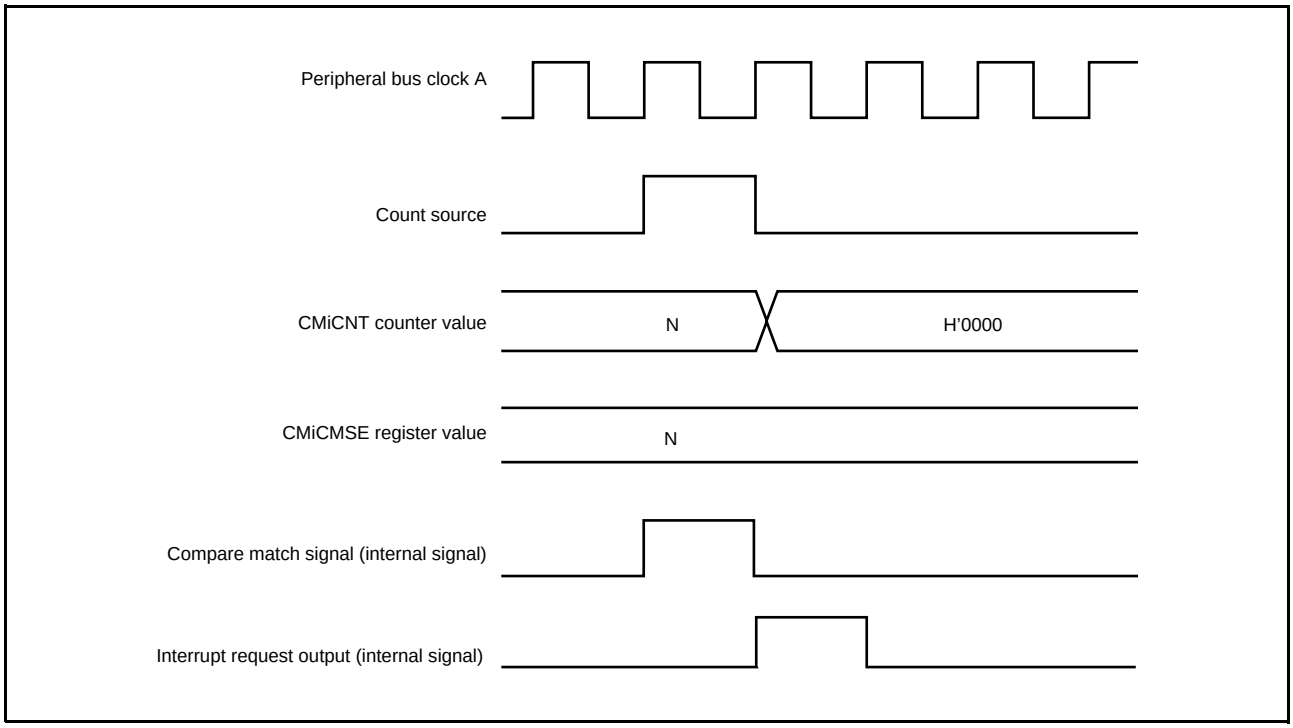


Figure 14.4 Timing of CMTi Interrupt Request Output

14.5 Notes on Compare Match Timer (CMT)

Take note of the following conflicts or operations to be generated when the CMT is activated.

14.5.1 Conflict between Write to CMiCNT Counter and Compare Match

When the compare match signal is generated while writing to the CMiCNT counter, clearing the CMiCNT counter has priority over writing to it. Then the write to the CMiCNT counter is not performed. (i = 0 to 5)
Figure 14.5 shows Conflict between Write to CMiCNT Counter and Compare Match.

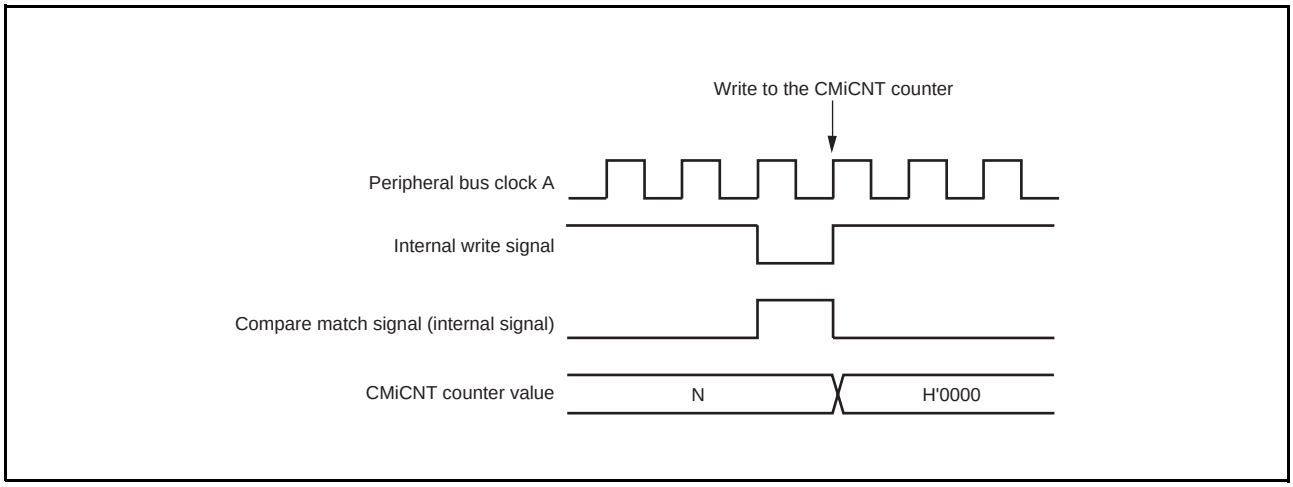


Figure 14.5 Conflict between Write to CMiCNT Counter and Compare Match

14.5.2 Conflict between Write to CMiCNT Counter and Count-up

Even when a count-up occurs while writing to the CMiCNT counter, the write to the CMiCNT counter has priority over the count-up and the count-up is not performed. (i = 0 to 5)
Figure 14.6 shows Conflict between Write to CMiCNT Counter and Count-up.

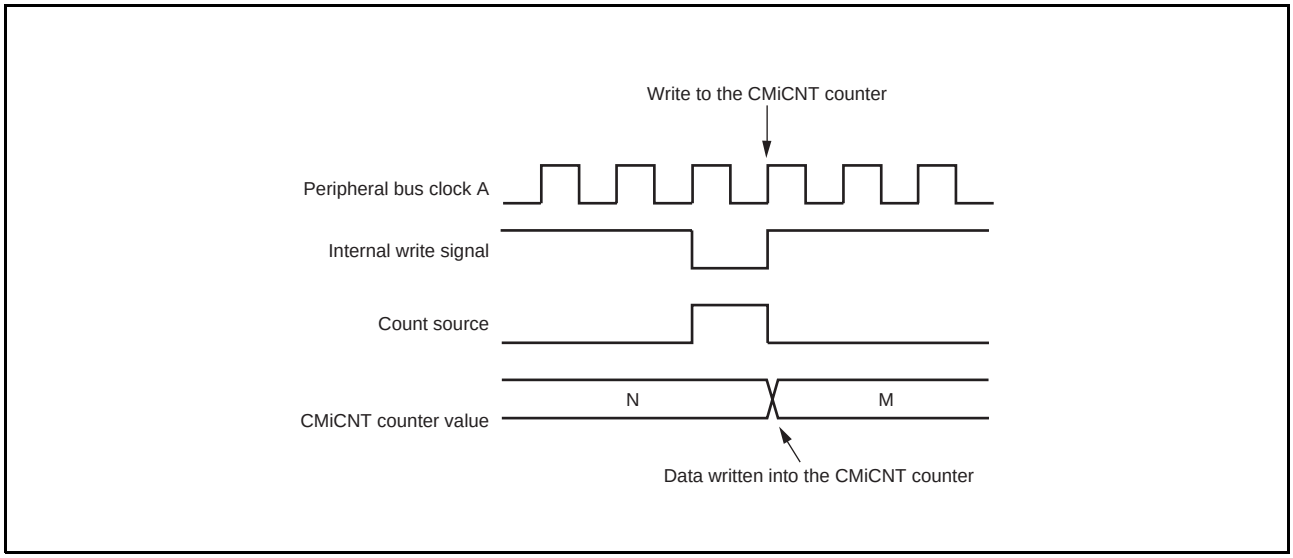


Figure 14.6 Conflict between Write to CMiCNT Counter and Count-up

14.5.3 Notes on Write Operation for CMiCR

- Any write operation for the CMiCR register must be performed before the CMT is started (when the CMT has stopped).
- You can change the CKS1 and CKS0 bits of the CMiCR register while the CMT is operating. For details, see Figure 14.7, Example of Procedure for Writing to CMiCR Register.
- You can also change the CMIE bit of the CMiCR register while the CMT is operating. To do this, use the INTEN bit of the ICRi register. For details about the INTEN bit, see section 8, Interrupt Controller (INTC).

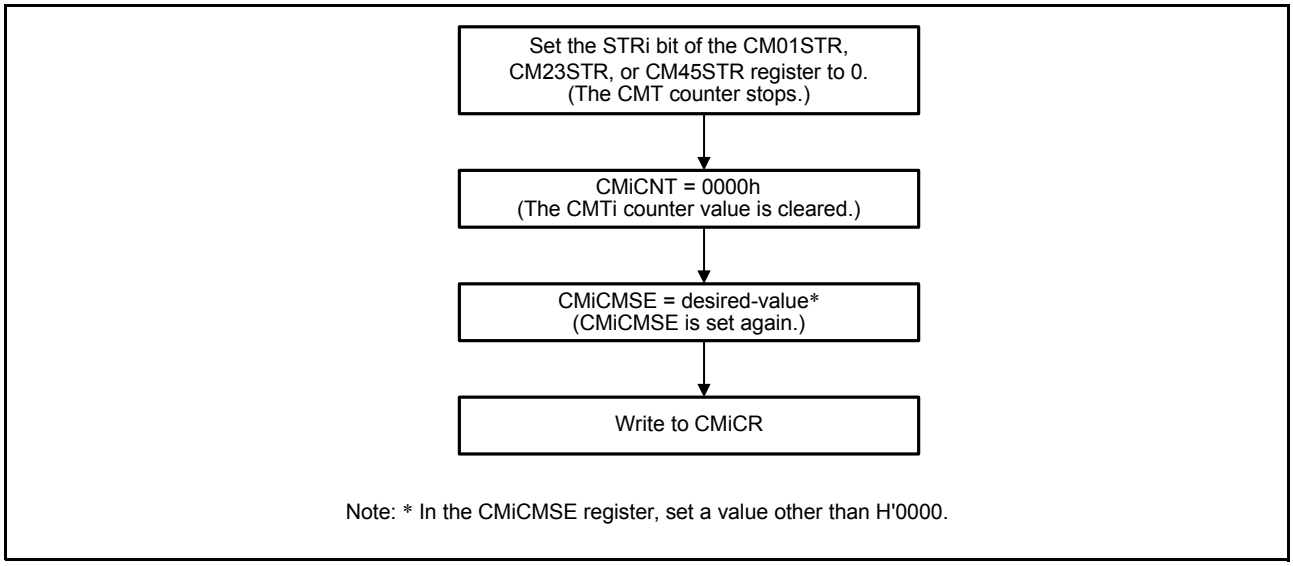


Figure 14.7 Example of Procedure for Writing to CMiCR Register

14.5.4 Notes on Count Clearance and Interrupt Generation when Counting is Stopped

When counting is stopped by setting the STRi bit of the CM01STR, CM23STR, or CM45STR register to 0, if the value of the stopped CMiCNT and the value of the CMiCMSE register match, count clearance and an interrupt occur the next time the count source is input.

If you do not want an interrupt to occur after stopping counting, disable generation of CMT interrupts by setting the INTEN bit of the ICRi register to 0 before setting the CM01STR, CM23STR, or CM45STR register.

In addition, do not set the same value for both CMiCNT and the CMiCMSE register when the counting has stopped. If such setting is attempted, count clearance and an interrupt occur the next time the count source is input.

15. Timer Pulse Unit (TPU)

This LSI has four 16-bit timer pulse units (TPU1 to TPU4). Units 2 and 3 (TPU2 and TPU3) support cascade connection. This chapter describes an LSI which has 16 timer pulse units.

15.1 Introduction

Table 15.1 lists TPU Functions and Channels and Table 15.2 lists TPU Functions.

Table 15.1 TPU Functions and Channels

Function			Channel
Input capture			4 channels/UNIT
	Digital filter function		4 channels/UNIT
Output compare			4 channels/UNIT
	Single-phase waveform output mode		4 channels/UNIT
	PWM mode	PWM mode 1	2 channels/UNIT
		PWM mode 2	4 channels/UNIT

Table 15.2 TPU Functions

Item	Description	
	Unit i (i = 0, 2, 4, 6, 8, 10, 12, or 14)	Unit i (i = 1, 3, 5, 7, 9, 11, 13, or 15)
Count sources	Selected by the CSS bit in the TPiCR register b2 b1 b0 0 0 0 : Peripheral bus clock A is not divided. 0 0 1 : Peripheral bus clock A divided by 4 0 1 0 : Peripheral bus clock A divided by 16 0 1 1 : Peripheral bus clock A divided by 64 1 0 0 : Peripheral bus clock A divided by 256 1 0 1 : Peripheral bus clock A divided by 1024 1 1 0 : Peripheral bus clock A divided by 4096 1 1 1 : Unit i + 1 overflow	Selected by the CSS bit in the TPiCR register b2 b1 b0 0 0 0 : Peripheral bus clock A is not divided. 0 0 1 : Peripheral bus clock A divided by 4 0 1 0 : Peripheral bus clock A divided by 16 0 1 1 : Peripheral bus clock A divided by 64 1 0 0 : Peripheral bus clock A divided by 256 1 0 1 : Peripheral bus clock A divided by 1024 1 1 0 : Peripheral bus clock A divided by 4096 1 1 1 : Setting prohibited
General registers	TPiGR0, TPiGR1, TPiGR2, TPiGR3	
I/O pins	TPjA, TPjB, TPjC, TPjD *	
Output level	Low level, High level, Toggle Duty 0% or 100% output is available	
TPiCNT counter reset sources	Compare match, Input capture	
Compare match	Low level output, High level output, Toggle output	
PWM mode	Available	
Phase shift mode	Available	
Input capture function	Available	
Buffer function	Available	
One-shot output mode	4 channels	
A/D conversion timing generator	4 channels: TPiCNT counter reset of each unit	
Interrupt sources	6 sources/unit • Compare match/input capture of each channel TPiGR0, TPiGR1, TPiGR2, TPiGR3 • TPiCNT counter overflow • TPiCNT counter reset of each unit	
DMA sources	5 sources/unit • Compare match/input capture of each channel TPiGR0, TPiGR1, TPiGR2, TPiGR3 • TPiCNT counter reset of each unit	

Note: * j = 00 to 15

Figure 15.1 and Figure 15.2 show the TPU block diagrams. For details on the internal unit, refer to Figure 15.2.

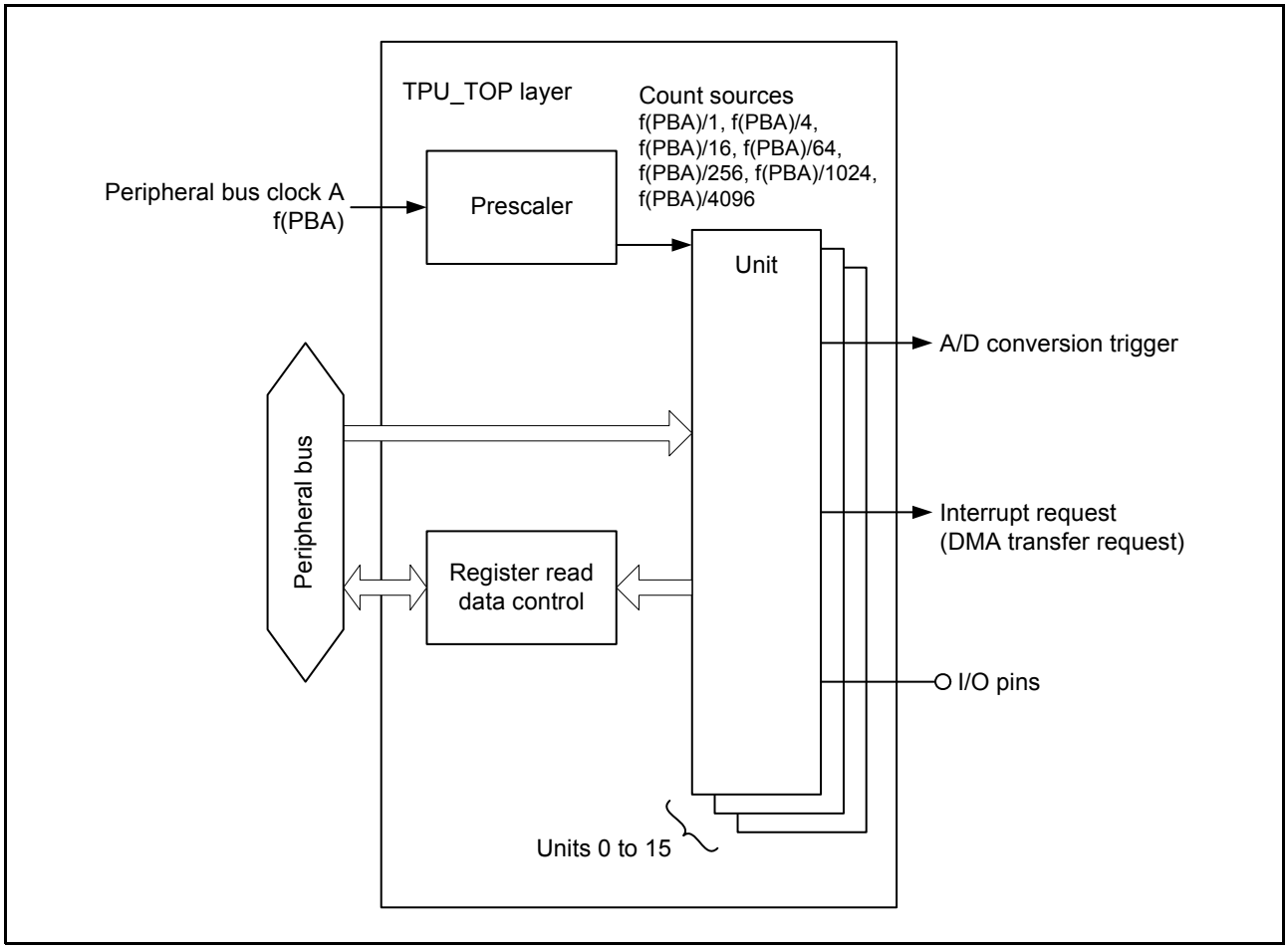
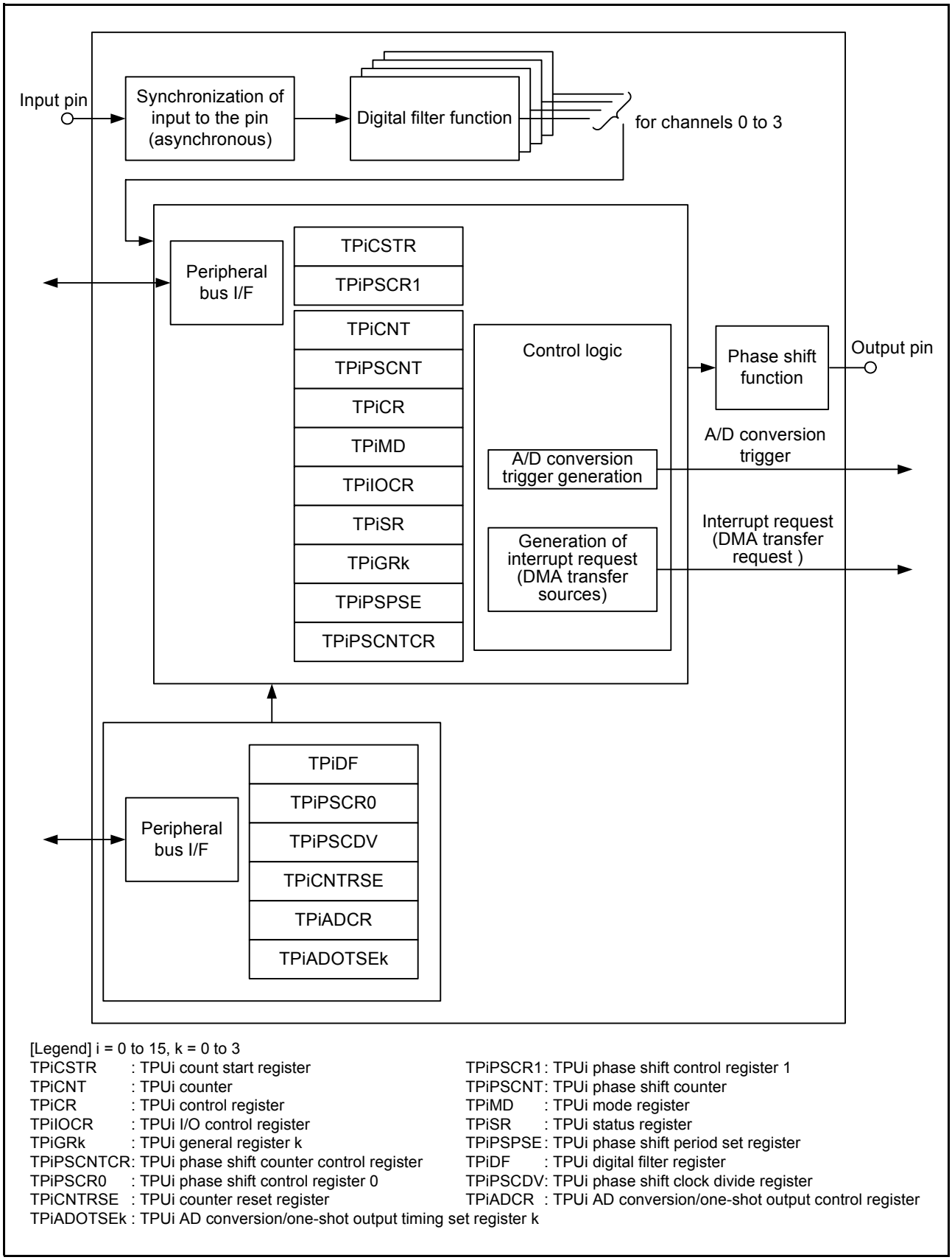


Figure 15.1 TPU Block Diagram (1)



[Legend] i = 0 to 15, k = 0 to 3

TPiCSTR	: TPUi count start register	TPiPSCR1	: TPUi phase shift control register 1
TPiCNT	: TPUi counter	TPiPSCNT	: TPUi phase shift counter
TPiCR	: TPUi control register	TPiMD	: TPUi mode register
TPiIOCR	: TPUi I/O control register	TPiSR	: TPUi status register
TPiGRk	: TPUi general register k	TPiPSPSE	: TPUi phase shift period set register
TPiPSCNTCR	: TPUi phase shift counter control register	TPiDF	: TPUi digital filter register
TPiPSCR0	: TPUi phase shift control register 0	TPiPSCDV	: TPUi phase shift clock divide register
TPiCNTRSE	: TPUi counter reset register	TPiADCR	: TPUi AD conversion/one-shot output control register
TPiADOTSEk	: TPUi AD conversion/one-shot output timing set register k		

Figure 15.2 TPU Block Diagram (2)

Table 15.3 lists TPU I/O Pins.

Table 15.3 TPU I/O Pins

Pin Name	I/O	Description
TP00A to TP00D	I/O	Pins in unit 0 for input capture input/output compare output/PWM output
TP01A to TP01D*1	I/O	Pins in unit 1 for input capture input/output compare output/PWM output
TP02A to TP02D*2	I/O	Pins in unit 2 for input capture input/output compare output/PWM output
TP03A to TP03D	I/O	Pins in unit 3 for input capture input/output compare output/PWM output
TP04A to TP04D*3	I/O	Pins in unit 4 for input capture input/output compare output/PWM output
TP05A to TP05D	I/O	Pins in unit 5 for input capture input/output compare output/PWM output
TP06A to TP06D	I/O	Pins in unit 6 for input capture input/output compare output/PWM output
TP07A to TP07D	I/O	Pins in unit 7 for input capture input/output compare output/PWM output
TP08A to TP08D	I/O	Pins in unit 8 for input capture input/output compare output/PWM output
TP09A to TP09D	I/O	Pins in unit 9 for input capture input/output compare output/PWM output
TP10A to TP10D	I/O	Pins in unit 10 for input capture input/output compare output/PWM output
TP11A to TP11D	I/O	Pins in unit 11 for input capture input/output compare output/PWM output
TP12A to TP12D	I/O	Pins in unit 12 for input capture input/output compare output/PWM output
TP13A to TP13D	I/O	Pins in unit 13 for input capture input/output compare output/PWM output
TP14A to TP14D	I/O	Pins in unit 14 for input capture input/output compare output/PWM output
TP15A to TP15D	I/O	Pins in unit 15 for input capture input/output compare output/PWM output

Notes: 1. The TP01D pin cannot be used in the SH72A0 Group.
2. The TP02A and TP02B pins cannot be used in the SH72A0 Group.
3. The TP04C and TP04D pins cannot be used in the SH72A0 Group.

Note: • In this chapter the TPU I/O pins in the unit i and the channel k are described below as TPjx.

		Pin Name TPjx			
Unit i (i = 0 to 15)	k = 0 to 3 x = A, B, C, or D j = 00 to 15	Channel 0 k = 0 x = A	Channel 1 k = 1 x = B	Channel 2 k = 2 x = C	Channel 3 k = 3 x = D
i = 0	j = 00	TP00A	TP00B	TP00C	TP00D
i = 1	j = 01	TP01A	TP01B	TP01C	TP01D
i = 2	j = 02	TP02A	TP02B	TP02C	TP02D
i = 3	j = 03	TP03A	TP03B	TP03C	TP03D
i = 4	j = 04	TP04A	TP04B	TP04C	TP04D
i = 5	j = 05	TP05A	TP05B	TP05C	TP05D
i = 6	j = 06	TP06A	TP06B	TP06C	TP06D
i = 7	j = 07	TP07A	TP07B	TP07C	TP07D
i = 8	j = 08	TP08A	TP08B	TP08C	TP08D
i = 9	j = 09	TP09A	TP09B	TP09C	TP09D
i = 10	j = 10	TP10A	TP10B	TP10C	TP10D
i = 11	j = 11	TP11A	TP11B	TP11C	TP11D
i = 12	j = 12	TP12A	TP12B	TP12C	TP12D
i = 13	j = 13	TP13A	TP13B	TP13C	TP13D
i = 14	j = 14	TP14A	TP14B	TP14C	TP14D
i = 15	j = 15	TP15A	TP15B	TP15C	TP15D

15.2 Registers

Table 15.4 to Table 15.12 list the TPU registers.

Table 15.4 TPU Registers (1)

Unit	Register Name	Symbol	After Reset	Address	Access Size
Unit 0	TPU0 control register	TP0CR	H'00	H'FFFE 4000	8
	TPU0 mode register	TP0MD	H'C0	H'FFFE 4001	8
	TPU0 status register	TP0SR	H'C0	H'FFFE 4003	8
	TPU0 counter	TP0CNT	H'0000	H'FFFE 4004	16
	TPU0 general register 0	TP0GR0	H'FFFF	H'FFFE 4006	16
	TPU0 general register 1	TP0GR1	H'FFFF	H'FFFE 4008	16
	TPU0 general register 2	TP0GR2	H'FFFF	H'FFFE 400A	16
	TPU0 general register 3	TP0GR3	H'FFFF	H'FFFE 400C	16
	TPU0 count start register	TP0CSTR	H'00	H'FFFE 4050	8
	TPU0 I/O control register	TP0IOCR	H'0000	H'FFFE 400E	8, 16
	TPU0 digital filter register	TP0DF	H'00	H'FFFE 4032	8
	TPU0 AD conversion/one-shot output control register	TP0ADCR	H'00	H'FFFE 4034	8
	TPU0 AD conversion/one-shot output timing set register 0	TP0ADOTSE0	H'0000	H'FFFE 4036	8, 16
	TPU0 AD conversion/one-shot output timing set register 1	TP0ADOTSE1	H'0000	H'FFFE 4038	8, 16
	TPU0 AD conversion/one-shot output timing set register 2	TP0ADOTSE2	H'0000	H'FFFE 403A	8, 16
	TPU0 AD conversion/one-shot output timing set register 3	TP0ADOTSE3	H'0000	H'FFFE 403C	8, 16
	TPU0 counter reset register	TP0CNTRSE	H'0000	H'FFFE 403E	8, 16
	TPU0 phase shift control register 0	TP0PSCR0	H'00	H'FFFE 4030	8
	TPU0 phase shift control register 1	TP0PSCR1	H'00	H'FFFE 4051	8
	TPU0 phase shift clock divide register	TP0PSCDV	H'00	H'FFFE 4031	8
	TPU0 phase shift counter	TP0PSCNT	H'00	H'FFFE 4024	16
	TPU0 phase shift period set register	TP0PSPSE	H'FFFF	H'FFFE 4028	16
	TPU0 phase shift counter control register	TP0PSCNTCR	H'00	H'FFFE 4020	8
Unit 1	TPU1 control register	TP1CR	H'00	H'FFFE 4100	8
	TPU1 mode register	TP1MD	H'C0	H'FFFE 4101	8
	TPU1 status register	TP1SR	H'C0	H'FFFE 4103	8
	TPU1 counter	TP1CNT	H'0000	H'FFFE 4104	16
	TPU1 general register 0	TP1GR0	H'FFFF	H'FFFE 4106	16
	TPU1 general register 1	TP1GR1	H'FFFF	H'FFFE 4108	16
	TPU1 general register 2	TP1GR2	H'FFFF	H'FFFE 410A	16
	TPU1 general register 3	TP1GR3	H'FFFF	H'FFFE 410C	16
	TPU1 count start register	TP1CSTR	H'00	H'FFFE 4150	8
	TPU1 I/O control register	TP1IOCR	H'0000	H'FFFE 410E	8, 16
	TPU1 digital filter register	TP1DF	H'00	H'FFFE 4132	8
	TPU1 AD conversion/one-shot output control register	TP1ADCR	H'00	H'FFFE 4134	8
	TPU1 AD conversion/one-shot output timing set register 0	TP1ADOTSE0	H'0000	H'FFFE 4136	8, 16
	TPU1 AD conversion/one-shot output timing set register 1	TP1ADOTSE1	H'0000	H'FFFE 4138	8, 16
	TPU1 AD conversion/one-shot output timing set register 2	TP1ADOTSE2	H'0000	H'FFFE 413A	8, 16
	TPU1 AD conversion/one-shot output timing set register 3	TP1ADOTSE3	H'0000	H'FFFE 413C	8, 16

Table 15.5 TPU Registers (2)

Unit	Register Name	Symbol	After Reset	Address	Access Size
Unit 1	TPU1 counter reset register	TP1CNTRSE	H'0000	H'FFFE 413E	8, 16
	TPU1 phase shift control register 0	TP1PSCR0	H'00	H'FFFE 4130	8
	TPU1 phase shift control register 1	TP1PSCR1	H'00	H'FFFE 4151	8
	TPU1 phase shift clock divide register	TP1PSCDV	H'00	H'FFFE 4131	8
	TPU1 phase shift counter	TP1PSCNT	H'00	H'FFFE 4124	16
	TPU1 phase shift period set register	TP1PSPSE	H'FFFF	H'FFFE 4128	16
	TPU1 phase shift counter control register	TP1PSCNTCR	H'00	H'FFFE 4120	8
Unit 2	TPU2 control register	TP2CR	H'00	H'FFFE 4200	8
	TPU2 mode register	TP2MD	H'C0	H'FFFE 4201	8
	TPU2 status register	TP2SR	H'C0	H'FFFE 4203	8
	TPU2 counter	TP2CNT	H'0000	H'FFFE 4204	16
	TPU2 general register 0	TP2GR0	H'FFFF	H'FFFE 4206	16
	TPU2 general register 1	TP2GR1	H'FFFF	H'FFFE 4208	16
	TPU2 general register 2	TP2GR2	H'FFFF	H'FFFE 420A	16
	TPU2 general register 3	TP2GR3	H'FFFF	H'FFFE 420C	16
	TPU2 count start register	TP2CSTR	H'00	H'FFFE 4250	8
	TPU2 I/O control register	TP2IOCR	H'0000	H'FFFE 420E	8, 16
	TPU2 digital filter register	TP2DF	H'00	H'FFFE 4232	8
	TPU2 AD conversion/one-shot output control register	TP2ADCR	H'00	H'FFFE 4234	8
	TPU2 AD conversion/one-shot output timing set register 0	TP2ADOTSE0	H'0000	H'FFFE 4236	8, 16
	TPU2 AD conversion/one-shot output timing set register 1	TP2ADOTSE1	H'0000	H'FFFE 4238	8, 16
	TPU2 AD conversion/one-shot output timing set register 2	TP2ADOTSE2	H'0000	H'FFFE 423A	8, 16
	TPU2 AD conversion/one-shot output timing set register 3	TP2ADOTSE3	H'0000	H'FFFE 423C	8, 16
	TPU2 counter reset register	TP2CNTRSE	H'0000	H'FFFE 423E	8, 16
	TPU2 phase shift control register 0	TP2PSCR0	H'00	H'FFFE 4230	8
	TPU2 phase shift control register 1	TP2PSCR1	H'00	H'FFFE 4251	8
	TPU2 phase shift clock divide register	TP2PSCDV	H'00	H'FFFE 4231	8
	TPU2 phase shift counter	TP2PSCNT	H'00	H'FFFE 4224	16
	TPU2 phase shift period set register	TP2PSPSE	H'FFFF	H'FFFE 4228	16
	TPU2 phase shift counter control register	TP2PSCNTCR	H'00	H'FFFE 4220	8
Unit 3	TPU3 control register	TP3CR	H'00	H'FFFE 4300	8
	TPU3 mode register	TP3MD	H'C0	H'FFFE 4301	8
	TPU3 status register	TP3SR	H'C0	H'FFFE 4303	8
	TPU3 counter	TP3CNT	H'0000	H'FFFE 4304	16
	TPU3 general register 0	TP3GR0	H'FFFF	H'FFFE 4306	16
	TPU3 general register 1	TP3GR1	H'FFFF	H'FFFE 4308	16
	TPU3 general register 2	TP3GR2	H'FFFF	H'FFFE 430A	16
	TPU3 general register 3	TP3GR3	H'FFFF	H'FFFE 430C	16
	TPU3 count start register	TP3CSTR	H'00	H'FFFE 4350	8
	TPU3 I/O control register	TP3IOCR	H'0000	H'FFFE 430E	8, 16
	TPU3 digital filter register	TP3DF	H'00	H'FFFE 4332	8
	TPU3 AD conversion/one-shot output control register	TP3ADCR	H'00	H'FFFE 4334	8

Table 15.6 TPU Registers (3)

Unit	Register Name	Symbol	After Reset	Address	Access Size
Unit 3	TPU3 AD conversion/one-shot output timing set register 0	TP3ADOTSE0	H'0000	H'FFFE 4336	8, 16
	TPU3 AD conversion/one-shot output timing set register 1	TP3ADOTSE1	H'0000	H'FFFE 4338	8, 16
	TPU3 AD conversion/one-shot output timing set register 2	TP3ADOTSE2	H'0000	H'FFFE 433A	8, 16
	TPU3 AD conversion/one-shot output timing set register 3	TP3ADOTSE3	H'0000	H'FFFE 433C	8, 16
	TPU3 counter reset register	TP3CNTRSE	H'0000	H'FFFE 433E	8, 16
	TPU3 phase shift control register 0	TP3PSCR0	H'00	H'FFFE 4330	8
	TPU3 phase shift control register 1	TP3PSCR1	H'00	H'FFFE 4351	8
	TPU3 phase shift clock divide register	TP3PSCDV	H'00	H'FFFE 4331	8
	TPU3 phase shift counter	TP3PSCNT	H'00	H'FFFE 4324	16
	TPU3 phase shift period set register	TP3PSPSE	H'FFFF	H'FFFE 4328	16
	TPU3 phase shift counter control register	TP3PSCNTCR	H'00	H'FFFE 4320	8
Unit 4	TPU4 control register	TP4CR	H'00	H'FFFE 4400	8
	TPU4 mode register	TP4MD	H'C0	H'FFFE 4401	8
	TPU4 status register	TP4SR	H'C0	H'FFFE 4403	8
	TPU4 counter	TP4CNT	H'0000	H'FFFE 4404	16
	TPU4 general register 0	TP4GR0	H'FFFF	H'FFFE 4406	16
	TPU4 general register 1	TP4GR1	H'FFFF	H'FFFE 4408	16
	TPU4 general register 2	TP4GR2	H'FFFF	H'FFFE 440A	16
	TPU4 general register 3	TP4GR3	H'FFFF	H'FFFE 440C	16
	TPU4 count start register	TP4CSTR	H'00	H'FFFE 4450	8
	TPU4 I/O control register	TP4IOCR	H'0000	H'FFFE 440E	8, 16
	TPU4 digital filter register	TP4DF	H'00	H'FFFE 4432	8
	TPU4 AD conversion/one-shot output control register	TP4ADCR	H'00	H'FFFE 4434	8
	TPU4 AD conversion/one-shot output timing set register 0	TP4ADOTSE0	H'0000	H'FFFE 4436	8, 16
	TPU4 AD conversion/one-shot output timing set register 1	TP4ADOTSE1	H'0000	H'FFFE 4438	8, 16
	TPU4 AD conversion/one-shot output timing set register 2	TP4ADOTSE2	H'0000	H'FFFE 443A	8, 16
	TPU4 AD conversion/one-shot output timing set register 3	TP4ADOTSE3	H'0000	H'FFFE 443C	8, 16
	TPU4 counter reset register	TP4CNTRSE	H'0000	H'FFFE 443E	8, 16
	TPU4 phase shift control register 0	TP4PSCR0	H'00	H'FFFE 4430	8
	TPU4 phase shift control register 1	TP4PSCR1	H'00	H'FFFE 4451	8
	TPU4 phase shift clock divide register	TP4PSCDV	H'00	H'FFFE 4431	8
	TPU4 phase shift counter	TP4PSCNT	H'00	H'FFFE 4424	16
	TPU4 phase shift period set register	TP4PSPSE	H'FFFF	H'FFFE 4428	16
	TPU4 phase shift counter control register	TP4PSCNTCR	H'00	H'FFFE 4420	8
Unit 5	TPU5 control register	TP5CR	H'00	H'FFFE 4500	8
	TPU5 mode register	TP5MD	H'C0	H'FFFE 4501	8
	TPU5 status register	TP5SR	H'C0	H'FFFE 4503	8
	TPU5 counter	TP5CNT	H'0000	H'FFFE 4504	16
	TPU5 general register 0	TP5GR0	H'FFFF	H'FFFE 4506	16
	TPU5 general register 1	TP5GR1	H'FFFF	H'FFFE 4508	16
	TPU5 general register 2	TP5GR2	H'FFFF	H'FFFE 450A	16
	TPU5 general register 3	TP5GR3	H'FFFF	H'FFFE 450C	16

Table 15.7 TPU Registers (4)

Unit	Register Name	Symbol	After Reset	Address	Access Size
Unit 5	TPU5 count start register	TP5CSTR	H'00	H'FFFE 4550	8
	TPU5 I/O control register	TP5IOCR	H'0000	H'FFFE 450E	8, 16
	TPU5 digital filter register	TP5DF	H'00	H'FFFE 4532	8
	TPU5 AD conversion/one-shot output control register	TP5ADCR	H'00	H'FFFE 4534	8
	TPU5 AD conversion/one-shot output timing set register 0	TP5ADOTSE0	H'0000	H'FFFE 4536	8, 16
	TPU5 AD conversion/one-shot output timing set register 1	TP5ADOTSE1	H'0000	H'FFFE 4538	8, 16
	TPU5 AD conversion/one-shot output timing set register 2	TP5ADOTSE2	H'0000	H'FFFE 453A	8, 16
	TPU5 AD conversion/one-shot output timing set register 3	TP5ADOTSE3	H'0000	H'FFFE 453C	8, 16
	TPU5 counter reset register	TP5CNTRSE	H'0000	H'FFFE 453E	8, 16
	TPU5 phase shift control register 0	TP5PSCR0	H'00	H'FFFE 4530	8
	TPU5 phase shift control register 1	TP5PSCR1	H'00	H'FFFE 4551	8
	TPU5 phase shift clock divide register	TP5PSCDV	H'00	H'FFFE 4531	8
	TPU5 phase shift counter	TP5PSCNT	H'00	H'FFFE 4524	16
	TPU5 phase shift period set register	TP5PSPSE	H'FFFF	H'FFFE 4528	16
	TPU5 phase shift counter control register	TP5PSCNTCR	H'00	H'FFFE 4520	8
Unit 6	TPU6 control register	TP6CR	H'00	H'FFFE 4600	8
	TPU6 mode register	TP6MD	H'C0	H'FFFE 4601	8
	TPU6 status register	TP6SR	H'C0	H'FFFE 4603	8
	TPU6 counter	TP6CNT	H'0000	H'FFFE 4604	16
	TPU6 general register 0	TP6GR0	H'FFFF	H'FFFE 4606	16
	TPU6 general register 1	TP6GR1	H'FFFF	H'FFFE 4608	16
	TPU6 general register 2	TP6GR2	H'FFFF	H'FFFE 460A	16
	TPU6 general register 3	TP6GR3	H'FFFF	H'FFFE 460C	16
	TPU6 count start register	TP6CSTR	H'00	H'FFFE 4650	8
	TPU6 I/O control register	TP6IOCR	H'0000	H'FFFE 460E	8, 16
	TPU6 digital filter register	TP6DF	H'00	H'FFFE 4632	8
	TPU6 AD conversion/one-shot output control register	TP6ADCR	H'00	H'FFFE 4634	8
	TPU6 AD conversion/one-shot output timing set register 0	TP6ADOTSE0	H'0000	H'FFFE 4636	8, 16
	TPU6 AD conversion/one-shot output timing set register 1	TP6ADOTSE1	H'0000	H'FFFE 4638	8, 16
	TPU6 AD conversion/one-shot output timing set register 2	TP6ADOTSE2	H'0000	H'FFFE 463A	8, 16
	TPU6 AD conversion/one-shot output timing set register 3	TP6ADOTSE3	H'0000	H'FFFE 463C	8, 16
	TPU6 counter reset register	TP6CNTRSE	H'0000	H'FFFE 463E	8, 16
	TPU6 phase shift control register 0	TP6PSCR0	H'00	H'FFFE 4630	8
	TPU6 phase shift control register 1	TP6PSCR1	H'00	H'FFFE 4651	8
	TPU6 phase shift clock divide register	TP6PSCDV	H'00	H'FFFE 4631	8
	TPU6 phase shift counter	TP6PSCNT	H'00	H'FFFE 4624	16
	TPU6 phase shift period set register	TP6PSPSE	H'FFFF	H'FFFE 4628	16
	TPU6 phase shift counter control register	TP6PSCNTCR	H'00	H'FFFE 4620	8
Unit 7	TPU7 control register	TP7CR	H'00	H'FFFE 4700	8
	TPU7 mode register	TP7MD	H'C0	H'FFFE 4701	8
	TPU7 status register	TP7SR	H'C0	H'FFFE 4703	8
	TPU7 counter	TP7CNT	H'0000	H'FFFE 4704	16

Table 15.8 TPU Registers (5)

Unit	Register Name	Symbol	After Reset	Address	Access Size
Unit 7	TPU7 general register 0	TP7GR0	H'FFFF	H'FFFE 4706	16
	TPU7 general register 1	TP7GR1	H'FFFF	H'FFFE 4708	16
	TPU7 general register 2	TP7GR2	H'FFFF	H'FFFE 470A	16
	TPU7 general register 3	TP7GR3	H'FFFF	H'FFFE 470C	16
	TPU7 count start register	TP7CSTR	H'00	H'FFFE 4750	8
	TPU7 I/O control register	TP7IOCR	H'0000	H'FFFE 470E	8, 16
	TPU7 digital filter register	TP7DF	H'00	H'FFFE 4732	8
	TPU7 AD conversion/one-shot output control register	TP7ADCR	H'00	H'FFFE 4734	8
	TPU7 AD conversion/one-shot output timing set register 0	TP7ADOTSE0	H'0000	H'FFFE 4736	8, 16
	TPU7 AD conversion/one-shot output timing set register 1	TP7ADOTSE1	H'0000	H'FFFE 4738	8, 16
	TPU7 AD conversion/one-shot output timing set register 2	TP7ADOTSE2	H'0000	H'FFFE 473A	8, 16
	TPU7 AD conversion/one-shot output timing set register 3	TP7ADOTSE3	H'0000	H'FFFE 473C	8, 16
	TPU7 counter reset register	TP7CNTRSE	H'0000	H'FFFE 473E	8, 16
	TPU7 phase shift control register 0	TP7PSCR0	H'00	H'FFFE 4730	8
	TPU7 phase shift control register 1	TP7PSCR1	H'00	H'FFFE 4751	8
	TPU7 phase shift clock divide register	TP7PSCDV	H'00	H'FFFE 4731	8
	TPU7 phase shift counter	TP7PSCNT	H'00	H'FFFE 4724	16
	TPU7 phase shift period set register	TP7PSPSE	H'FFFF	H'FFFE 4728	16
	TPU7 phase shift counter control register	TP7PSCNTCR	H'00	H'FFFE 4720	8
Unit 8	TPU8 control register	TP8CR	H'00	H'FFFE 4800	8
	TPU8 mode register	TP8MD	H'C0	H'FFFE 4801	8
	TPU8 status register	TP8SR	H'C0	H'FFFE 4803	8
	TPU8 counter	TP8CNT	H'0000	H'FFFE 4804	16
	TPU8 general register 0	TP8GR0	H'FFFF	H'FFFE 4806	16
	TPU8 general register 1	TP8GR1	H'FFFF	H'FFFE 4808	16
	TPU8 general register 2	TP8GR2	H'FFFF	H'FFFE 480A	16
	TPU8 general register 3	TP8GR3	H'FFFF	H'FFFE 480C	16
	TPU8 count start register	TP8CSTR	H'00	H'FFFE 4850	8
	TPU8 I/O control register	TP8IOCR	H'0000	H'FFFE 480E	8, 16
	TPU8 digital filter register	TP8DF	H'00	H'FFFE 4832	8
	TPU8 AD conversion/one-shot output control register	TP8ADCR	H'00	H'FFFE 4834	8
	TPU8 AD conversion/one-shot output timing set register 0	TP8ADOTSE0	H'0000	H'FFFE 4836	8, 16
	TPU8 AD conversion/one-shot output timing set register 1	TP8ADOTSE1	H'0000	H'FFFE 4838	8, 16
	TPU8 AD conversion/one-shot output timing set register 2	TP8ADOTSE2	H'0000	H'FFFE 483A	8, 16
	TPU8 AD conversion/one-shot output timing set register 3	TP8ADOTSE3	H'0000	H'FFFE 483C	8, 16
	TPU8 counter reset register	TP8CNTRSE	H'0000	H'FFFE 483E	8, 16
	TPU8 phase shift control register 0	TP8PSCR0	H'00	H'FFFE 4830	8
	TPU8 phase shift control register 1	TP8PSCR1	H'00	H'FFFE 4851	8
	TPU8 phase shift clock divide register	TP8PSCDV	H'00	H'FFFE 4831	8
	TPU8 phase shift counter	TP8PSCNT	H'00	H'FFFE 4824	16
	TPU8 phase shift period set register	TP8PSPSE	H'FFFF	H'FFFE 4828	16
	TPU8 phase shift counter control register	TP8PSCNTCR	H'00	H'FFFE 4820	8

Table 15.9 TPU Registers (6)

Unit	Register Name	Symbol	After Reset	Address	Access Size
Unit 9	TPU9 control register	TP9CR	H'00	H'FFFE 4900	8
	TPU9 mode register	TP9MD	H'C0	H'FFFE 4901	8
	TPU9 status register	TP9SR	H'C0	H'FFFE 4903	8
	TPU9 counter	TP9CNT	H'0000	H'FFFE 4904	16
	TPU9 general register 0	TP9GR0	H'FFFF	H'FFFE 4906	16
	TPU9 general register 1	TP9GR1	H'FFFF	H'FFFE 4908	16
	TPU9 general register 2	TP9GR2	H'FFFF	H'FFFE 490A	16
	TPU9 general register 3	TP9GR3	H'FFFF	H'FFFE 490C	16
	TPU9 count start register	TP9CSTR	H'00	H'FFFE 4950	8
	TPU9 I/O control register	TP9IOCR	H'0000	H'FFFE 490E	8, 16
	TPU9 digital filter register	TP9DF	H'00	H'FFFE 4932	8
	TPU9 AD conversion/one-shot output control register	TP9ADCR	H'00	H'FFFE 4934	8
	TPU9 AD conversion/one-shot output timing set register 0	TP9ADOTSE0	H'0000	H'FFFE 4936	8, 16
	TPU9 AD conversion/one-shot output timing set register 1	TP9ADOTSE1	H'0000	H'FFFE 4938	8, 16
	TPU9 AD conversion/one-shot output timing set register 2	TP9ADOTSE2	H'0000	H'FFFE 493A	8, 16
	TPU9 AD conversion/one-shot output timing set register 3	TP9ADOTSE3	H'0000	H'FFFE 493C	8, 16
	TPU9 counter reset register	TP9CNTRSE	H'0000	H'FFFE 493E	8, 16
	TPU9 phase shift control register 0	TP9PSCR0	H'00	H'FFFE 4930	8
	TPU9 phase shift control register 1	TP9PSCR1	H'00	H'FFFE 4951	8
	TPU9 phase shift clock divide register	TP9PSCDV	H'00	H'FFFE 4931	8
	TPU9 phase shift counter	TP9PSCNT	H'00	H'FFFE 4924	16
	TPU9 phase shift period set register	TP9PSPSE	H'FFFF	H'FFFE 4928	16
	TPU9 phase shift counter control register	TP9PSCNTCR	H'00	H'FFFE 4920	8
Unit 10	TPU10 control register	TP10CR	H'00	H'FFFE 4A00	8
	TPU10 mode register	TP10MD	H'C0	H'FFFE 4A01	8
	TPU10 status register	TP10SR	H'C0	H'FFFE 4A03	8
	TPU10 counter	TP10CNT	H'0000	H'FFFE 4A04	16
	TPU10 general register 0	TP10GR0	H'FFFF	H'FFFE 4A06	16
	TPU10 general register 1	TP10GR1	H'FFFF	H'FFFE 4A08	16
	TPU10 general register 2	TP10GR2	H'FFFF	H'FFFE 4A0A	16
	TPU10 general register 3	TP10GR3	H'FFFF	H'FFFE 4A0C	16
	TPU10 count start register	TP10CSTR	H'00	H'FFFE 4A50	8
	TPU10 I/O control register	TP10IOCR	H'0000	H'FFFE 4A0E	8, 16
	TPU10 digital filter register	TP10DF	H'00	H'FFFE 4A32	8
	TPU10 AD conversion/one-shot output control register	TP10ADCR	H'00	H'FFFE 4A34	8
	TPU10 AD conversion/one-shot output timing set register 0	TP10ADOTSE0	H'0000	H'FFFE 4A36	8, 16
	TPU10 AD conversion/one-shot output timing set register 1	TP10ADOTSE1	H'0000	H'FFFE 4A38	8, 16
	TPU10 AD conversion/one-shot output timing set register 2	TP10ADOTSE2	H'0000	H'FFFE 4A3A	8, 16
	TPU10 AD conversion/one-shot output timing set register 3	TP10ADOTSE3	H'0000	H'FFFE 4A3C	8, 16
	TPU10 counter reset register	TP10CNTRSE	H'0000	H'FFFE 4A3E	8, 16
	TPU10 phase shift control register 0	TP10PSCR0	H'00	H'FFFE 4A30	8
	TPU10 phase shift control register 1	TP10PSCR1	H'00	H'FFFE 4A51	8

Table 15.10 TPU Registers (7)

Unit	Register Name	Symbol	After Reset	Address	Access Size
Unit 10	TPU10 phase shift clock divide register	TP10PSCDV	H'00	H'FFFE 4A31	8
	TPU10 phase shift counter	TP10PSCNT	H'00	H'FFFE 4A24	16
	TPU10 phase shift period set register	TP10PSPSE	H'FFFF	H'FFFE 4A28	16
	TPU10 phase shift counter control register	TP10PSCNTCR	H'00	H'FFFE 4A20	8
Unit 11	TPU11 control register	TP11CR	H'00	H'FFFE 4B00	8
	TPU11 mode register	TP11MD	H'C0	H'FFFE 4B01	8
	TPU11 status register	TP11SR	H'C0	H'FFFE 4B03	8
	TPU11 counter	TP11CNT	H'0000	H'FFFE 4B04	16
	TPU11 general register 0	TP11GR0	H'FFFF	H'FFFE 4B06	16
	TPU11 general register 1	TP11GR1	H'FFFF	H'FFFE 4B08	16
	TPU11 general register 2	TP11GR2	H'FFFF	H'FFFE 4B0A	16
	TPU11 general register 3	TP11GR3	H'FFFF	H'FFFE 4B0C	16
	TPU11 count start register	TP11CSTR	H'00	H'FFFE 4B50	8
	TPU11 I/O control register	TP11IOCR	H'0000	H'FFFE 4B0E	8, 16
	TPU11 digital filter register	TP11DF	H'00	H'FFFE 4B32	8
	TPU11 AD conversion/one-shot output control register	TP11ADCR	H'00	H'FFFE 4B34	8
	TPU11 AD conversion/one-shot output timing set register 0	TP11ADOTSE0	H'0000	H'FFFE 4B36	8, 16
	TPU11 AD conversion/one-shot output timing set register 1	TP11ADOTSE1	H'0000	H'FFFE 4B38	8, 16
	TPU11 AD conversion/one-shot output timing set register 2	TP11ADOTSE2	H'0000	H'FFFE 4B3A	8, 16
	TPU11 AD conversion/one-shot output timing set register 3	TP11ADOTSE3	H'0000	H'FFFE 4B3C	8, 16
	TPU11 counter reset register	TP11CNTRSE	H'0000	H'FFFE 4B3E	8, 16
	TPU11 phase shift control register 0	TP11PSCR0	H'00	H'FFFE 4B30	8
	TPU11 phase shift control register 1	TP11PSCR1	H'00	H'FFFE 4B51	8
	TPU11 phase shift clock divide register	TP11PSCDV	H'00	H'FFFE 4B31	8
	TPU11 phase shift counter	TP11PSCNT	H'00	H'FFFE 4B24	16
	TPU11 phase shift period set register	TP11PSPSE	H'FFFF	H'FFFE 4B28	16
	TPU11 phase shift counter control register	TP11PSCNTCR	H'00	H'FFFE 4B20	8
Unit 12	TPU12 control register	TP12CR	H'00	H'FFFE 4C00	8
	TPU12 mode register	TP12MD	H'C0	H'FFFE 4C01	8
	TPU12 status register	TP12SR	H'C0	H'FFFE 4C03	8
	TPU12 counter	TP12CNT	H'0000	H'FFFE 4C04	16
	TPU12 general register 0	TP12GR0	H'FFFF	H'FFFE 4C06	16
	TPU12 general register 1	TP12GR1	H'FFFF	H'FFFE 4C08	16
	TPU12 general register 2	TP12GR2	H'FFFF	H'FFFE 4C0A	16
	TPU12 general register 3	TP12GR3	H'FFFF	H'FFFE 4C0C	16
	TPU12 count start register	TP12CSTR	H'00	H'FFFE 4C50	8
	TPU12 I/O control register	TP12IOCR	H'0000	H'FFFE 4C0E	8, 16
	TPU12 digital filter register	TP12DF	H'00	H'FFFE 4C32	8
	TPU12 AD conversion/one-shot output control register	TP12ADCR	H'00	H'FFFE 4C34	8
	TPU12 AD conversion/one-shot output timing set register 0	TP12ADOTSE0	H'0000	H'FFFE 4C36	8, 16
	TPU12 AD conversion/one-shot output timing set register 1	TP12ADOTSE1	H'0000	H'FFFE 4C38	8, 16
	TPU12 AD conversion/one-shot output timing set register 2	TP12ADOTSE2	H'0000	H'FFFE 4C3A	8, 16

Table 15.11 TPU Registers (8)

Unit	Register Name	Symbol	After Reset	Address	Access Size
Unit 12	TPU12 AD conversion/one-shot output timing set register 3	TP12ADOTSE3	H'0000	H'FFFE 4C3C	8, 16
	TPU12 counter reset register	TP12CNTRSE	H'0000	H'FFFE 4C3E	8, 16
	TPU12 phase shift control register 0	TP12PSCR0	H'00	H'FFFE 4C30	8
	TPU12 phase shift control register 1	TP12PSCR1	H'00	H'FFFE 4C51	8
	TPU12 phase shift clock divide register	TP12PSCDV	H'00	H'FFFE 4C31	8
	TPU12 phase shift counter	TP12PSCNT	H'00	H'FFFE 4C24	16
	TPU12 phase shift period set register	TP12PSPSE	H'FFFF	H'FFFE 4C28	16
	TPU12 phase shift counter control register	TP12PSCNTCR	H'00	H'FFFE 4C20	8
Unit 13	TPU13 control register	TP13CR	H'00	H'FFFE 4D00	8
	TPU13 mode register	TP13MD	H'C0	H'FFFE 4D01	8
	TPU13 status register	TP13SR	H'C0	H'FFFE 4D03	8
	TPU13 counter	TP13CNT	H'0000	H'FFFE 4D04	16
	TPU13 general register 0	TP13GR0	H'FFFF	H'FFFE 4D06	16
	TPU13 general register 1	TP13GR1	H'FFFF	H'FFFE 4D08	16
	TPU13 general register 2	TP13GR2	H'FFFF	H'FFFE 4D0A	16
	TPU13 general register 3	TP13GR3	H'FFFF	H'FFFE 4D0C	16
	TPU13 count start register	TP13CSTR	H'00	H'FFFE 4D50	8
	TPU13 I/O control register	TP13IOCR	H'0000	H'FFFE 4D0E	8, 16
	TPU13 digital filter register	TP13DF	H'00	H'FFFE 4D32	8
	TPU13 AD conversion/one-shot output control register	TP13ADCR	H'00	H'FFFE 4D34	8
	TPU13 AD conversion/one-shot output timing set register 0	TP13ADOTSE0	H'0000	H'FFFE 4D36	8, 16
	TPU13 AD conversion/one-shot output timing set register 1	TP13ADOTSE1	H'0000	H'FFFE 4D38	8, 16
	TPU13 AD conversion/one-shot output timing set register 2	TP13ADOTSE2	H'0000	H'FFFE 4D3A	8, 16
	TPU13 AD conversion/one-shot output timing set register 3	TP13ADOTSE3	H'0000	H'FFFE 4D3C	8, 16
	TPU13 counter reset register	TP13CNTRSE	H'0000	H'FFFE 4D3E	8, 16
	TPU13 phase shift control register 0	TP13PSCR0	H'00	H'FFFE 4D30	8
	TPU13 phase shift control register 1	TP13PSCR1	H'00	H'FFFE 4D51	8
	TPU13 phase shift clock divide register	TP13PSCDV	H'00	H'FFFE 4D31	8
	TPU13 phase shift counter	TP13PSCNT	H'00	H'FFFE 4D24	16
	TPU13 phase shift period set register	TP13PSPSE	H'FFFF	H'FFFE 4D28	16
	TPU13 phase shift counter control register	TP13PSCNTCR	H'00	H'FFFE 4D20	8
Unit 14	TPU14 control register	TP14CR	H'00	H'FFFE 4E00	8
	TPU14 mode register	TP14MD	H'C0	H'FFFE 4E01	8
	TPU14 status register	TP14SR	H'C0	H'FFFE 4E03	8
	TPU14 counter	TP14CNT	H'0000	H'FFFE 4E04	16
	TPU14 general register 0	TP14GR0	H'FFFF	H'FFFE 4E06	16
	TPU14 general register 1	TP14GR1	H'FFFF	H'FFFE 4E08	16
	TPU14 general register 2	TP14GR2	H'FFFF	H'FFFE 4E0A	16
	TPU14 general register 3	TP14GR3	H'FFFF	H'FFFE 4E0C	16
	TPU14 count start register	TP14CSTR	H'00	H'FFFE 4E50	8
	TPU14 I/O control register	TP14IOCR	H'0000	H'FFFE 4E0E	8, 16
	TPU14 digital filter register	TP14DF	H'00	H'FFFE 4E32	8

Table 15.12 TPU Registers (9)

Unit	Register Name	Symbol	After Reset	Address	Access Size
Unit 14	TPU14 AD conversion/one-shot output control register	TP14ADCR	H'00	H'FFFE 4E34	8
	TPU14 AD conversion/one-shot output timing set register 0	TP14ADOTSE0	H'0000	H'FFFE 4E36	8, 16
	TPU14 AD conversion/one-shot output timing set register 1	TP14ADOTSE1	H'0000	H'FFFE 4E38	8, 16
	TPU14 AD conversion/one-shot output timing set register 2	TP14ADOTSE2	H'0000	H'FFFE 4E3A	8, 16
	TPU14 AD conversion/one-shot output timing set register 3	TP14ADOTSE3	H'0000	H'FFFE 4E3C	8, 16
	TPU14 counter reset register	TP14CNTRSE	H'0000	H'FFFE 4E3E	8, 16
	TPU14 phase shift control register 0	TP14PSCR0	H'00	H'FFFE 4E30	8
	TPU14 phase shift control register 1	TP14PSCR1	H'00	H'FFFE 4E51	8
	TPU14 phase shift clock divide register	TP14PSCDV	H'00	H'FFFE 4E31	8
	TPU14 phase shift counter	TP14PSCNT	H'00	H'FFFE 4E24	16
	TPU14 phase shift period set register	TP14PSPSE	H'FFFF	H'FFFE 4E28	16
	TPU14 phase shift counter control register	TP14PSCNTCR	H'00	H'FFFE 4E20	8
Unit 15	TPU15 control register	TP15CR	H'00	H'FFFE 4F00	8
	TPU15 mode register	TP15MD	H'C0	H'FFFE 4F01	8
	TPU15 status register	TP15SR	H'C0	H'FFFE 4F03	8
	TPU15 counter	TP15CNT	H'0000	H'FFFE 4F04	16
	TPU15 general register 0	TP15GR0	H'FFFF	H'FFFE 4F06	16
	TPU15 general register 1	TP15GR1	H'FFFF	H'FFFE 4F08	16
	TPU15 general register 2	TP15GR2	H'FFFF	H'FFFE 4F0A	16
	TPU15 general register 3	TP15GR3	H'FFFF	H'FFFE 4F0C	16
	TPU15 count start register	TP15CSTR	H'00	H'FFFE 4F50	8
	TPU15 I/O control register	TP15IOCR	H'0000	H'FFFE 4F0E	8, 16
	TPU15 digital filter register	TP15DF	H'00	H'FFFE 4F32	8
	TPU15 AD conversion/one-shot output control register	TP15ADCR	H'00	H'FFFE 4F34	8
	TPU15 AD conversion/one-shot output timing set register 0	TP15ADOTSE0	H'0000	H'FFFE 4F36	8, 16
	TPU15 AD conversion/one-shot output timing set register 1	TP15ADOTSE1	H'0000	H'FFFE 4F38	8, 16
	TPU15 AD conversion/one-shot output timing set register 2	TP15ADOTSE2	H'0000	H'FFFE 4F3A	8, 16
	TPU15 AD conversion/one-shot output timing set register 3	TP15ADOTSE3	H'0000	H'FFFE 4F3C	8, 16
	TPU15 counter reset register	TP15CNTRSE	H'0000	H'FFFE 4F3E	8, 16
	TPU15 phase shift control register 0	TP15PSCR0	H'00	H'FFFE 4F30	8
	TPU15 phase shift control register 1	TP15PSCR1	H'00	H'FFFE 4F51	8
	TPU15 phase shift clock divide register	TP15PSCDV	H'00	H'FFFE 4F31	8
	TPU15 phase shift counter	TP15PSCNT	H'00	H'FFFE 4F24	16
	TPU15 phase shift period set register	TP15PSPSE	H'FFFF	H'FFFE 4F28	16
	TPU15 phase shift counter control register	TP15PSCNTCR	H'00	H'FFFE 4F20	8

15.2.1 TPUi Control Register (TPiCR) (i = 0 to 15)

Address: TP0CR: H'FFFE 4000, TP1CR: H'FFFE 4100, TP2CR: H'FFFE 4200, TP3CR: H'FFFE 4300, TP4CR: H'FFFE 4400, TP5CR: H'FFFE 4500, TP6CR: H'FFFE 4600, TP7CR: H'FFFE 4700, TP8CR: H'FFFE 4800, TP9CR: H'FFFE 4900, TP10CR: H'FFFE 4A00, TP11CR: H'FFFE 4B00, TP12CR: H'FFFE 4C00, TP13CR: H'FFFE 4D00, TP14CR: H'FFFE 4E00, TP15CR: H'FFFE 4F00

	b7	b6	b5	b4	b3	b2	b1	b0
	TRS[2:0]			CES[1:0]		CSS[2:0]		
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b5	TRS[2:0]	TPiCNT Counter Reset Source Select Bits	b7 b6 b5 0 0 0 : No reset source 0 0 1 : Compare match or input capture of channel 0 0 1 0 : Compare match or input capture of channel 1 0 1 1 : Compare match between the TPiPSCNT counter and the TPiPSPSE register 1 0 0 : Reset triggered by the TPiCNTRSE register setting 1 0 1 : Compare match or input capture of channel 2 1 1 0 : Compare match or input capture of channel 3 1 1 1 : Compare match between the TPiPSCNT counter and the TPiPSPSE register	R/W
b4, b3	CES[1:0]	Count Edge Select Bits	b4 b3 0 0 : Rising edge 0 1 : Falling edge 1 0 : Both edges 1 1 : Both edges	R/W
b2 to b0	CSS[2:0]	Count Source Select Bits	b2 b1 b0 0 0 0 : Peripheral bus clock A is not divided. 0 0 1 : Peripheral bus clock A divided by 4 0 1 0 : Peripheral bus clock A divided by 16 0 1 1 : Peripheral bus clock A divided by 64 1 0 0 : Peripheral bus clock A divided by 256 1 0 1 : Peripheral bus clock A divided by 1024 1 1 0 : Peripheral bus clock A divided by 4096 1 1 1 : Unit i + 1 overflow when i = 0, 2, 4, 6, 8, 10, 12, or 14 Setting prohibited when i = 1, 3, 5, 7, 9, 11, 13, or 15	R/W

Note: • i = 0 to 15

TRS Bit

The TPiCNT counter reset source is selected by the TRS bit. Set the GR2FS bit in the TPiMD register to 0 (the TPiGR2 register is used for the input capture/output compare) when the TRS bit is B'101.

Set the GR3FS bit in the TPiMD register to 0 (the TPiGR3 register is used for the input capture/output compare) when the TRS bit is B'110.

In phase shift mode, set the TRS bit to B'011 or B'111.

In PWM mode 2, set the TRS bit to B'100 and configure the TPiCNT counter cycle in the TPiCNTRSE register.

When the TRS bit is B'001, B'010, B'101, or B'110, waveforms generated from the compare match are not output at the corresponding output pin.

CES Bit

A counted edge is selected by the CES bit. The setting value is disabled and the rising edge is selected when the peripheral bus clock A undivided is selected as a count source. The counted edge can be selected when an input clock is the peripheral bus clock A divided by 4 or slower.

Set to 01 or 00 when an overflow in another unit is selected as a count source.

CSS Bit

The TPiCNT counter count source is selected by the CSS bit.

15.2.2 TPUi Mode Register (TPiMD) (i = 0 to 15)

Address: TP0MD: H'FFFE 4001, TP1MD: H'FFFE 4101, TP2MD: H'FFFE 4201, TP3MD: H'FFFE 4301, TP4MD: H'FFFE 4401, TP5MD: H'FFFE 4501, TP6MD: H'FFFE 4601, TP7MD: H'FFFE 4701, TP8MD: H'FFFE 4801, TP9MD: H'FFFE 4901, TP10MD: H'FFFE 4A01, TP11MD: H'FFFE 4B01, TP12MD: H'FFFE 4C01, TP13MD: H'FFFE 4D01, TP14MD: H'FFFE 4E01, TP15MD: H'FFFE 4F01

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	GR3FS	GR2FS	—	MD[2:0]		
After Reset	1	1	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7, b6	—	Reserved	These bits are read as 1. The write value should be 1.	R
b5	GR3FS	TPiGR3 Function Select Bit	0: TPiGR3 register is used for input capture/output compare. 1: TPiGR3 register is used as buffer register.	R/W
b4	GR2FS	TPiGR2 Function Select Bit	0: TPiGR2 register is used for input capture/output compare. 1: TPiGR2 register is used as buffer register.	R/W
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R
b2 to b0	MD[2:0]	Mode Select Bits	b2 b1 b0 0 0 0 : Timer mode 0 0 1 : Setting prohibited 0 1 0 : Pulse width modulation mode 1 0 1 1 : Pulse width modulation mode 2 1 0 0 : Setting prohibited 1 0 1 : Setting prohibited 1 1 0 : Setting prohibited 1 1 1 : Setting prohibited	R/W

Note: • i = 0 to 15

GR3FS Bit

Whether the TPiGR3 register is used for the input capture/output compare or as the buffer register for the TPiGR1 register is selected by the GR3FS bit.

The TPiGR3 register cannot be used for the input capture/output compare when the TPiGR3 register is used as the buffer register.

GR2FS Bit

Whether the TPiGR2 register is used for the input capture/output compare or as the buffer register for the TPiGR0 register is selected by using the GR2FS bit.

The TPiGR2 register cannot be used for the input capture/output compare when the TPiGR2 register is used as the buffer register.

MD Bit

Operating mode is set by the MD bit.

15.2.3 TPUi Status Register (TPiSR) (i = 0 to 15)

Address: TP0SR: H'FFFE 4003, TP1SR: H'FFFE 4103, TP2SR: H'FFFE 4203, TP3SR: H'FFFE 4303, TP4SR: H'FFFE 4403, TP5SR: H'FFFE 4503, TP6SR: H'FFFE 4603, TP7SR: H'FFFE 4703, TP8SR: H'FFFE 4803, TP9SR: H'FFFE 4903, TP10SR: H'FFFE 4A03, TP11SR: H'FFFE 4B03, TP12SR: H'FFFE 4C03, TP13SR: H'FFFE 4D03, TP14SR: H'FFFE 4E03, TP15SR: H'FFFE 4F03

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	TOVF	—	—	—	—
After Reset	1	1	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7, b6	—	Reserved	These bits are read as 1. The write value should be 1.	R
b5	—	Reserved	This bit is read as 0. The write value should be 0.	R
b4	TOVF	Overflow Flag	0: No overflow has occurred. 1: Overflow has occurred.	R/W
b3 to b0	—	Reserved	These bits are read as undefined value. The write value should be 0.	R

Note: • i = 0 to 15

TOVF Flag

The TOVF flag is a status flag indicating that the TPiCNT counter overflow has occurred.

Condition of being set to 0:

- When 0 is written to the TOVF flag after the TOVF flag is read while the flag is set to 1

Condition of being set to 1:

- When the TPiCNT counter value is in the overflow (H'FFFF → H'0000)

15.2.4 TPUi Counter (TPiCNT) (i = 0 to 15)

Address: TP0CNT: H'FFFE 4004, TP1CNT: H'FFFE 4104, TP2CNT: H'FFFE 4204, TP3CNT: H'FFFE 4304, TP4CNT: H'FFFE 4404, TP5CNT: H'FFFE 4504, TP6CNT: H'FFFE 4604, TP7CNT: H'FFFE 4704, TP8CNT: H'FFFE 4804, TP9CNT: H'FFFE 4904, TP10CNT: H'FFFE 4A04, TP11CNT: H'FFFE 4B04, TP12CNT: H'FFFE 4C04, TP13CNT: H'FFFE 4D04, TP14CNT: H'FFFE 4E04, TP15CNT: H'FFFE 4F04

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Description	Setting Range	R/W
b15 to b0	When the CST3 bit in the TPiCSTR register is 1 to start counting: • A read returns the counter value. • When a value is written, the counter starts counting from this value. When the CST3 bit in the TPiCSTR register is 0 to stop counting and the MTUST bit in the TPiCSTR register is 0: • A read after a write is done returns the written value. When the CST3 bit in the TPiCSTR register is 0 to stop counting and the MTUST bit in the TPiCSTR register is 1: • A read returns the counter value. • When a value is written, the counter starts counting from this value.	H'0000 to H'FFFF	R/W

Note: • i = 0 to 15

15.2.5 TPUi General Register k (TPiGRk) (i = 0 to 15; k = 0 to 3)

Address

TP0GR0: H'FFFE 4006, TP0GR1: H'FFFE 4008, TP0GR2: H'FFFE 400A, TP0GR3: H'FFFE 400C,
TP1GR0: H'FFFE 4106, TP1GR1: H'FFFE 4108, TP1GR2: H'FFFE 410A, TP1GR3: H'FFFE 410C,
TP2GR0: H'FFFE 4206, TP2GR1: H'FFFE 4208, TP2GR2: H'FFFE 420A, TP2GR3: H'FFFE 420C,
TP3GR0: H'FFFE 4306, TP3GR1: H'FFFE 4308, TP3GR2: H'FFFE 430A, TP3GR3: H'FFFE 430C,
TP4GR0: H'FFFE 4406, TP4GR1: H'FFFE 4408, TP4GR2: H'FFFE 440A, TP4GR3: H'FFFE 440C,

TP5GR0: H'FFFE 4506, TP5GR1: H'FFFE 4508, TP5GR2: H'FFFE 450A, TP5GR3: H'FFFE 450C,
TP6GR0: H'FFFE 4606, TP6GR1: H'FFFE 4608, TP6GR2: H'FFFE 460A, TP6GR3: H'FFFE 460C,
TP7GR0: H'FFFE 4706, TP7GR1: H'FFFE 4708, TP7GR2: H'FFFE 470A, TP7GR3: H'FFFE 470C,
TP8GR0: H'FFFE 4806, TP8GR1: H'FFFE 4808, TP8GR2: H'FFFE 480A, TP8GR3: H'FFFE 480C,
TP9GR0: H'FFFE 4906, TP9GR1: H'FFFE 4908, TP9GR2: H'FFFE 490A, TP9GR3: H'FFFE 490C,

TP10GR0: H'FFFE 4A06, TP10GR1: H'FFFE 4A08, TP10GR2: H'FFFE 4A0A, TP10GR3: H'FFFE 4A0C,
TP11GR0: H'FFFE 4B06, TP11GR1: H'FFFE 4B08, TP11GR2: H'FFFE 4B0A, TP11GR3: H'FFFE 4B0C,
TP12GR0: H'FFFE 4C06, TP12GR1: H'FFFE 4C08, TP12GR2: H'FFFE 4C0A, TP12GR3: H'FFFE 4C0C,
TP13GR0: H'FFFE 4D06, TP13GR1: H'FFFE 4D08, TP13GR2: H'FFFE 4D0A, TP13GR3: H'FFFE 4D0C,
TP14GR0: H'FFFE 4E06, TP14GR1: H'FFFE 4E08, TP14GR2: H'FFFE 4E0A, TP14GR3: H'FFFE 4E0C,
TP15GR0: H'FFFE 4F06, TP15GR1: H'FFFE 4F08, TP15GR2: H'FFFE 4F0A, TP15GR3: H'FFFE 4F0C

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit	Description	Setting Range	R/W
b15 to b0	When used for input capture: • When input capture occurs, the value in the TPiCNT counter is stored. When used for output compare: • When matching the TPiCNT counter value, waveforms are output at the TPjx pin.	H'0000 to H'FFFF	R/W

Note: • i = 0 to 15, j = 00 to 15; x = A, B, C, or D

15.2.6 TPUi Count Start Register (TPiCSTR) (i = 0 to 15)

Address: TP0CSTR: H'FFFE 4050, TP1CSTR: H'FFFE 4150, TP2CSTR: H'FFFE 4250, TP3CSTR: H'FFFE 4350, TP4CSTR: H'FFFE 4450, TP5CSTR: H'FFFE 4550, TP6CSTR: H'FFFE 4650, TP7CSTR: H'FFFE 4750, TP8CSTR: H'FFFE 4850, TP9CSTR: H'FFFE 4950, TP10CSTR: H'FFFE 4A50, TP11CSTR: H'FFFE 4B50, TP12CSTR: H'FFFE 4C50, TP13CSTR: H'FFFE 4D50, TP14CSTR: H'FFFE 4E50, TP15CSTR: H'FFFE 4F50

	b7	b6	b5	b4	b3	b2	b1	b0
	—	MTUST	CST5	—	CST3	—	—	—
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R
b6	MTUST	MTU-III_TPU Synchronous Start Select Bit	0: MTU-III_TPU synchronous start function is not used. 1: MTU-III_TPU synchronous start function is used.	R/W
b5	CST5	TPiPSCNT Count Start Bit	0: Count stops. 1: Count starts.	R/W
b4	—	Reserved	This bit is read as 0. The write value should be 0.	R
b3	CST3	TPiCNT Count Start Bit	0: Count stops. 1: Count starts.	R/W
b2 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

Note: • i = 0 to 15

MTUST Bit

This bit selects whether or not to use the MTU-III_TPU synchronous start function. When the MTUST bit is 1 and 1 is written to the CST0 bit in the MT01234STR register, the TPU starts counting. When the MTUST bit is 1 and 0 is written to the CST0 bit in the MT01234STR register, the TPU stops counting.

CST5 Bit

This bit selects whether the TPUi phase shift counter (TPiPSCNT) starts or stops counting.

CST3 Bit

Whether the TPUi counter (TPiCNT) starts or stops counting is selected by the CST3 bit. The counter stops by setting the CST3 bit to 0 when the TPjx pin is used in output mode, but the TPjx pin holds the output level. The pin output level becomes the initial value by a write to the TPiIOCR register when the CST3 bit is 0.

The counter does not start by setting the CST3 bit to 1 when the MTUST bit is 1. When the MTUST bit is 1 and then 1 is written to the CST0 bit in the MT01234 start register (MT01234STR) of the MTU-III, the CST3 bit is not set to 1 even if the TPU starts counting.

15.2.7 TPUi I/O Control Register (TPiIOCR) (i = 0 to 15)

Address: TP0IOCR: H'FFFE 400E, TP1IOCR: H'FFFE 410E, TP2IOCR: H'FFFE 420E, TP3IOCR: H'FFFE 430E, TP4IOCR: H'FFFE 440E, TP5IOCR: H'FFFE 450E, TP6IOCR: H'FFFE 460E, TP7IOCR: H'FFFE 470E, TP8IOCR: H'FFFE 480E, TP9IOCR: H'FFFE 490E, TP10IOCR: H'FFFE 4A0E, TP11IOCR: H'FFFE 4B0E, TP12IOCR: H'FFFE 4C0E, TP13IOCR: H'FFFE 4D0E, TP14IOCR: H'FFFE 4E0E, TP15IOCR: H'FFFE 4F0E

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IO1[3:0]				IO0[3:0]				IO3[3:0]				IO2[3:0]			
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 b12	IO1[3:0]	Channel 1 Function Select Bits	Set output compare value/trigger for input capture. (Refer to Table 15.13 to Table 15.15)	R/W
b11 to b8	IO0[3:0]	Channel 0 Function Select Bits	Set output compare value/trigger for input capture. (Refer to Table 15.13 to Table 15.15)	R/W
b7 to b4	IO3[3:0]	Channel 3 Function Select Bits	Set output compare value/trigger for input capture. (Refer to Table 15.13 to Table 15.15)	R/W
b3 to b0	IO2[3:0]	Channel 2 Function Select Bits	Set output compare value/trigger for input capture. (Refer to Table 15.13 to Table 15.15)	R/W

Table 15.13, Table 15.14, and Table 15.15 list Bit Settings to Use TPiGRk Register for Output Compare, Bit Settings to Use TPiGRk Register for Input Capture, and Combinations of IOk Bit in TPiIOCR Register and TPU I/O Pin, respectively.

Table 15.13 Bit Settings to Use TPiGRk Register for Output Compare

IOk Bit in TPiIOCR Register	Output from TPjx Pin
0000	Output disabled
0001	Initial output value: low level, compare match: low level output
0010	Initial output value: low level, compare match: high level output
0011	Initial output value: low level, compare match: toggle output
0100	Output disabled
0101	Initial output value: high level, compare match: low level output
0110	Initial output value: high level, compare match: high level output
0111	Initial output value: high level, compare match: toggle output

Note: • i = 0 to 15; k = 0 to 3; j = 00 to 15; x = A, B, C, or D

Table 15.14 Bit Settings to Use TPiGRk Register for Input Capture

IOk Bit in TPiIOCR Register	Trigger for Input Capture
1000	Rising edge of the signal input to the TPjx pin
1001	Falling edge of the signal input to the TPjx pin
1010	Both edges of the signal input to the TPjx pin
1011	Both edges of the signal input to the TPjx pin
1100 to 1111	Setting prohibited

Note: • i = 0 to 15; k = 0 to 3; j = 00 to 15; x = A, B, C, or D

Table 15.15 Combinations of IOk Bit in TPiIOCR Register and TPU I/O Pin

IOk Bit in TPiIOCR Register	TPU I/O Pin
IO0 bit	TPjA pin
IO1 bit	TPjB pin
IO2 bit	TPjC pin
IO3 bit	TPjD pin

Note: • i = 0 to 15; k = 0 to 3; j = 00 to 15

15.2.8 TPUi Digital Filter Register (TPiDF) (i = 0 to 15)

Address TP0DF: H'FFFE 4032, TP1DF: H'FFFE 4132, TP2DF: H'FFFE 4232, TP3DF: H'FFFE 4332, TP4DF: H'FFFE 4432, TP5DF: H'FFFE 4532, TP6DF: H'FFFE 4632, TP7DF: H'FFFE 4732, TP8DF: H'FFFE 4832, TP9DF: H'FFFE 4932, TP10DF: H'FFFE 4A32, TP11DF: H'FFFE 4B32, TP12DF: H'FFFE 4C32, TP13DF: H'FFFE 4D32, TP14DF: H'FFFE 4E32, TP15DF: H'FFFE 4F32

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	DFE3	DFE2	DFE1	DFE0
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R
b3	DFE3	Channel 3 Digital Filter Function Enable Bit	0: Digital filter function is not used. 1: Digital filter function is used.	R/W
b2	DFE2	Channel 2 Digital Filter Function Enable Bit	0: Digital filter function is not used. 1: Digital filter function is used.	R/W
b1	DFE1	Channel 1 Digital Filter Function Enable Bit	0: Digital filter function is not used. 1: Digital filter function is used.	R/W
b0	DFE0	Channel 0 Digital Filter Function Enable Bit	0: Digital filter function is not used. 1: Digital filter function is used.	R/W

15.2.9 TPUi AD Conversion/One-Shot Output Control Register (TPiADCR) (i = 0 to 15)

Address: TP0ADCR: H'FFFE 4034, TP1ADCR: H'FFFE 4134, TP2ADCR: H'FFFE 4234, TP3ADCR: H'FFFE 4334, TP4ADCR: H'FFFE 4434, TP5ADCR: H'FFFE 4534, TP6ADCR: H'FFFE 4634, TP7ADCR: H'FFFE 4734, TP8ADCR: H'FFFE 4834, TP9ADCR: H'FFFE 4934, TP10ADCR: H'FFFE 4A34, TP11ADCR: H'FFFE 4B34, TP12ADCR: H'FFFE 4C34, TP13ADCR: H'FFFE 4D34, TP14ADCR: H'FFFE 4E34, TP15ADCR: H'FFFE 4F34

	b7	b6	b5	b4	b3	b2	b1	b0
	ADTE3[1:0]		ADTE2[1:0]		ADTE1[1:0]		ADTE0[1:0]	
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7, b6	ADTE3[1:0]	Channel 3 AD Conversion Trigger/ One-shot Output Select Bits	b7 b6 0 0 : A/D conversion trigger function and one-shot output function are not used. 0 1 : A/D conversion trigger function is used. 1 0 : One-shot output function is used. 1 1 : Setting prohibited	R/W
b5, b4	ADTE2[1:0]	Channel 2 AD Conversion Trigger/ One-shot Output Select Bits	b5 b4 0 0 : A/D conversion trigger function and one-shot output function are not used. 0 1 : A/D conversion trigger function is used. 1 0 : One-shot output function is used. 1 1 : Setting prohibited	R/W
b3, b2	ADTE1[1:0]	Channel 1 AD Conversion Trigger/ One-shot Output Select Bits	b3 b2 0 0 : A/D conversion trigger function and one-shot output function are not used. 0 1 : A/D conversion trigger function is used. 1 0 : One-shot output function is used. 1 1 : Setting prohibited	R/W
b1, b0	ADTE0[1:0]	Channel 0 AD Conversion Trigger/ One-shot Output Select Bits	b1 b0 0 0 : A/D conversion trigger function and one-shot output function are not used. 0 1 : A/D conversion trigger function is used. 1 0 : One-shot output function is used. 1 1 : Setting prohibited	R/W

15.2.10 TPUi AD Conversion/One-Shot Output Timing Set Register k (TPiADOTSEk) (i = 0 to 15; k = 0 to 3)

Address

TP0ADOTSE0: H'FFFE 4036, TP0ADOTSE1: H'FFFE 4038, TP0ADOTSE2: H'FFFE 403A, TP0ADOTSE3: H'FFFE 403C,
TP1ADOTSE0: H'FFFE 4136, TP1ADOTSE1: H'FFFE 4138, TP1ADOTSE2: H'FFFE 413A, TP1ADOTSE3: H'FFFE 413C,
TP2ADOTSE0: H'FFFE 4236, TP2ADOTSE1: H'FFFE 4238, TP2ADOTSE2: H'FFFE 423A, TP2ADOTSE3: H'FFFE 423C,
TP3ADOTSE0: H'FFFE 4336, TP3ADOTSE1: H'FFFE 4338, TP3ADOTSE2: H'FFFE 433A, TP3ADOTSE3: H'FFFE 433C,
TP4ADOTSE0: H'FFFE 4436, TP4ADOTSE1: H'FFFE 4438, TP4ADOTSE2: H'FFFE 443A, TP4ADOTSE3: H'FFFE 443C,

TP5ADOTSE0: H'FFFE 4536, TP5ADOTSE1: H'FFFE 4538, TP5ADOTSE2: H'FFFE 453A, TP5ADOTSE3: H'FFFE 453C,
TP6ADOTSE0: H'FFFE 4636, TP6ADOTSE1: H'FFFE 4638, TP6ADOTSE2: H'FFFE 463A, TP6ADOTSE3: H'FFFE 463C,
TP7ADOTSE0: H'FFFE 4736, TP7ADOTSE1: H'FFFE 4738, TP7ADOTSE2: H'FFFE 473A, TP7ADOTSE3: H'FFFE 473C,
TP8ADOTSE0: H'FFFE 4836, TP8ADOTSE1: H'FFFE 4838, TP8ADOTSE2: H'FFFE 483A, TP8ADOTSE3: H'FFFE 483C,
TP9ADOTSE0: H'FFFE 4936, TP9ADOTSE1: H'FFFE 4938, TP9ADOTSE2: H'FFFE 493A, TP9ADOTSE3: H'FFFE 493C,

TP10ADOTSE0: H'FFFE 4A36, TP10ADOTSE1: H'FFFE 4A38, TP10ADOTSE2: H'FFFE 4A3A, TP10ADOTSE3: H'FFFE 4A3C,
TP11ADOTSE0: H'FFFE 4B36, TP11ADOTSE1: H'FFFE 4B38, TP11ADOTSE2: H'FFFE 4B3A, TP11ADOTSE3: H'FFFE 4B3C,
TP12ADOTSE0: H'FFFE 4C36, TP12ADOTSE1: H'FFFE 4C38, TP12ADOTSE2: H'FFFE 4C3A, TP12ADOTSE3: H'FFFE 4C3C,
TP13ADOTSE0: H'FFFE 4D36, TP13ADOTSE1: H'FFFE 4D38, TP13ADOTSE2: H'FFFE 4D3A, TP13ADOTSE3: H'FFFE 4D3C,
TP14ADOTSE0: H'FFFE 4E36, TP14ADOTSE1: H'FFFE 4E38, TP14ADOTSE2: H'FFFE 4E3A, TP14ADOTSE3: H'FFFE 4E3C,
TP15ADOTSE0: H'FFFE 4F36, TP15ADOTSE1: H'FFFE 4F38, TP15ADOTSE2: H'FFFE 4F3A, TP15ADOTSE3: H'FFFE 4F3C

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Description	Setting Range	R/W
b15 to b0	A/D conversion timing/one-shot output timing is set.	H'0001 to H'FFFF	R/W

Note: • The value to be set is “H'0001” or above when the one-shot output timing function is selected.

When the A/D conversion trigger function is selected in the TPUi AD conversion/one-shot output control register (TPiADCR), this register setting value indicates the A/D conversion timing. When the setting value matches the TPiCNT counter value, the A/D conversion start trigger is generated.

When the one-shot output function is selected in the TPUi AD conversion/one-shot output control register (TPiADCR), this register setting value indicates the one-shot output timing. Then the TPiGRk stores the value that the setting value plus the TPiCNT counter value at the time makes.

15.2.11 TPUi Counter Reset Register (TPiCNTRSE)

Address: TP0CNTRSE: H'FFFE 403E, TP1CNTRSE: H'FFFE 413E, TP2CNTRSE: H'FFFE 423E, TP3CNTRSE: H'FFFE 433E, TP4CNTRSE: H'FFFE 443E, TP5CNTRSE: H'FFFE 453E, TP6CNTRSE: H'FFFE 463E, TP7CNTRSE: H'FFFE 473E, TP8CNTRSE: H'FFFE 483E, TP9CNTRSE: H'FFFE 493E, TP10CNTRSE: H'FFFE 4A3E, TP11CNTRSE: H'FFFE 4B3E, TP12CNTRSE: H'FFFE 4C3E, TP13CNTRSE: H'FFFE 4D3E, TP14CNTRSE: H'FFFE 4E3E, TP15CNTRSE: H'FFFE 4F3E

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Description	Setting Range	R/W
b15 to b0	TPiCNT counter reset	H'0000 to H'FFFF	R/W

Note: • i = 0 to 15

When the TRS bit in the TPiCR register is B'100 and the setting value of this register matches the value of the TPiCNT counter, the TPiCNT counter is reset. Then the A/D conversion trigger is output at the A/D converter regardless of the type of the TPiADCR register setting. At the register of the A/D converter, select whether or not to use the A/D conversion caused by the TPiCNT counter reset of the TPU. Then an interrupt request is output at an interrupt controller.

15.2.12 TPUi Phase Shift Control Register 0 (TPiPSCR0) (i = 0 to 15)

Address: TP0PSCR0: H'FFFE 4030, TP1PSCR0: H'FFFE 4130, TP2PSCR0: H'FFFE 4230, TP3PSCR0: H'FFFE 4330, TP4PSCR0: H'FFFE 4430, TP5PSCR0: H'FFFE 4530, TP6PSCR0: H'FFFE 4630, TP7PSCR0: H'FFFE 4730, TP8PSCR0: H'FFFE 4830, TP9PSCR0: H'FFFE 4930, TP10PSCR0: H'FFFE 4A30, TP11PSCR0: H'FFFE 4B30, TP12PSCR0: H'FFFE 4C30, TP13PSCR0: H'FFFE 4D30, TP14PSCR0: H'FFFE 4E30, TP15PSCR0: H'FFFE 4F30

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	PSM
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R
b0	PSM	Phase Shift Mode Select Bit 1	0: Phase shift mode (channel shift mode) is not used. 1: Phase shift mode (channel shift mode) is used.	R/W

15.2.13 TPUi Phase Shift Control Register 1 (TPiPSCR1) (i = 0 to 15)

Address: TP0PSCR1: H'FFFE 4051, TP1PSCR1: H'FFFE 4151, TP2PSCR1: H'FFFE 4251, TP3PSCR1: H'FFFE 4351, TP4PSCR1: H'FFFE 4451, TP5PSCR1: H'FFFE 4551, TP6PSCR1: H'FFFE 4651, TP7PSCR1: H'FFFE 4751, TP8PSCR1: H'FFFE 4851, TP9PSCR1: H'FFFE 4951, TP10PSCR1: H'FFFE 4A51, TP11PSCR1: H'FFFE 4B51, TP12PSCR1: H'FFFE 4C51, TP13PSCR1: H'FFFE 4D51, TP14PSCR1: H'FFFE 4E51, TP15PSCR1: H'FFFE 4F51

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	TSRE5	—	TSRE3	—	—	—
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R
b5	TSRE5	Phase Shift Mode Select Bit 5	0: Phase shift mode (channel shift mode) is not used. 1: Phase shift mode (channel shift mode) is used.	R/W
b4	—	Reserved	This bit is read as 0. The write value should be 0.	R
b3	TSRE3	Phase Shift Mode Select Bit 3	0: Phase shift mode (channel shift mode) is not used. 1: Phase shift mode (channel shift mode) is used.	R/W
b2 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

15.2.14 TPUi Phase Shift Clock Divide Register (TPiPSCDV) (i = 0 to 15)

Address: TP0PSCDV: H'FFFE 4031, TP1PSCDV: H'FFFE 4131, TP2PSCDV: H'FFFE 4231, TP3PSCDV: H'FFFE 4331, TP4PSCDV: H'FFFE 4431, TP5PSCDV: H'FFFE 4531, TP6PSCDV: H'FFFE 4631, TP7PSCDV: H'FFFE 4731, TP8PSCDV: H'FFFE 4831, TP9PSCDV: H'FFFE 4931, TP10PSCDV: H'FFFE 4A31, TP11PSCDV: H'FFFE 4B31, TP12PSCDV: H'FFFE 4C31, TP13PSCDV: H'FFFE 4D31, TP14PSCDV: H'FFFE 4E31, TP15PSCDV: H'FFFE 4F31

	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	0	0	0	0	0	0	0	0

Bit	Description	Setting Range	R/W
b7 to b0	If the setting value is n, the count source for the TPiCNT counter is divided by n. The cycle divided by n is a phase shift time.	H'00 to H'FF	R/W

Note: • i = 0 to 15

15.2.15 TPUi Phase Shift Counter (TPiPSCNT) (i = 0 to 15)

Address: TP0PSCNT: H'FFFE 4024, TP1PSCNT: H'FFFE 4124, TP2PSCNT: H'FFFE 4224, TP3PSCNT: H'FFFE 4324, TP4PSCNT: H'FFFE 4424, TP5PSCNT: H'FFFE 4524, TP6PSCNT: H'FFFE 4624, TP7PSCNT: H'FFFE 4724, TP8PSCNT: H'FFFE 4824, TP9PSCNT: H'FFFE 4924, TP10PSCNT: H'FFFE 4A24, TP11PSCNT: H'FFFE 4B24, TP12PSCNT: H'FFFE 4C24, TP13PSCNT: H'FFFE 4D24, TP14PSCNT: H'FFFE 4E24, TP15PSCNT: H'FFFE 4F24

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Description	Setting Range	R/W
b15 to b0	When the CST5 bit in the TPiCSTR register is 1 to start counting: • A read returns the counter value. • When a value is written, the counter starts counting from this value. When the CST5 bit is 0 to stop counting: • A read after a write is done returns the written value.	H'0000 to H'FFFF	R/W

Note: • i = 0 to 15

15.2.16 TPUi Phase Shift Period Set Register (TPiPSPSE) (i = 0 to 15)

Address: TP0PSPSE: H'FFFE 4028, TP1PSPSE: H'FFFE 4128, TP2PSPSE: H'FFFE 4228, TP3PSPSE: H'FFFE 4328, TP4PSPSE: H'FFFE 4428, TP5PSPSE: H'FFFE 4528, TP6PSPSE: H'FFFE 4628, TP7PSPSE: H'FFFE 4728, TP8PSPSE: H'FFFE 4828, TP9PSPSE: H'FFFE 4928, TP10PSPSE: H'FFFE 4A28, TP11PSPSE: H'FFFE 4B28, TP12PSPSE: H'FFFE 4C28, TP13PSPSE: H'FFFE 4D28, TP14PSPSE: H'FFFE 4E28, TP15PSPSE: H'FFFE 4F28

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit	Description	Setting Range	R/W
b15 to b0	The setting value is equivalent to the cycle of PWM mode 2.	H'0000 to H'FFFF	R/W

15.2.17 TPUi Phase Shift Counter Control Register (TPiPSCNTCR) (i = 0 to 15)

Address: TP0PSCNTCR: H'FFFE 4020, TP1PSCNTCR: H'FFFE 4120, TP2PSCNTCR: H'FFFE 4220, TP3PSCNTCR: H'FFFE 4320, TP4PSCNTCR: H'FFFE 4420, TP5PSCNTCR: H'FFFE 4520, TP6PSCNTCR: H'FFFE 4620, TP7PSCNTCR: H'FFFE 4720, TP8PSCNTCR: H'FFFE 4820, TP9PSCNTCR: H'FFFE 4920, TP10PSCNTCR: H'FFFE 4A20, TP11PSCNTCR: H'FFFE 4B20, TP12PSCNTCR: H'FFFE 4C20, TP13PSCNTCR: H'FFFE 4D20, TP14PSCNTCR: H'FFFE 4E20, TP15PSCNTCR: H'FFFE 4F20

	b7	b6	b5	b4	b3	b2	b1	b0
	—	TRS[1:0]		CES[1:0]		CSS[2:0]		
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R
b6, b5	TRS[1:0]	TPiCNT Reset Source Select Bits	b6 b5 0 0 : No reset source 0 1 : Setting prohibited 1 0 : Compare match between the TPiPSPSE register and the TPiPSCNT counter 1 1 : No reset source	R/W
b4, b3	CES[1:0]	Count Edge Select Bits	b4 b3 0 0 : Rising edge 0 1 : Falling edge 1 0 : Both edges 1 1 : Both edges	R/W
b2 to b0	CSS[2:0]	Count Source Select Bits	b2 b1 b0 0 0 0 : Peripheral bus clock A is not divided. 0 0 1 : Peripheral bus clock A divided by 4 0 1 0 : Peripheral bus clock A divided by 16 0 1 1 : Peripheral bus clock A divided by 64 1 0 0 : Peripheral bus clock A divided by 256 1 0 1 : Peripheral bus clock A divided by 1024 1 1 0 : Peripheral bus clock A divided by 4096 1 1 1 : Setting prohibited	R/W

Note: • i = 0 to 15

TRS Bit

The TPiCNT counter reset source is selected by the TRS bit. In phase shift mode, set the TRS bit to B'10.

CES Bit

A counted edge is selected by the CES bit. The setting value is disabled and a rising edge is selected when the peripheral bus clock A undivided is selected as a count source. The counted edge can be selected when an input clock is the peripheral bus clock A divided by 4 or slower.

CSS Bit

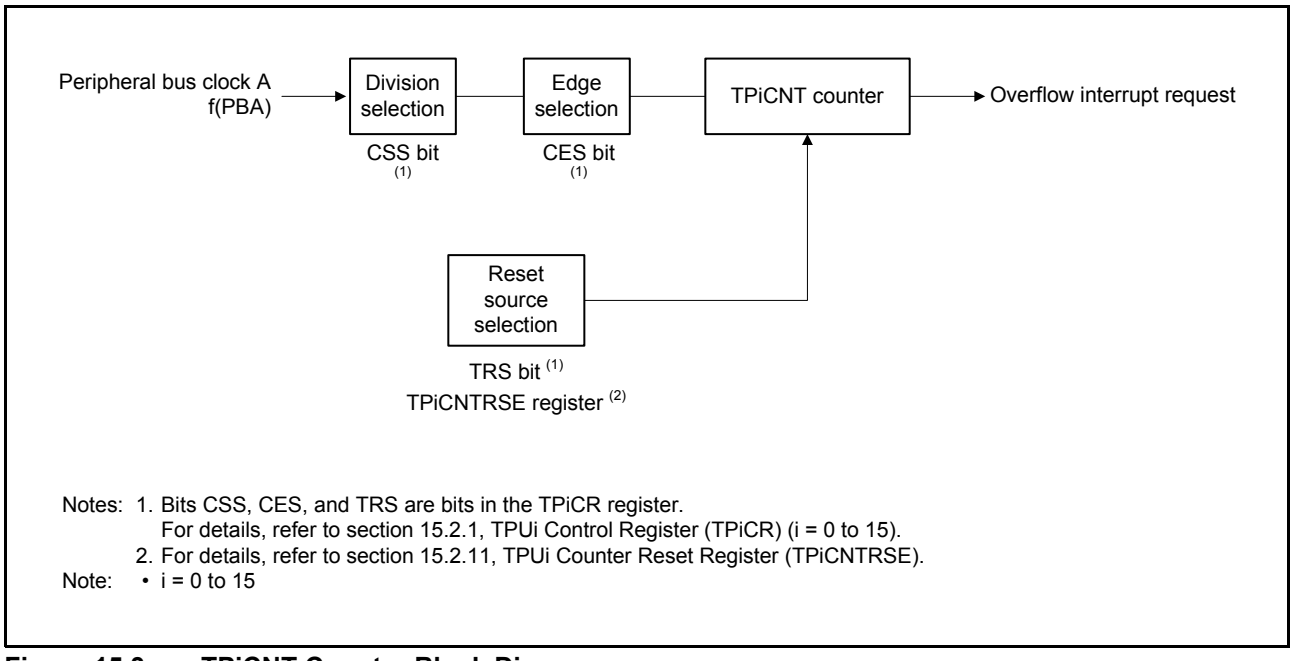
The TPiCNT counter count source is selected by the CSS bit.

15.3 TPiCNT Counter

The TPiCNT counter performs increment operations, free-running count operations and cycle count operations. Table 15.16 lists TPiCNT Counter Specifications and Figure 15.3 shows a TPiCNT Counter Block Diagram.

Table 15.16 TPiCNT Counter Specifications

Item	Description	
	Unit i (i = 0, 2, 4, 6, 8, 10, 12, or 14)	Unit i (i = 1, 3, 5, 7, 9, 11, 13, or 15)
Count sources	Selected by the CSS bit in the TPiCR register b2 b1 b0 0 0 0 : Peripheral bus clock A is not divided 0 0 1 : Peripheral bus clock A divided by 4 0 1 0 : Peripheral bus clock A divided by 16 0 1 1 : Peripheral bus clock A divided by 64 1 0 0 : Peripheral bus clock A divided by 256 1 0 1 : Peripheral bus clock A divided by 1024 1 1 0 : Peripheral bus clock A divided by 4096 1 1 1 : Unit i + 1 overflow	Selected by the CSS bit in the TPiCR register b2 b1 b0 0 0 0 : Peripheral bus clock A is not divided 0 0 1 : Peripheral bus clock A divided by 4 0 1 0 : Peripheral bus clock A divided by 16 0 1 1 : Peripheral bus clock A divided by 64 1 0 0 : Peripheral bus clock A divided by 256 1 0 1 : Peripheral bus clock A divided by 1024 1 1 0 : Peripheral bus clock A divided by 4096 1 1 1 : Setting prohibited
Counting operation	Increment	
Count start condition	The CST3 bit in the TPiCSTR register is set to 1 (count starts).	
Count stop condition	The CST3 bit in the TPiCSTR register is set to 0 (count stops).	
TPiCNT counter reset condition	Reset under the condition set by the TRS bit in the TPiCR register. Reset triggered by the TPiCNTRSE register setting	
Value when TPiCNT counter is reset	H'0000	
Interrupt request generation timing	TPiCNT counter overflow	
Read from TPiCNT counter	A read returns the counter value when the TPiCNT counter is counting. A read returns the counter value while the TPiCNT counter is halted.	
Write to TPiCNT counter	Refer to section 15.10.1, Collision of Write Signal to TPiCNT Counter and Various Signals	



Notes: 1. Bits CSS, CES, and TRS are bits in the TPiCR register.
For details, refer to section 15.2.1, TPUi Control Register (TPiCR) (i = 0 to 15).
2. For details, refer to section 15.2.11, TPUi Counter Reset Register (TPiCNTRSE).
Note: • i = 0 to 15

Figure 15.3 TPiCNT Counter Block Diagram

Table 15.17 lists register settings associated with the count operation.

Table 15.17 Count Operation Settings

TPiCNT Counter Operation	Register	Bit	Setting
Free-running operation	TPiCR	CSS	Select the count source.
	TPiCSTR	CST3	Set to 1 (count starts).
Cycle count operation	TPiCR	CSS	Select the count source.
	TPiCR	TRS	Select the reset source
	TPiCSTR	CST3	Set to 1 (count starts).

Note: • i = 0 to 15

15.3.1 Free-Running Operation and Cycle Count Operation

All the TPiCNT counters in the TPU are set as free-running counters after reset. The TPiCNT counter is incremented as the free-running counter by setting its corresponding bit in the TPiCSTR register to 1. The TOVF bit in the TPiSR register is set to 1 when the TPiCNT counter overflows (H'FFFF → H'0000), and the TPiCNT counter overflow interrupt request is generated.

The TPiCNT counter continues to be incremented from H'0000 after overflows. Figure 15.4 shows an Operation Example of TPiCNT Counter as Free-Running Counter.

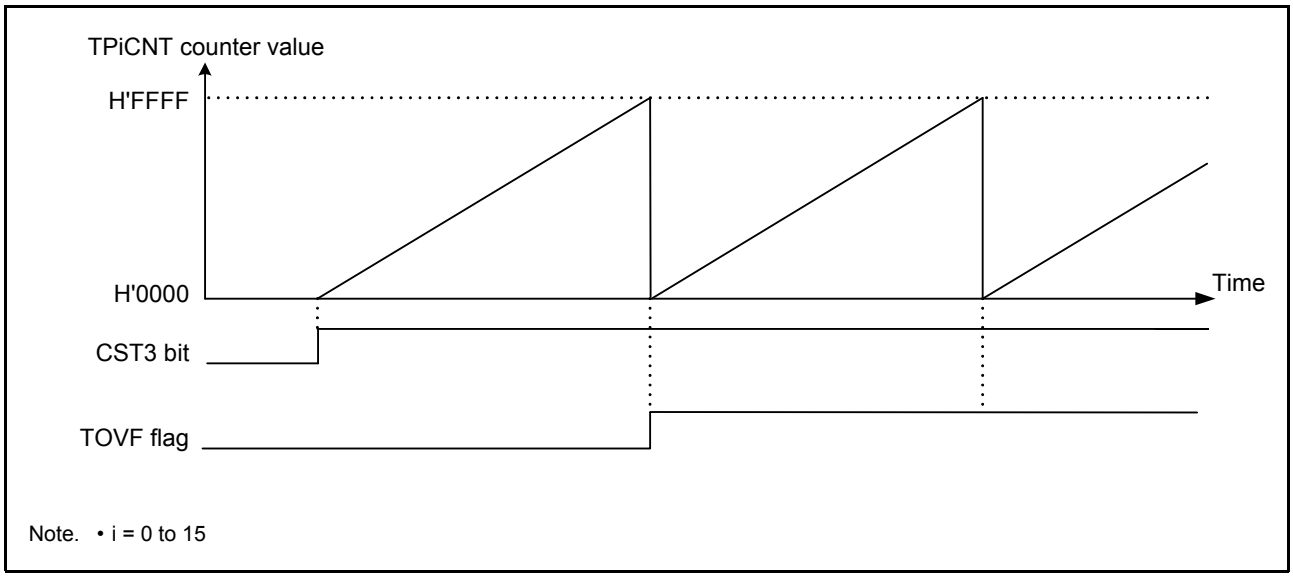


Figure 15.4 Operation Example of TPiCNT Counter as Free-Running Counter

When a compare match is selected as the TPiCNT counter reset source, the TPiCNT counter in the corresponding unit performs a cycle count operation. The TPiCNT counter reset by the compare match is selected by using the TRS bit in the TPiCR register, and the CST3 bit in the TPiCSTR register is set to 1 to increment the corresponding TPiCNT counter as a cycle counter.

The TPiCNT counter is reset to H'0000 and a compare match interrupt request in the TPiGRk register is generated when the counter value corresponds to the TPiGRk register value. The TPiCNT counter continues to be incremented from H'0000 after the compare match. Figure 15.5 shows an Operation Example of TPiCNT Counter as Cycle Counter

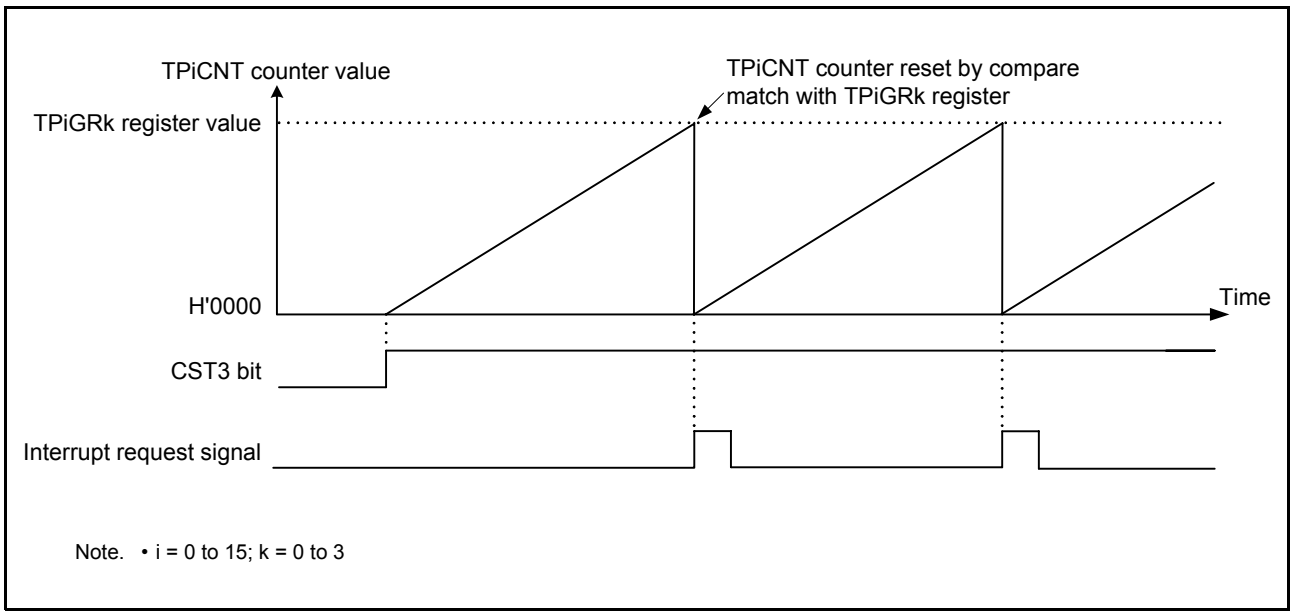


Figure 15.5 Operation Example of TPiCNT Counter as Cycle Counter

15.3.2 TPiCNT Counter Count Timing

Figure 15.6 shows Count Timing of Internal Count Source.

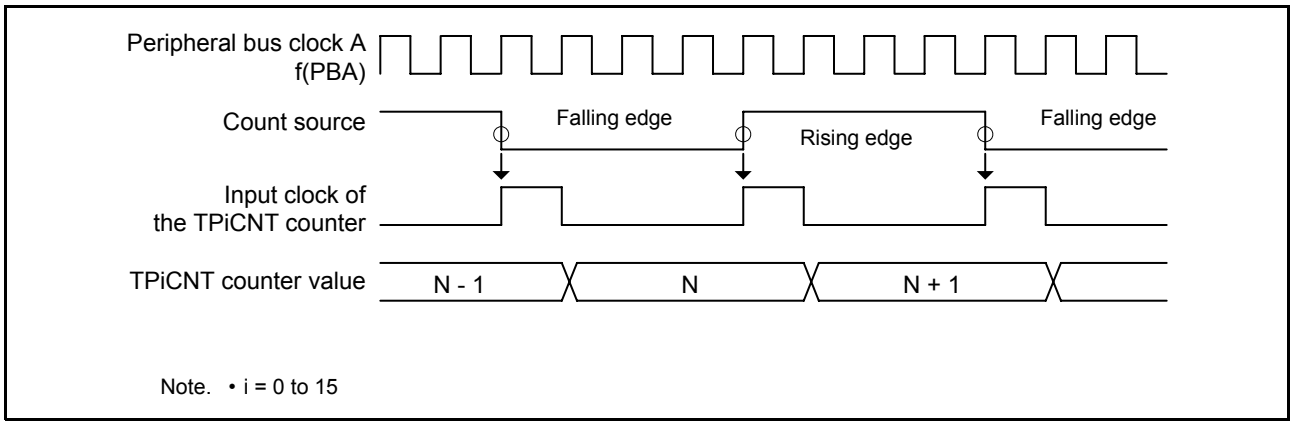


Figure 15.6 Count Timing of Internal Count Source

Figure 15.7 shows TPiCNT Counter Overflow Timing.

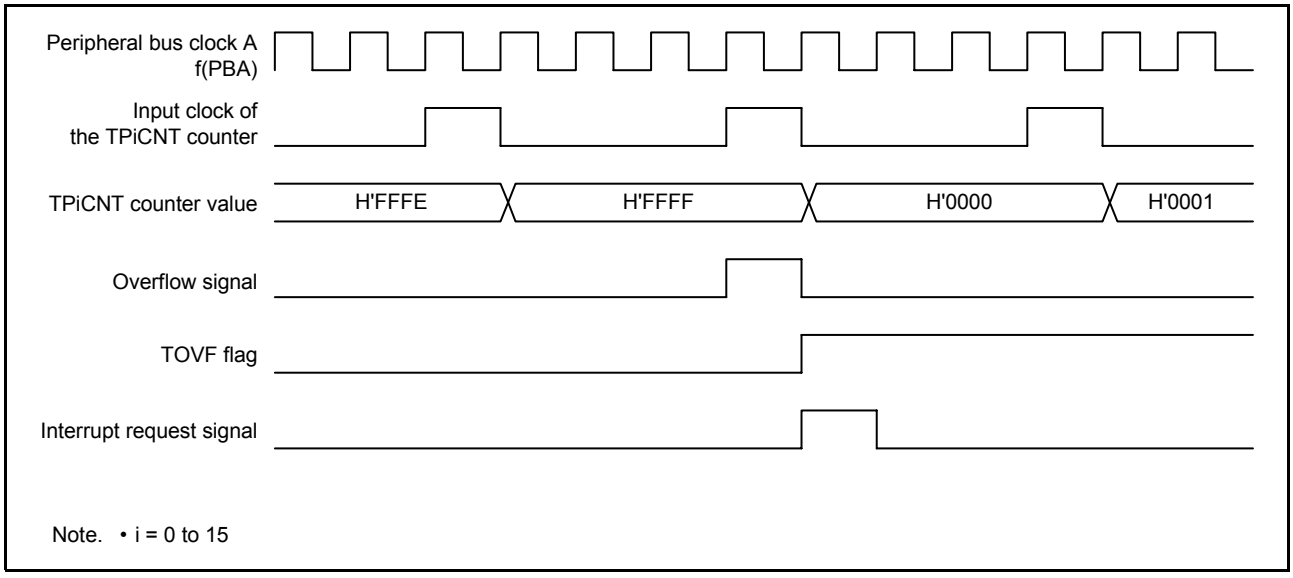


Figure 15.7 TPiCNT Counter Overflow Timing

15.4 Cascade Connection

The cascade connection is a function to connect the TPiCNT counters in two units to operate as a 32-bit counter. With this function, the overflow of the TPiCNT (i + 1) counter can be counted as count source for the TPiCNT counter by setting the CSS bit in the TPiCR register (i = 0, 2, 4, 6, 8, 10, 12, or 14) to B'111 (the overflow of the unit (i + 1)). Table 15.18 lists Cascade Connection Settings.

Table 15.18 Cascade Connection Settings

Combination	16 Low-Order Bits	16 High-Order Bits	Setting
TP0CNT and TP1CNT	TP1CNT	TP0CNT	Set the CSS bit in the TP0CR register to B'111. Set the MD bit in the TP0MD register to B'000 (timer mode).
TP2CNT and TP3CNT	TP3CNT	TP2CNT	Set the CSS bit in the TP2CR register to B'111. Set the MD bit in the TP2MD register to B'000 (timer mode).
TP4CNT and TP5CNT	TP5CNT	TP4CNT	Set the CSS bit in the TP4CR register to B'111. Set the MD bit in the TP4MD register to B'000 (timer mode).
TP6CNT and TP7CNT	TP7CNT	TP6CNT	Set the CSS bit in the TP6CR register to B'111. Set the MD bit in the TP6MD register to B'000 (timer mode).
TP8CNT and TP9CNT	TP9CNT	TP8CNT	Set the CSS bit in the TP8CR register to B'111. Set the MD bit in the TP8MD register to B'000 (timer mode).
TP10CNT and TP11CNT	TP11CNT	TP10CNT	Set the CSS bit in the TP10CR register to B'111. Set the MD bit in the TP10MD register to B'000 (timer mode).
TP12CNT and TP13CNT	TP13CNT	TP12CNT	Set the CSS bit in the TP12CR register to B'111. Set the MD bit in the TP12MD register to B'000 (timer mode).
TP14CNT and TP15CNT	TP15CNT	TP14CNT	Set the CSS bit in the TP14CR register to B'111. Set the MD bit in the TP14MD register to B'000 (timer mode).

In Figure 15.8, the TP0CNT counter is counted by the TP1CNT counter overflows, registers TP0GR0 and TP1GR0 are set as the input capture registers, and rising edges of signals input to pins TP00A and TP01A are set as triggers for the input capture.

The falling edges are simultaneously input to pins TP00A and TP01A to store 16 high-order bits (the value of the TP0CNT counter) into the TP0GR0 register and 16 low-order bits (the value of the TP1CNT counter) into the TP1GR0 register.

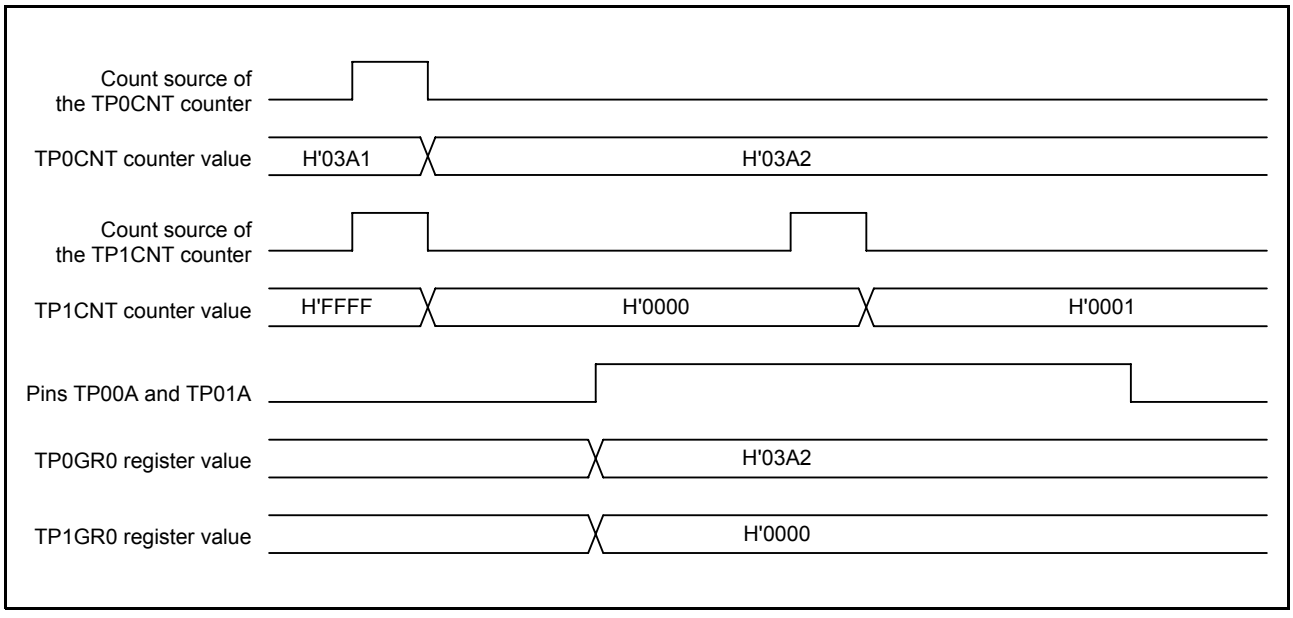


Figure 15.8 Cascade Connection Example

15.5 Input Capture Function

The value of the TPiCNT counter can be transferred to the TPiGRk register by detecting edges of a signal input to the TPU pin. The edges to be detected can be selected from rising edge/falling edge/both edges. Table 15.19 lists Input Capture Specifications and Table 15.20 lists Register Settings Associated with Input Capture Function.

Table 15.19 Input Capture Specifications

Item	Description
Measurement channel	Channels 0, 1, 2, and 3
Trigger input	Rising edge/falling edge/both edges of the TPjx pin
Measurement start condition	Set the IOk bit in the TPiIOCR register as the trigger source for the input capture.
Measurement stop condition	Set the IOk bit in the TPiIOCR register to the value except the trigger source for the input capture.
Time measurement timing	Every time a trigger is input
Interrupt request generation timing	Trigger input timing
TPjx pin function	Trigger input
Selectable function	• Digital filter function The input level of the TPjx pin is determined for every count source cycle of the TPiCNT counter and the signals holding the same level during three sequential cycles are passed as the trigger input level. • Buffer function The TPiGR2 register and the TPiGR3 register can be used as the buffer register for the TPiGR0 register and the TPiGR1 register, respectively. When the trigger is input, the value of the TPiCNT counter is transferred to the TPiGR0 or TPiGR1 register, and at the same time the value stored in the TPiGR0 register and the TPiGR1 register are transferred to the TPiGR2 register and the TPiGR3 register, respectively.

Note: • i = 0 to 15; k = 0 to 3; j = 00 to 15; x = A, B, C, or D

Table 15.20 Register Settings Associated with Input Capture Function

Register	Bit	Setting
TPiIOCR	IOk	Select the trigger.
TPiDF	—	Select the digital function filter.
TPiMD	GR2FS, GR3FS	Buffer select function

Note: • i = 0 to 15; k = 0 to 3

Figure 15.9 shows an Input Capture Operation Example set as below:

- Select the rising edge and falling edge of the TPjA pin.
- Select the falling edge of the TPjB pin.
- Reset the TPiCNT counter at the falling edge of the TPjB pin.

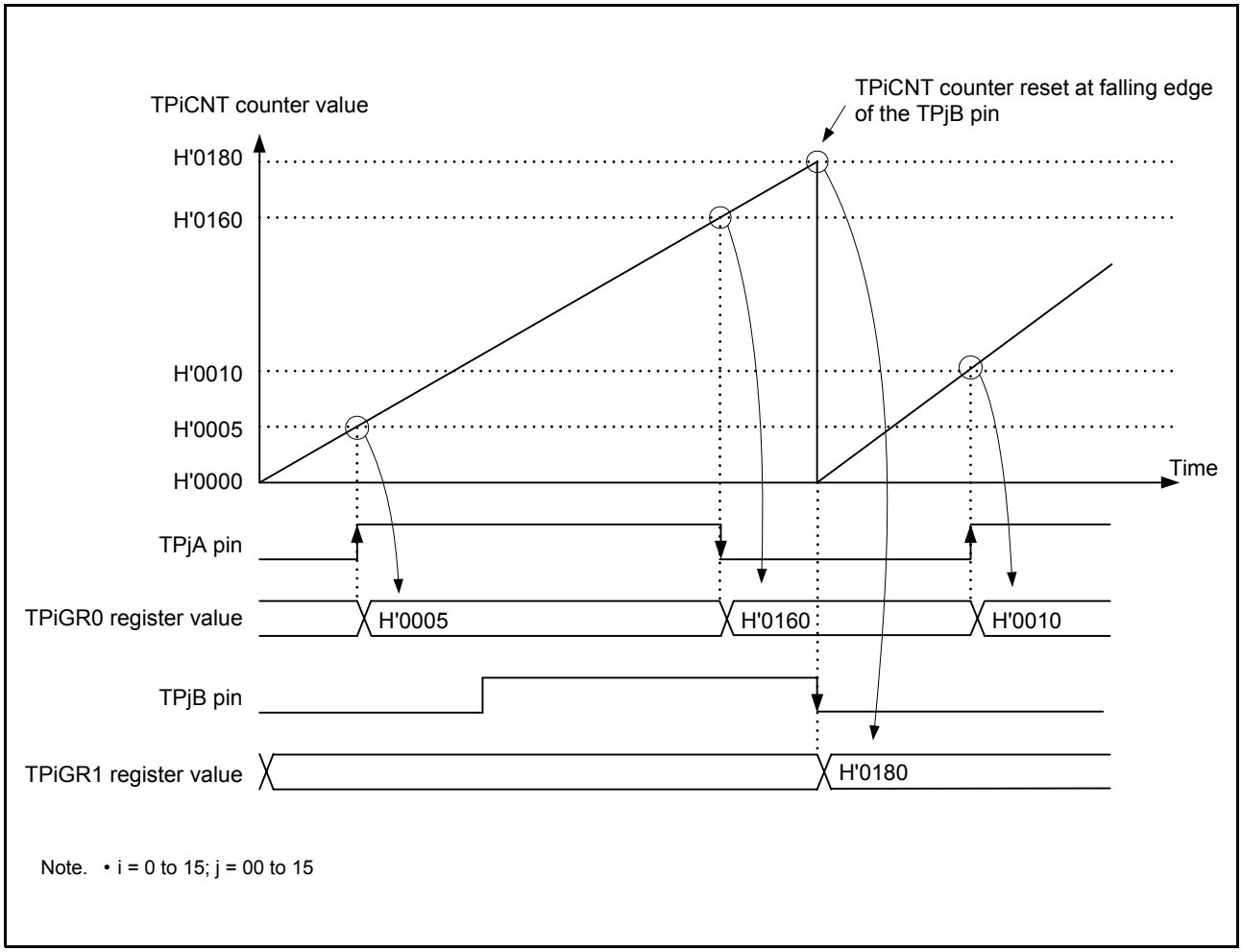
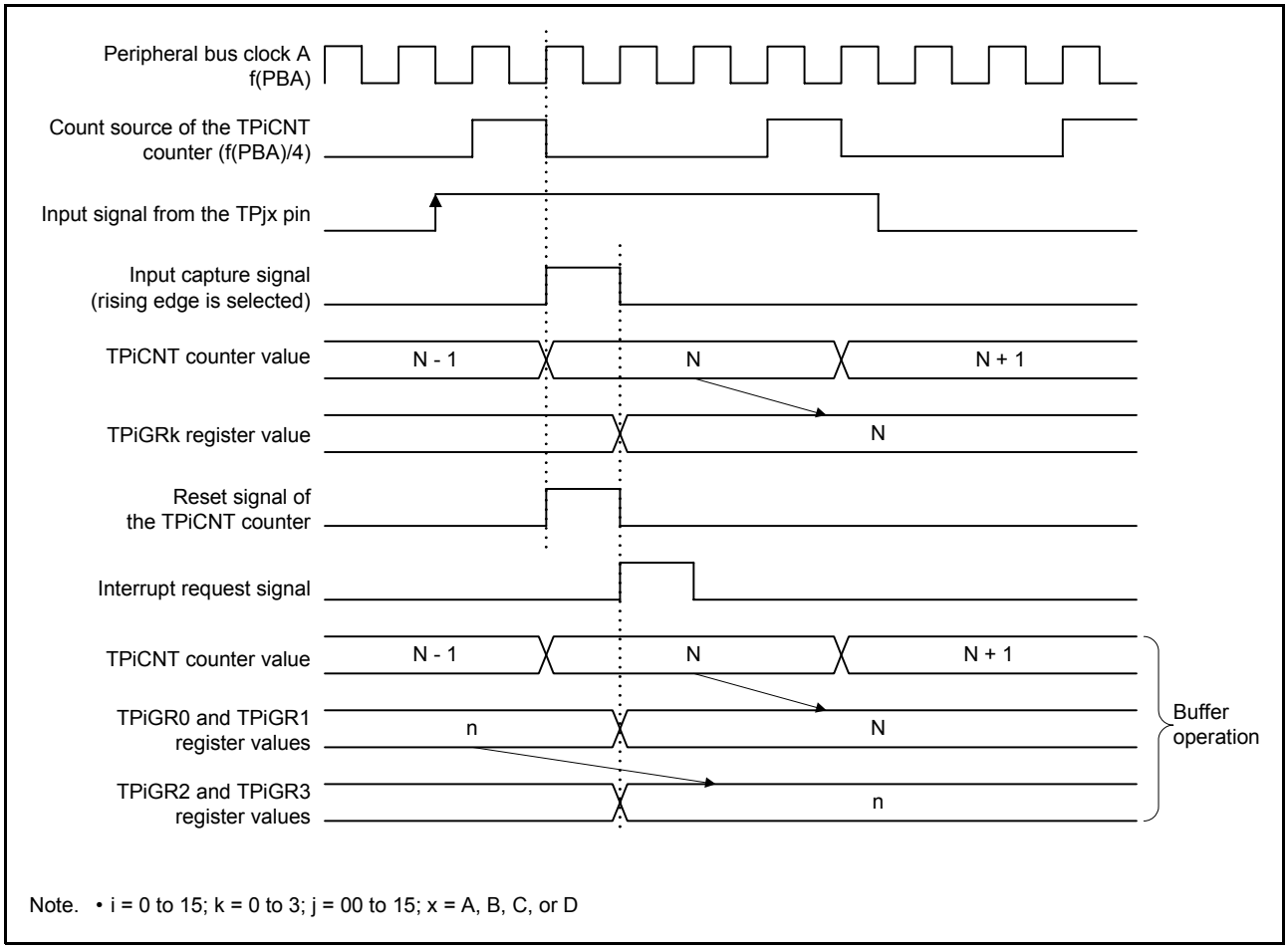


Figure 15.9 Input Capture Operation Example

Figure 15.10 shows Input Capture Timing.



Note. • i = 0 to 15; k = 0 to 3; j = 00 to 15; x = A, B, C, or D

Figure 15.10 Input Capture Timing

15.5.1 Buffer Function

This function can be used in the input capture and output compare. In this section, the buffer function is used in the input capture. Table 15.21 lists Register Combinations and Table 15.22 lists Buffer Associated Register Settings.

Table 15.21 Register Combinations

TPU General Register	Buffer Register
TPiGR0	TPiGR2
TPiGR1	TPiCR3

Note: • i = 0 to 15

Table 15.22 Buffer Associated Register Settings

Register	Bit	Description
TPiIOCR	IOk	Select the trigger for the input capture.
TPiMD	GR2FS, GR3FS	Select the buffer register.

Note: • i = 0 to 15; k = 0 to 3

In Figure 15.11, when a trigger is input from the TPjA pin, the value of the TPiCNT counter is transferred to the TPiGR0 register, and at the same time the value stored in the TPiGR0 register is transferred to the TPiGR2 register.

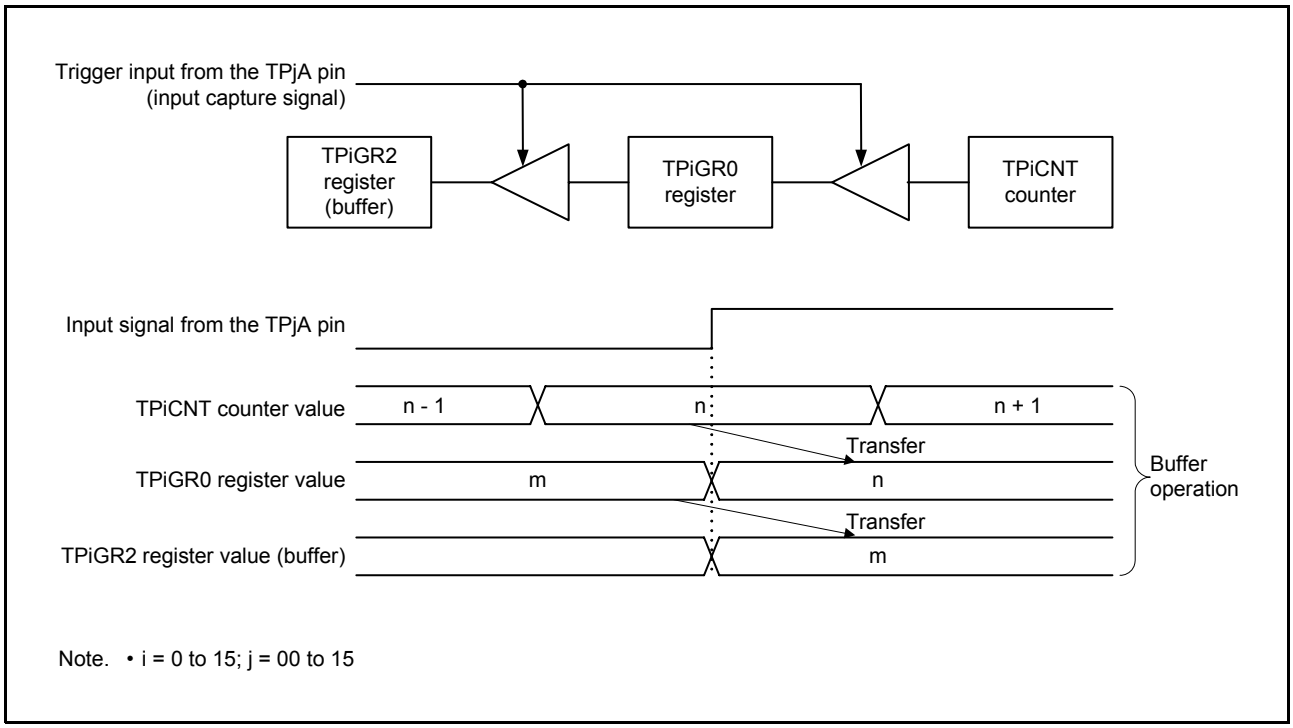


Figure 15.11 Buffer Operation Example

Figure 15.12 shows an application of the buffer function set as below:

- Set both edges input to the TPjA pin as the input capture trigger.
- Reset the TPiCNT counter by the input capture.
- Use registers TPiGR0 and TPiGR2 for the buffer function.

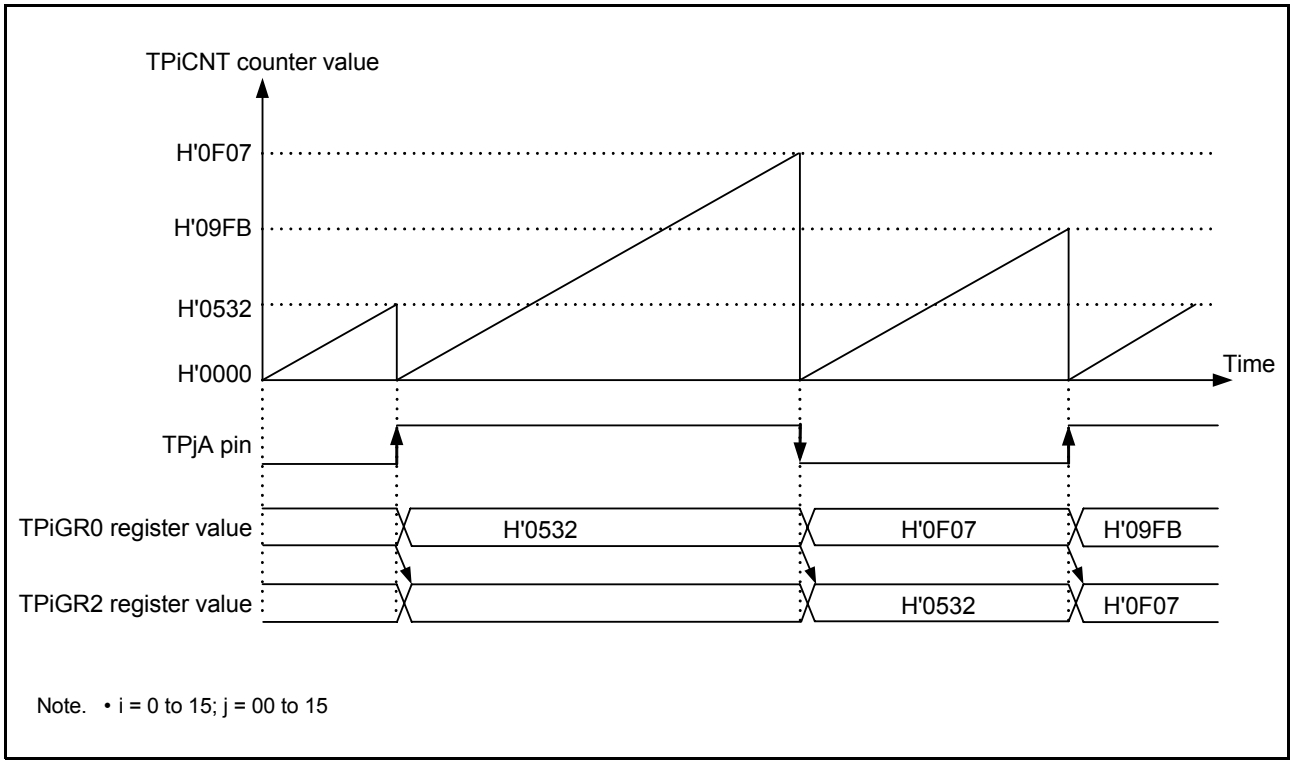


Figure 15.12 Buffer Function Application Example

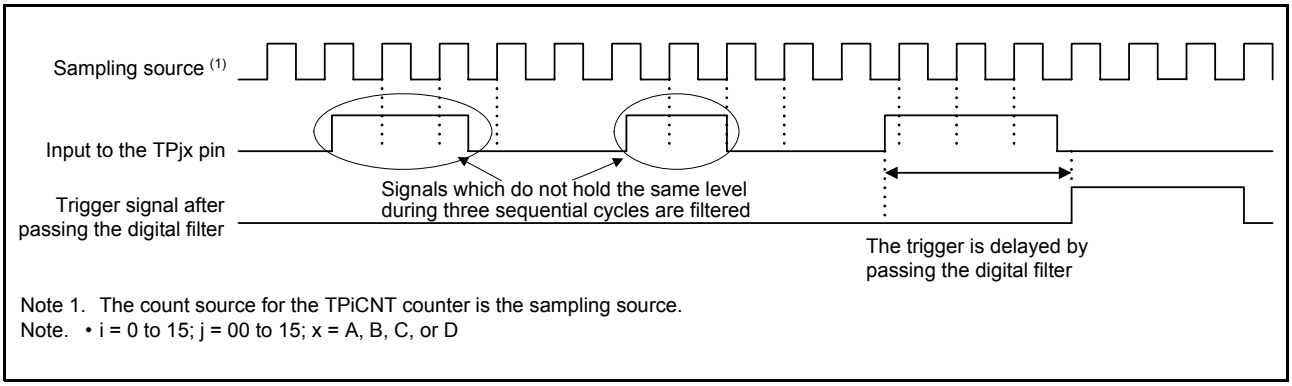
15.5.2 Digital Filter Function

This function determines the input level of the TPjx pin for every count source cycle of the TPiCNT counter and passes the signals holding the same level during three sequential cycles as the trigger input level. Table 15.23 lists Register Settings Associated with Digital Filter Function, and Figure 15.13 shows Trigger Signal When Using Digital Filter.

Table 15.23 Register Settings Associated with Digital Filter Function

Register	Bit	Description
TPiDF	DFEk	Select the digital filter function.

Note: • i = 0 to 15; k = 0 to 3



Note 1. The count source for the TPiCNT counter is the sampling source.
Note. • i = 0 to 15; j = 00 to 15; x = A, B, C, or D

Figure 15.13 Trigger Signal When Using Digital Filter

15.6 Output Compare Function

Waveforms are output from the TPjx pin by the compare match between the TPiCNT counter and the TPiGRk register. The output compare has the following two modes:

- Single-phase waveform output mode
- PWM mode

15.6.1 Single-Phase Waveform Output Mode

The value set by the IOk bit in the TPiIOCR register is output from the TPjx pin when the value of the TPiCNT counter matches that of the TPiGRk register. The output level can be selected from the low level output/high level output/toggle output. Table 15.24 lists Specifications of Single-Phase Waveform Output Mode and Table 15.25 lists Register Settings Associated with Output Compare Function.

Table 15.24 Specifications of Single-Phase Waveform Output Mode

Item	Description
Output waveform	• Free-running operation Set the TSRE bit in the TPiPSCR1 register to 0. When the TRS bit in the TPiCR register is B'000 or B'100 (reset is not used) Cycle: $\frac{65536}{T}$ Initial value level: $\frac{n}{T}$ T: The count source for the TPiCNT counter n: The setting value of the TPiGRk register (H'0000 to H'FFFF) • The TPiCNT counter is reset when the value of the TPiCNT counter matches that of the TPiGRk register. Cycle: $\frac{n + 1}{T}$ T: The count source for the TPiCNT counter n: The setting value of the TPiGRk register (H'0000 to H'FFFF)
Waveform output start condition	The IOk bit in the TPiIOCR register is set to the output compare value.
Waveform output stop condition	The IOk bit in the TPiIOCR register is set to the value except the output compare
Interrupt request generation timing	When the compare match between the TPiCNT counter and the TPiGRk register occurs
TPjx pin	Pulse signal output
Selectable function	• Buffer function The TPiGR2 register and the TPiGR3 register can be used as the buffer register for the TPiGR0 register and the TPiGR1 register, respectively. When the compare match occurs, the values stored in the TPiGR2 register and the TPiGR3 register are transferred to the TPiGR0 register and the TPiGR1 register, respectively.

Note: • i = 0 to 15; k = 0 to 3

Table 15.25 Register Settings Associated with Output Compare Function

Register	Bit	Setting
TPiIOCR	IOk	Select the initial output value and the output level when the compare match occurs.
TPiMD	GR2FS, GR3FS	Buffer select function

Note: • i = 0 to 15; k = 0 to 3

Figure 15.14 shows Operation Example of Low Level Output/High Level Output.
In figure 15.14, the TPiCNT counter operates as a free-running counter, 1 is output from the TPjA pin by the compare match between the TPiCNT counter and the TPiGR0 register, and 0 is output from the TPjB pin by the compare match between the TPiCNT counter and the TPiGR1 register. The pin level is fixed by matching the output level set by the IOk bit in the TPiIOCR register with the pin output level.

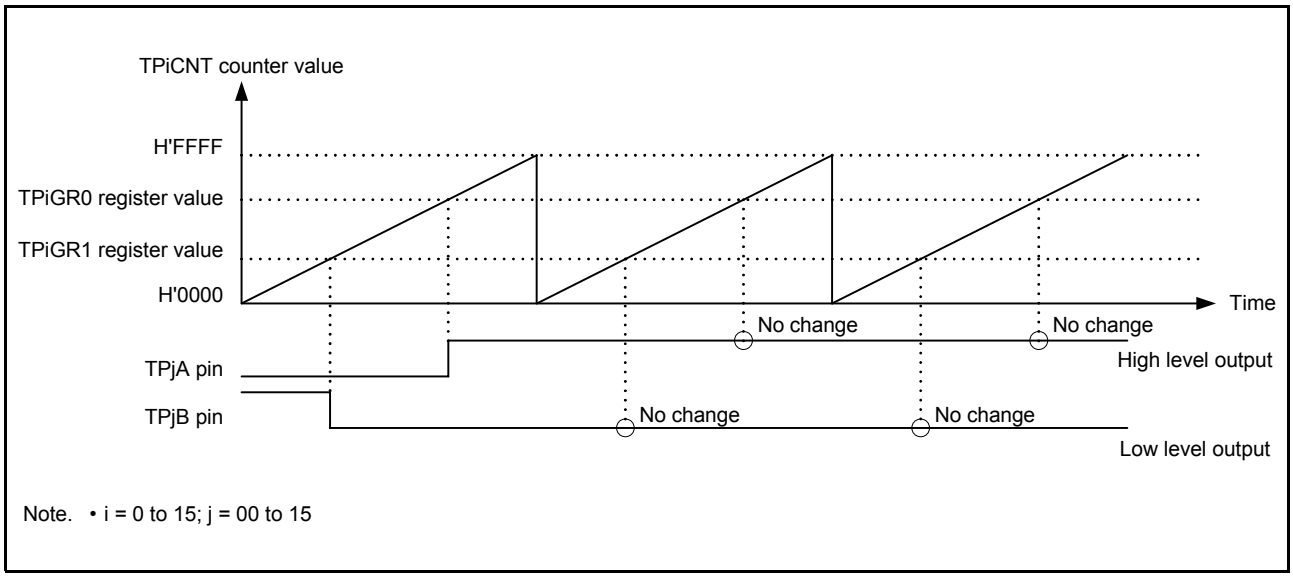


Figure 15.14 Operation Example of Low Level Output/High Level Output

Figure 15.15 shows a Toggle Output Operation Example.
In Figure 15.15, the TPiCNT counter operates as a cycle counter (TPiCNT counter reset by the compare match with the TPiGR1 register), and the output level is set to the toggle output by the compare match between the TPiGR0 register and the TPiGR1 register.

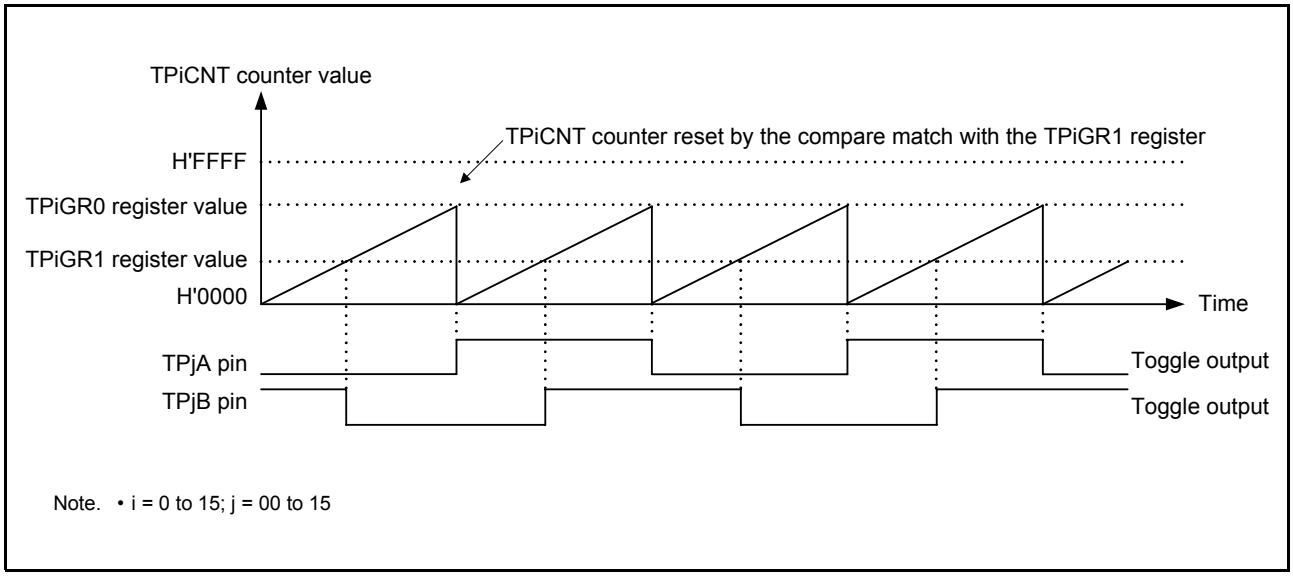


Figure 15.15 Toggle Output Operation Example

Figure 15.16 shows Output Compare Timing.

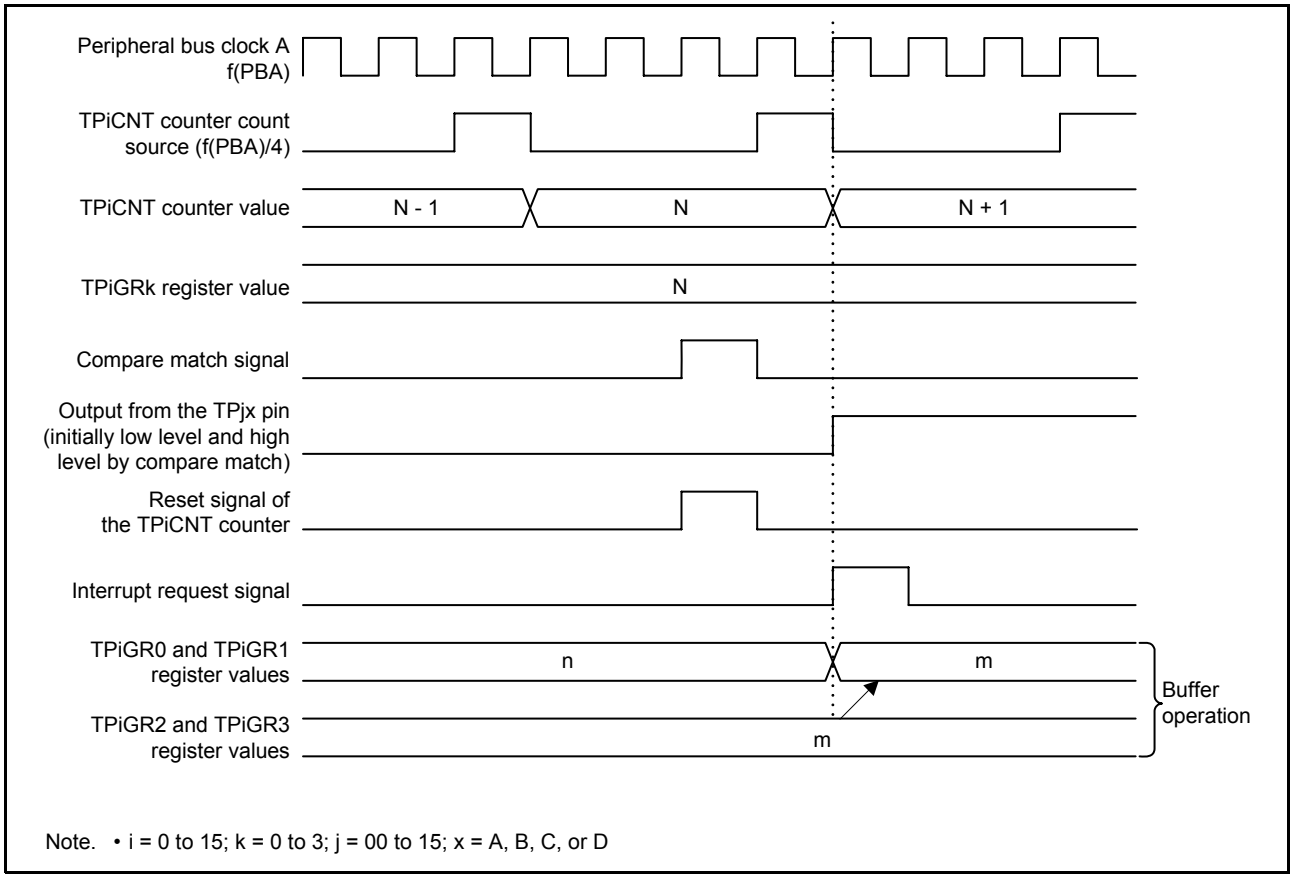


Figure 15.16 Output Compare Timing

15.6.2 Buffer Function

This function can be used in the input capture and output compare. In this section, the buffer function is used in the output compare.

Table 15.26 lists Register Combinations and Table 15.27 lists Buffer Associated Register Settings.

Table 15.26 Register Combinations

TPU General Register	Buffer Register
TPiGR0	TPiGR2
TPiGR1	TPiCR3

Note: • i = 0 to 15

Table 15.27 Buffer Associated Register Settings

Register	Bit	Description
TPiIOCR	IOk	Select the initial output value and the output level when the compare match occurs.
TPiMD	GR2FS, GR3FS	Select the buffer function.

Note: • i = 0 to 15; k = 0 to 3

Figure 15.17 shows a Buffer Operation Example.

In Figure 15.17, the value of the TPiGR2 register is transferred to the TPiGR0 register when the compare match between the TP0CNT counter and the TPiGR0 register occurs.

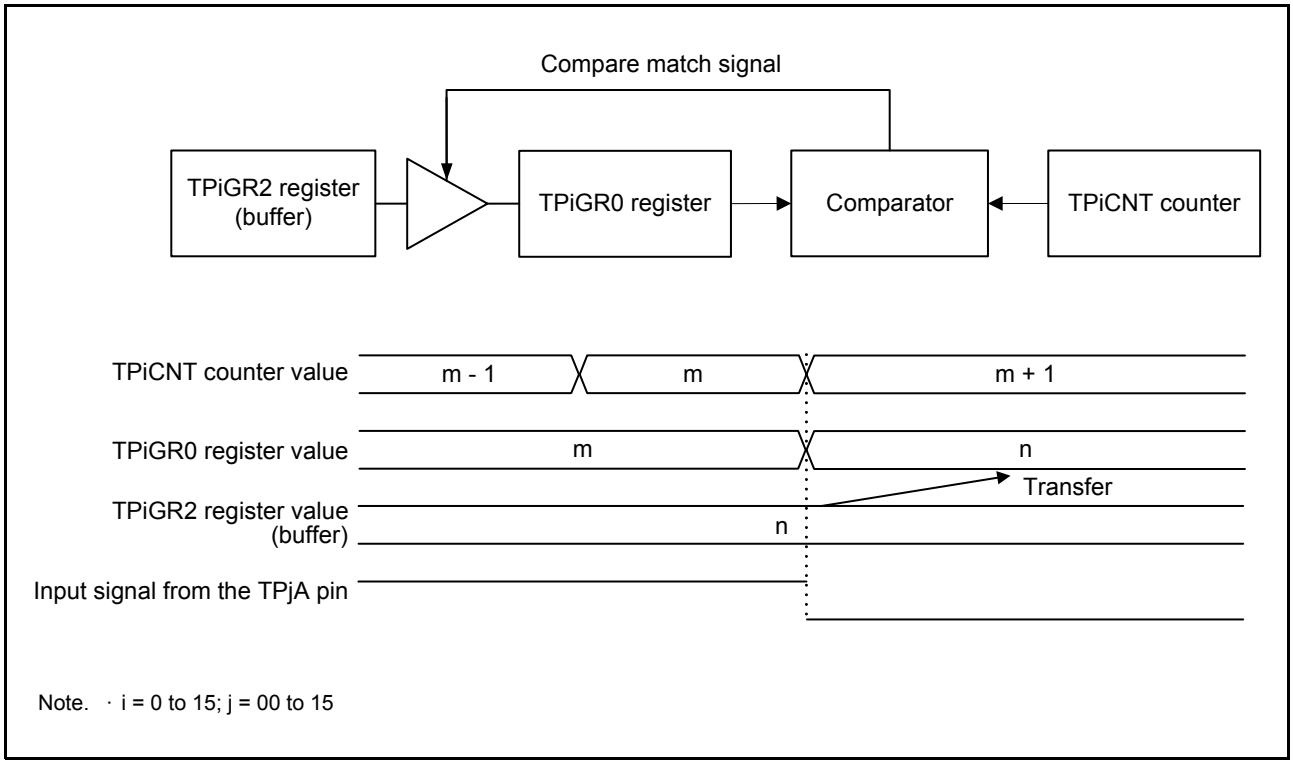


Figure 15.17 Buffer Operation Example

Figure 15.18 shows an application of the buffer function set as below:

- Set to PWM mode 1.
- The TPiCNT counter is reset by the compare match between the TPiCNT counter and the TPiGR1 register.
- The output level from the TPjA pin becomes high by the compare match between the TPiCNT counter and the TPiGR0 register.
- Use registers TPiGR0 and TPiGR2 for the buffer function.

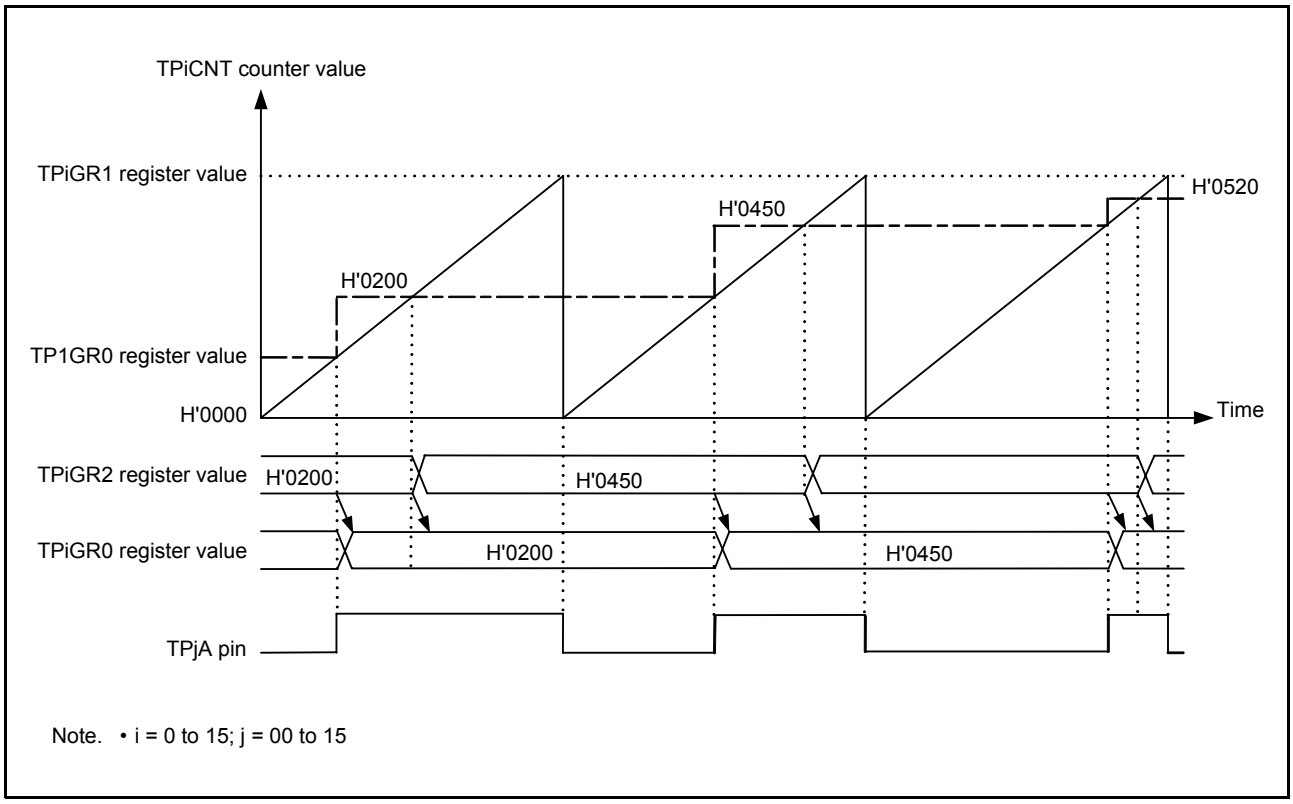


Figure 15.18 Buffer Function Application Example

15.6.3 PWM Mode

PWM mode outputs PWM waveforms at the TPjx pin and has two modes, PWM mode 1 and PWM mode 2. The output level can be selected from the low level output/high level output/toggle output, and the PWM waveform with duty rates from 0% to 100% can be output.

A cycle can be set by setting the compare match between the TPiCNT counter and the TPiGRk register as a counter reset source.

(1) PWM Mode 1

PWM mode 1 uses two TPiGRk registers and generates PWM waveforms. In PWM mode 1, PWM waveforms of up to thirty-two channels can be generated.

Table 15.28 lists TPiGRk Register Combinations and Output Pins.

Table 15.28 TPiGRk Register Combinations and Output Pins

TPiCNT Counter	TPiGRk Register	Output Pin
TPiCNT	TPiGR0	TPjA
	TPiGR1	
	TPiGR2	TPjC
	TPiGR3	

Note: • i = 0 to 15; k = 0 to 3; j = 00 to 15

Table 15.29 lists Specifications of PWM Mode 1.

Table 15.29 Specifications of PWM Mode 1

Item	Description
Output waveform *1	• Free-running operation *2 Set the TSRE bit in the TPiPSCR1 register to 0. When the TRS bit in the TPiCR register is B'000 or B'100 (reset is not used) High level width: $\frac{n - m}{T}$ Low level value: $\frac{m}{T}$ *3 + $\frac{65536 - n}{T}$ *4 T: The count source for the TPiCNT counter m: Initially low level output, and high level output by the compare match The setting value of the TPiGRk register (H'0000 to H'FFFF) n: Initially low level output, and high level output by the compare match The setting value of the TPiGRk register (H'0000 to H'FFFF) • The TPiCNT counter is reset when the value of the TPiCNT counter matches the value of the TPiGRk register. Cycle: $\frac{n + 1}{T}$ T: The count source for the TPiCNT counter n: The setting value of the TPiGRk register (H'0000 to H'FFFF)
Waveform output start condition	• The IOk bit in the TPiIOCR register is set to the output compare value. • The MD bit in the TPiMD register is set to B'010 (PWM mode 1).
Waveform output stop condition	The IOk bit in the TPiIOCR register is set to the value except the output compare.
Interrupt request generation timing	When the compare match between the TPiCNT counter and the TPiGRk register occurs
TPjx pin	Pulse signal output
Selectable function	• Buffer function The TPiGR2 register and the TPiGR3 register can be used as the buffer register for the TPiGR0 register and the TPiGR1 register, respectively. When the compare match occurs, the values stored in the TPiGR2 register and the TPiGR3 register are transferred to the TPiGR0 register and the TPiGR1 register, respectively.

Notes: 1. If the values of the TPiGRk registers used in pairs are same, the output level is fixed even though the compare match with the TPiCNT counter occurs.
2. Figure 15.4 shows an operation example.
3. Time from when the TPiCNT counter is reset until when the output level becomes high.
4. Time from when the output level becomes low until when the TPiCNT counter is reset.

Note: • i = 0 to 15; k = 0 to 3; j = 00 to 15; x = A, B, C, or D

Table 15.30 lists Register Settings Associated with PWM Mode 1.

Table 15.30 Register Settings Associated with PWM Mode 1

Register	Bit	Setting
TPiIOCR	IOk	Select the initial output value and the output level when the compare match occurs.
TPiMD	MD	Select PWM mode 1.
	GR2FS, GR3FS	Buffer select function

Note: • i = 0 to 15; k = 0 to 3

Figure 15.19 shows an Operation Example of PWM Mode 1.

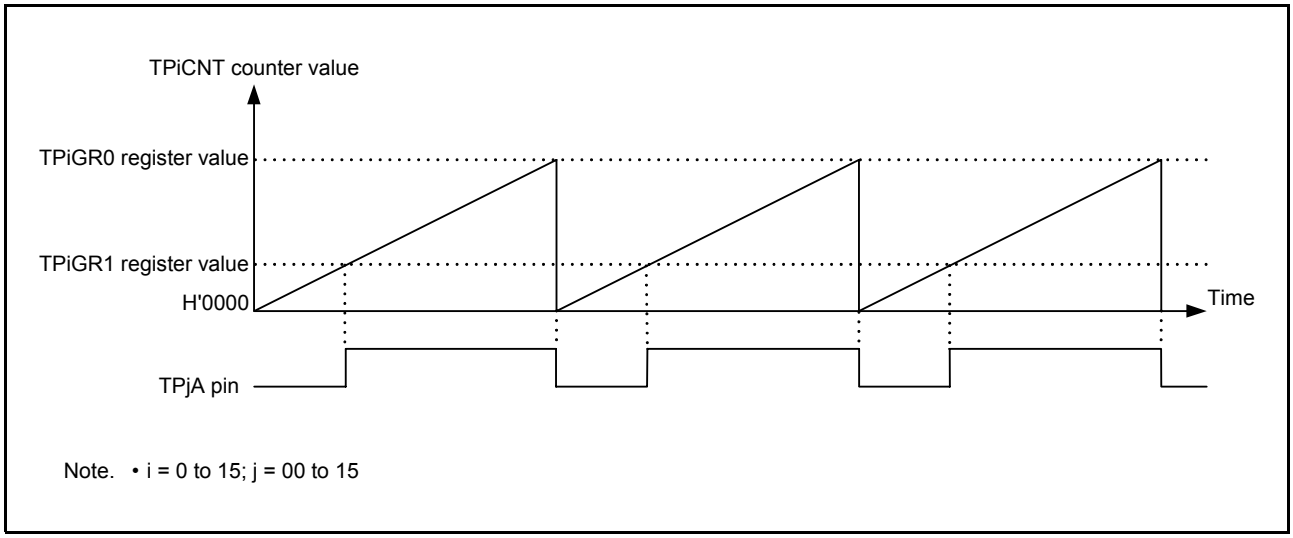


Figure 15.19 Operation Example of PWM Mode 1

(2) PWM Mode 2

PWM mode 2 generates PWM waveforms by using the TPiCNTRSE register as the cycle setting register and the TPiGRk register as the waveform generating register. In PWM mode 2, PWM outputs of up to sixty-four channels are possible.

Table 15.31 lists correspondences between the TPiGRk register and output pins.

Table 15.31 TPiGRk Register Combinations and Output Pins

TPiCNT Counter	TPiGRk	Output Pin
TPiCNT	TPiGR0	TPjA
	TPiGR1	TPjB
	TPiGR2	TPjC
	TPiGR3	TPjD

Note: • i = 0 to 15; k = 0 to 3; j = 00 to 15

Table 15.32 lists Specifications of PWM Mode 2.

Table 15.32 Specifications of PWM Mode 2

Item	Description
Output waveform	The TPiCNT counter is reset when the value of the TPiCNT counter matches that of the TPiCNTRSE register. Cycle: $\frac{n + 1}{T}$ Initial output level width: $\frac{m}{T}$ Inverted level width: $\frac{n - m}{T}$ T: The count source for the TPiCNT counter m: The setting value of the TPiGRk register (H'0000 to H'FFFF) n: The setting value of the TPiCNTRSE register used for the cycle setting (H'0000 to H'FFFF)
Waveform output start condition	• The IOk bit in the TPiIOCR register is set to the output compare value. • The MD bit in the TPiMD register is set to B'011 (PWM mode 2).
Waveform output stop condition	The IOk bit in the TPiIOCR register is set to the value except the output compare value.
Interrupt request generation timing	When the compare match between the TPiCNT counter and the TPiGRk register occurs
TPjx pin	Pulse signal output
Selectable function	• Buffer function The TPiGR2 register and the TPiGR3 register can be used as the buffer register for the TPiGR0 register and the TPiGR1 register, respectively. When the compare match occurs, the values stored in the TPiGR2 register and the TPiGR3 register are transferred to the TPiGR0 register and the TPiGR1 register, respectively. • Phase shift mode (Channel shift mode) PWM waveforms are output synchronously with the phase shift clock for each channel

Note: • i = 0 to 15; k = 0 to 3; j = 00 to 15; x = A, B, C, or D

Table 15.33 lists Register Settings Associated with PWM Mode 2.

Table 15.33 Register Settings Associated with PWM Mode 2

Register	Bit	Setting
TPiIOCR	IOk	Select the initial output value and the output level when the compare match occurs.
TPiMD	MD	Select PWM mode 2.
	GR2FS, GR3FS	Buffer select function
TPiPSCDV	—	Set the phase shift time.
TPiPSCR0	PSM	Set the phase shift mode.
TPiCNTRSE	—	Cycle setting

Note: • i = 0 to 15; k = 0 to 3

Figure 15.20 shows an Operation Example of PWM Mode 2.

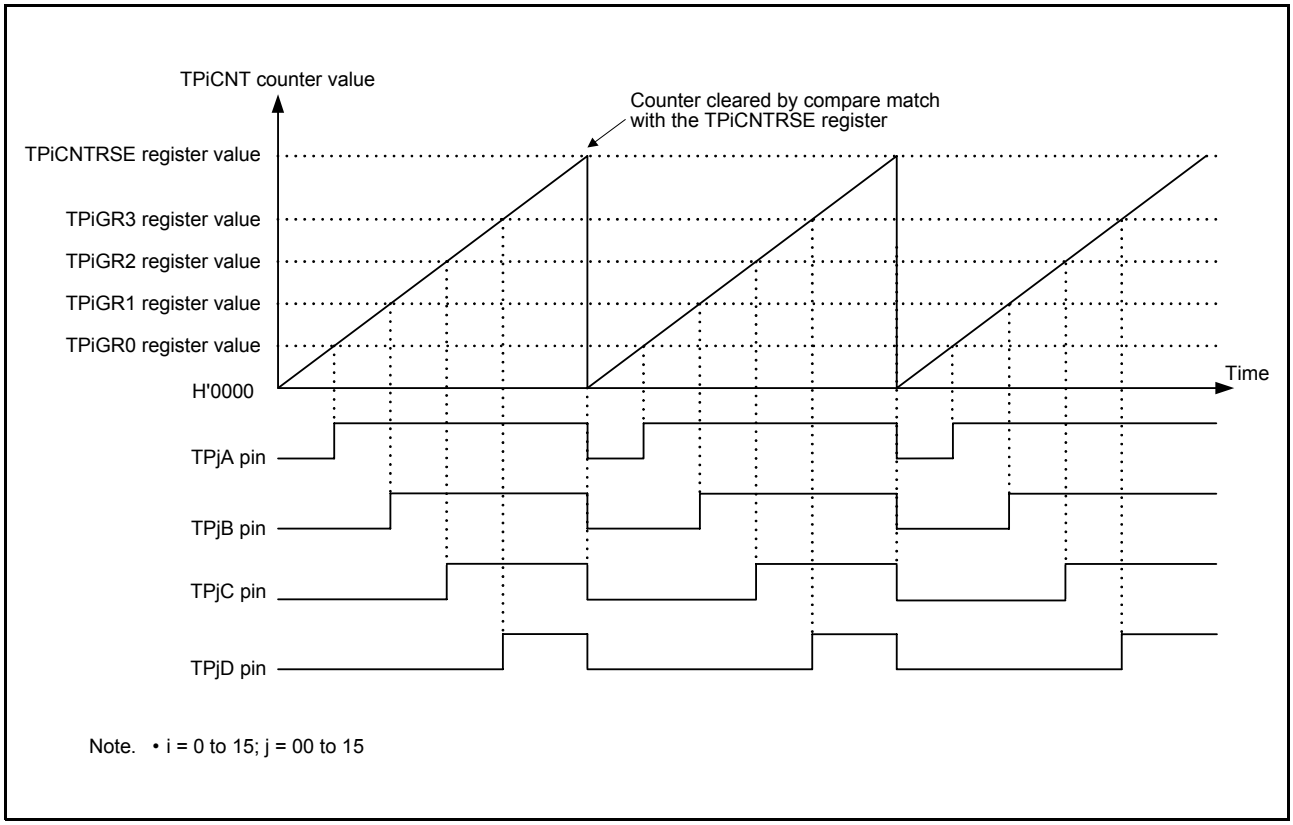


Figure 15.20 Operation Example of PWM Mode 2

15.6.4 Phase Shift Mode (Channel Shift Mode)

This function shifts and outputs PWM waveforms, and enables to reduce switching noise and instantaneous current consumption. The function is supposed to operate the TPU under the following conditions:

- Set all channels to PWM mode 2. The TPiPSPSE register is the cycle setting register when using the phase shift function
- Set bits TSRE3 and TSRE5 in the TPiPSCR1 register to 1
- Set the count sources in registers TPiCR and TPiPSCNTCR to the same frequency
- Set the TRS bit in the TPiPSCNTCR register to B'10 for the TPiPSCNT counter reset source.

This function enables to output the low level initially, output the high level by shifting a PWM waveform per channel by 1 cycle of the phase shift clock, and output the low level by using the output compare. Output ordering is TPjA, TPjB, TPjC, and TPjD. The phase shift clock is the clock with the TPiPSCNT counter count source divided by the value set in the TPiPSCDV register.

Figure 15.21 shows a Block Diagram of Phase Shift Mode.

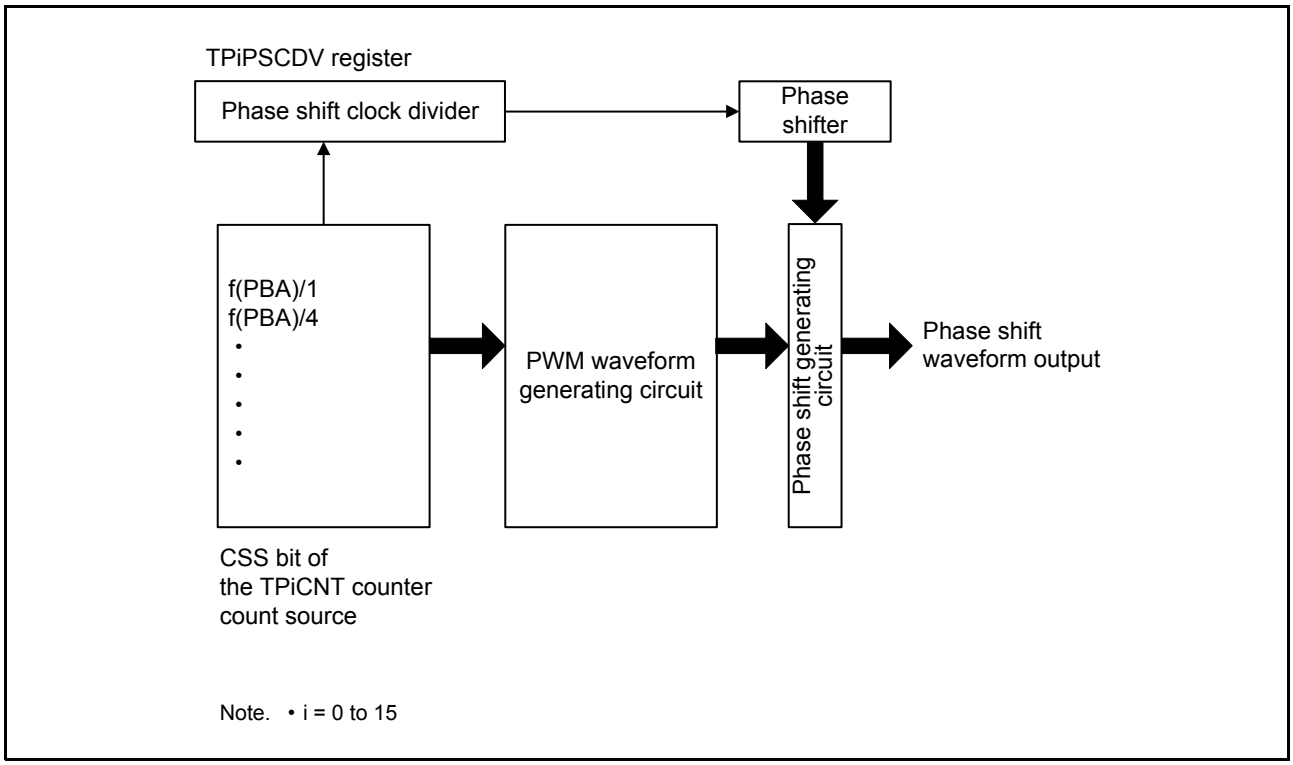


Figure 15.21 Block Diagram of Phase Shift Mode

Table 15.34 lists Register Settings Associated with Channel Shift Mode.

Table 15.34 Register Settings Associated with Channel Shift Mode

Register	Bit	Setting
TPiCSTR	CST3, CST5	Select whether counters TPiCNT and TPiPSCNT start or stop counting.
TPiMD	MD	Select PWM mode 2.
TPiPSCR0	PSM	Select channel shift mode.
TPiCR, TPiPSCNTCR	CSS, CES, TRS	Select the count sources, the count edges, and the reset sources for counters TPiCNT and TPiPSCNT.
TPiGRk	—	Set the output waveform width.
TPiIOCR	IOk	Set to B'0101 (initially high level output, and low level output by compare match).
TPiPSCR1	TSRE	Select phase shift mode.
TPiPSCDV	—	Set the phase shift clock division.

Note: • i = 0 to 15; k = 0 to 3

Figure 15.22 shows TPiPSCNT Counter Operation Example.
When the compare match between the TPiPSPSE register and the TPiPSCNT counter is selected by the TRS bit in the TPiPSCNTCR register and 1 is set to the CST5 bit in the TPiCSTR register, the corresponding TPiPSCNT counter starts being incremented.
When the count value matches the TPiPSPSE register, the TPiPSCNT counter is reset to H'0000.

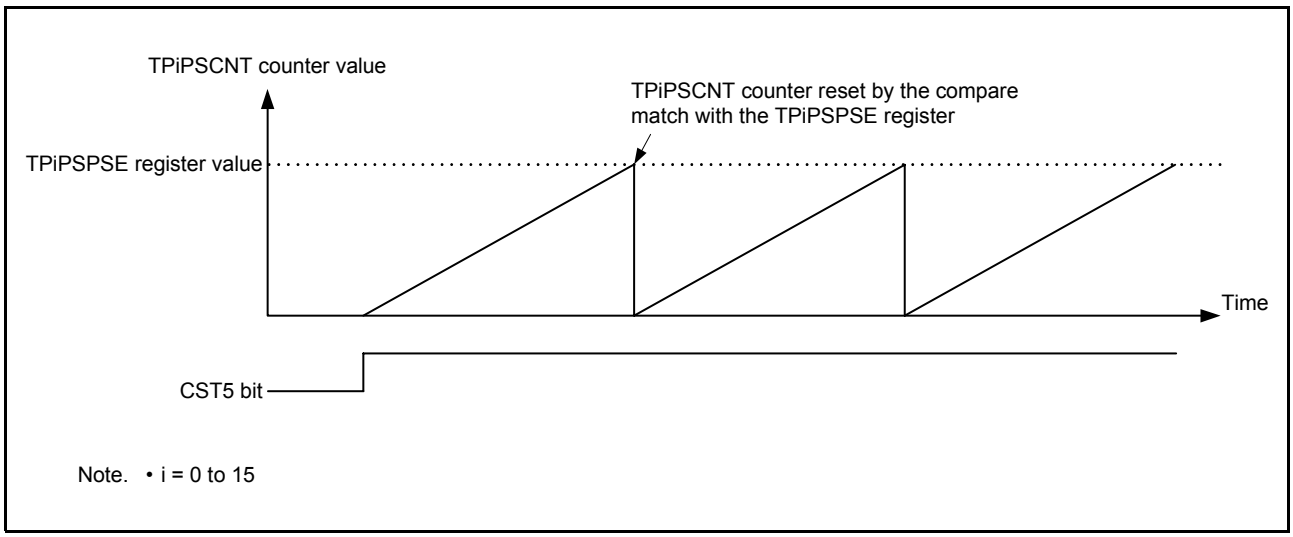


Figure 15.22 TPiPSCNT Counter Operation Example

In phase shift mode, the TPiCNT counter is also reset by the compare match between the TPiPSCNT counter and the TPiPSPSE register shown in Figure 15.22.

Figure 15.23 shows an Operating Procedure for Channel Shift Mode.

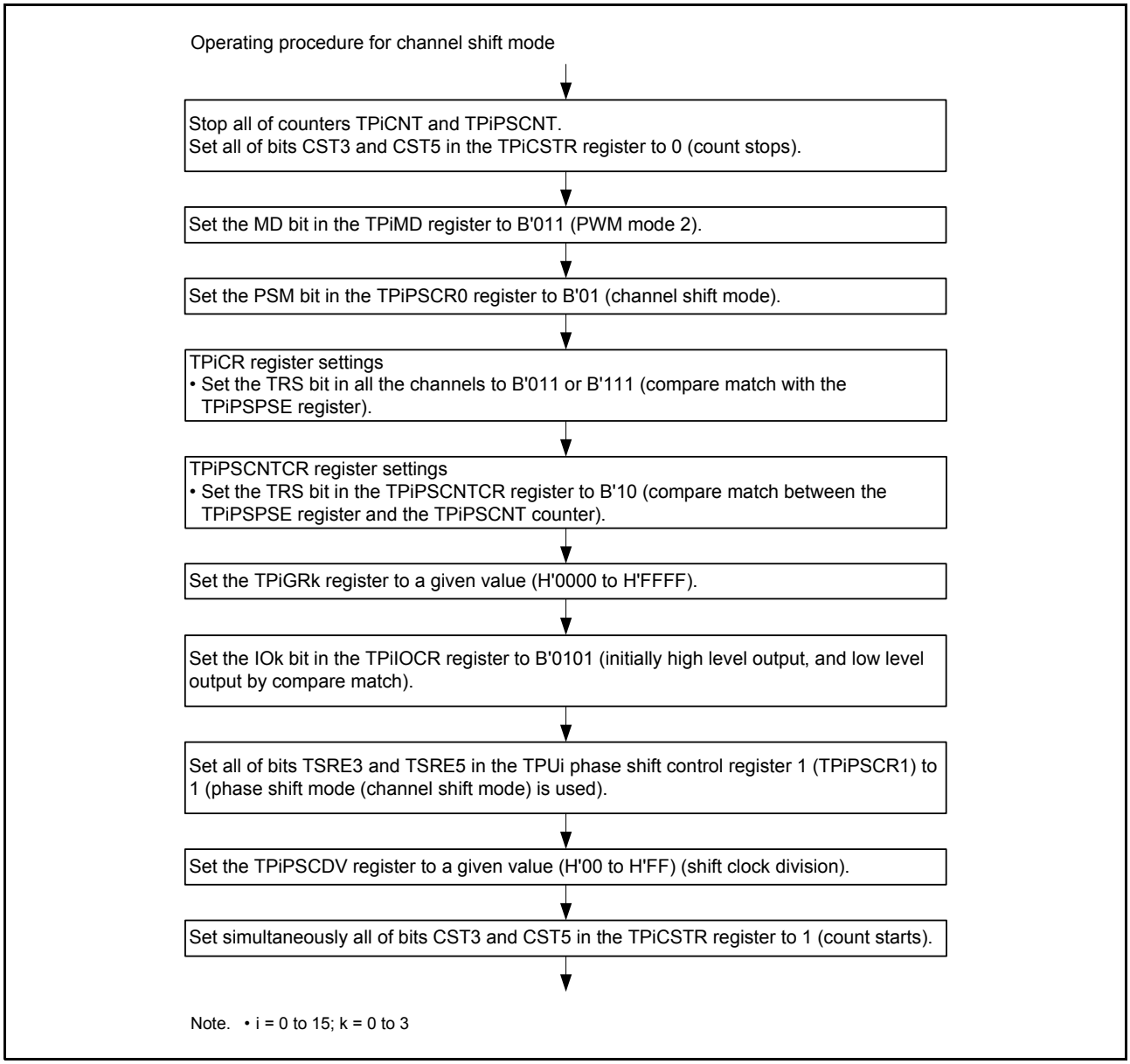


Figure 15.23 Operating Procedure for Channel Shift Mode

Figure 15.24 shows an Operation Example of Channel Shift Mode.

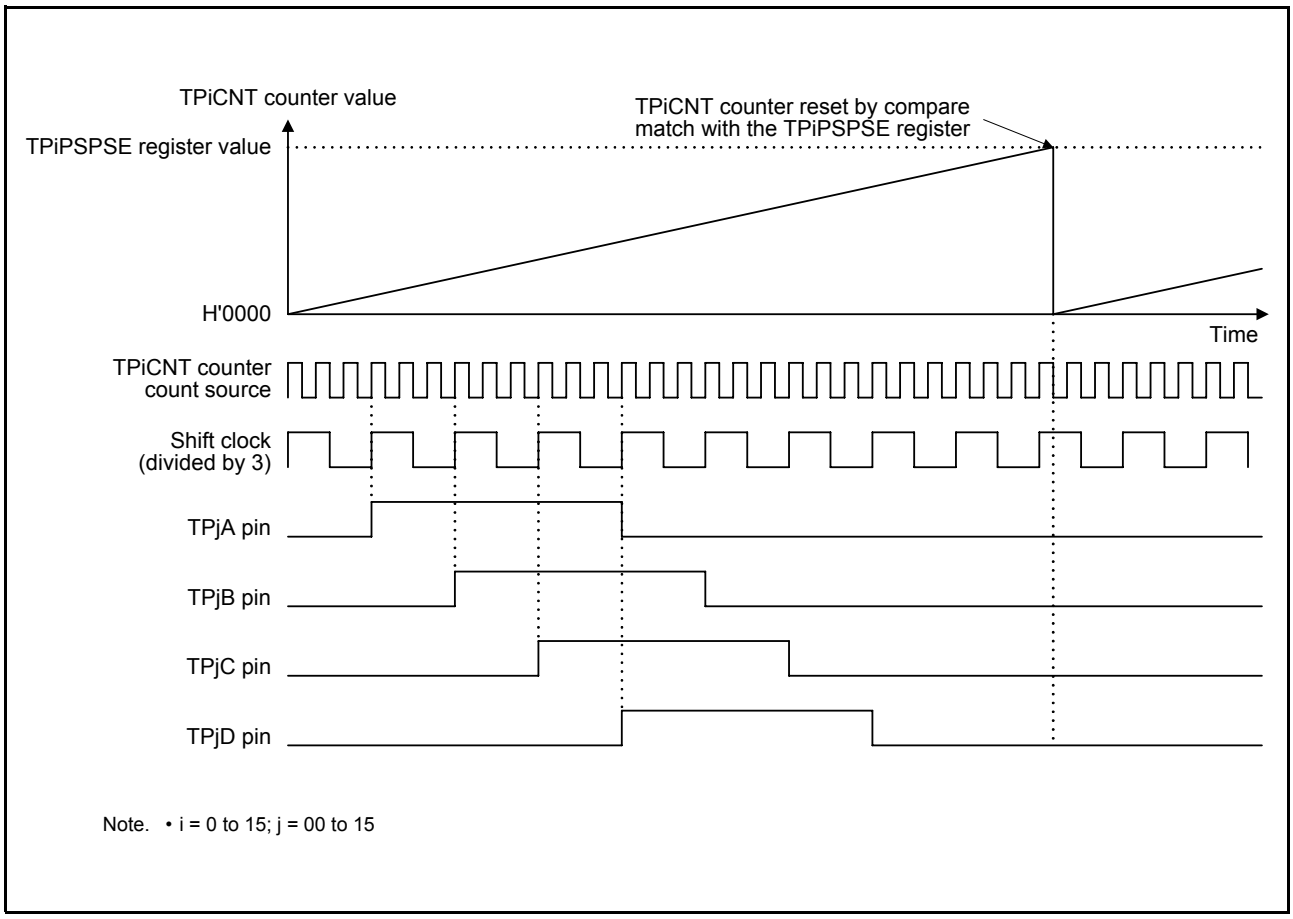


Figure 15.24 Operation Example of Channel Shift Mode

15.7 A/D Conversion Timing Generate Function

This function enables to generate the A/D conversion trigger signal by using phase shift mode. The A/D conversion start trigger signal is generated by the compare match between the TPUi AD conversion/one-shot output timing set register k (TPiADOTSEk) and the TPiCNT counter. Table 15.35 lists Register Settings Associated with AD Conversion Timing Generate Function.

Table 15.35 Register Settings Associated with AD Conversion Timing Generate Function

Register	Bit	Description
TPiADCR	ADTE	Select to use the A/D conversion trigger function of the corresponding channel.
TPiADOTSEk	—	Set the A/D conversion timing of the corresponding channel.

Note: • i = 0 to 15; k = 0 to 3

Figure 15.25 shows Operation Example of A/D Conversion Timing Generate Function.

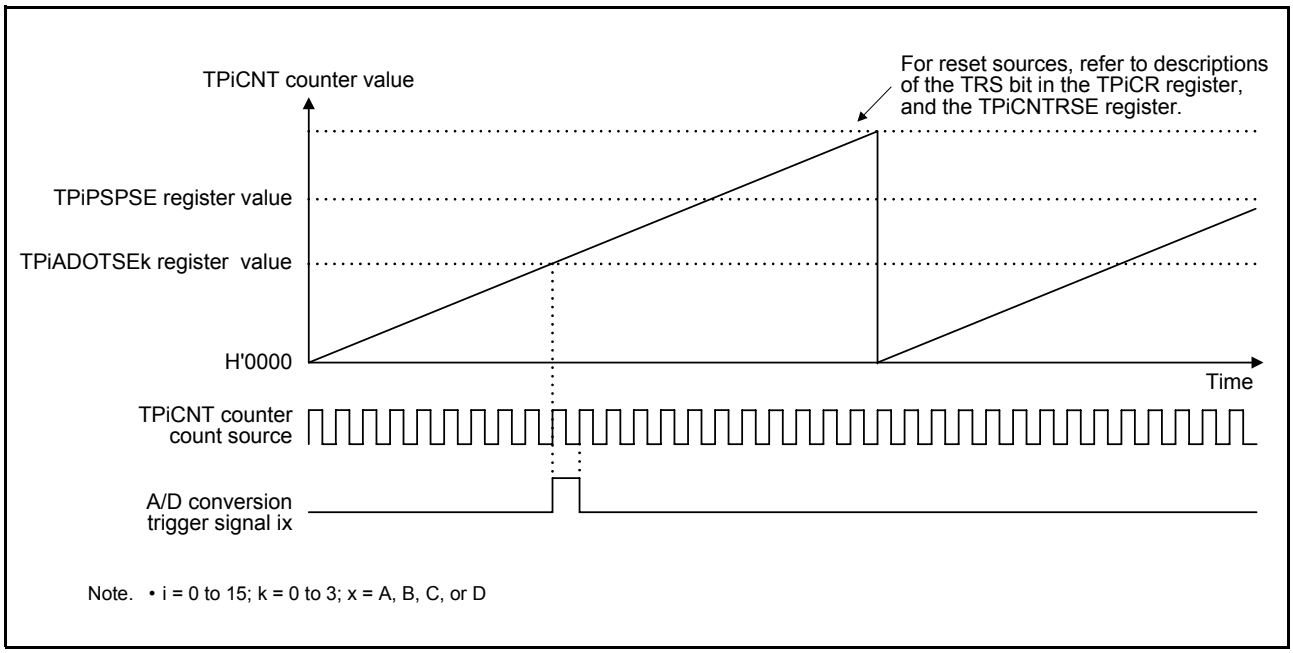


Figure 15.25 Operation Example of A/D Conversion Timing Generate Function

15.8 One-shot Output Function

This function allows the value written into the TPiADOTSEk register to be added to the value of the TPiCNT counter at the time and the total value to be stored in the TPiGRk register. One of the function applications allows the output compare output to be executed after the compare match event with the TPiGR register occurs by using the compare match function and then the time is set in the TPiADOTSEk register.
Table 15.36 lists Register Settings Associated with One-shot Output Function.

Table 15.36 Register Settings Associated with One-shot Output Function

Register	Description
TPiADOTSEk	Set the one-shot timing of the corresponding channel.
TPiADCR	Select to use the one-shot output function.

Note: • Settings for the event output function are required separately.
• i = 0 to 15; k = 0 to 3

Figure 15.26 shows an Operating Procedure of One-shot Output Function.

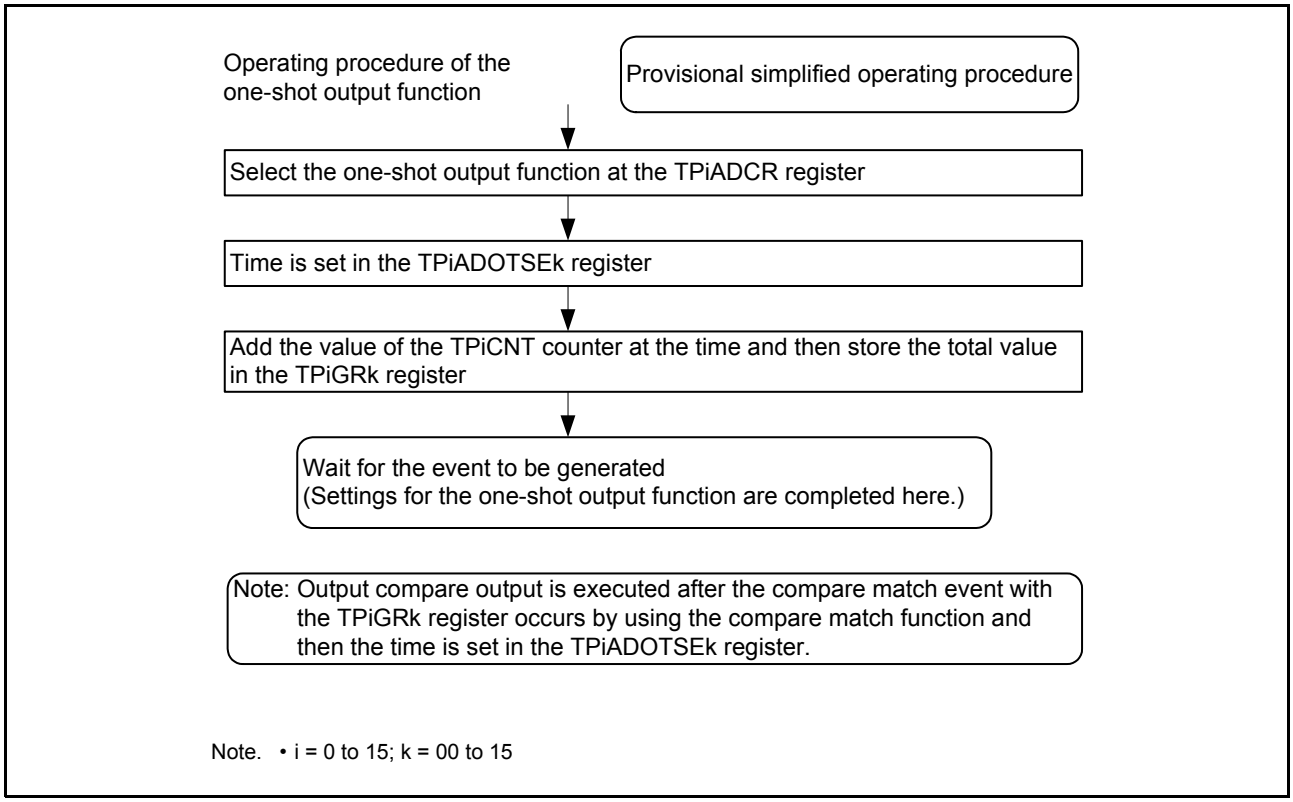
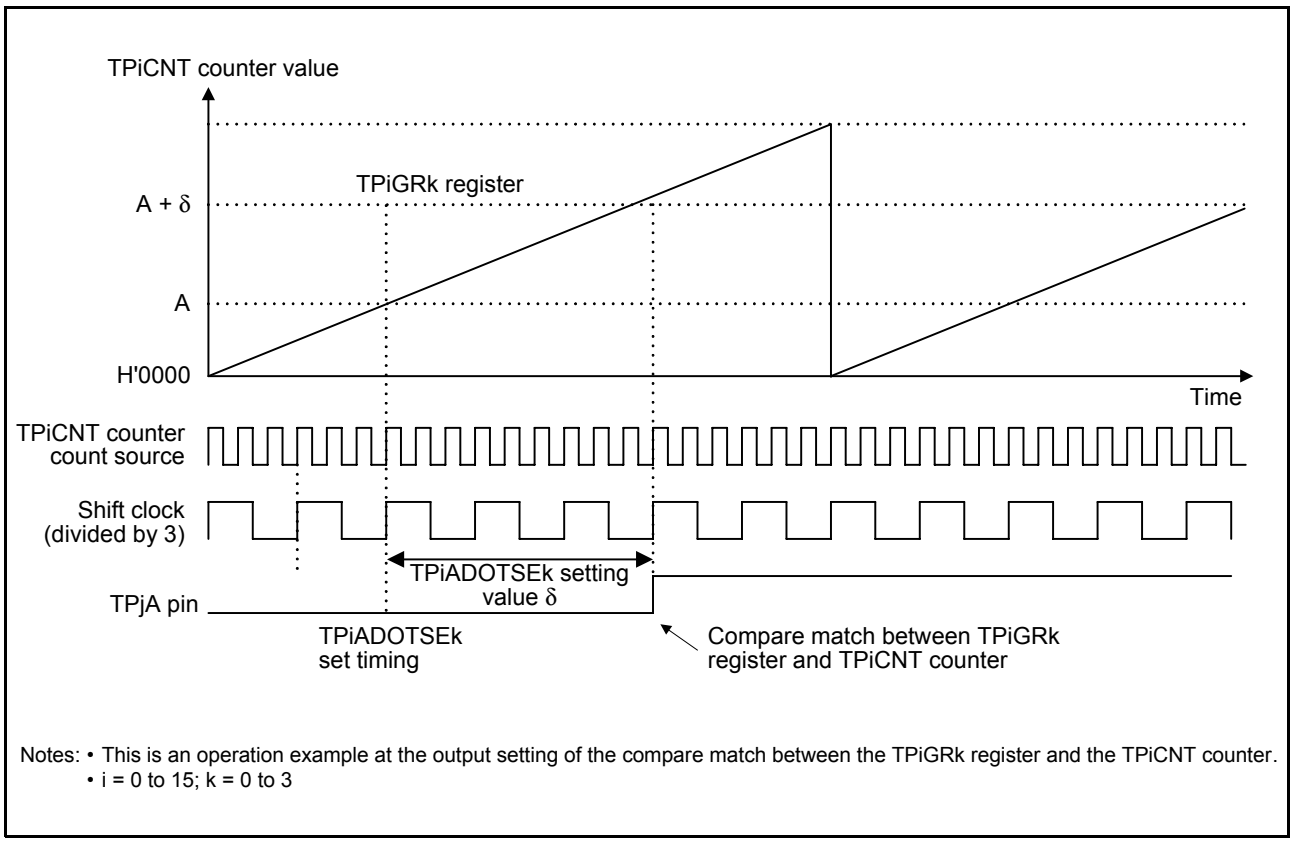


Figure 15.26 Operating Procedure of One-shot Output Function

Figure 15.27 shows an Operation Example of One-shot Output Function.



Notes: • This is an operation example at the output setting of the compare match between the TPiGRk register and the TPiCNT counter.
• i = 0 to 15; k = 0 to 3

Figure 15.27 Operation Example of One-shot Output Function

15.9 Interrupt Sources

The TPU interrupt sources are the TPiGRk interrupt (the input capture/compare match) of channel k for each unit i, the TPUi counter overflow interrupt, and the TPUi counter reset interrupt. Each interrupt source has the interrupt control register and the interrupt priority level can be set independently. The TPU itself owns six interrupt sources per unit and ninety interrupt sources in total.
Table 15.37 lists TPU Interrupt Sources.

Table 15.37 TPU Interrupt Sources

Interrupt Source	DMAC activation (O: YES; —: NO)															
	Unit															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
TPiGR0 interrupt	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O
TPiGR1 interrupt	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O
TPiGR2 interrupt	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O
TPiGR3 interrupt	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O
Overflow interrupt of the TPUi counter	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Reset interrupt of the TPUi counter by the TPiCNTRSE register setting	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O

Note: • i = 0 to 15

15.10 Notes on TPU

15.10.1 Collision of Write Signal to TPiCNT Counter and Various Signals

- If colliding with the reset signal of the TPiCNT counter, the TPiCNT counter is reset and the value is not written into the TPiCNT counter.
- If colliding with the increment timing, the value is written into the TPiCNT counter and the TPiCNT counter does not increment.
- If colliding with the overflow timing, the value is written into the TPiCNT counter and the overflow flag is not set.

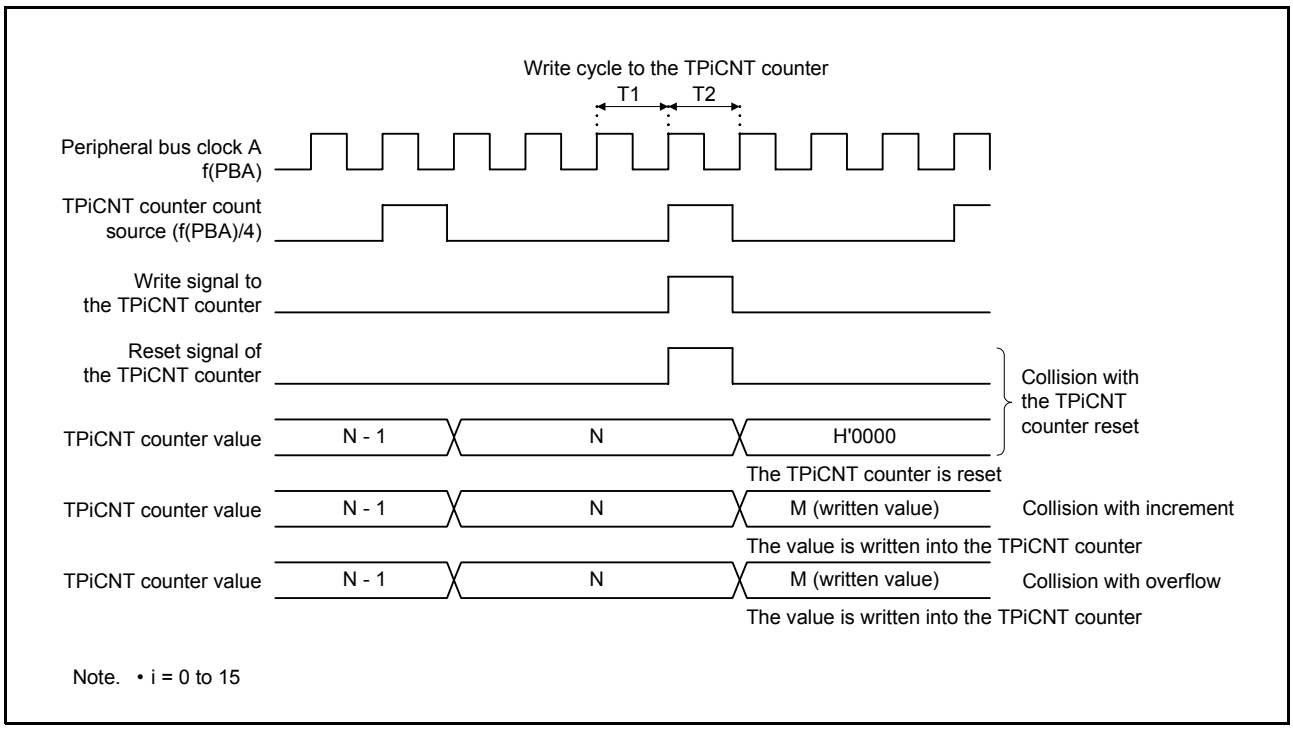


Figure 15.28 Collision of Write Signal to TPiCNT Counter and Various Signals

15.10.2 Collision of Write Signal to TPiGRk register and Various Signals

- If colliding with the signal of the compare match with the TPiCNT counter, the value is written into the TPiGRk register and the compare match signal is not generated.
- If colliding with the input capture signal, the input capture occurs and the value is not written into the TPiGRk register.

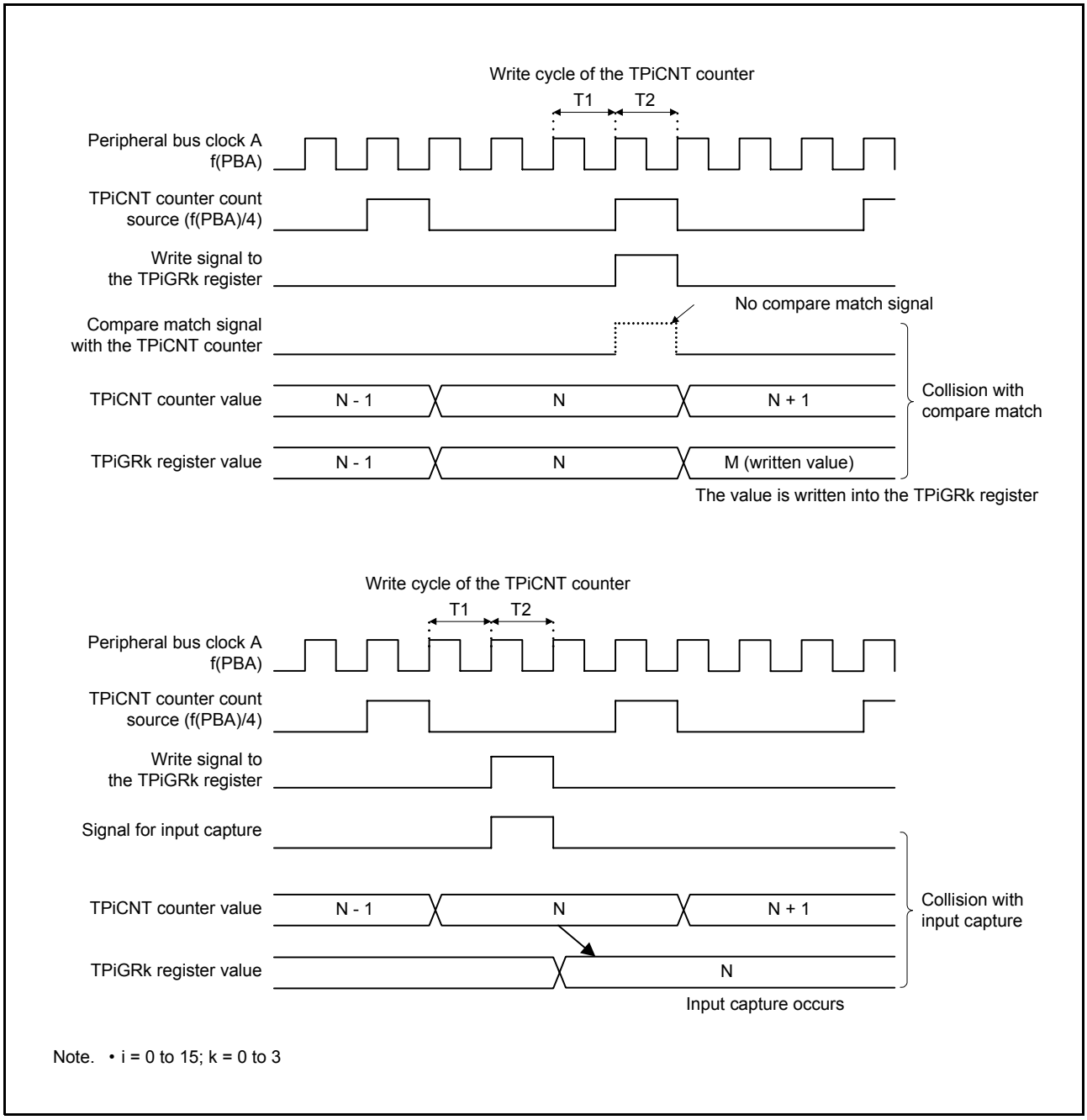


Figure 15.29 Collision of Write Signal to the TPiGRk Register and Various Signals

15.10.3 Collision of Read Signal of TPiGRk Register and Input Capture Signal

If the read signal of the TPiGRk register collides with the input capture signal, the value before the input capture is read.

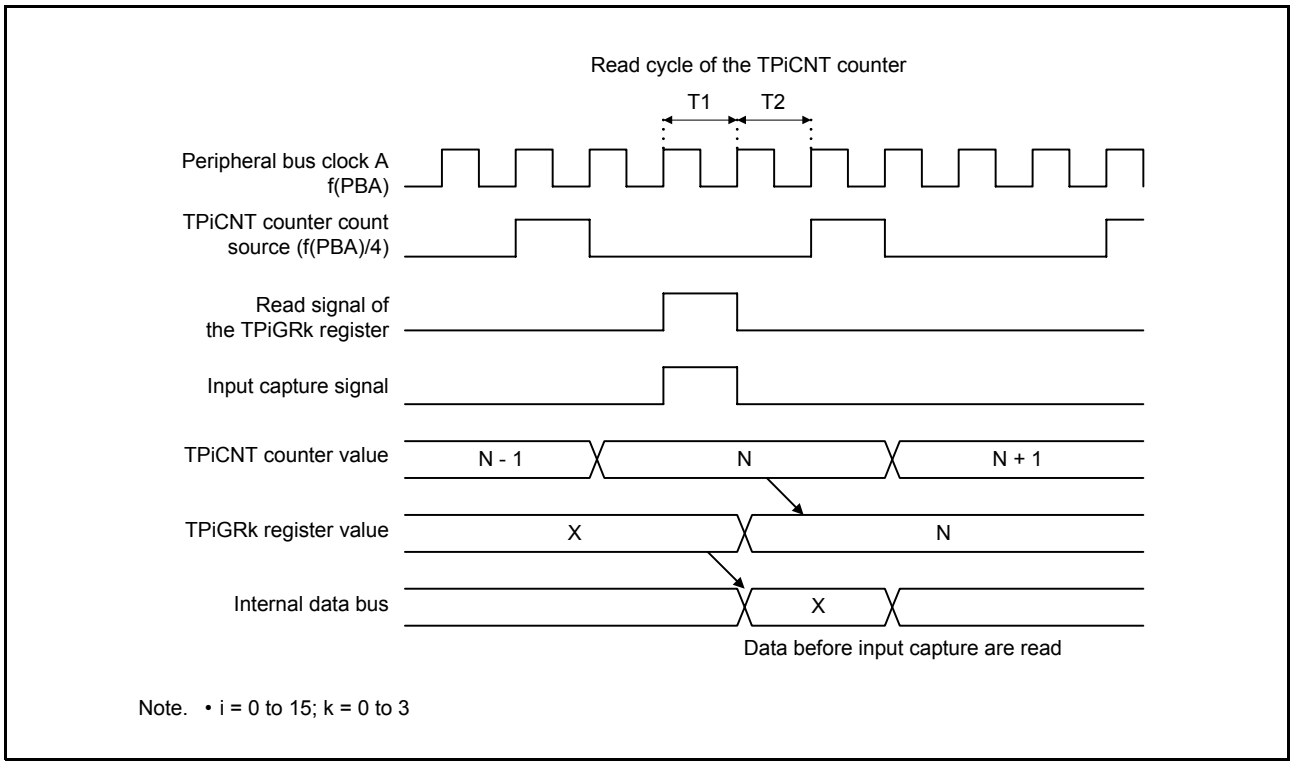


Figure 15.30 Collision of Read Signal of TPiGRk Register and Input Capture Signal

15.10.4 Collision of Write Signal to Buffer Register and Various Signals

- If colliding with the compare match signal, the data transferred to the TPiGRk register by the buffer operation are the data before writing.
- If colliding with the input capture signal, the buffer operation is operated and the value is not written into the buffer register.

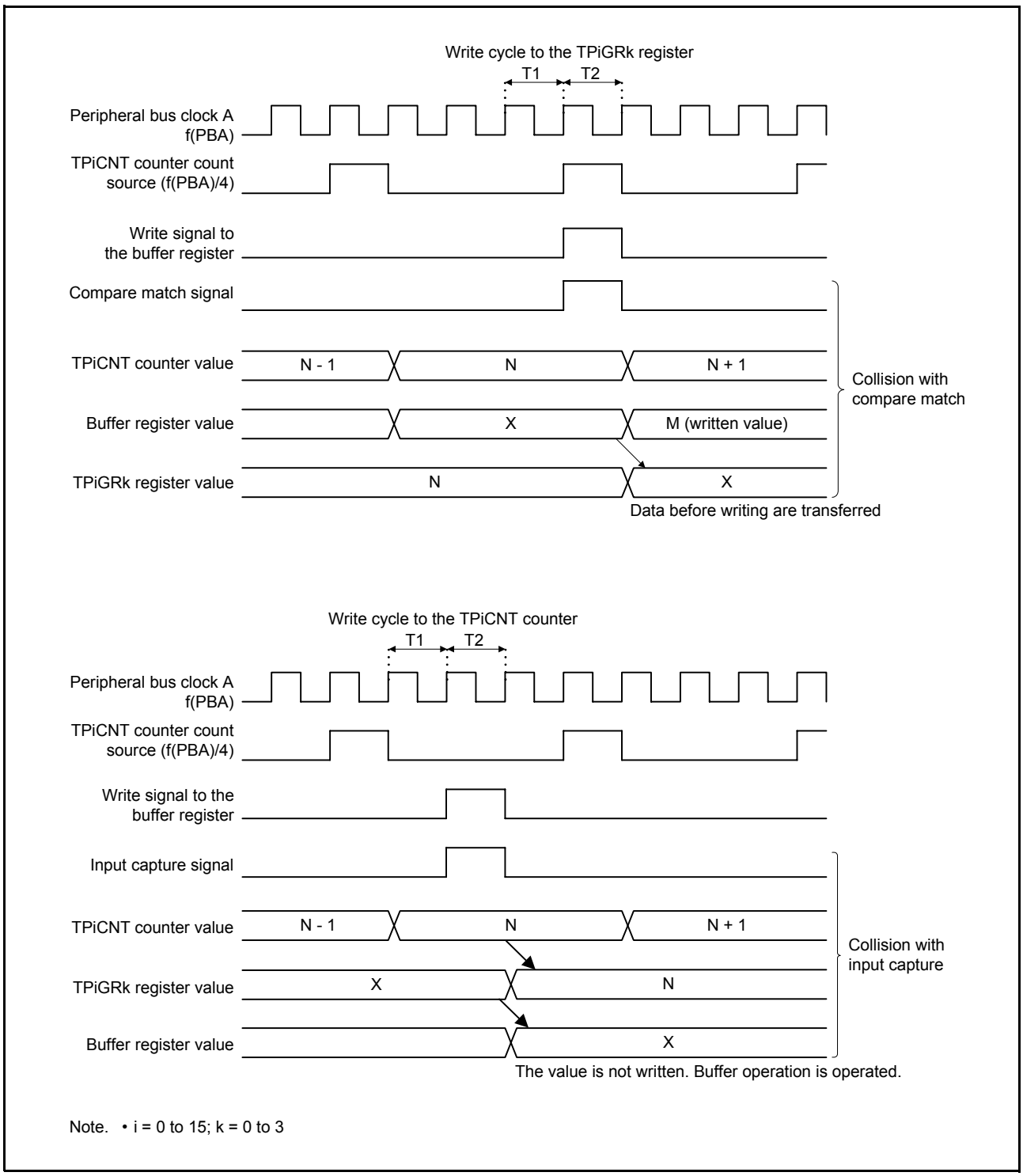


Figure 15.31 Collision of Write Signal to Buffer Register and Various Signals

15.10.5 Collision of Overflow Signal and Reset Signal of TPiCNT Counter

If the overflow signal collides with the reset signal of the TPiCNT counter, the TOVF flag is set and the TPiCNT counter is reset. In Figure 15.32, the TPiGRk register is set to H'FFFF and the TPiCNT counter is reset by the compare match.

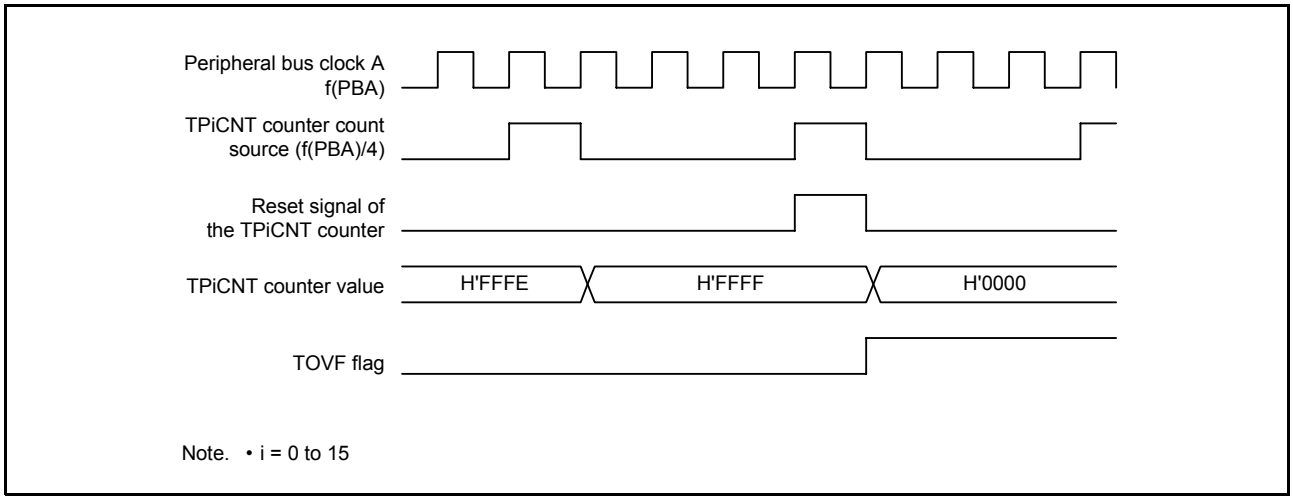


Figure 15.32 Collision of Overflow Signal and Reset Signal of TPiCNT Counter

16. Multi-Function Timer Pulse Unit (MTU-III)

This MCU has an on-chip multi-function timer pulse unit (MTU-III) that comprises eight 16-bit timer channels (channels 0 to 7).

16.1 Features

- Maximum 24 pulse input/output lines and three pulse input lines in MTU-III
- Selection of six to eight counter input clocks for each channel (four clocks for channel 5)
- The following operations can be set for channels 0 to 4, 6, and 7 in MTU-III:
  - Waveform output at compare match
  - Input capture function
  - Counter clear operation
  - Multiple timer counters (MTCNT) can be written to simultaneously
  - Simultaneous clearing by compare match and input capture is possible
  - Register simultaneous input/output is possible by synchronous counter operation
  - A maximum 12-phase PWM output is possible in combination with synchronous operation
- Buffer operation settable for channels 0, 3, 4, 6, and 7. In complementary PWM mode, data can be transferred to the temporary register at the crest and trough of the MTCNT counter value or when a buffer register (MT4GRD and MT7GRD) is written to.
- Phase counting mode settable independently for each of channels 1 and 2
- Cascade connection operation
- Fast access via internal 16-bit bus
- 38 interrupt sources in MTU-III
- Automatic transfer of register data (automatic transfer from the buffer register to the timer register during buffer operation)
- A/D converter start trigger can be generated.
- A/D converter start request delaying function enables the A/D converter to be started with any desired timing and to be synchronized with PWM output.
- A total of 12-phase output, which includes six phases each for positive and negative waveforms (three complementary PWM and three reset-synchronized PWM waveforms), is obtained by linked operation of channels 3 and 4 and channels 6 and 7.
- Brushless DC motor (AC synchronous motor) drive mode using complementary PWM output and reset-synchronized PWM output is settable by linked operation of channels 0, 3, and 4, and the selection of two types of waveform outputs (chopping and level) is possible.
- Double buffer function settable for channels 3 and 4 and channels 6 and 7 in complementary PWM mode
- Channel 5 can be used as dead time compensation counters by specifying external low pulse measurement for the counter operation.
- In complementary PWM mode, interrupts at the crest and trough of the counter value and A/D converter start triggers can be skipped. (The interrupts to be skipped can be selected from those at the crest or trough of the counter value, and the interrupt skipping period can be specified by the compare match count between MT4ADSRCSEA and MT4ADSRCSEB (MT7ADSRCSEA and MT7ADSRCSEB).)

Table 16.1 MTU-III Functions (1)

Item		Channel 0	Channel 1	Channel 2	Channel 3	Channel 4	Channel 5	Channel 6	Channel 7
Count clock		f(PBB)/1 f(PBB)/4 f(PBB)/16 f(PBB)/64 TCLKA TCLKB TCLKC TCLKD	f(PBB)/1 f(PBB)/4 f(PBB)/16 f(PBB)/64 f(PBB)/256 TCLKA TCLKB	f(PBB)/1 f(PBB)/4 f(PBB)/16 f(PBB)/64 f(PBB)/1024 TCLKA TCLKB TCLKC	f(PBB)/1 f(PBB)/4 f(PBB)/16 f(PBB)/64 f(PBB)/256 f(PBB)/1024 TCLKA TCLKB	f(PBB)/1 f(PBB)/4 f(PBB)/16 f(PBB)/64 f(PBB)/256 f(PBB)/1024 TCLKA TCLKB	f(PBB)/1 f(PBB)/4 f(PBB)/16 f(PBB)/64	f(PBB)/1 f(PBB)/4 f(PBB)/16 f(PBB)/64 f(PBB)/256 f(PBB)/1024	f(PBB)/1 f(PBB)/4 f(PBB)/16 f(PBB)/64 f(PBB)/256 f(PBB)/1024
MT general registers (MTGR)		MT0GRA MT0GRB MT0GRE	MT1GRA MT1GRB	MT2GRA MT2GRB	MT3GRA MT3GRB	MT4GRA MT4GRB	MT5GRU MT5GRV MT5GRW	MT6GRA MT6GRB	MT7GRA MT7GRB
MT general registers/ buffer registers		MT0GRC MT0GRD MT0GRF	—	—	MT3GRC MT3GRD MT3GRE	MT4GRC MT4GRD MT4GRE MT4GRF	—	MT6GRC MT6GRD MT6GRE	MT7GRC MT7GRD MT7GRE MT7GRF
I/O pins		TIOC0A TIOC0B TIOC0C TIOC0D *	TIOC1A * TIOC1B	TIOC2A TIOC2B *	TIOC3A TIOC3B TIOC3C TIOC3D	TIOC4A TIOC4B TIOC4C TIOC4D	Input pin TIC5U TIC5V TIC5W *	TIOC6A* TIOC6B TIOC6C* TIOC6D	TIOC7A TIOC7B TIOC7C TIOC7D
Counter clear function		MTGR compare match or input capture							
Compare match output	0 output	√	√	√	√	√	—	√	√
	1 output	√	√	√	√	√	—	√	√
	Toggle output	√	√	√	√	√	—	√	√
Input capture function		√	√	√	√	√	√	√	√
Synchronous operation		√	√	√	√	√	—	√	√
PWM mode 1		√	√	√	√	√	—	√	√
PWM mode 2		√	√	√	—	—	—	—	—
Complementary PWM mode		—	—	—	√	√	—	√	√
Reset-synchronized PWM mode		—	—	—	√	√	—	√	√
Brushless DC motor control		√	—	—	√	√	—	—	—
Phase counting mode		—	√	√	—	—	—	—	—
Buffer operation		√	—	—	√	√	—	√	√
Dead time compensation function		—	—	—	—	—	√	—	—

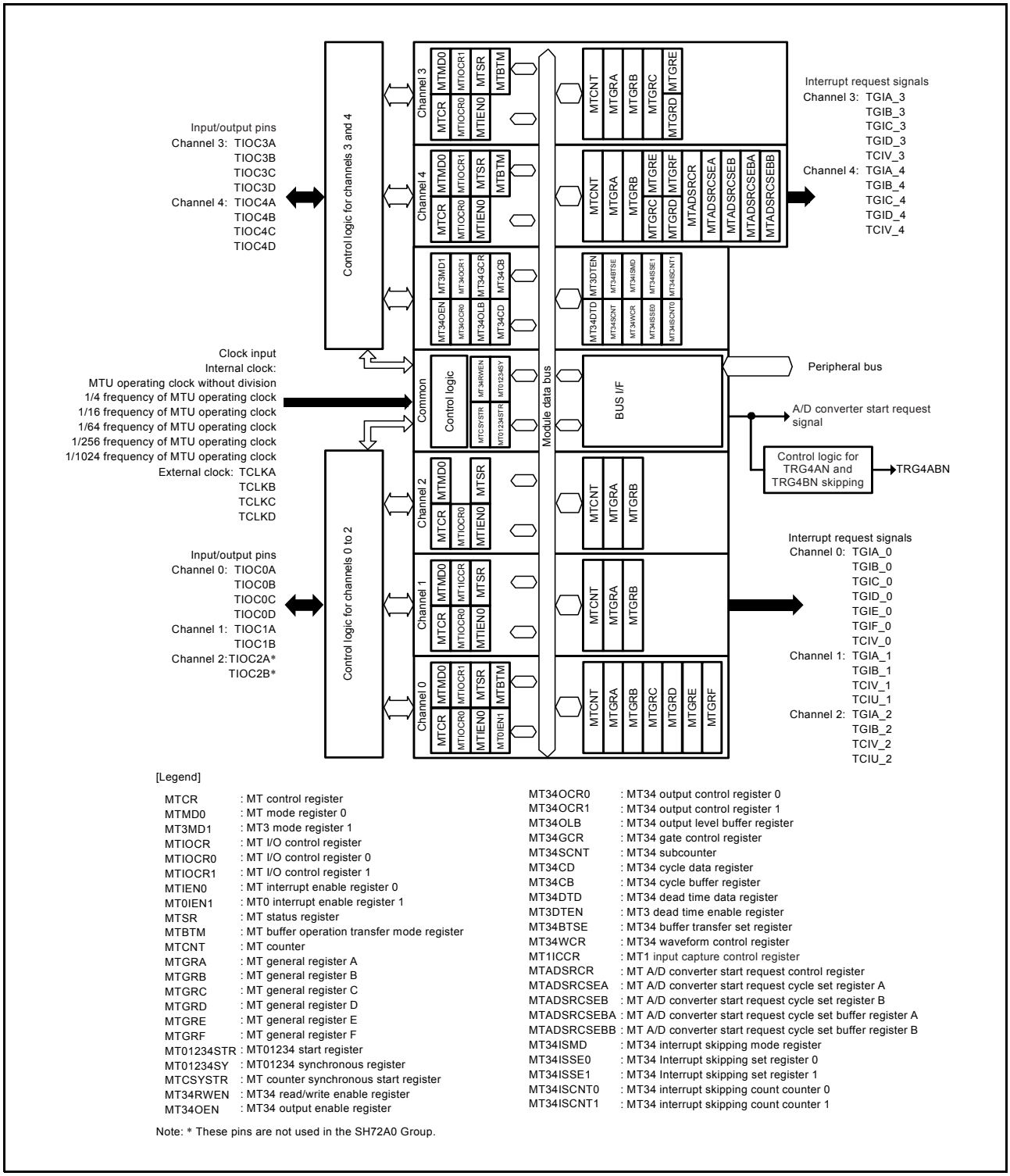
[Legend]
√: Possible
—: Not possible
Note: * These pins are not used in the SH72A0 Group.

Table 16.2 MTU-III Functions (2)

Item	Channel 0	Channel 1	Channel 2	Channel 3	Channel 4	Channel 5	Channel 6	Channel 7
Interrupt sources	7 sources •Compare match or input capture 0A •Compare match or input capture 0B •Compare match or input capture 0C •Compare match or input capture 0D •Compare match 0E •Compare match 0F •Overflow	4 sources •Compare match or input capture 1A •Compare match or input capture 1B •Overflow •Underflow	4 sources •Compare match or input capture 2A •Compare match or input capture 2B •Overflow •Underflow	5 sources •Compare match or input capture 3A •Compare match or input capture 3B •Compare match or input capture 3C •Compare match or input capture 3D •Overflow	5 sources •Compare match or input capture 4A •Compare match or input capture 4B •Compare match or input capture 4C •Compare match or input capture 4D •Overflow/ underflow (only in complementary PWM mode)	3 sources •Compare match or input capture 5U •Compare match or input capture 5V •Compare match or input capture 5W	5 sources •Compare match or input capture 6A •Compare match or input capture 6B •Compare match or input capture 6C •Compare match or input capture 6D •Overflow	5 sources •Compare match or input capture 7A •Compare match or input capture 7B •Compare match or input capture 7C •Compare match or input capture 7D •Overflow/ underflow (only in complementary PWM mode)
DMAC activation	MTGR compare match or input capture							
A/D converter start trigger	MT0GRA compare match or input capture MT0GRE compare match	MT1GRA compare match or input capture	MT2GRA compare match or input capture	MT3GRA compare match or input capture	MT4GRA compare match or input capture MT4CNT underflow (trough) in complementary PWM mode	—	MT6GRA compare match or input capture	MT7GRA compare match or input capture MT7CNT underflow (trough) in complementary PWM mode
A/D converter start request delaying function	—	—	—	—	•A/D converter start request at a match between MT4ADSRCSEA and MT4CNT •A/D converter start request at a match between MT4ADSRCSEB and MT4CNT	—	—	•A/D converter start request at a match between MT7ADSRCSEA and MT7CNT •A/D converter start request at a match between MT7ADSRCSEB and MT7CNT
Interrupt skipping function 1	—	—	—	Skips MT3GRA compare match interrupts	Skips TCIV_4 interrupts	—	Skips MT6GRA compare match interrupts	Skips TCIV_7 interrupts
Interrupt skipping function 2	—	—	—	—	Skips interrupts according to the count of MT4ADSRCSEA and MT4CNT or MT4ADSRCSEB and MT4CNT compare matches	—	—	Skips interrupts according to the count of MT7ADSRCSEA and MT7CNT or MT7ADSRCSEB and MT7CNT compare matches

[Legend]
—: Not possible

Figure 16.1 and figure 16.2 show block diagrams of the MTU-III.



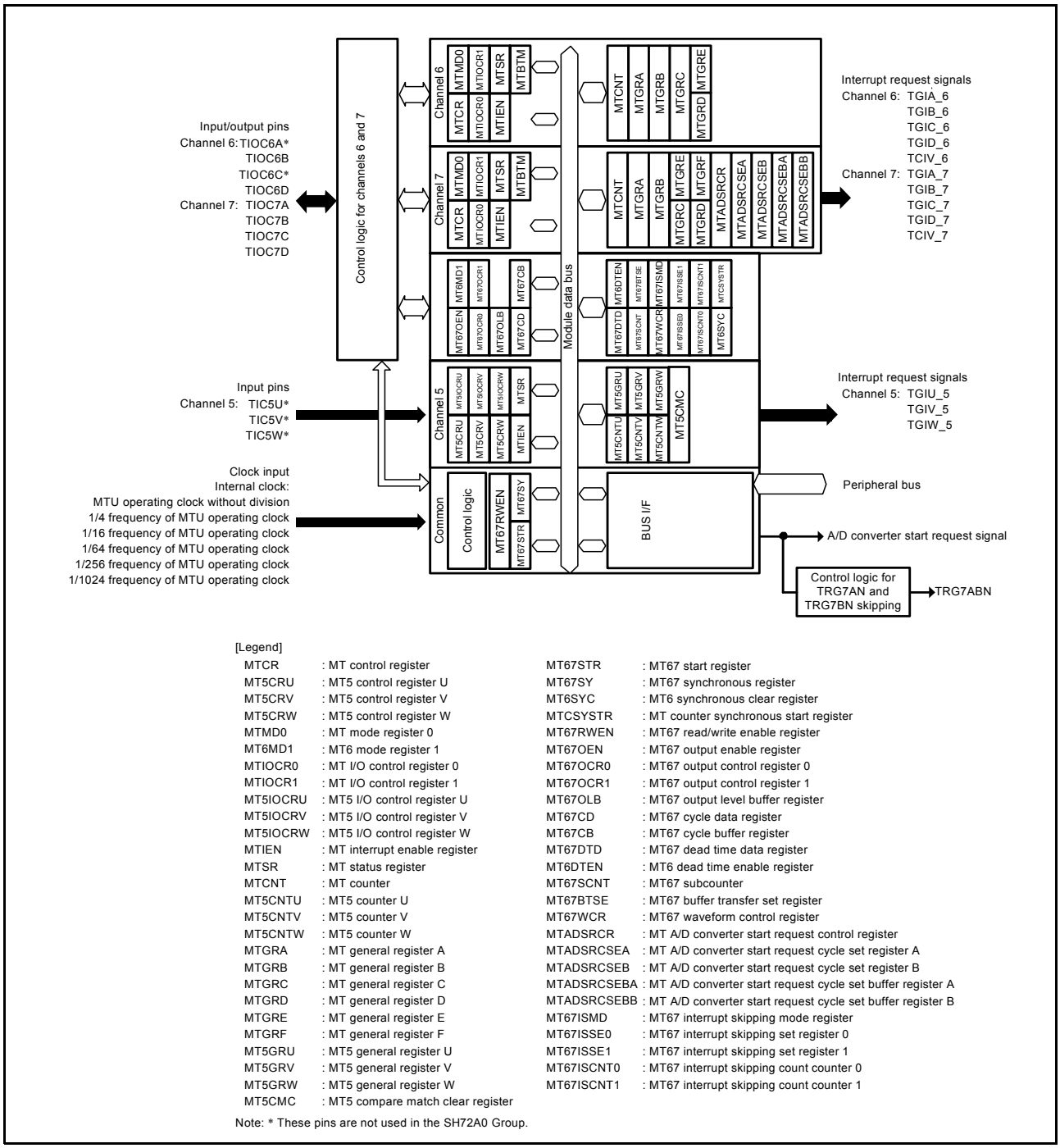
[Legend]

MTCR	: MT control register
MTMD0	: MT mode register 0
MT3MD1	: MT3 mode register 1
MTIOCR	: MT I/O control register
MTIOCR0	: MT I/O control register 0
MTIOCR1	: MT I/O control register 1
MTIEN0	: MT interrupt enable register 0
MT0IEN1	: MT0 interrupt enable register 1
MTSR	: MT status register
MTBTM	: MT buffer operation transfer mode register
MTCNT	: MT counter
MTGRA	: MT general register A
MTGRB	: MT general register B
MTGRC	: MT general register C
MTGRD	: MT general register D
MTGRE	: MT general register E
MTGRF	: MT general register F
MT01234STR	: MT01234 start register
MT01234SY	: MT01234 synchronous register
MTCSYSTR	: MT counter synchronous start register
MT34RWEN	: MT34 read/write enable register
MT34OEN	: MT34 output enable register

MT34OCR0	: MT34 output control register 0
MT34OCR1	: MT34 output control register 1
MT34OLB	: MT34 output level buffer register
MT34GCR	: MT34 gate control register
MT34SCNT	: MT34 subcounter
MT34CD	: MT34 cycle data register
MT34CB	: MT34 cycle buffer register
MT34DTD	: MT34 dead time data register
MT3DTEN	: MT3 dead time enable register
MT34BTSE	: MT34 buffer transfer set register
MT34WCR	: MT34 waveform control register
MT1ICCR	: MT1 input capture control register
MTADSRCR	: MT A/D converter start request control register
MTADSRCSEA	: MT A/D converter start request cycle set register A
MTADSRCSEB	: MT A/D converter start request cycle set register B
MTADSRCSEBA	: MT A/D converter start request cycle set buffer register A
MTADSRCSEBB	: MT A/D converter start request cycle set buffer register B
MT34ISMD	: MT34 interrupt skipping mode register
MT34ISSE0	: MT34 Interrupt skipping set register 0
MT34ISSE1	: MT34 Interrupt skipping set register 1
MT34ISCNT0	: MT34 interrupt skipping count counter 0
MT34ISCNT1	: MT34 interrupt skipping count counter 1

Note: * These pins are not used in the SH72A0 Group.

Figure 16.1 Block Diagram A of MTU-III (Channels 0 to 4)



[Legend]

MTCR	: MT control register	MT67STR	: MT67 start register
MT5CRU	: MT5 control register U	MT67SY	: MT67 synchronous register
MT5CRV	: MT5 control register V	MT6SYC	: MT6 synchronous clear register
MT5CRW	: MT5 control register W	MTCSYSTR	: MT counter synchronous start register
MTMD0	: MT mode register 0	MT67RWEN	: MT67 read/write enable register
MT6MD1	: MT6 mode register 1	MT67OEN	: MT67 output enable register
MTIOCR0	: MT I/O control register 0	MT67OCR0	: MT67 output control register 0
MTIOCR1	: MT I/O control register 1	MT67OCR1	: MT67 output control register 1
MT5IOCRU	: MT5 I/O control register U	MT67OLB	: MT67 output level buffer register
MT5IOCRV	: MT5 I/O control register V	MT67CD	: MT67 cycle data register
MT5IOCRW	: MT5 I/O control register W	MT67CB	: MT67 cycle buffer register
MTIEN	: MT interrupt enable register	MT67DTD	: MT67 dead time data register
MTSR	: MT status register	MT6DTEN	: MT6 dead time enable register
MTCNT	: MT counter	MT67SCNT	: MT67 subcounter
MT5CNTU	: MT5 counter U	MT67BTSE	: MT67 buffer transfer set register
MT5CNTV	: MT5 counter V	MT67WCR	: MT67 waveform control register
MT5CNTW	: MT5 counter W	MTADSRCR	: MT A/D converter start request control register
MTGRA	: MT general register A	MTADSRCSEA	: MT A/D converter start request cycle set register A
MTGRB	: MT general register B	MTADSRCSEB	: MT A/D converter start request cycle set register B
MTGRC	: MT general register C	MTADSRCSEBA	: MT A/D converter start request cycle set buffer register A
MTGRD	: MT general register D	MTADSRCSEBB	: MT A/D converter start request cycle set buffer register B
MTGRE	: MT general register E	MT67ISMD	: MT67 interrupt skipping mode register
MTGRF	: MT general register F	MT67ISSE0	: MT67 interrupt skipping set register 0
MT5GRU	: MT5 general register U	MT67ISSE1	: MT67 interrupt skipping set register 1
MT5GRV	: MT5 general register V	MT67ISCNT0	: MT67 interrupt skipping count counter 0
MT5GRW	: MT5 general register W	MT67ISCNT1	: MT67 interrupt skipping count counter 1
MT5CMC	: MT5 compare match clear register		

Note: * These pins are not used in the SH72A0 Group.

Figure 16.2 Block Diagram B of MTU-III (Channels 5 to 7)

16.1.1 Input/Output Pins

Table 16.3 shows the pin configuration of the MTU-III.

Table 16.3 MTU-III Pin Configuration

Channel	Pin Name	I/O	Function
Common	TCLKA	Input	External clock A input pin (Channel 1 phase counting mode A phase input)
	TCLKB	Input	External clock B input pin (Channel 1 phase counting mode B phase input)
	TCLKC*	Input	External clock C input pin (Channel 2 phase counting mode A phase input)
	TCLKD*	Input	External clock D input pin (Channel 2 phase counting mode B phase input)
0	TIOC0A	I/O	MT0GRA input capture input/output compare output/PWM output pin
	TIOC0B	I/O	MT0GRB input capture input/output compare output/PWM output pin
	TIOC0C	I/O	MT0GRC input capture input/output compare output/PWM output pin
	TIOC0D*	I/O	MT0GRD input capture input/output compare output/PWM output pin
1	TIOC1A*	I/O	MT1GRA input capture input/output compare output/PWM output pin
	TIOC1B	I/O	MT1GRB input capture input/output compare output/PWM output pin
2	TIOC2A*	I/O	MT2GRA input capture input/output compare output/PWM output pin
	TIOC2B*	I/O	MT2GRB input capture input/output compare output/PWM output pin
3	TIOC3A	I/O	MT3GRA input capture input/output compare output/PWM output pin
	TIOC3B	I/O	MT3GRB input capture input/output compare output/PWM output pin
	TIOC3C	I/O	MT3GRC input capture input/output compare output/PWM output pin
	TIOC3D	I/O	MT3GRD input capture input/output compare output/PWM output pin
4	TIOC4A	I/O	MT4GRA input capture input/output compare output/PWM output pin
	TIOC4B	I/O	MT4GRB input capture input/output compare output/PWM output pin
	TIOC4C	I/O	MT4GRC input capture input/output compare output/PWM output pin
	TIOC4D	I/O	MT4GRD input capture input/output compare output/PWM output pin
5	TIC5U*	Input	MT5GRU input capture input/external pulse input pin
	TIC5V*	Input	MT5GRV input capture input/external pulse input pin
	TIC5W*	Input	MT5GRW input capture input/external pulse input pin
6	TIOC6A*	Input	MT6GRA input capture input/output compare output/PWM output pin
	TIOC6B	Input	MT6GRB input capture input/output compare output/PWM output pin
	TIOC6C*	Input	MT6GRC input capture input/output compare output/PWM output pin
	TIOC6D	Input	MT6GRD input capture input/output compare output/PWM output pin
7	TIOC7A	Input	MT7GRA input capture input/output compare output/PWM output pin
	TIOC7B	Input	MT7GRB input capture input/output compare output/PWM output pin
	TIOC7C	Input	MT7GRC input capture input/output compare output/PWM output pin
	TIOC7D	Input	MT7GRD input capture input/output compare output/PWM output pin

Note: * These pins are not used in the SH72A0 Group.

16.2 Register Descriptions

The MTU-III has the following registers.

Table 16.4 List of MTU-III Registers (1)

Register Name	Symbol	R/W	After Reset	Address	Access Size
MT3 control register	MT3CR	R/W	H'00	H'FFFF 8200	8, 16, 32
MT4 control register	MT4CR	R/W	H'00	H'FFFF 8201	8
MT3 mode register 0	MT3MD0	R/W	H'00	H'FFFF 8202	8, 16
MT4 mode register 0	MT4MD0	R/W	H'00	H'FFFF 8203	8
MT3 I/O control register 0	MT3IOCR0	R/W	H'00	H'FFFF 8204	8, 16, 32
MT3 I/O control register 1	MT3IOCR1	R/W	H'00	H'FFFF 8205	8
MT4 I/O control register 0	MT4IOCR0	R/W	H'00	H'FFFF 8206	8, 16
MT4 I/O control register 1	MT4IOCR1	R/W	H'00	H'FFFF 8207	8
MT3 interrupt enable register 0	MT3IEN0	R/W	H'00	H'FFFF 8208	8, 16
MT4 interrupt enable register 0	MT4IEN0	R/W	H'00	H'FFFF 8209	8
MT34 output enable register	MT34OEN	R/W	H'C0	H'FFFF 820A	8
MT34 gate control register	MT34GCR	R/W	H'80	H'FFFF 820D	8
MT34 output control register 0	MT34OCR0	R/W	H'00	H'FFFF 820E	8, 16
MT34 output control register 1	MT34OCR1	R/W	H'00	H'FFFF 820F	8
MT3 counter	MT3CNT	R/W	H'0000	H'FFFF 8210	16, 32
MT4 counter	MT4CNT	R/W	H'0000	H'FFFF 8212	16
MT34 cycle data register	MT34CD	R/W	H'FFFF	H'FFFF 8214	16, 32
MT34 dead time data register	MT34DTD	R/W	H'FFFF	H'FFFF 8216	16
MT3 general register A	MT3GRA	R/W	H'FFFF	H'FFFF 8218	16, 32
MT3 general register B	MT3GRB	R/W	H'FFFF	H'FFFF 821A	16
MT4 general register A	MT4GRA	R/W	H'FFFF	H'FFFF 821C	16, 32
MT4 general register B	MT4GRB	R/W	H'FFFF	H'FFFF 821E	16
MT34 subcounter	MT34SCNT	R	H'0000	H'FFFF 8220	16, 32
MT34 cycle buffer register	MT34CB	R/W	H'FFFF	H'FFFF 8222	16
MT3 general register C	MT3GRC	R/W	H'FFFF	H'FFFF 8224	16, 32
MT3 general register D	MT3GRD	R/W	H'FFFF	H'FFFF 8226	16
MT4 general register C	MT4GRC	R/W	H'FFFF	H'FFFF 8228	16, 32
MT4 general register D	MT4GRD	R/W	H'FFFF	H'FFFF 822A	16
MT3 status register 0	MT3SR0	R/W	H'C0	H'FFFF 822C	8, 16
MT4 status register 0	MT4SR0	R/W	H'C0	H'FFFF 822D	8
MT34 interrupt skipping setting register 0	MT34ISSE0	R/W	H'00	H'FFFF 8230	8, 16
MT34 interrupt skipping count counter 0	MT34ISCNT0	R	H'00	H'FFFF 8231	8
MT34 buffer transfer setting register	MT34BTSE	R/W	H'00	H'FFFF 8232	8
MT3 dead time enable register	MT3DTEN	R/W	H'01	H'FFFF 8234	8
MT34 output level buffer register	MT34OLB	R/W	H'00	H'FFFF 8236	8
MT3 buffer operation transfer mode register	MT3BTM	R/W	H'00	H'FFFF 8238	8, 16
MT4 buffer operation transfer mode register	MT4BTM	R/W	H'00	H'FFFF 8239	8
MT34 interrupt skipping mode register	MT34ISMD	R/W	H'00	H'FFFF 823A	8
MT4 interrupt skipping setting register 1	MT4ISSE1	R/W	H'00	H'FFFF 823B	8
MT4 interrupt skipping count counter 1	MT4ISCNT1	R	H'00	H'FFFF 823C	8

Table 16.5 List of MTU-III Registers (2)

Register Name	Symbol	R/W	After Reset	Address	Access Size
MT4 A/D converter start request control register	MT4ADSRCR	R/W	H'0000	H'FFFF 8240	16
MT4 A/D converter start request cycle setting register A	MT4ADSRCSEA	R/W	H'FFFF	H'FFFF 8244	16, 32
MT4 A/D converter start request cycle setting register B	MT4ADSRCSEB	R/W	H'FFFF	H'FFFF 8246	16
MT4 A/D converter start request cycle setting buffer register A	MT4ADSRCSEBA	R/W	H'FFFF	H'FFFF 8248	16, 32
MT4 A/D converter start request cycle setting buffer register B	MT4ADSRCSEBB	R/W	H'FFFF	H'FFFF 824A	16
MT34 waveform control register	MT34WCR	R/W	H'00	H'FFFF 8260	8
MT3 mode register 1	MT3MD1	R/W	H'00	H'FFFF 8270	8
MT3 general register E	MT3GRE	R/W	H'FFFF	H'FFFF 8272	16
MT4 general register E	MT4GRE	R/W	H'FFFF	H'FFFF 8274	16
MT4 general register F	MT4GRF	R/W	H'FFFF	H'FFFF 8276	16
MT01234 start register	MT01234STR	R/W	H'00	H'FFFF 8280	8, 16
MT01234 synchronous register	MT01234SY	R/W	H'00	H'FFFF 8281	8
MT counter synchronization start register	MTCSYSTR	R/W	H'00	H'FFFF 8282	8
MT34 read/write enable register	MT34RWEN	R/W	H'01	H'FFFF 8284	8
MT0 control register	MT0CR	R/W	H'00	H'FFFF 8300	8, 16, 32
MT0 mode register 0	MT0MD0	R/W	H'00	H'FFFF 8301	8
MT0 I/O control register 0	MT0IOCR0	R/W	H'00	H'FFFF 8302	8, 16
MT0 I/O control register 1	MT0IOCR1	R/W	H'00	H'FFFF 8303	8
MT0 interrupt enable register 0	MT0IEN0	R/W	H'00	H'FFFF 8304	8, 16, 32
MT0 status register 0	MT0SR0	R/W	H'C0	H'FFFF 8305	8
MT0 counter	MT0CNT	R/W	H'0000	H'FFFF 8306	16
MT0 general register A	MT0GRA	R/W	H'FFFF	H'FFFF 8308	16, 32
MT0 general register B	MT0GRB	R/W	H'FFFF	H'FFFF 830A	16
MT0 general register C	MT0GRC	R/W	H'FFFF	H'FFFF 830C	16, 32
MT0 general register D	MT0GRD	R/W	H'FFFF	H'FFFF 830E	16
MT0 general register E	MT0GRE	R/W	H'FFFF	H'FFFF 8320	16, 32
MT0 general register F	MT0GRF	R/W	H'FFFF	H'FFFF 8322	16
MT0 interrupt enable register 1	MT0IEN1	R/W	H'00	H'FFFF 8324	8, 16
MT0 status register 1	MT0SR1	R/W	H'C0	H'FFFF 8325	8
MT0 buffer operation transfer mode register	MT0BTM	R/W	H'00	H'FFFF 8326	8
MT1 control register	MT1CR	R/W	H'00	H'FFFF 8380	8, 16
MT1 mode register 0	MT1MD0	R/W	H'00	H'FFFF 8381	8
MT1 I/O control register 0	MT1IOCR0	R/W	H'00	H'FFFF 8382	8
MT1 interrupt enable register 0	MT1IEN0	R/W	H'00	H'FFFF 8384	8, 16, 32
MT1 status register 0	MT1SR0	R/W	H'C0	H'FFFF 8385	8
MT1 counter	MT1CNT	R/W	H'0000	H'FFFF 8386	16
MT1 general register A	MT1GRA	R/W	H'FFFF	H'FFFF 8388	16, 32
MT1 general register B	MT1GRB	R/W	H'FFFF	H'FFFF 838A	16
MT1 input capture control register	MT1ICCR	R/W	H'00	H'FFFF 8390	8
MT2 control register	MT2CR	R/W	H'00	H'FFFF 8400	8, 16
MT2 mode register 0	MT2MD0	R/W	H'00	H'FFFF 8401	8
MT2 I/O control register 0	MT2IOCR0	R/W	H'00	H'FFFF 8402	8
MT2 interrupt enable register 0	MT2IEN0	R/W	H'00	H'FFFF 8404	8, 16, 32

Table 16.6 List of MTU-III Registers (3)

Register Name	Symbol	R/W	After Reset	Address	Access Size
MT2 status register 0	MT2SR0	R/W	H'C0	H'FFFF 8405	8
MT2 counter	MT2CNT	R/W	H'0000	H'FFFF 8406	16
MT2 general register A	MT2GRA	R/W	H'FFFF	H'FFFF 8408	16, 32
MT2 general register B	MT2GRB	R/W	H'FFFF	H'FFFF 840A	16
MT6 control register	MT6CR	R/W	H'00	H'FFFF 9200	8, 16, 32
MT7 control register	MT7CR	R/W	H'00	H'FFFF 9201	8
MT6 mode register 0	MT6MD0	R/W	H'00	H'FFFF 9202	8, 16
MT7 mode register 0	MT7MD0	R/W	H'00	H'FFFF 9203	8
MT6 I/O control register 0	MT6IOCR0	R/W	H'00	H'FFFF 9204	8, 16, 32
MT6 I/O control register 1	MT6IOCR1	R/W	H'00	H'FFFF 9205	8
MT7 I/O control register 0	MT7IOCR0	R/W	H'00	H'FFFF 9206	8, 16
MT7 I/O control register 1	MT7IOCR1	R/W	H'00	H'FFFF 9207	8
MT6 interrupt enable register 0	MT6IEN0	R/W	H'00	H'FFFF 9208	8, 16
MT7 interrupt enable register 0	MT7IEN0	R/W	H'00	H'FFFF 9209	8
MT67 output enable register	MT67OEN	R/W	H'C0	H'FFFF 920A	8
MT67 output control register 0	MT67OCR0	R/W	H'00	H'FFFF 920E	8, 16
MT67 output control register 1	MT67OCR1	R/W	H'00	H'FFFF 920F	8
MT6 counter	MT6CNT	R/W	H'0000	H'FFFF 9210	16, 32
MT7 counter	MT7CNT	R/W	H'0000	H'FFFF 9212	16
MT67 cycle data register	MT67CD	R/W	H'FFFF	H'FFFF 9214	16, 32
MT67 dead time data register	MT67DTD	R/W	H'FFFF	H'FFFF 9216	16
MT6 general register A	MT6GRA	R/W	H'FFFF	H'FFFF 9218	16, 32
MT6 general register B	MT6GRB	R/W	H'FFFF	H'FFFF 921A	16
MT7 general register A	MT7GRA	R/W	H'FFFF	H'FFFF 921C	16, 32
MT7 general register B	MT7GRB	R/W	H'FFFF	H'FFFF 921E	16
MT67 subcounter	MT67SCNT	R	H'0000	H'FFFF 9220	16, 32
MT67 cycle buffer register	MT67CB	R/W	H'FFFF	H'FFFF 9222	16
MT6 general register C	MT6GRC	R/W	H'FFFF	H'FFFF 9224	16, 32
MT6 general register D	MT6GRD	R/W	H'FFFF	H'FFFF 9226	16
MT7 general register C	MT7GRC	R/W	H'FFFF	H'FFFF 9228	16, 32
MT7 general register D	MT7GRD	R/W	H'FFFF	H'FFFF 922A	16
MT6 status register 0	MT6SR0	R/W	H'C0	H'FFFF 922C	8, 16
MT7 status register 0	MT7SR0	R/W	H'C0	H'FFFF 922D	8
MT67 interrupt skipping setting register 0	MT67ISSE0	R/W	H'00	H'FFFF 9230	8, 16
MT67 interrupt skipping count counter 0	MT67ISCNT0	R	H'00	H'FFFF 9231	8
MT67 buffer transfer setting register	MT67BTSE	R/W	H'00	H'FFFF 9232	8
MT6 dead time enable register	MT6DTEN	R/W	H'01	H'FFFF 9234	8
MT67 output level buffer register	MT67OLB	R/W	H'00	H'FFFF 9236	8
MT6 buffer operation transfer mode register	MT6BTM	R/W	H'00	H'FFFF 9238	8, 16
MT7 buffer operation transfer mode register	MT7BTM	R/W	H'00	H'FFFF 9239	8
MT67 interrupt skipping mode register	MT67ISMD	R/W	H'00	H'FFFF 923A	8
MT7 interrupt skipping setting register 1	MT7ISSE1	R/W	H'00	H'FFFF 923B	8
MT7 interrupt skipping count counter 1	MT7ISCNT1	R/W	H'00	H'FFFF 923C	8

Table 16.7 List of MTU-III Registers (4)

Register Name	Symbol	R/W	After Reset	Address	Access Size
MT7 A/D converter start request control register	MT7ADSRCR	R/W	H'0000	H'FFFF 9240	16
MT7 A/D converter start request cycle setting register A	MT7ADSRCSEA	R/W	H'FFFF	H'FFFF 9244	16, 32
MT7 A/D converter start request cycle setting register B	MT7ADSRCSEB	R/W	H'FFFF	H'FFFF 9246	16
MT7 A/D converter start request cycle setting buffer register A	MT7ADSRCSEBA	R/W	H'FFFF	H'FFFF 9248	16, 32
MT7 A/D converter start request cycle setting buffer register B	MT7ADSRCSEBB	R/W	H'FFFF	H'FFFF 924A	16
MT6 synchronization clear register	MT6SYC	R/W	H'00	H'FFFF 9250	8
MT67 waveform control register	MT67WCR	R/W	H'00	H'FFFF 9260	8
MT6 mode register 1	MT6MD1	R/W	H'00	H'FFFF 9270	8
MT6 general register E	MT6GRE	R/W	H'FFFF	H'FFFF 9272	16
MT7 general register E	MT7GRE	R/W	H'FFFF	H'FFFF 9274	16
MT7 general register F	MT7GRF	R/W	H'FFFF	H'FFFF 9276	16
MT67 start register	MT67STR	R/W	H'00	H'FFFF 9280	8, 16
MT67 synchronous register	MT67SY	R/W	H'00	H'FFFF 9281	8
MT67 read/write enable register	MT67RWEN	R/W	H'01	H'FFFF 9284	8
MT5 counter U	MT5CNTU	R/W	H'0000	H'FFFF 9480	16, 32
MT5 general register U	MT5GRU	R/W	H'FFFF	H'FFFF 9482	16
MT5 control register U	MT5CRU	R/W	H'00	H'FFFF 9484	8
MT5 I/O control register U	MT5IOCRU	R/W	H'00	H'FFFF 9486	8
MT5 counter V	MT5CNTV	R/W	H'0000	H'FFFF 9490	16, 32
MT5 general register V	MT5GRV	R/W	H'FFFF	H'FFFF 9492	16
MT5 control register V	MT5CRV	R/W	H'00	H'FFFF 9494	8
MT5 I/O control register V	MT5IOCRV	R/W	H'00	H'FFFF 9496	8
MT5 counter W	MT5CNTW	R/W	H'0000	H'FFFF 94A0	16, 32
MT5 general register W	MT5GRW	R/W	H'FFFF	H'FFFF 94A2	16
MT5 control register W	MT5CRW	R/W	H'00	H'FFFF 94A4	8
MT5 I/O control register W	MT5IOCRW	R/W	H'00	H'FFFF 94A6	8
MT5 status register 0	MT5SR0	R/W	H'00	H'FFFF 94B0	8
MT5 interrupt enable register 0	MT5IEN0	R/W	H'00	H'FFFF 94B2	8
MT5 start register	MT5STR	R/W	H'00	H'FFFF 94B4	8
MT5 compare match clear register	MT5CMC	R/W	H'00	H'FFFF 94B6	8
MT3467 waveform switchover register	MT3467WSW	R/W	H'00	H'FFFF 9500	8
MT1 waveform input capture/output compare switchover enable register A	MT1WIOSWENA	R/W	H'00	H'FFFF 9510	8
MT1 waveform input capture/output compare switchover enable register B	MT1WIOSWENB	R/W	H'00	H'FFFF 9512	8
MT2 waveform input capture/output compare switchover enable register A	MT2WIOSWENA	R/W	H'00	H'FFFF 9520	8
MT2 waveform input capture/output compare switchover enable register B	MT2WIOSWENB	R/W	H'00	H'FFFF 9522	8

Bus Master Interface

The timer counters (MTCNT), general registers (MTGR), timer sub-counters (MTSCNT), timer cycle buffer registers (MTCB), timer dead time data registers (MTDTD), timer cycle data registers (MTCD), timer A/D converter start request control registers (MTADSRCR), timer A/D converter start request cycle set registers (MTADSRCSE), and timer A/D converter start request cycle set buffer registers (MTADSRCSEB) are 16-bit registers. As the data bus to the bus master is 16 bits wide, these registers can be read and written to in 16-bit units. They cannot be read from or written to in 8-bit units; 16-bit access must always be used.
All registers other than the above registers are 8-bit registers. As the data bus to the CPU is 16 bits wide, these registers can be read and written to in 16-bit units. They can also be read from or written to in 8-bit units.

In this chapter, register name in each channel are described as follows:

(1) MTi control register (MTiCR) (i = 0 to 4, 6, 7), MT5 control register U (MT5CRU), MT5 control register V (MT5CRV), MT5 control register W (MT5CRW): MTCR registers
(2) MTi mode register 0 (MTiMD0) (i = 0 to 4, 6, 7): MTMD0 registers
(3) MT3 mode register 1 (MT3MD1), MT6 mode register 1 (MT6MD1): MTMD1 registers
(4) MTi I/O control register 0 (MTiIOCR0) (i = 0 to 4, 6, 7), MTj I/O control register 1 (MTjIOCR1) (j = 0, 3, 4, 6, 7), MT5 I/O control register U (MT5IOCRU), MT5 I/O control register V (MT5IOCRV), MT5 I/O control register W (MT5IOCRW): MTIOCR registers
(5) MT5 compare match clear register (MT5CMC): MT5CMC register
(6) MTi interrupt enable register 0 (MTiIEN0) (i = 0 to 4, 6, 7), MT0 interrupt enable register 1 (MT0IEN1), MT5 interrupt enable register 0 (MT5IEN0): MTIEN registers
(7) MTi status register 0 (MTiSR0) (i = 0 to 4, 6, 7), MT0 status register 1 (MT0SR1), MT5 status register 0 (MT5SR0): MTSR registers
(8) MTj buffer operation transfer mode register (MTjBTM) (j = 0, 3, 4, 6, 7): MTBTM registers
(9) MT1 input capture control register (MT1ICCR): MT1ICCR register
(10) MT6 synchronous clear register (MT6SYC): MT6SYC register
(11) MTi counter (MTiCNT) (i = 0 to 4, 6, 7), MT5 counter U (MT5CNTU), MT5 counter V (MT5CNTV), MT5 counter W (MT5CNTW): MTCNT counters
(12) MTi general register A, B (MTiGRA, MTiGRB) (i = 0 to 4, 6, 7), MTj general register C, D, E (MTjGRC, MTjGRD, MTjGRE) (j = 0, 3, 4, 6, 7), MTk general register F (MTkGRF) (k = 0, 4, 7), MT5 general register U, V, W (MT5GRU, MT5GRV, MT5GRW): MTGR registers
(13) MT01234 start register (MT01234STR), MT67 start register (MT67STR), MT5 start register (MT5STR): MTSTR registers
(14) MT01234 synchronous register (MT01234SY), MT67 synchronous register (MT67SY): MTSY registers
(15) MT counter synchronous start register (MTCSYSTR): MTCSYSTR register
(16) MT34 read/write enable register (MT34RWEN), MT67 read/write enable register (MT67RWEN): MTRWEN registers
(17) MT34 output enable register (MT34OEN), MT67 output enable register (MT67OEN): MTOEN registers
(18) MT34 output control register 0 (MT34OCR0), MT67 output control register 0 (MT67OCR0): MTOCR0 registers
(19) MT34 output control register 1 (MT34OCR1), MT67 output control register 1 (MT67OCR1): MTOCR1 registers
(20) MT34 output level buffer register (MT34OLB), MT67 output level buffer register (MT67OLB): MTOLB registers
(21) MT34 gate control register (MT34GCR): MT34GCR register
(22) MT34 subcounter (MT34SCNT), MT67 subcounter (MT67SCNT): MTSCNT counters
(23) MT34 cycle data register (MT34CD), MT67 cycle data register (MT67CD): MTCD registers
(24) MT34 cycle buffer register (MT34CB), MT67 cycle buffer register (MT67CB): MTCB registers
(25) MT34 dead time data register (MT34DTD), MT67 dead time data register (MT67DTD): MTDTD registers
(26) MT3 dead time enable register (MT3DTEN), MT6 dead time enable register (MT6DTEN): MTDTEN registers

(27) MT34 buffer transfer set register (MT34BTSE), MT67 buffer transfer set register (MT67BTSE): MTBTSE registers
(28) MT34 waveform control register (MT34WCR), MT67 waveform control register (MT67WCR): MTWCR registers
(29) MT4 A/D converter start request control register (MT4ADSRCR), MT7 A/D converter start request control register (MT7ADSRCR): MTADSRCR registers
(30) MT4 A/D converter start request cycle set register A (MT4ADSRCSEA), MT7 A/D converter start request cycle set register A (MT7ADSRCSEA), MT4 A/D converter start request cycle set register B (MT4ADSRCSEB), MT7 A/D converter start request cycle set register B (MT7ADSRCSEB): MTADSRCSE registers
(31) MT4 A/D converter start request cycle set buffer register A (MT4ADSRCSEBA), MT7 A/D converter start request cycle set buffer register A (MT7ADSRCSEBA), MT4 A/D converter start request cycle set buffer register B (MT4ADSRCSEBB), MT7 A/D converter start request cycle set buffer register B (MT7ADSRCSEBB): MTADSRCSEB registers
(32) MT34 interrupt skipping mode register (MT34ISMD), MT67 interrupt skipping mode register (MT67ISMD): MTISMD registers
(33) MT34 interrupt skipping set register 0 (MT34ISSE0), MT67 interrupt skipping set register 0 (MT67ISSE0): MTISSE0 registers
(34) MT34 interrupt skipping count counter 0 (MT34ISCNT0), MT67 interrupt skipping count counter 0 (MT67ISCNT0): MTISCNT0 counters
(35) MT4 interrupt skipping set register 1 (MT4ISSE1), MT7 interrupt skipping set register 1 (MT7ISSE1): MTISSE1 registers
(36) MT4 interrupt skipping count counter 1 (MT4ISCNT1), MT7 interrupt skipping count counter 1 (MT7ISCNT1): MTISCNT1 counters
(37) MT3467 waveform switch register (MT3467WSW): MT3467WSW register
(38) MT1 waveform input capture/output compare switch enable register A (MT1WIOSWENA): MT1WIOSWENA register
(39) MT1 waveform input capture/output compare switch enable register B (MT1WIOSWENB): MT1WIOSWENB register
(40) MT2 waveform input capture/output compare switch enable register A (MT2WIOSWENA): MT2WIOSWENA register
(41) MT2 waveform input capture/output compare switch enable register B (MT2WIOSWENB): MT2WIOSWENB register

16.2.1 MTi Control Register (MTiCR) (i = 0 to 4, 6, and 7), MT5 Control Register U (MT5CRU), MT5 Control Register V (MT5CRV), MT5 Control Register W (MT5CRW)

Address MT0CR: H'FFFF 8300, MT1CR: H'FFFF 8380, MT2CR: H'FFFF 8400, MT3CR: H'FFFF 8200, MT4CR: H'FFFF 8201, MT6CR: H'FFFF 9200, MT7CR: H'FFFF 9201

	b7	b6	b5	b4	b3	b2	b1	b0
	CCLR [2:0]			CKEG [1:0]		TPSC [2:0]		
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b5	CCLR [2:0]	Counter Clear	These bits select the clearing source of the MTCNT counter. See table 16.8 and table 16.9.	R/W
b4, b3	CKEG [1:0]	Clock Edge	These bits select the input clock edge. When the input clock is counted using both edges, the input clock period is halved. (e.g. Both edges of 1/4 frequency of MTU operating clock = rising edge of 1/2 frequency of MTU operating clock). If phase counting mode is used on channels 1 and 2, this setting is ignored and the phase counting mode setting has priority. Internal clock edge selection is valid when the input clock is 1/4 frequency of the MTU operating clock or slower. When the MTU operating clock is selected without division, or the overflow/underflow of another channel is selected for the input clock, although values can be written, counter operation compiles with the initial value. b4 b3 0 0 : Count at rising edge 0 1 : Count at falling edge 1 0 : Count at both edges 1 1 : Count at both edges	R/W
b2 to b0	TPSC [2:0]	Time Prescaler	These bits select the MTCNT counter clock. The clock source can be selected independently for each channel. See table 16.10 to table 16.13.	R/W

Address MT5CRU : H'FFFF 9484, MT5CRV : H'FFFF 9494, MT5CRW : H'FFFF 94A4

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	TPSC [1:0]	
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b2	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b1 to b0	TPSC [1:0]	Time Prescaler	These bits select the MTCNT counter clock. The clock source can be selected independently for each channel. See table 16.14.	R/W

MTCR is an 8-bit register that controls the MTCNT counter of each channel. MTCR register settings should be conducted only when MTCNT counter operation is stopped.

Table 16.8 CCLR Bits (Channels 0, 3, 4, 6, and 7)

Channel	Bit 7	Bit 6	Bit 5	Description
	CCLR2	CCLR1	CCLR0	
0, 3, 4 6, 7	0	0	0	MTCNT counter clearing disabled
	0	0	1	MTCNT counter cleared by MTGRA compare match/input capture*3
	0	1	0	MTCNT counter cleared by MTGRB compare match/input capture
	0	1	1	MTCNT counter cleared by counter clearing for another channel performing synchronous clearing/synchronous operation*1
	1	0	0	MTCNT counter clearing disabled
	1	0	1	MTCNT counter cleared by MTGRC compare match/input capture*2 *3
	1	1	0	MTCNT counter cleared by MTGRD compare match/input capture*2
	1	1	1	MTCNT counter cleared by counter clearing for another channel performing synchronous clearing/synchronous operation*1

Notes: 1. Synchronous operation is selected by setting the SYNC bit in MTSY to 1.
2. When MTGRC or MTGRD is used as a buffer register, the MTCNT counter is not cleared because the buffer register setting has priority, and compare match/input capture does not occur.
3. The input capture function cannot be used in channel 6 of the SH72A0 Group.

Table 16.9 CCLR Bits (Channels 1 and 2)

Channel	Bit 7	Bit 6	Bit 5	Description
	Reserved*2	CCLR1	CCLR0	
1, 2	0	0	0	MTCNT counter clearing disabled
	0	0	1	MTCNT counter cleared by MTGRA compare match/input capture*3
	0	1	0	MTCNT counter cleared by MTGRB compare match/input capture*3
	0	1	1	MTCNT counter cleared by counter clearing for another channel performing synchronous clearing/synchronous operation*1

Notes: 1. Synchronous operation is selected by setting the SYNC bit in MTSY to 1.
2. In channels 1 and 2, bit 7 is reserved. It is always read as 0 and cannot be modified.
3. The input capture function cannot be used in channel 2 of the SH72A0 Group.

Table 16.10 TPSC Bits (Channel 0)

Channel	Bit 2	Bit 1	Bit 0	Description
	TPSC2	TPSC1	TPSC0	
0	0	0	0	Internal Clock: counts on the MTU operating clock without frequency division
	0	0	1	Internal Clock: counts on 1/4 frequency of the MTU operating clock
	0	1	0	Internal Clock: counts on 1/16 frequency of the MTU operating clock
	0	1	1	Internal Clock: counts on 1/64 frequency of the MTU operating clock
	1	0	0	External clock: counts on TCLKA pin input
	1	0	1	External clock: counts on TCLKB pin input
	1	1	0	External clock: counts on TCLKC pin input
	1	1	1	External clock: counts on TCLKD pin input

Table 16.11 TPSC Bits (Channel 1)

Channel	Bit 2	Bit 1	Bit 0	Description
	TPSC2	TPSC1	TPSC0	
1	0	0	0	Internal Clock: counts on the MTU operating clock without frequency division
	0	0	1	Internal Clock: counts on 1/4 frequency of the MTU operating clock
	0	1	0	Internal Clock: counts on 1/16 frequency of the MTU operating clock
	0	1	1	Internal Clock: counts on 1/64 frequency of the MTU operating clock
	1	0	0	External clock: counts on TCLKA pin input
	1	0	1	External clock: counts on TCLKB pin input
	1	1	0	Internal Clock: counts on 1/256 frequency of the MTU operating clock
	1	1	1	Counts on the MT2CNT counter overflow/underflow

Note: This setting is ignored when channel 1 is in phase counting mode.

Table 16.12 TPSC Bits (Channel 2)

Channel	Bit 2	Bit 1	Bit 0	Description
	TPSC2	TPSC1	TPSC0	
2	0	0	0	Internal Clock: counts on the MTU operating clock without frequency division
	0	0	1	Internal Clock: counts on 1/4 frequency of the MTU operating clock
	0	1	0	Internal Clock: counts on 1/16 frequency of the MTU operating clock
	0	1	1	Internal Clock: counts on 1/64 frequency of the MTU operating clock
	1	0	0	External clock: counts on TCLKA pin input
	1	0	1	External clock: counts on TCLKB pin input
	1	1	0	External clock: counts on TCLKC pin input
	1	1	1	Internal Clock: counts on 1/1024 frequency of the MTU operating clock

Note: This setting is ignored when channel 2 is in phase counting mode.

Table 16.13 TPSC Bits (Channels 3, 4, 6, and 7)

Channel	Bit 2	Bit 1	Bit 0	Description
	TPSC2	TPSC1	TPSC0	
3, 4, 6, 7	0	0	0	Internal Clock: counts on the peripheral function clock B without frequency division
	0	0	1	Internal Clock: counts on 1/4 frequency of the MTU operating clock
	0	1	0	Internal Clock: counts on 1/16 frequency of the MTU operating clock
	0	1	1	Internal Clock: counts on 1/64 frequency of the MTU operating clock
	1	0	0	Internal Clock: counts on 1/256 frequency of the MTU operating clock
	1	0	1	Internal Clock: counts on 1/1024 frequency of the MTU operating clock
	1	1	0	External clock: counts on TCLKA pin input*
	1	1	1	External clock: counts on TCLKB pin input*

Note: * Setting prohibited in channels 6 and 7.

Table 16.14 TPSC Bits (Channel 5)

Channel	Bit 1	Bit 0	Description
	TPSC1	TPSC0	
5	0	0	Internal Clock: counts on the MTU operating clock without frequency division
	0	1	Internal Clock: counts on 1/4 frequency of the MTU operating clock
	1	0	Internal Clock: counts on 1/16 frequency of the MTU operating clock
	1	1	Internal Clock: counts on 1/64 frequency of the MTU operating clock

Note: In channel 5, bits 7 to 2 are reserved. These bits are always read as 0. The write value should always be 0.

16.2.2 MTi Mode Register 0 (MTiMD0) (i = 0 to 4, 6, and 7)

Address MT0MD0: H'FFFF 8301

	b7	b6	b5	b4	b3	b2	b1	b0
	—	BFE	BFB	BFA	MD [3:0]			
After Reset	0	0	0	0	0	0	0	0

Address MT1MD0: H'FFFF 8381, MT2MD0: H'FFFF 8401

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	MD [3:0]			
After Reset	0	0	0	0	0	0	0	0

Address MT3MD0: H'FFFF 8202, MT4MD0: H'FFFF 8203, MT6MD0: H'FFFF 9202, MT7MD0: H'FFFF 9203

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	BFB	BFA	MD [3:0]			
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	—	Reserved	This bit is always read as 0. The write value should always be 0.	R
b6	BFE	Buffer Operation E	Specifies whether MT0GRE and MT0GRF are to operate in the normal way or to be used together for buffer operation. 0: MT0GRE and MT0GRF operate normally 1: MT0GRE and MT0GRF used together for buffer operation In channels 1 to 4, 6, and 7, bit 6 is reserved. This bit is always read as 0 and the write value should always be 0.	R/W
b5	BFB	Buffer Operation B	Specifies whether MTGRB is to operate in the normal way, or MTGRB and MTGRD are to be used together for buffer operation. When MTGRD is used as a buffer register, MTGRD input capture/ output compare do not take place. 0: MTGRB and MTGRD operate normally 1: MTGRB and MTGRD used together for buffer operation In channels 1 and 2, which have no MTGRD, bit 5 is reserved. This bit is always read as 0. The write value should always be 0.	R/W
b4	BFA	Buffer Operation A	Specifies whether MTGRA is to operate in the normal way, or MTGRA and MTGRC are to be used together for buffer operation. When MTGRC is used as a buffer register, MTGRC input capture/ output compare do not take place. 0: MTGRA and MTGRC operate normally 1: MTGRA and MTGRC used together for buffer operation In channels 1 and 2, which have no MTGRC, bit 4 is reserved. This bit is always read as 0 and the write value should always be 0.	R/W
b3 to b0	MD [3:0]	Mode	These bits are used to set the timer operating mode. See table 16.15.	R/W

MTMD0 register settings should be changed only when MTCNT counter operation is stopped.

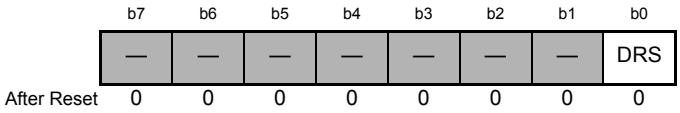
Table 16.15 Setting of Operation Mode by MD Bits

Bit 3	Bit 2	Bit 1	Bit 0	Description
MD3	MD2	MD1	MD0	
0	0	0	0	Normal operation
0	0	0	1	Setting prohibited
0	0	1	0	PWM mode 1
0	0	1	1	PWM mode 2*1
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
1	0	0	0	Reset-synchronized PWM mode*3
1	0	0	1	Setting prohibited
1	0	1	x	Setting prohibited
1	1	0	0	Setting prohibited
1	1	0	1	Complementary PWM mode 1 (transmit at crest)*3
1	1	1	0	Complementary PWM mode 2 (transmit at trough)*3
1	1	1	1	Complementary PWM mode 3 (transmit at crest and trough)*3

[Legend]
x: Don't care
Notes: 1. PWM mode 2 cannot be set for channels 3, 4, 6, and 7.
2. Phase counting mode cannot be set for channels 0, 3, 4, 6, and 7.
3. Reset-synchronized PWM mode and complementary PWM mode can only be set for channels 3 and 6.
When channels 3 and 6 are set to reset-synchronized PWM mode or complementary PWM mode, the channel 4 and 7 settings become ineffective and automatically conform to the channel 3 and 6 settings. However, do not set channels 4 and 7 to reset-synchronized PWM mode or complementary PWM mode.
Reset-synchronized PWM mode and complementary PWM mode cannot be set for channels 0, 1, and 2.

16.2.3 MT3 Mode Register 1 (MT3MD1), MT6 Mode Register 1 (MT6MD1)

Address MT3MD1: H'FFFF 8270, MT6MD1: H'FFFF 9270



Bit	Symbol	Bit Name	Description	R/W
b7 to b1	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b0	DRS	Double Buffer Select	Enables or disables the double buffer function in complementary PWM mode. 0: Double buffer function is disabled 1: Double buffer function is enabled	R/W

MTMD1 is an 8-bit register that sets the double buffer function in complementary PWM mode 3 (transfer at the crest and trough of the counter value). MTMD1 register settings should be changed only when MTCNT counter operation is stopped.

16.2.4 MTi I/O Control Register 0 (MTiIOCR0) (i = 0 to 4, 6, and 7)
MTj I/O Control Register 1 (MTjIOCR1) (j = 0 ,3, 4, 6, and 7)
MT5 I/O Control Register U (MT5IOCRU),
MT5 I/O Control Register V (MT5IOCRV),
MT5 I/O Control Register W (MT5IOCRW)

Address MT0IOCR0: H'FFFF 8302, MT1IOCR0: H'FFFF 8382, MT2IOCR0: H'FFFF 8402, MT3IOCR0: H'FFFF 8204, MT4IOCR0: H'FFFF 8206, MT6IOCR0: H'FFFF 9204, MT7IOCR0: H'FFFF 9206

	b7	b6	b5	b4	b3	b2	b1	b0
	IOB [3:0]				IOA [3:0]			
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b4	IOB [3:0]	I/O Control B	These bits set the function of MTGRB. See the following tables: MT0IOCR0: Table 16.16 MT1IOCR0: Table 16.18 MT2IOCR0: Table 16.19 MT3IOCR0: Table 16.20 MT4IOCR0: Table 16.22 MT6IOCR0: Table 16.24 MT7IOCR0: Table 16.26	R/W
b3 to b0	IOA [3:0]	I/O Control A	These bits set the function of MTGRA. See the following tables: MT0IOCR0: Table 16.28 MT1IOCR0: Table 16.30 MT2IOCR0: Table 16.31 MT3IOCR0: Table 16.32 MT4IOCR0: Table 16.34 MT6IOCR0: Table 16.36 MT7IOCR0: Table 16.38	R/W

Address MT0IOCR1: H'FFFF 8303, MT3IOCR1: H'FFFF 8205, MT4IOCR1: H'FFFF 8207, MT6IOCR1: H'FFFF 9205, MT7IOCR1: H'FFFF 9207

	b7	b6	b5	b4	b3	b2	b1	b0
	IOD [3:0]				IOC [3:0]			
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b4	IOD [3:0]	I/O Control D	These bits set the function of MTGRD. See the following tables: MT0IOCR1: Table 16.17 MT3IOCR1: Table 16.21 MT4IOCR1: Table 16.23 MT6IOCR1: Table 16.25 MT7IOCR1: Table 16.27	R/W
b3 to b0	IOC [3:0]	I/O Control C	These bits set the function of MTGRC. See the following tables: MT0IOCR1: Table 16.29 MT3IOCR1: Table 16.33 MT4IOCR1: Table 16.35 MT6IOCR1: Table 16.37 MT7IOCR1: Table 16.39	R/W

Address MT5IOCRU: H'FFFF 9486, MT5IOCRV: H'FFFF 9496, MT5IOCRW: H'FFFF 94A6

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	IOC [4:0]				
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b5	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b4 to b0	IOC [4:0]	I/O Control C	These bits set the function of MT5GRU, MT5GRV, and MT5GRW. See table 16.40.	R/W

MTIOCR is an 8-bit register that controls MTGR. Note that the MTIOCR function is affected by the MTMD0 setting. The initial output specified by MTIOCR is valid when the counter is stopped (the CST bit in MT01234STR or MT67STR is cleared to 0). Note also that, in PWM mode 2, the output at the point at which the counter is cleared to 0 is specified.

When MTGRC or MTGRD is designated for buffer operation, this setting is invalid and the register operates as a buffer register.

Table 16.16 MT0IOCR0 (Channel 0)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB3	IOB2	IOB1	IOB0	MT0GRB Function	TIOC0B Pin Function
0	0	0	0	Output compare register	Output retained*
0	0	0	1		Initial output is 0 0 output at compare match
0	0	1	0		Initial output is 0 1 output at compare match
0	0	1	1		Initial output is 0 Toggle output at compare match
0	1	0	0		Output retained
0	1	0	1		Initial output is 1 0 output at compare match
0	1	1	0		Initial output is 1 1 output at compare match
0	1	1	1		Initial output is 1 Toggle output at compare match
1	0	0	0	Input capture register	Input capture at rising edge
1	0	0	1		Input capture at falling edge
1	0	1	x		Input capture at both edges
1	1	x	x		Capture input source is channel 1/count clock Input capture at the MT1CNT counter count-up/count-down

[Legend]
x: Don't care
Note: * After reset, 0 is output until MTIOCR is set.

Table 16.17 MT0IOCR1 (Channel 0)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOD3	IOD2	IOD1	IOD0	MT0GRD Function	TIOC0D Pin Function
0	0	0	0	Output compare register*2	Output retained*1
0	0	0	1		Initial output is 0 0 output at compare match
0	0	1	0		Initial output is 0 1 output at compare match
0	0	1	1		Initial output is 0 Toggle output at compare match
0	1	0	0		Output retained
0	1	0	1		Initial output is 1 0 output at compare match
0	1	1	0		Initial output is 1 1 output at compare match
0	1	1	1		Initial output is 1 Toggle output at compare match
1	0	0	0	Input capture register*2	Input capture at rising edge
1	0	0	1		Input capture at falling edge
1	0	1	x		Input capture at both edges
1	1	x	x		Capture input source is channel 1/count clock Input capture at the MT1CNT counter count-up/count-down

[Legend]
x: Don't care
Notes: 1. After reset, 0 is output until MTIOCR is set.
2. When the BFB bit in MT0MD0 is set to 1 and MT0GRD is used as a buffer register, this setting is invalid and input capture/output compare is not generated.
Note: This function cannot be used in the SH72A0 Group.

Table 16.18 MT1IOCR0 (Channel 1)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB3	IOB2	IOB1	IOB0	MT1GRB Function	TIOC1B Pin Function
0	0	0	0	Output compare register	Output retained*
0	0	0	1		Initial output is 0 0 output at compare match
0	0	1	0		Initial output is 0 1 output at compare match
0	0	1	1		Initial output is 0 Toggle output at compare match
0	1	0	0		Output retained
0	1	0	1		Initial output is 1 0 output at compare match
0	1	1	0		Initial output is 1 1 output at compare match
0	1	1	1		Initial output is 1 Toggle output at compare match
1	0	0	0	Input capture register	Input capture at rising edge
1	0	0	1		Input capture at falling edge
1	0	1	x		Input capture at both edges
1	1	x	x		Input capture at the MT0GRC compare match/input capture

[Legend]
x: Don't care
Note: * After reset, 0 is output until MTIOCR is set.

Table 16.19 MT2IOCR0 (Channel 2)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB3	IOB2	IOB1	IOB0	MT2GRB Function	TIOC2B Pin Function
0	0	0	0	Output compare register	Output retained*
0	0	0	1		Initial output is 0 0 output at compare match
0	0	1	0		Initial output is 0 1 output at compare match
0	0	1	1		Initial output is 0 Toggle output at compare match
0	1	0	0		Output retained
0	1	0	1		Initial output is 1 0 output at compare match
0	1	1	0		Initial output is 1 1 output at compare match
0	1	1	1		Initial output is 1 Toggle output at compare match
1	x	0	0	Input capture register	Input capture at rising edge
1	x	0	1		Input capture at falling edge
1	x	1	x		Input capture at both edges

[Legend]
x: Don't care
Note: * After reset, 0 is output until MTIOCR is set.
Note: This function cannot be used in the SH72A0 Group.

Table 16.20 MT3IOCR0 (Channel 3)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB3	IOB2	IOB1	IOB0	MT3GRB Function	TIOC3B Pin Function
0	0	0	0	Output compare register	Output retained*
0	0	0	1		Initial output is 0 0 output at compare match
0	0	1	0		Initial output is 0 1 output at compare match
0	0	1	1		Initial output is 0 Toggle output at compare match
0	1	0	0		Output retained
0	1	0	1		Initial output is 1 0 output at compare match
0	1	1	0		Initial output is 1 1 output at compare match
0	1	1	1		Initial output is 1 Toggle output at compare match
1	x	0	0	Input capture register	Input capture at rising edge
1	x	0	1		Input capture at falling edge
1	x	1	x		Input capture at both edges

[Legend]
x: Don't care
Note: * After reset, 0 is output until MTIOCR is set.

Table 16.21 MT3IOCR1 (Channel 3)

Bit 7	Bit 7	Bit 5	Bit 4	Description	
IOD3	IOD2	IOD1	IOD0	MT3GRD Function	TIOC3D Pin Function
0	0	0	0	Output compare register*2	Output retained*1
0	0	0	1		Initial output is 0 0 output at compare match
0	0	1	0		Initial output is 0 1 output at compare match
0	0	1	1		Initial output is 0 Toggle output at compare match
0	1	0	0		Output retained
0	1	0	1		Initial output is 1 0 output at compare match
0	1	1	0		Initial output is 1 1 output at compare match
0	1	1	1		Initial output is 1 Toggle output at compare match
1	x	0	0	Input capture register*2	Input capture at rising edge
1	x	0	1		Input capture at falling edge
1	x	1	x		Input capture at both edges

[Legend]
x: Don't care
Notes: 1. After reset, 0 is output until MTIOCR is set.
2. When the BFB bit in MT3MD0 is set to 1 and MT3GRD is used as a buffer register, this setting is invalid and input capture/ output compare is not generated.

Table 16.22 MT4IOCR0 (Channel 4)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB3	IOB2	IOB1	IOB0	MT4GRB Function	TIOC4B Pin Function
0	0	0	0	Output compare register	Output retained*
0	0	0	1		Initial output is 0 0 output at compare match
0	0	1	0		Initial output is 0 1 output at compare match
0	0	1	1		Initial output is 0 Toggle output at compare match
0	1	0	0		Output retained
0	1	0	1		Initial output is 1 0 output at compare match
0	1	1	0		Initial output is 1 1 output at compare match
0	1	1	1		Initial output is 1 Toggle output at compare match
1	x	0	0	Input capture register	Input capture at rising edge
1	x	0	1		Input capture at falling edge
1	x	1	x		Input capture at both edges

[Legend]
x: Don't care
Note: * After reset, 0 is output until MTIOCR is set.

Table 16.23 MT4IOCR1 (Channel 4)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOD3	IOD2	IOD1	IOD0	MT4GRD Function	TIOC4D Pin Function
0	0	0	0	Output compare register*2	Output retained*1
0	0	0	1		Initial output is 0 0 output at compare match
0	0	1	0		Initial output is 0 1 output at compare match
0	0	1	1		Initial output is 0 Toggle output at compare match
0	1	0	0		Output retained
0	1	0	1		Initial output is 1 0 output at compare match
0	1	1	0		Initial output is 1 1 output at compare match
0	1	1	1		Initial output is 1 Toggle output at compare match
1	x	0	0	Input capture register*2	Input capture at rising edge
1	x	0	1		Input capture at falling edge
1	x	1	x		Input capture at both edges

[Legend]
x: Don't care
Notes: 1. After reset, 0 is output until MTIOCR is set.
2. When the BFB bit in MT4MD0 is set to 1 and MT4GRD is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 16.24 MT6IOCR0 (Channel 6)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB3	IOB2	IOB1	IOB0	MT6GRB Function	TIOC6B Pin Function
0	0	0	0	Output compare register	Output retained*
0	0	0	1		Initial output is 0 0 output at compare match
0	0	1	0		Initial output is 0 1 output at compare match
0	0	1	1		Initial output is 0 Toggle output at compare match
0	1	0	0		Output retained
0	1	0	1		Initial output is 1 0 output at compare match
0	1	1	0		Initial output is 1 1 output at compare match
0	1	1	1		Initial output is 1 Toggle output at compare match
1	x	0	0	Input capture register	Input capture at rising edge
1	x	0	1		Input capture at falling edge
1	x	1	x		Input capture at both edges

[Legend]
x: Don't care
Note: * After reset, 0 is output until MTIOCR is set.

Table 16.25 MT6IOCR0 (Channel 6)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOD3	IOD2	IOD1	IOD0	MT6GRD Function	TIOC6D Pin Function
0	0	0	0	Output compare register*2	Output retained*1
0	0	0	1		Initial output is 0 0 output at compare match
0	0	1	0		Initial output is 0 1 output at compare match
0	0	1	1		Initial output is 0 Toggle output at compare match
0	1	0	0		Output retained
0	1	0	1		Initial output is 1 0 output at compare match
0	1	1	0		Initial output is 1 1 output at compare match
0	1	1	1		Initial output is 1 Toggle output at compare match
1	x	0	0	Input capture register*2	Input capture at rising edge
1	x	0	1		Input capture at falling edge
1	x	1	x		Input capture at both edges

[Legend]
x: Don't care
Notes: 1. After reset, 0 is output until MTIOCR is set.
2. When the BFB bit in MT6MD0 is set to 1 and MT6GRD is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 16.26 MT7IOCR0 (Channel 7)

Bit 7	Bit 6	Bit 5	Bit 4	Description	
IOB3	IOB2	IOB1	IOB0	MT7GRB Function	TIOC7B Pin Function
0	0	0	0	Output compare register	Output retained*
0	0	0	1		Initial output is 0 0 output at compare match
0	0	1	0		Initial output is 0 1 output at compare match
0	0	1	1		Initial output is 0 Toggle output at compare match
0	1	0	0		Output retained
0	1	0	1		Initial output is 1 0 output at compare match
0	1	1	0		Initial output is 1 1 output at compare match
0	1	1	1		Initial output is 1 Toggle output at compare match
1	x	0	0	Input capture register	Input capture at rising edge
1	x	0	1		Input capture at falling edge
1	x	1	x		Input capture at both edges

[Legend]
x: Don't care
Note: * After reset, 0 is output until MTIOCR is set.

Table 16.27 MT7IOCR1 (Channel 7)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOD3	IOD2	IOD1	IOD0	MT7GRD Function	TIOC7D Pin Function
0	0	0	0	Output compare register*2	Output retained*1
0	0	0	1		Initial output is 0 0 output at compare match
0	0	1	0		Initial output is 0 1 output at compare match
0	0	1	1		Initial output is 0 Toggle output at compare match
0	1	0	0		Output retained
0	1	0	1		Initial output is 1 0 output at compare match
0	1	1	0		Initial output is 1 1 output at compare match
0	1	1	1		Initial output is 1 Toggle output at compare match
1	x	0	0	Input capture register*2	Input capture at rising edge
1	x	0	1		Input capture at falling edge
1	x	1	x		Input capture at both edges

[Legend]
x: Don't care
Notes: 1. After reset, 0 is output until MTIOCR is set.
2. When the BFB bit in MT7MD0 is set to 1 and MT7GRD is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 16.28 MT0IOCR0 (Channel 0)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA3	IOA2	IOA1	IOA0	MT0GRA Function	TIOC0A Pin Function
0	0	0	0	Output compare register	Output retained*
0	0	0	1		Initial output is 0 0 output at compare match
0	0	1	0		Initial output is 0 1 output at compare match
0	0	1	1		Initial output is 0 Toggle output at compare match
0	1	0	0		Output retained
0	1	0	1		Initial output is 1 0 output at compare match
0	1	1	0		Initial output is 1 1 output at compare match
0	1	1	1		Initial output is 1 Toggle output at compare match
1	0	0	0	Input capture register	Input capture at rising edge
1	0	0	1		Input capture at falling edge
1	0	1	x		Input capture at both edges
1	1	x	x		Capture input source is channel 1/count clock Input capture at the MT1CNT counter count-up/count-down

[Legend]
x: Don't care
Note: * After reset, 0 is output until MTIOCR is set.

Table 16.29 MT0IOCR1 (Channel 0)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOC3	IOC2	IOC1	IOC0	MT0GRC Function	TIOC0C Pin Function
0	0	0	0	Output compare register*2	Output retained*1
0	0	0	1		Initial output is 0 0 output at compare match
0	0	1	0		Initial output is 0 1 output at compare match
0	0	1	1		Initial output is 0 Toggle output at compare match
0	1	0	0		Output retained
0	1	0	1		Initial output is 1 0 output at compare match
0	1	1	0		Initial output is 1 1 output at compare match
0	1	1	1		Initial output is 1 Toggle output at compare match
1	0	0	0	Input capture register*2	Input capture at rising edge
1	0	0	1		Input capture at falling edge
1	0	1	x		Input capture at both edges
1	1	x	x		Capture input source is channel 1/count clock Input capture at the MT1CNT counter count-up/count-down

[Legend]
x: Don't care
Notes: 1. After reset, 0 is output until MTIOCR is set.
2. When the BFA bit in MT0MD0 is set to 1 and MT0GRC is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 16.30 MT1IOCR0 (Channel 1)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA3	IOA2	IOA1	IOA0	MT1GRA Function	TIOC1A Pin Function
0	0	0	0	Output compare register	Output retained*
0	0	0	1		Initial output is 0 0 output at compare match
0	0	1	0		Initial output is 0 1 output at compare match
0	0	1	1		Initial output is 0 Toggle output at compare match
0	1	0	0		Output retained
0	1	0	1		Initial output is 1 0 output at compare match
0	1	1	0		Initial output is 1 1 output at compare match
0	1	1	1		Initial output is 1 Toggle output at compare match
1	0	0	0	Input capture register	Input capture at rising edge
1	0	0	1		Input capture at falling edge
1	0	1	x		Input capture at both edges
1	1	x	x		Input capture at generation of MT0GRA compare match/input capture

[Legend]
x: Don't care
Note: * After reset, 0 is output until MTIOCR is set.
Note: This function cannot be used in the SH72A0 Group.

Table 16.31 MT2IOCR0 (Channel 2)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA3	IOA2	IOA1	IOA0	MT2GRA Function	TIOC2A Pin Function
0	0	0	0	Output compare register	Output retained*
0	0	0	1		Initial output is 0 0 output at compare match
0	0	1	0		Initial output is 0 1 output at compare match
0	0	1	1		Initial output is 0 Toggle output at compare match
0	1	0	0		Output retained
0	1	0	1		Initial output is 1 0 output at compare match
0	1	1	0		Initial output is 1 1 output at compare match
0	1	1	1		Initial output is 1 Toggle output at compare match
1	x	0	0	Input capture register	Input capture at rising edge
1	x	0	1		Input capture at falling edge
1	x	1	x		Input capture at both edges

[Legend]
x: Don't care
Note: * After reset, 0 is output until MTIOCR is set.
Note: This function cannot be used in the SH72A0 Group.

Table 16.32 MT3IOCR0 (Channel 3)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA3	IOA2	IOA1	IOA0	MT3GRA Function	TIOC3A Pin Function
0	0	0	0	Output compare register	Output retained*
0	0	0	1		Initial output is 0 0 output at compare match
0	0	1	0		Initial output is 0 1 output at compare match
0	0	1	1		Initial output is 0 Toggle output at compare match
0	1	0	0		Output retained
0	1	0	1		Initial output is 1 0 output at compare match
0	1	1	0		Initial output is 1 1 output at compare match
0	1	1	1		Initial output is 1 Toggle output at compare match
1	x	0	0	Input capture register	Input capture at rising edge
1	x	0	1		Input capture at falling edge
1	x	1	x		Input capture at both edges

[Legend]
x: Don't care
Note: * After reset, 0 is output until MTIOCR is set.

Table 16.33 MT3IOCR1 (Channel 3)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOC3	IOC2	IOC1	IOC0	MT3GRC Function	TIOC3C Pin Function
0	0	0	0	Output compare register*2	Output retained*1
0	0	0	1		Initial output is 0 0 output at compare match
0	0	1	0		Initial output is 0 1 output at compare match
0	0	1	1		Initial output is 0 Toggle output at compare match
0	1	0	0		Output retained
0	1	0	1		Initial output is 1 0 output at compare match
0	1	1	0		Initial output is 1 1 output at compare match
0	1	1	1		Initial output is 1 Toggle output at compare match
1	x	0	0	Input capture register*2	Input capture at rising edge
1	x	0	1		Input capture at falling edge
1	x	1	x		Input capture at both edges

[Legend]
x: Don't care
Notes: 1. After reset, 0 is output until MTIOCR is set.
2. When the BFA bit in MT3MD0 is set to 1 and MT3GRD is used as a buffer register, this setting is invalid and input capture/ output compare is not generated.

Table 16.34 MT4IOCR0 (Channel 4)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA3	IOA2	IOA1	IOA0	MT4GRA Function	TIOC4A Pin Function
0	0	0	0	Output compare register	Output retained*
0	0	0	1		Initial output is 0 0 output at compare match
0	0	1	0		Initial output is 0 1 output at compare match
0	0	1	1		Initial output is 0 Toggle output at compare match
0	1	0	0		Output retained
0	1	0	1		Initial output is 1 0 output at compare match
0	1	1	0		Initial output is 1 1 output at compare match
0	1	1	1		Initial output is 1 Toggle output at compare match
1	x	0	0	Input capture register	Input capture at rising edge
1	x	0	1		Input capture at falling edge
1	x	1	x		Input capture at both edges

[Legend]
x: Don't care
Note: * After reset, 0 is output until MTIOCR is set.

Table 16.35 MT4IOCR1 (Channel 4)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOC3	IOC2	IOC1	IOC0	MT4GRC Function	TIOC4C Pin Function
0	0	0	0	Output compare register*2	Output retained*1
0	0	0	1		Initial output is 0 0 output at compare match
0	0	1	0		Initial output is 0 1 output at compare match
0	0	1	1		Initial output is 0 Toggle output at compare match
0	1	0	0		Output retained
0	1	0	1		Initial output is 1 0 output at compare match
0	1	1	0		Initial output is 1 1 output at compare match
0	1	1	1		Initial output is 1 Toggle output at compare match
1	x	0	0	Input capture register*2	Input capture at rising edge
1	x	0	1		Input capture at falling edge
1	x	1	x		Input capture at both edges

[Legend]
x: Don't care
Notes: 1. After reset, 0 is output until MTIOCR is set.
2. When the BFA bit in MT4MD0 is set to 1 and MT4GRC is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 16.36 MT6IOCR0 (Channel 6)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA3	IOA2	IOA1	IOA0	MT6GRA Function	TIOC6A Pin Function
0	0	0	0	Output compare register	Output retained*
0	0	0	1		Initial output is 0 0 output at compare match
0	0	1	0		Initial output is 0 1 output at compare match
0	0	1	1		Initial output is 0 Toggle output at compare match
0	1	0	0		Output retained
0	1	0	1		Initial output is 1 0 output at compare match
0	1	1	0		Initial output is 1 1 output at compare match
0	1	1	1		Initial output is 1 Toggle output at compare match
1	x	0	0	Input capture register	Input capture at rising edge
1	x	0	1		Input capture at falling edge
1	x	1	x		Input capture at both edges

[Legend]
x: Don't care
Note: * After reset, 0 is output until MTIOCR is set.
Note: This function cannot be used in the SH72A0 Group.

Table 16.37 MT6IOCR1 (Channel 6)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOC3	IOC2	IOC1	IOC0	MT6GRC Function	TIOC6C Pin Function
0	0	0	0	Output compare register*2	Output retained*1
0	0	0	1		Initial output is 0 0 output at compare match
0	0	1	0		Initial output is 0 1 output at compare match
0	0	1	1		Initial output is 0 Toggle output at compare match
0	1	0	0		Output retained
0	1	0	1		Initial output is 1 0 output at compare match
0	1	1	0		Initial output is 1 1 output at compare match
0	1	1	1		Initial output is 1 Toggle output at compare match
1	x	0	0	Input capture register*2	Input capture at rising edge
1	x	0	1		Input capture at falling edge
1	x	1	x		Input capture at both edges

[Legend]
x: Don't care
Notes: 1. After reset, 0 is output until MTIOCR is set.
2. When the BFA bit in MT6MD0 is set to 1 and MT6GRC is used as a buffer register, this setting is invalid and input capture/output compare is not generated.
Note: This function cannot be used in the SH72A0 Group.

Table 16.38 MT7IOCR0 (Channel 7)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOA3	IOA2	IOA1	IOA0	MT7GRA Function	TIOC7A Pin Function
0	0	0	0	Output compare register	Output retained*
0	0	0	1		Initial output is 0 0 output at compare match
0	0	1	0		Initial output is 0 1 output at compare match
0	0	1	1		Initial output is 0 Toggle output at compare match
0	1	0	0		Output retained
0	1	0	1		Initial output is 1 0 output at compare match
0	1	1	0		Initial output is 1 1 output at compare match
0	1	1	1		Initial output is 1 Toggle output at compare match
1	x	0	0	Input capture register	Input capture at rising edge
1	x	0	1		Input capture at falling edge
1	x	1	x		Input capture at both edges

[Legend]
x: Don't care
Note: * After reset, 0 is output until MTIOCR is set.

Table 16.39 MT7IOCR1 (Channel 7)

Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOC3	IOC2	IOC1	IOC0	MT7GRC Function	TIOC7C Pin Function
0	0	0	0	Output compare register*2	Output retained*1
0	0	0	1		Initial output is 0 0 output at compare match
0	0	1	0		Initial output is 0 1 output at compare match
0	0	1	1		Initial output is 0 Toggle output at compare match
0	1	0	0		Output retained
0	1	0	1		Initial output is 1 0 output at compare match
0	1	1	0		Initial output is 1 1 output at compare match
0	1	1	1		Initial output is 1 Toggle output at compare match
1	x	0	0	Input capture register*2	Input capture at rising edge
1	x	0	1		Input capture at falling edge
1	x	1	x		Input capture at both edges

[Legend]
x: Don't care
Notes: 1. After reset, 0 is output until MTIOCR is set.
2. When the BFA bit in MT7MD0 is set to 1 and MT7GRC is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 16.40 MT5IOCRU, MT5IOCRV, and MT5IOCRW (Channel 5)

Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Description	
IOC4	IOC3	IOC2	IOC1	IOC0	MT5GRU, MT5GRV, and MT5GRW Function	TIC5U, TIC5V, and TIC5W Pin Function
0	0	0	0	0	Output compare register	Output compare
0	0	0	0	1		Setting prohibited
0	0	0	1	x		Setting prohibited
0	0	1	x	x		Setting prohibited
0	1	x	x	x		Setting prohibited
1	0	0	0	0	Input capture register	Setting prohibited
1	0	0	0	1		Input capture at rising edge
1	0	0	1	0		Input capture at falling edge
1	0	0	1	1		Input capture at both edges
1	0	1	x	x		Setting prohibited
1	1	0	0	0		Setting prohibited
1	1	0	0	1		Measurement of low pulse width of external input signal Capture at trough
1	1	0	1	0		Measurement of low pulse width of external input signal Capture at crest
1	1	0	1	1		Measurement of low pulse width of external input signal Capture at crest and trough
1	1	1	0	0		Setting prohibited
1	1	1	0	1		Measurement of high pulse width of external input signal Capture at trough
1	1	1	1	0		Measurement of high pulse width of external input signal Capture at crest
1	1	1	1	1		Measurement of high pulse width of external input signal Capture at crest and trough

[Legend]
x: Don't care
Note: This function cannot be used in the SH72A0 Group.

16.2.5 MT5 Compare Match Clear Register (MT5CMC)

Address H'FFFF 94B6

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	CMPCLR5U	CMPCLR5V	CMPCLR5W
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b3	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b2	CMPCLR5U	MTCNT Counter Compare Clear 5U	Enables or disables requests to clear the MT5CNTU counter at MT5GRU compare match or input capture. 0: Disables MT5CNTU counter to be cleared to H'0000 1: Enables MT5CNTU counter to be cleared to H'0000	R/W
b1	CMPCLR5V	MTCNT Counter Compare Clear 5V	Enables or disables requests to clear the MT5CNTV counter at MT5GRV compare match or input capture. 0: Disables MT5CNTV to be cleared to H'0000 1: Enables MT5CNTV to be cleared to H'0000	R/W
b0	CMPCLR5W	MTCNT Counter Compare Clear 5W	Enables or disables requests to clear the MT5CNTW counter at MT5GRW compare match or input capture. 0: Disables MT5CNTW to be cleared to H'0000 1: Enables MT5CNTW to be cleared to H'0000	R/W

MT5CMC is an 8-bit register that specifies requests to clear counters MT5CNTU, MT5CNTV, and MT5CNTW.

16.2.6 MTi Interrupt Enable Register 0 (MTiIEN0) (i = 0 to 4, 6, 7), MT0 Interrupt Enable Register 1 (MT0IEN1), MT5 Interrupt Enable Register 0 (MT5IEN0)

Address MT1IEN0: H'FFFF 8384, MT2IEN0: H'FFFF 8404

	b7	b6	b5	b4	b3	b2	b1	b0
	TTGE	—	TCIEU	TCIEV	—	—	TGIEB	TGIEA
After Reset	0	0	0	0	0	0	0	0

Address MT0IEN0: H'FFFF 8304, MT3IEN0: H'FFFF 8208, MT6IEN0: H'FFFF 9208

	b7	b6	b5	b4	b3	b2	b1	b0
	TTGE	—	—	TCIEV	TGIED	TGIEC	TGIEB	TGIEA
After Reset	0	0	0	0	0	0	0	0

Address MT4IEN0: H'FFFF 8209, MT7IEN0: H'FFFF 9209

	b7	b6	b5	b4	b3	b2	b1	b0
	TTGE	TTGE2	—	TCIEV	TGIED	TGIEC	TGIEB	TGIEA
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	TTGE	A/D Converter Start Request Enable	Enables or disables generation of A/D converter start requests by MTGRA input capture/compare match. 0: A/D converter start request generation disabled 1: A/D converter start request generation enabled	R/W
b6	TTGE2	A/D Converter Start Request Enable 2	Enables or disables generation of A/D converter start requests by MT4CNT (MT7CNT) counter underflow (trough) in complementary PWM mode. 0: A/D converter start request generation disabled 1: A/D converter start request generation enabled In channels 0 to 3 and 6, bit 6 is reserved. It is always read as 0 and the write value should always be 0.	R/W
b5	TCIEU	Underflow Interrupt Enable	Enables or disables interrupt requests (TCIU) by the TCFU flag when the TCFU flag in MTSR is set to 1 in channels 1 and 2. 0: Generation of interrupt requests (TCIU) by the TCFU flag disabled 1: Generation of interrupt requests (TCIU) by the TCFU flag enabled In channels 0, 3, 4, 6, and 7, bit 5 is reserved. It is always read as 0 and the write value should always be 0.	R/W
b4	TCIEV	Overflow Interrupt Enable	Enables or disables interrupt requests (TCIV) by the TCFV flag when the TCFV flag in MTSR is set to 1. 0: Generation of interrupt requests (TCIV) by the TCFV flag disabled 1: Generation of interrupt requests (TCIV) by the TCFV flag enabled	R/W
b3	TGIED	TGR Interrupt Enable D	Enables or disables interrupt requests (TGID) by the TGFD flag when the TGFD flag in MTSR is set to 1 in channels 0, 3, 4, 6 and 7. 0: Generation of interrupt requests (TGID) by the TGFD flag disabled 1: Generation of interrupt requests (TGID) by the TGFD flag enabled In channels 1 and 2, bit 3 is reserved. It is always read as 0 and the write value should always be 0.	R/W
b2	TGIEC	TGR Interrupt Enable C	Enables or disables interrupt requests (TGIC) by the TGFC flag when the TGFC flag in MTSR is set to 1 in channels 0, 3, 4, 6 and 7. 0: Generation of interrupt requests (TGIC) by the TGFC flag disabled 1: Generation of interrupt requests (TGIC) by the TGFC flag enabled In channels 1 and 2, bit 2 is reserved. It is always read as 0 and the write value should always be 0.	R/W

Bit	Symbol	Bit Name	Description	R/W
b1	TGIEB	TGR Interrupt Enable B	Enables or disables interrupt requests (TGIB) by the TGFB flag when the TGFB flag in MTSR is set to 1. 0: Generation of interrupt requests (TGIB) by the TGFB flag disabled 1: Generation of interrupt requests (TGIB) by the TGFB flag enabled	R/W
b0	TGIEA	TGR Interrupt Enable A	Enables or disables interrupt requests (TGIA) by the TGFA flag when the TGFA flag in MTSR is set to 1. 0: Generation of interrupt requests (TGIA) by the TGFA flag disabled 1: Generation of interrupt requests (TGIA) by the TGFA flag enabled	R/W

Address MT0IEN1: H'FFFF 8324

	b7	b6	b5	b4	b3	b2	b1	b0
	TTGE2	—	—	—	—	—	TGIEF	TGIEE
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	TTGE2	A/D Converter Start Request Enable 2	Enables or disables A/D converter start requests by compare match of the MT0CNT counter and MT0GRE. 0: A/D converter start request generation disabled 1: A/D converter start request generation enabled	R/W
b6 to b2	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b1	TGIEF	TGR Interrupt Enable F	Enables or disables interrupt requests by compare match of the MT0CNT counter and MT0GRF. 0: Generation of interrupt requests (TGIF) by the TGFF flag disabled 1: Generation of interrupt requests (TGIF) by the TGFF flag enabled	R/W
b0	TGIEE	TGR Interrupt Enable E	Enables or disables interrupt requests by compare match of the MT0CNT counter and MT0GRE. 0: Generation of interrupt requests (TGIE) by the TGFE flag disabled 1: Generation of interrupt requests (TGIE) by the TGFE flag enabled	R/W

Address MT5IEN0: H'FFFF 94B2

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	TGIE5U	TGIE5V	TGIE5W
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b3	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b2	TGIE5U	TGR Interrupt Enable 5U	Enables or disables interrupt requests (TGIU_5) by the CMFU5 flag when the CMFU5 flag in MT5SR0 is set to 1. 0: Generation of interrupt request (TGIU_5) disabled 1: Generation of interrupt request (TGIU_5) enabled	R/W
b1	TGIE5V	TGR Interrupt Enable 5V	Enables or disables interrupt requests (TGIV_5) by the CMFV5 flag when the CMFV5 flag in MT5SR0 is set to 1. 0: Generation of interrupt request (TGIV_5) disabled 1: Generation of interrupt request (TGIV_5) enabled	R/W
b0	TGIE5W	TGR Interrupt Enable 5W	Enables or disables interrupt requests (TGIW_5) by the CMFW5 flag when the CMFW5 flag in MT5SR0 is set to 1. 0: Generation of interrupt request (TGIW_5) disabled 1: Generation of interrupt request (TGIW_5) enabled	R/W

MTIEN is an 8-bit register that controls enabling or disabling interrupt requests for each channel.

16.2.7 MTi Status Register 0 (MTiSR0) (i = 0 to 4, 6, and 7), MT0 Status Register 1 (MT0SR1), MT5 Status Register 0 (MT5SR0)

Address MT0SR0: H'FFFF 8305

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	TCFV	TGFD	TGFC	TGFB	TGFA
After Reset	1	1	0	0	0	0	0	0

Address MT1SR0: H'FFFF 8385, MT2SR0: H'FFFF 8405

	b7	b6	b5	b4	b3	b2	b1	b0
	TCFD	—	TCFU	TCFV	—	—	TGFB	TGFA
After Reset	1	1	0	0	0	0	0	0

Address MT3SR0: H'FFFF 822C, MT4SR0: H'FFFF 822D, MT6SR0: H'FFFF 922C, MT7SR0: H'FFFF 922D

	b7	b6	b5	b4	b3	b2	b1	b0
	TCFD	—	—	TCFV	TGFD	TGFC	TGFB	TGFA
After Reset	1	1	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	TCFD	Count Direction Flag	Status flag that indicates the direction in which the MTCNT counter counts in channels 1 to 4, 6, and 7. 0: MTCNT counter counts down 1: MTCNT counter counts up In channel 0, bit 7 is reserved. It is always read as 1 and the write value should always be 1.	R
b6	—	Reserved	This bit is always read as 0. The write value should always be 0.	R
b5	TCFU	Underflow Flag	Status flag that indicates that the MTCNT counter underflow has occurred when channels 1 and 2 are set to phase counting mode. Only 0 can be written, for flag clearing. [Condition of being set to 1] • When the MTCNT counter value underflows (changes from H'0000 to H'FFFF) [Condition of being set to 0] • When 0 is written to the TCFU flag after reading TCFU = 1 In channels 0, 3, 4, 6, and 7, bit 5 is reserved. It is always read as 0 and the write value should always be 0.	R/W*
b4	TCFV	Overflow Flag	Status flag that indicates that MTCNT overflow has occurred. [Condition of being set to 1] • When the MTCNT counter value overflows (changes from H'FFFF to H'0000) • In channel 4 or 7 when the MT4CNT or MT7CNT counter value changes to H'0000 after down-counting in complementary PWM mode [Condition of being set to 0] • When 0 is written to the TCFV flag after reading TCFV = 1	R/W*

Bit	Symbol	Bit Name	Description	R/W
b3	TGFD	Input Capture/Output Compare Flag D	Status flag that indicates the occurrence of MTGRD input capture or compare match in channels 0, 3, 4, 6, and 7. Only 0 can be written, for flag clearing. [Conditions of being set to 1] • When MTCNT = MTGRD and MTGRD is functioning as output compare register • When the MTCNT counter value is transferred to MTGRD by input capture signal and MTGRD is functioning as input capture register [Conditions of being set to 0] • When 0 is written to the TGFD flag after reading TGFD = 1 • When the transfer acknowledge signal is received from DMAC after DMAC is activated by the TGID interrupt In channels 1 and 2, bit 3 is reserved. It is always read as 0 and the write value should always be 0.	R/W*
b2	TGFC	Input Capture/Output Compare Flag C	Status flag that indicates the occurrence of MTGRC input capture or compare match in channels 0, 3, 4, 6, and 7. Only 0 can be written, for flag clearing. [Conditions of being set to 1] • When MTCNT = MTGRC and MTGRC is functioning as output compare register • When the MTCNT counter value is transferred to MTGRC by input capture signal and MTGRC is functioning as input capture register [Conditions of being set to 0] • When 0 is written to the TGFC flag after reading TGFC = 1 • When the transfer acknowledge signal is received from DMAC after DMAC is activated by the TGIC interrupt	R/W*
b1	TGFB	Input Capture/Output Compare Flag B	Status flag that indicates the occurrence of MTGRB input capture or compare match. Only 0 can be written, for flag clearing. [Conditions of being set to 1] • When MTCNT = MTGRB and MTGRB is functioning as output compare register • When the MTCNT counter value is transferred to MTGRB by input capture signal and MTGRB is functioning as input capture register [Conditions of being set to 0] • When 0 is written to the TGFB flag after reading TGFB = 1 • When the transfer acknowledge signal is received from DMAC after DMAC is activated by the TGIB interrupt	R/W*
b0	TGFA	Input Capture/Output Compare Flag A	Status flag that indicates the occurrence of MTGRA input capture or compare match. Only 0 can be written, for flag clearing. [Conditions of being set to 1] • When MTCNT = MTGRA and MTGRA is functioning as output compare register • When the MTCNT counter value is transferred to MTGRA by input capture signal and MTGRA is functioning as input capture register [Conditions of being set to 0] • When 0 is written to the TGFA flag after reading TGFA = 1 • When the transfer acknowledge signal is received from DMAC after DMAC is activated by the TGIA interrupt	R/W*

Note: * Only 0 can be written to clear the status. When a value other than 0 is written to, the flag retains the status before the writing, and the value remains unchanged.

Address MT0SR1: H'FFFF 8325

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	TGFF	TGFE
After Reset	1	1	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7, b6	—	Reserved	These bits are always read as 1. The write value should always be 1.	R
b5 to b2	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b1	TGFF	Compare Match Flag F	Status flag that indicates the occurrence of compare match between the MT0CNT counter and MT0GRF. [Condition of being set to 1] • When MT0CNT = MT0GRF and MTGRF is functioning as compare register [Condition of being set to 0] • When 0 is written to the TGFF flag after reading TGFF = 1	R/W*
b0	TGFE	Compare Match Flag E	Status flag that enables or disables interrupt requests to be occurred by compare match between the MT0CNT counter and MT0GRE. [Condition of being set to 1] • MT0CNT = MT0GRE and MTGRE is functioning as compare register [Condition of being set to 0] • When 0 is written to the TGFE flag after reading TGFE = 1	R/W*

Note: * Only 0 can be written to clear the status. When a value other than 0 is written to, the flag retains the status before the writing, and the value remains unchanged.

Address MT5SR0: H'FFFF 94B0

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	CMFU5	CMFV5	CMFW5
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b3	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b2	CMFU5	Compare Match/Input Capture Flag 5U	Status flag that indicates the occurrence of MT5GRU input capture or compare match. Only 0 can be written, for flag clearing. [Conditions of being set to 1] • When MT5CNTU = MT5GRU and MT5GRU is functioning as compare match register • When the MT5CNTU counter value is transferred to MT5GRU by input capture signal and MT5GRU is functioning as input capture register • When the MT5CNTU counter value is transferred to MT5GRU and MT5GRU is functioning as the pulse width measurement of the external input signal [Conditions of being set to 0] • When 0 is written to the CMFU5 flag after reading CMFU5 = 1 • When the transfer acknowledge signal is received from DMAC after DMAC is activated by a TGIV_5 interrupt.	R/W*1

Bit	Symbol	Bit Name	Description	R/W
b1	CMFV5	Compare Match/Input Capture Flag 5V	Status flag that indicates the occurrence of MT5GRV input capture or compare match. Only 0 can be written, for flag clearing. [Conditions of being set to 1] • When MT5CNTV = MT5GRV and MT5GRV is functioning as compare match register • When the MT5CNTV counter value is transferred to MT5GRV by input capture signal and MT5GRV is functioning as input capture register • When the MT5CNTV counter value is transferred to MT5GRV and MT5GRV is functioning as the pulse width measurement of the external input signal*2 [Conditions of being set to 0] • When 0 is written to the CMFV5 flag after reading CMFV5 = 1 • When the transfer acknowledge signal is received from DMAC after DMAC is activated by a TGIV_5 interrupt.	R/W*1
b0	CMFW5	Compare Match/Input Capture Flag 5W	Status flag that indicates the occurrence of MT5GRW input capture or compare match. Only 0 can be written, for flag clearing. [Conditions of being set to 1] • When MT5CNTW = MT5GRW and MT5GRW is functioning as compare match register • When the MT5CNTW counter value is transferred to MT5GRW by input capture signal and MT5GRW is functioning as input capture register • When the MT5CNTW counter value is transferred to MT5GRW and MT5GRW is functioning as the pulse width measurement of the external input signal [Conditions of being set to 0] • When 0 is written to the CMFW5 flag after reading CMFW5 = 1 • When the transfer acknowledge signal is received from DMAC after DMAC is activated by a TGIW_5 interrupt.	R/W*1

Notes: 1. Only 0 can be written to clear the status. When a value other than 0 is written to, the flag retains the status before the writing, and the value remains unchanged.
2. The transfer timing is specified by the IOC bit of MT5IOCRU, MT5IOCRV, and MT5IOCRW.

MTSR is an 8-bit register that indicates the status of each channel.

16.2.8 MTj Buffer Operation Transfer Mode Register (MTjBTM) (j = 0, 3, 4, 6, and 7)

Address MT0BTM: H'FFFF 8326

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	TTSE	TTSB	TTSA
After Reset	0	0	0	0	0	0	0	0

Address MT3BTM: H'FFFF 8238, MT4BTM: H'FFFF 8239, MT6BTM: H'FFFF 9238, MT7BTM: H'FFFF 9239

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	TTSB	TTSA
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b3	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b2	TTSE	Timing Select E	Specifies the timing for transferring data from MT0GRF to MT0GRE when they are used together for buffer operation. 0: When compare match E occurs in channel 0 1: When the MT0CNT counter is cleared In channels 3, 4, 6, and 7, bit 2 is reserved. It is always read as 0 and the write value should always be 0.	R/W
b1	TTSB	Timing Select B	Specifies the timing for transferring data from MTGRD to MTGRB when they are used together for buffer operation of each channel. 0: When compare match B occurs in each channel 1: When the MTCNT counter is cleared in each channel	R/W
b0	TTSA	Timing Select A	Specifies the timing for transferring data from MTGRC to MTGRA when they are used together for buffer operation of each channel. 0: When compare match A occurs in each channel 1: When the MTCNT counter is cleared in each channel	R/W

MTBTM is an 8-bit register that specifies the timing for transferring data from buffer registers to MTGR in PWM mode.

16.2.9 MT1 Input Capture Control Register (MT1ICCR)

Address H'FFFF 8390

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	I2BE	I2AE	I1BE	I1AE
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b4	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b3	I2BE	Input Capture Enable	Specifies whether to include the TIOC2B pin in the MT1GRB input capture conditions. 0: Does not include the input capture conditions 1: Includes the input capture conditions*	R/W
b2	I2AE	Input Capture Enable	Specifies whether to include the TIOC2A pin in the MT1GRA input capture conditions. 0: Does not include the input capture conditions 1: Includes the input capture conditions*	R/W
b1	I1BE	Input Capture Enable	Specifies whether to include the TIOC1B pin in the MT2GRB input capture conditions. 0: Does not include the input capture conditions 1: Includes the input capture conditions	R/W
b0	I1AE	Input Capture Enable	Specifies whether to include the TIOC1A pin in the MT2GRA input capture conditions. 0: Does not include the input capture conditions 1: Includes the input capture conditions*	R/W

Note: * This function cannot be used in the SH72A0 Group.

MT1ICCR is an 8-bit register that specifies input capture conditions when counters MT1CNT and MT2CNT are cascaded.

16.2.10 MT6 Synchronization Clear Register (MT6SYC)

Address H'FFFF 9250

	b7	b6	b5	b4	b3	b2	b1	b0
	CE0A	CE0B	CE0C	CE0D	CE1A	CE1B	CE2A	CE2B
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	CE0A	Clear Enable 0A	Enables or disables counter clearing when the TGFA flag of MT0SR0 is set. 0: Disables counter clearing by the TGFA flag 1: Enables counter clearing by the TGFA flag	R/W
b6	CE0B	Clear Enable 0B	Enables or disables counter clearing when the TGFB flag of MT0SR0 is set. 0: Disables counter clearing by the TGFB flag 1: Enables counter clearing by the TGFB flag	R/W
b5	CE0C	Clear Enable 0C	Enables or disables counter clearing when the TGFC flag of MT0SR0 is set. 0: Disables counter clearing by the TGFC flag 1: Enables counter clearing by the TGFC flag	R/W
b4	CE0D	Clear Enable 0D	Enables or disables counter clearing when the TGFD flag of MT0SR0 is set. 0: Disables counter clearing by the TGFD flag 1: Enables counter clearing by the TGFD flag	R/W
b3	CE1A	Clear Enable 1A	Enables or disables counter clearing when the TGFA flag of MT1SR0 is set. 0: Disables counter clearing by the TGFA flag 1: Enables counter clearing by the TGFA flag	R/W
b2	CE1B	Clear Enable 1B	Enables or disables counter clearing when the TGFB flag of MT1SR0 is set. 0: Disables counter clearing by the TGFB flag 1: Enables counter clearing by the TGFB flag	R/W
b1	CE2A	Clear Enable 2A	Enables or disables counter clearing when the TGFA flag of MT2SR0 is set. 0: Disables counter clearing by the TGFA flag 1: Enables counter clearing by the TGFA flag	R/W
b0	CE2B	Clear Enable 2B	Enables or disables counter clearing when the TGFB flag of MT2SR0 is set. 0: Disables counter clearing by the TGFB flag 1: Enables counter clearing by the TGFB flag	R/W

MT6SYC is an 8-bit register that specifies synchronous clear conditions for counters MT6CNT and MT7CNT of MTU-III.

16.2.11 MTi Counter (MTiCNT) (i = 0 to 4, 6, and 7), MT5 Counters U, V, and W (MT5CNTU, MT5CNTV, MT5CNTW)

Address: MT0CNT: H'FFFF 8306, MT1CNT: H'FFFF 8386, MT2CNT: H'FFFF 8406, MT3CNT: H'FFFF 8210, MT4CNT: H'FFFF 8212, MT6CNT: H'FFFF 9210, MT7CNT: H'FFFF 9212, MT5CNTU: H'FFFF 9480, MT5CNTV: H'FFFF 9490, MT5CNTW: H'FFFF 94A0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Note: * The MTCNT counter must not be accessed in eight bits; they should always be accessed in 16-bit units.

The MTCNT counter is a 16-bit readable/writable counter.

The MTCNT counter must not be accessed in eight bits; they should always be accessed in 16-bit units.

16.2.12 MTi General Registers A, B (MTiGRA, MTiGRB) (i = 0 to 4, 6, and 7), MTj General Registers C, D, and E (MTjGRC, MTjGRD, and MTjGRE) (j = 0, 3, 4, 6, and 7), MTk General Register F (MTkGRF) (k = 0, 4, and 7), MT5 General Registers U, V, and W (MT5GRU, MT5GRV, MT5GRW)

Address
MT0GRA: H'FFFF 8308, MT0GRB: H'FFFF 830A, MT0GRC: H'FFFF 830C, MT0GRD: H'FFFF 830E,
MT0GRE: H'FFFF 8320, MT0GRF: H'FFFF 8322,
MT1GRA: H'FFFF 8388, MT1GRB: H'FFFF 838A,
MT2GRA: H'FFFF 8408, MT2GRB: H'FFFF 840A,
MT3GRA: H'FFFF 8218, MT3GRB: H'FFFF 821A, MT3GRC: H'FFFF 8224, MT3GRD: H'FFFF 8226,
MT3GRE: H'FFFF 8272,
MT4GRA: H'FFFF 821C, MT4GRB: H'FFFF 821E, MT4GRC: H'FFFF 8228, MT4GRD: H'FFFF 822A,
MT4GRE: H'FFFF 8274, MT4GRF: H'FFFF 8276,
MT6GRA: H'FFFF 9218, MT6GRB: H'FFFF 921A, MT6GRC: H'FFFF 9224, MT6GRD: H'FFFF 9226,
MT6GRE: H'FFFF 9272,
MT7GRA: H'FFFF 921C, MT7GRB: H'FFFF 921E, MT7GRC: H'FFFF 9228, MT7GRD: H'FFFF 922A,
MT7GRE: H'FFFF 9274, MT7GRF: H'FFFF 9276,
MT5GRU: H'FFFF 9482, MT5GRV: H'FFFF 9492, MT5GRW: H'FFFF 94A2

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Note: * MTGR must not be accessed in eight bits; they should always be accessed in 16-bit units. MTGR is initialized to H'FFFF.

MTGR is an 16-bit readable/writable register. The MTU-III has 35 general registers, six for channel 0, two each for channels 1 and 2, five each for channels 3 and 6, six each for channels 4 and 7, and three for channel 5.

MTGRA, MTGRB, MTGRC, and MTGRD function as either output compare or input capture registers. MTGRC and MTGRD for channels 0, 3, 4, 6, and 7 can also be designated for operation as buffer registers. The combinations of MTGR and buffer registers are MTGRA and MTGRC, and MTGRB and MTGRD.

MT0GRE and MT0GRF function as compare registers. When the MT0CNT count matches the MT0GRE value, an A/D converter start request can be issued. MTGRF can also be designated for operation as a buffer register. The combination of MTGR and buffer registers is MTGRE and MTGRF.

MT5GRU, MT5GRV, and MT5GRW function as compare match, input capture, or external pulse width measurement registers.

16.2.13 MT01234 Start Register (MT01234STR), MT67 Start Register (MT67STR), MT5 Start Register (MT5STR)

Address MT01234STR : H'FFFF 8280

	b7	b6	b5	b4	b3	b2	b1	b0
	CST4	CST3	—	—	—	CST2	CST1	CST0
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	CST4	Counter Start 4	These bits select operation or stoppage for the MTCNT counter. If 0 is written to the CST bit during operation with the TIOC pin designated for output, the counter stops but the TIOC pin output compare output level is retained. If a write to MTIOCR0 is done when the CST bit is cleared to 0, the pin output level will be changed to the set initial output value. 0: MT4CNT and MT3CNT operation is stopped 1: MT4CNT and MT3CNT perform count operation	R/W
b6	CST3	Counter Start 3		R/W
b5 to b3	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b2	CST2	Counter Start 2	These bits select operation or stoppage for the MTCNT counter. If 0 is written to the CST bit during operation with the TIOC pin designated for output, the counter stops but the TIOC pin output compare output level is retained. If a write to MTIOCR0 is done when the CST bit is cleared to 0, the pin output level will be changed to the set initial output value. 0: MT2CNT to MT0CNT count operation is stopped 1: MT2CNT to MT0CNT perform count operation	R/W
b1	CST1	Counter Start 1		R/W
b0	CST0	Counter Start 0		R/W

Note: When the SCHi bit in the MTCSYSTR register is set to 1, the corresponding CSTi bit is also set to 1 automatically.

Address MT67STR: H'FFFF 9280

	b7	b6	b5	b4	b3	b2	b1	b0
	CST7	CST6	—	—	—	—	—	—
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	CST7	Counter Start 7	These bits select operation or stoppage for the MTCNT counter. If 0 is written to the CST bit during operation with the TIOC pin designated for output, the counter stops but the TIOC pin output compare output level is retained. If a write to MTIOCR0 is done when the CST bit is cleared to 0, the pin output level will be changed to the set initial output value. 0: MT7CNT and MT6CNT count operation is stopped 1: MT7CNT and MT6CNT perform count operation	R/W
b6	CST6	Counter Start 6		R/W
b5 to b0	—	Reserved	These bits are always read as 0. The write value should always be 0.	R

Note: When the SCHi bit in the MTCSYSTR register is set to 1, the corresponding CSTi bit is also set to 1 automatically.

Address MT5STR: H'FFFF 94B4

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	CSTU5	CSTV5	CSTW5
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b3	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b2	CSTU5	Counter Start U5	This bit selects operation or stoppage for MT5CNTU. 0: MT5CNTU count operation is stopped 1: MT5CNTU performs count operation	R/W
b1	CSTV5	Counter Start V5	This bit selects operation or stoppage for MT5CNTV. 0: MT5CNTV count operation is stopped 1: MT5CNTV performs count operation	R/W
b0	CSTW5	Counter Start W5	This bit selects operation or stoppage for MT5CNTW. 0: MT5CNTW count operation is stopped 1: MT5CNTW performs count operation	R/W

MT01234STR and MT67STR are 8-bit registers that select operation/stoppage of the MTCNT counter for channels 0 to 4, 6, and 7.

MT5STR selects operation/stoppage of the MT5CNTU, MT5CNTV, and MT5CNTW counters for channel 5.

When setting the operating mode in MTMD0 or setting the count clock in MTCR for the MTCNT counter, first stop the MTCNT counter.

16.2.14 MT01234 Synchronous Register (MT01234SY), MT67 Synchronous Register (MT67SY)

Address MT01234SY: H'FFFF 8281

	b7	b6	b5	b4	b3	b2	b1	b0
	SYNC4	SYNC3	—	—	—	SYNC2	SYNC1	SYNC0
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	SYNC4	Timer Synchronous Operation 4	These bits are used to select whether operation is independent of or synchronized with other channels. When synchronous operation is selected, the MTCNT counter synchronous presetting of multiple channels, and synchronous clearing by counter clearing on another channels, are possible. To set synchronous operation, the SYNC bits for at least two channels must be set to 1. To set synchronous clearing, in addition to the SYNC bit, the MTCNT counter clearing source must also be set by means of the CCLR bit in MTCR. 0: MT4CNT and MT3CNT operate independently (the MTCNT counter presetting/clearing is unrelated to other channels) 1: MT4CNT and MT3CNT perform synchronous operation (the MTCNT counter synchronous presetting/synchronous clearing is possible)	R/W
b6	SYNC3	Timer Synchronous Operation 3		R/W
b5 to b3	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b2	SYNC2	Timer Synchronous Operation 2	These bits are used to select whether operation is independent of or synchronized with other channels. When synchronous operation is selected, the MTCNT counter synchronous presetting of multiple channels, and synchronous clearing by counter clearing on another channels, are possible. To set synchronous operation, the SYNC bits for at least two channels must be set to 1. To set synchronous clearing, in addition to the SYNC bit, the MTCNT counter clearing source must also be set by means of the CCLR bit in MTCR. 0: MT2CNT to MT0CNT operate independently (the MTCNT counter presetting/clearing is unrelated to other channels) 1: MT2CNT to MT0CNT perform synchronous operation (the MTCNT counter synchronous presetting/synchronous clearing is possible)	R/W
b1	SYNC1	Timer Synchronous Operation 1		R/W
b0	SYNC0	Timer Synchronous Operation 0		R/W

Address MT67SY: H'FFFF 9281

	b7	b6	b5	b4	b3	b2	b1	b0
	SYNC7	SYNC6	—	—	—	—	—	—
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	SYNC7	Timer Synchronous Operation 7	These bits are used to select whether operation is independent of or synchronized with other channels. When synchronous operation is selected, the MTCNT counter synchronous presetting of multiple channels, and synchronous clearing by counter clearing on another channels, are possible. To set synchronous operation, the SYNC bits for at least two channels must be set to 1. To set synchronous clearing, in addition to the SYNC bit, the MTCNT counter clearing source must also be set by means of the CCLR bit in MTCR. 0: MT7CNT and MT6CNT operate independently (the MTCNT counter presetting/clearing is unrelated to other channels) 1: MT7CNT and MT6CNT perform synchronous operation (the MTCNT counter synchronous presetting/synchronous clearing is possible)	R/W
b6	SYNC6	Timer Synchronous Operation 6		R/W
b5 to b0	—	Reserved	These bits are always read as 0. The write value should always be 0.	R

MTSY is an 8-bit register that selects independent operation or synchronous operation for the MTCNT counters for channels 0 to 4, 6, and 7. A channel performs synchronous operation when the corresponding bit in MTSY is set to 1.

16.2.15 MT Counter Synchronization Start Register (MTCSYSTR)

Address H'FFFF 8282

	b7	b6	b5	b4	b3	b2	b1	b0
	SCH0	SCH1	SCH2	SCH3	SCH4	—	SCH6	SCH7
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	SCH0	Synchronous Start	Controls synchronous start of the MT0CNT counter. 0: Does not specify synchronous start for the MT0CNT counter 1: Specifies synchronous start for the MT0CNT counter [Condition of being set to 0] • When 1 is set to the CST0 bit of MT01234STR while SCH0 = 1	R/W*
b6	SCH1	Synchronous Start	Controls synchronous start of the MT1CNT counter. 0: Does not specify synchronous start for the MT1CNT counter 1: Specifies synchronous start for the MT1CNT counter [Condition of being set to 0] • When 1 is set to the CST1 bit of MT01234STR while SCH1 = 1	R/W*
b5	SCH2	Synchronous Start	Controls synchronous start of the MT2CNT counter. 0: Does not specify synchronous start for the MT2CNT counter 1: Specifies synchronous start for the MT2CNT counter [Condition of being set to 0] • When 1 is set to the CST2 bit of MT01234STR while SCH2 = 1	R/W*
b4	SCH3	Synchronous Start	Controls synchronous start of the MT3CNT counter. 0: Does not specify synchronous start for the MT3CNT counter 1: Specifies synchronous start for the MT3CNT counter [Condition of being set to 0] • When 1 is set to the CST3 bit of MT01234STR while SCH3 = 1	R/W*
b3	SCH4	Synchronous Start	Controls synchronous start of the MT4CNT counter. 0: Does not specify synchronous start for the MT4CNT counter 1: Specifies synchronous start for the MT4CNT counter [Condition of being set to 0] • When 1 is set to the CST4 bit of MT01234STR while SCH4 = 1	R/W*
b2	—	Reserved	This bit is always read as 0. The write value should always be 0.	R
b1	SCH6	Synchronous Start	Controls synchronous start of the MT6CNT counter. 0: Does not specify synchronous start for the MT6CNT counter 1: Specifies synchronous start for the MT6CNT counter [Condition of being set to 0] • When 1 is set to the CST6 bit of MT67STR while SCH6 = 1	R/W*
b0	SCH7	Synchronous Start	Controls synchronous start of the MT7CNT counter. 0: Does not specify synchronous start for the MT7CNT counter 1: Specifies synchronous start for the MT7CNT counter [Condition of being set to 0] • When 1 is set to the CST7 bit of MT67STR while SCH7 = 1	R/W*

Note: * Only 1 can be written to set the register. MTCSYSTR is automatically cleared after 1 is written to.

MTCSYSTR is an 8-bit register that specifies synchronous start of the MTU-III counters.

16.2.16 MT34 Read/Write Enable Register (MT34RWEN), MT67 Read/Write Enable Register (MT67RWEN)

Address MT34RWEN: H'FFFF 8284, MT67RWEN : H'FFFF 9284

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	RWE
After Reset	0	0	0	0	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b7 to b1	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b0	RWE	Read/Write Enable	Enables or disables access to the registers, which have write-protection capability against accidental modification. 0: Disables read/write access to the registers 1: Enables read/write access to the registers [Condition of being set to 0] • When 0 is written to the RWE bit after reading RWE = 1	R/W

The MTRWEN registers are 8-bit registers that enable or disable access to the registers and counters that have write-protection capability against accidental modification in channels 3, 4, 6, and 7.

- Registers and counters having write-protection capability against accidental modification (MT34RWEN)

22 registers: MT3CR, MT4CR, MT3MD0, MT4MD0, MT3IOCR0, MT4IOCR0, MT3IOCR1, MT4IOCR1, MT3IEN0, MT4IEN0, MT3GRA, MT4GRA, MT3GRB, MT4GRB, MT34OEN, MT34OCR0, MT34OCR1, MT34GCR, MT34CD, MT34DTD, MT3CNT, and MT4CNT

- Registers and counters having write-protection capability against accidental modification (MT67RWEN)

21 registers: MT6CR, MT7CR, MT6MD0, MT7MD0, MT6IOCR0, MT7IOCR0, MT6IOCR1, MT7IOCR1, MT6IEN0, MT7IEN0, MT6GRA, MT7GRA, MT6GRB, MT7GRB, MT67OEN, MT67OCR0, MT67OCR1, MT67CD, MT67DTD, MT6CNT, and MT7CNT

16.2.17 MT34 Output Enable Register (MT34OEN), MT67 Output Enable Register (MT67OEN)

Address MT34OEN: H'FFFF 820A

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	OE4D	OE4C	OE3D	OE4B	OE4A	OE3B
After Reset	1	1	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b6	—	Reserved	These bits are always read as 1. The write value should always be 1.	R
b5	OE4D	Master Enable TIOC4D	This bit enables/disables the TIOC4D pin MTU-III output. 0: MTU-III output is disabled (inactive level) 1: MTU-III output is enabled	R/W
b4	OE4C	Master Enable TIOC4C	This bit enables/disables the TIOC4C pin MTU-III output. 0: MTU-III output is disabled (inactive level) 1: MTU-III output is enabled	R/W
b3	OE3D	Master Enable TIOC3D	This bit enables/disables the TIOC3D pin MTU-III output. 0: MTU-III output is disabled (inactive level) 1: MTU-III output is enabled	R/W
b2	OE4B	Master Enable TIOC4B	This bit enables/disables the TIOC4B pin MTU-III output. 0: MTU-III output is disabled (inactive level) 1: MTU-III output is enabled	R/W
b1	OE4A	Master Enable TIOC4A	This bit enables/disables the TIOC4A pin MTU-III output. 0: MTU-III output is disabled (inactive level) 1: MTU-III output is enabled	R/W
b0	OE3B	Master Enable TIOC3B	This bit enables/disables the TIOC3B pin MTU-III output. 0: MTU-III output is disabled (inactive level) 1: MTU-III output is enabled	R/W

Note: The inactive level is determined by the settings in MT34OCR0 and MT67OCR0.
For details, refer to section section 16.2.18, MT34 Output Control Register 0 (MT34OCR0), MT67 Output Control Register 0 (MT67OCR0), and section section 16.2.19, MT34 Output Control Register 1 (MT34OCR1), MT67 Output Control Register 1 (MT67OCR1). Set these bits to 1 to enable MTU-III output in other than complementary PWM or reset-synchronized PWM mode. When these bits are set to 0, low level is output.

Address MT67OEN: H'FFFF 920A

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	OE7D	OE7C	OE6D	OE7B	OE7A	OE6B
After Reset	1	1	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b6	—	Reserved	These bits are always read as 1. The write value should always be 1.	R
b5	OE7D	Master Enable TIOC7D	This bit enables/disables the TIOC7D pin MTU-III output. 0: MTU-III output is disabled (inactive level) 1: MTU-III output is enabled	R/W
b4	OE7C	Master Enable TIOC7C	This bit enables/disables the TIOC7C pin MTU-III output. 0: MTU-III output is disabled (inactive level) 1: MTU-III output is enabled	R/W
b3	OE6D	Master Enable TIOC6D	This bit enables/disables the TIOC6D pin MTU-III output. 0: MTU-III output is disabled (inactive level) 1: MTU-III output is enabled	R/W
b2	OE7B	Master Enable TIOC7B	This bit enables/disables the TIOC7B pin MTU-III output. 0: MTU-III output is disabled (inactive level) 1: MTU-III output is enabled	R/W
b1	OE7A	Master Enable TIOC7A	This bit enables/disables the TIOC7A pin MTU-III output. 0: MTU-III output is disabled (inactive level) 1: MTU-III output is enabled	R/W
b0	OE6B	Master Enable TIOC6B	This bit enables/disables the TIOC6B pin MTU-III output. 0: MTU-III output is disabled (inactive level) 1: MTU-III output is enabled	R/W

Note: The inactive level is determined by the settings in MT34OCR1 and MT67OCR1.
For details, refer to section section 16.2.18, MT34 Output Control Register 0 (MT34OCR0), MT67 Output Control Register 0 (MT67OCR0), and section section 16.2.19, MT34 Output Control Register 1 (MT34OCR1), MT67 Output Control Register 1 (MT67OCR1). Set these bits to 1 to enable MTU-III output in other than complementary PWM or reset-synchronized PWM mode. When these bits are set to 0, low level is output.

The MTOEN registers are 8-bit registers that enable or disable output settings for output pins TIOC4D, TIOC4C, TIOC3D, TIOC4B, TIOC4A, TIOC3B, TIOC7D, TIOC7C, TIOC6D, TIOC7B, TIOC7A, and TIOC6B. These pins do not output correctly if the respective bits of MT34OEN and MT67OEN have not been set.

Set MT34OEN and MT67OEN of channels 3, 4, 6, and 7 prior to setting MTIOCR of channels 3, 4, 6, and 7.

Set MT34OEN when TCNT count operation of channels 3 and 4 is stopped.

Set MT67OEN when TCNT count operation of channels 6 and 7 is stoopped. (See figures 16.36 and 16.39)

16.2.18 MT34 Output Control Register 0 (MT34OCR0), MT67 Output Control Register 0 (MT67OCR0)

Address MT34OCR0: H'FFFF 820E, MT67OCR0: H'FFFF 920E

	b7	b6	b5	b4	b3	b2	b1	b0
	—	PSYE	—	—	TOCL	TOCS	OLSN	OLSP
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	—	Reserved	This bit is always read as 0 and the write value should always be 0.	R
b6	PSYE	PWM Synchronous Output Enable	This bit selects the enable/disable of toggle output synchronized with the PWM period. 0: Toggle output is disabled 1: Toggle output is enabled	R/W
b5 to b4	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b3	TOCL	TOC Register Write Protection*2	This bit selects the enable/disable of write access to the TOCS, OLSN, and OLSP bits in MT34OCR0 and MT67OCR0. 0: Write access to the TOCS, OLSN, and OLSP bits is enabled 1: Write access to the TOCS, OLSN, and OLSP bits is disabled	R/W*1
b2	TOCS	TOC Select	This bit selects either the MT34OCR0 (MT67OCR0) or MT34OCR1 (MT67OCR1) setting to be used for the output level in complementary PWM mode and reset-synchronized PWM mode. 0: MT34OCR0 (MT67OCR0) setting is selected 1: MT34OCR1 (MT67OCR1) setting is selected	R/W
b1	OLSN	Output Level Select N*3*4	This bit selects the negative phase output level in reset-synchronized PWM mode/complementary PWM mode. See table 16.41.	R/W
b0	OLSP	Output Level Select P*3	This bit selects the positive phase output level in reset-synchronized PWM mode/complementary PWM mode. See table 16.42.	R/W

Notes: 1. This bit can be set to 1 only once after reset. After 1 is written, 0 cannot be written to the bit.
2. Setting the TOCL bit to 1 prevents accidental modification when the CPU goes out of control.
3. Clearing the TOCS bit to 0 makes this bit setting valid.
4. When a dead time is not generated, the negative phase output level is the inverse of the positive phase. The OLSP and OLSN bits must be set to the same value.

The MTOCR0 registers are 8-bit registers that enable or disable PWM synchronized toggle output in complementary PWM mode/reset synchronized PWM mode, and controls output level inversion of PWM output.

Table 16.41 Output Level Select Function (1)

Bit 1	Function			
OLSN	Initial Output	Active Level	Compare Match Output	
			Up Count	Down Count
0	High level	Low level	High level	Low level
1	Low level	High level	Low level	High level

Note: • The negative phase waveform initial output value changes to active level after elapse of the dead time after count start.

Table 16.42 Output Level Select Function (2)

Bit 0	Function			
OLSP	Initial Output	Active Level	Compare Match Output	
			Up Count	Down Count
0	High level	Low level	High level	Low level
1	Low level	High level	Low level	High level

Figure 16.3 shows an example of complementary PWM mode output (1 phase) when OLSN = 1, OLSP = 1.

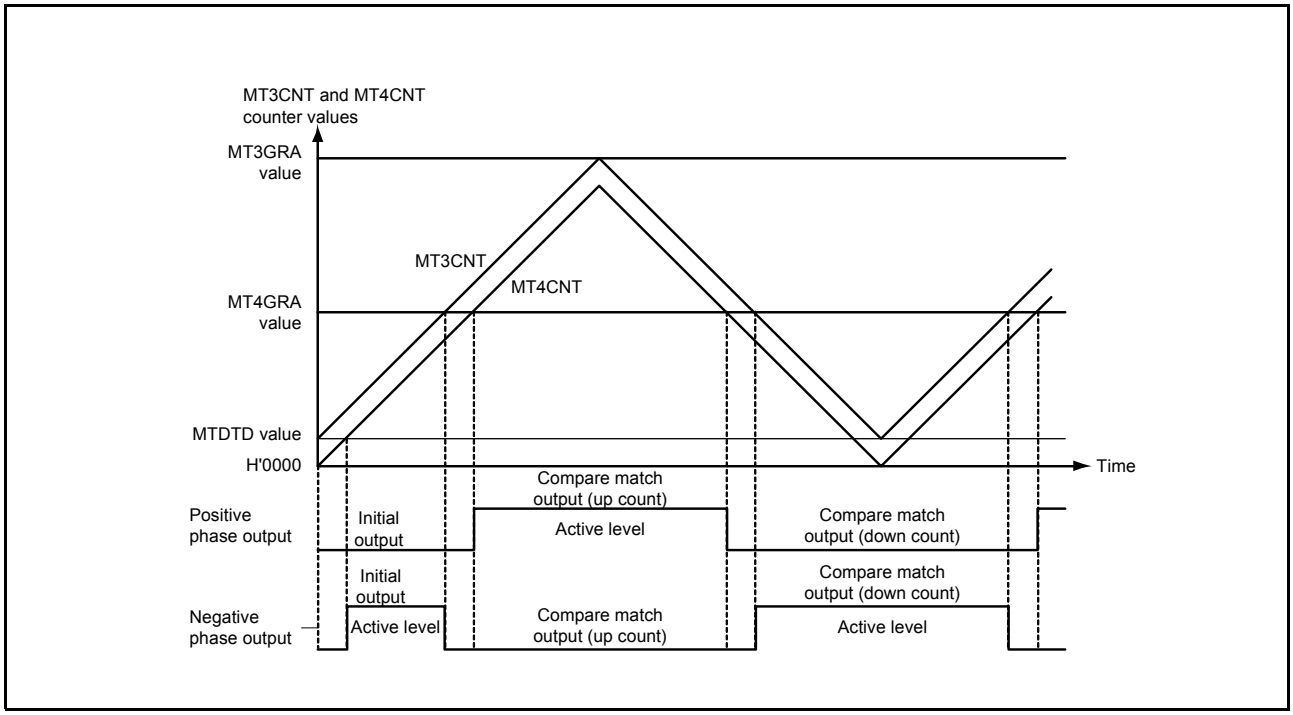


Figure 16.3 Complementary PWM Mode Output Level Example

16.2.19 MT34 Output Control Register 1 (MT34OCR1), MT67 Output Control Register 1 (MT67OCR1)

Address MT34OCR1: H'FFFF 820F, MT67OCR1: H'FFFF 920F

	b7	b6	b5	b4	b3	b2	b1	b0
	BF [1:0]		OLS3N	OLS3P	OLS2N	OLS2P	OLS1N	OLS1P
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b6	BF [1:0]	MTOLB Buffer Transfer Timing Select	Specifies the timing for transferring buffer from MTOLB to MTOCR1. See table 16.43.	R/W
b5	OLS3N	Output Level Select 3N*	This bit selects the output level on pins TIOC4D and TIOC7D in reset-synchronized PWM mode/complementary PWM mode. See table 16.44.	R/W
b4	OLS3P	Output Level Select 3P*	This bit selects the output level on pins TIOC4B and TIOC7B in reset-synchronized PWM mode/complementary PWM mode. See table 16.45.	R/W
b3	OLS2N	Output Level Select 2N*	This bit selects the output level on pins TIOC4C and TIOC7C in reset-synchronized PWM mode/complementary PWM mode. See table 16.46.	R/W
b2	OLS2P	Output Level Select 2P*	This bit selects the output level on pins TIOC4A and TIOC7A in reset-synchronized PWM mode/complementary PWM mode. See table 16.47.	R/W
b1	OLS1N	Output Level Select 1N*	This bit selects the output level on pins TIOC3D and TIOC6D in reset-synchronized PWM mode/complementary PWM mode. See table 16.48.	R/W
b0	OLS1P	Output Level Select 1P*	This bit selects the output level on pins TIOC3B and TIOC6B in reset-synchronized PWM mode/complementary PWM mode. See table 16.49.	R/W

Note: * When a dead time is not generated, the negative phase output level is the inverse of the positive phase. The OLSiP and OLSiN bits must be set to the same value. (i = 1, 2, 3)
Setting the TOCS bit in MT34OCR0 or MT67OCR0 to 1 makes this bit setting valid.

The MTOCR1 registers are 8-bit registers that control output level inversion of PWM output in complementary PWM mode and reset-synchronized PWM mode.

Table 16.43 BF Bits Setting

Bit 7	Bit 6	Description	
BF1	BF0	Complementary PWM Mode	Reset-Synchronized PWM Mode
0	0	Does not transfer data from the buffer register (MTOLB) to MTOCR1.	Does not transfer data from the buffer register (MTOLB) to MTOCR1.
0	1	Transfers data from the buffer register (MTOLB) to MTOCR1 at the crest of the MT4CNT (MT7CNT) counter.	Transfers data from the buffer register (MTOLB) to MTOCR1 when counters MT3CNT (MT6CNT) and MT4CNT (MT7CNT) are cleared.
1	0	Transfers data from the buffer register (MTOLB) to MTOCR1 at the trough of the MT4CNT (MT7CNT) counter.	Setting prohibited
1	1	Transfers data from the buffer register (MTOLB) to MTOCR1 at the crest and trough of the MT4CNT (MT7CNT) counter.	Setting prohibited

Table 16.44 Pins TIOC4D and TIOC7D Output Level Select Function

Bit 5	Function			
OLS3N	Initial Output	Active Level	Compare Match Output	
			Up Count	Down Count
0	High level	Low level	High level	Low level
1	Low level	High level	Low level	High level

Note: The negative phase waveform initial output value changes to active level after elapse of the dead time after count start.

Table 16.45 Pins TIOC4B and TIOC7B Output Level Select Function

Bit 4	Function			
OLS3P	Initial Output	Active Level	Compare Match Output	
			Up Count	Down Count
0	High level	Low level	Low level	High level
1	Low level	High level	High level	Low level

Table 16.46 Pins TIOC4C and TIOC7C Output Level Select Function

Bit 3	Function			
OLS2N	Initial Output	Active Level	Compare Match Output	
			Up Count	Down Count
0	High level	Low level	High level	Low level
1	Low level	High level	Low level	High level

Note: The negative phase waveform initial output value changes to active level after elapse of the dead time after count start.

Table 16.47 Pins TIOC4A and TIOC7A Output Level Select Function

Bit 2	Function			
OLS2P	Initial Output	Active Level	Compare Match Output	
			Up Count	Down Count
0	High level	Low level	Low level	High level
1	Low level	High level	High level	Low level

Table 16.48 Pins TIOC3D and TIOC6D Output Level Select Function

Bit 1	Function			
OLS1N	Initial Output	Active Level	Compare Match Output	
			Up Count	Down Count
0	High level	Low level	High level	Low level
1	Low level	High level	Low level	High level

Note: The negative phase waveform initial output value changes to active level after elapse of the dead time after count start.

Table 16.49 Pins TIOC3B and TIOC6B Output Level Select Function

Bit 0	Function			
OLS1P	Initial Output	Active Level	Compare Match Output	
			Up Count	Down Count
0	High level	Low level	Low level	High level
1	Low level	High level	High level	Low level

16.2.20 MT34 Output Level Buffer Register (MT34OLB), MT67 Output Level Buffer Register (MT67OLB)

Address MT34OLB: H'FFFF 8236, MT67OLB: H'FFFF 9236

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	OLS3N	OLS3P	OLS2N	OLS2P	OLS1N	OLS1P
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7, b6	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b5	OLS3N	Output Level Select 3N	Specifies the buffer value to be transferred to the OLS3N bit in MTOCR1.	R/W
b4	OLS3P	Output Level Select 3P	Specifies the buffer value to be transferred to the OLS3P bit in MTOCR1.	R/W
b3	OLS2N	Output Level Select 2N	Specifies the buffer value to be transferred to the OLS2N bit in MTOCR1.	R/W
b2	OLS2P	Output Level Select 2P	Specifies the buffer value to be transferred to the OLS2P bit in MTOCR1.	R/W
b1	OLS1N	Output Level Select 1N	Specifies the buffer value to be transferred to the OLS1N bit in MTOCR1.	R/W
b0	OLS1P	Output Level Select 1P	Specifies the buffer value to be transferred to the OLS1P bit in MTOCR1.	R/W

The MTOLB registers are 8-bit registers that function as buffers for MTOCR1 and specify the PWM output level in complementary PWM mode and reset-synchronized PWM mode.
Figure 16.4 shows an example of the PWM output level setting procedure in buffer operation.

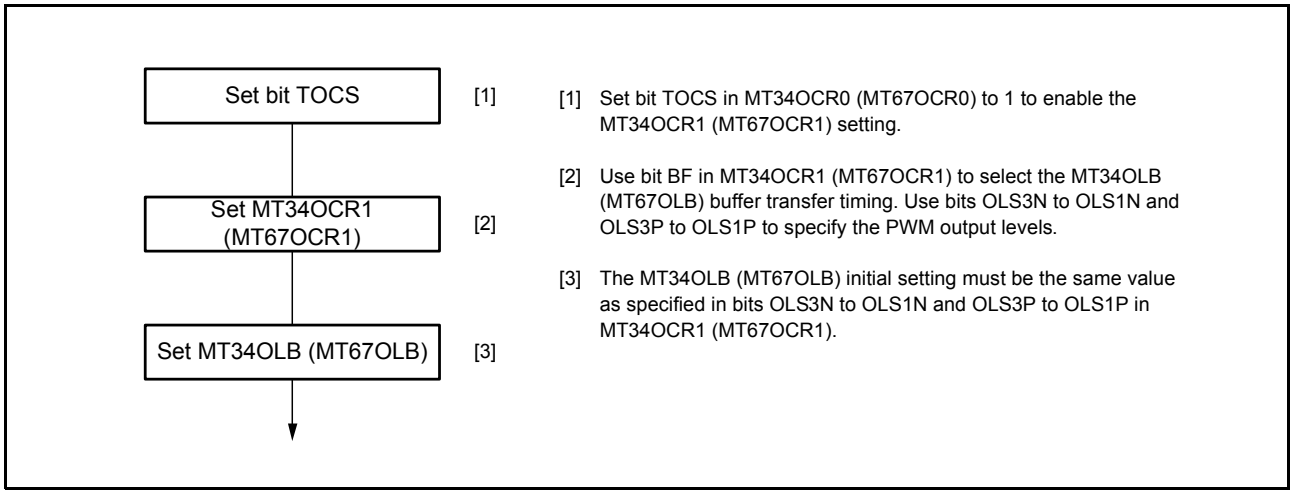


Figure 16.4 PWM Output Level Setting Procedure in Buffer Operation

16.2.21 MT34 Gate Control Register (MT34GCR)

Address H'FFFF 820D

	b7	b6	b5	b4	b3	b2	b1	b0
	—	BDC	N	P	FB	WF	VF	UF
After Reset	1	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	—	Reserved	This bit is always read as 1 and the write value should always be 1.	R
b6	BDC	Brushless DC Motor	This bit selects whether to make the functions of this register effective or ineffective. 0: Ordinary output 1: Functions of this register are made effective.	R/W
b5	N	Negative Phase Output (N) Control	This bit selects whether the level output or the reset-synchronized PWM/complementary PWM output while the negative pins (TIOC3D, TIOC4C, and TIOC4D) are output. 0: Level output 1: Reset synchronized PWM/complementary PWM output	R/W
b4	P	Positive Phase Output (P) Control	This bit selects whether the level output or the reset-synchronized PWM/complementary PWM output while the positive pins (TIOC3B, TIOC4A, and TIOC4B) are output. 0: Level output 1: Reset synchronized PWM/complementary PWM output	R/W
b3	FB	External Feedback Signal Enable	This bit selects whether the switching of the positive/ negative phase output is carried out automatically with channel 0 MTGRA, MTGRB, and MTGRC input capture signals, or by bits UF, VF, and WF of MT34GCR. 0: Input capture signal 1: Bits UF, VF, and WF	R/W
b2	WF	Output Phase Switch	These bits set the positive phase/negative phase output phase on or off state. The setting of these bits is valid only when the FB bit in this register is set to 1. In this case, the setting of bits 2 to 0 becomes a substitute for external inputs. See table 16.50.	R/W
b1	VF			R/W
b0	UF			R/W

MT34GCR is an 8-bit register that controls the waveform output necessary for brushless DC motor control in reset-synchronized PWM mode/complementary PWM mode. This register setting is ineffective for anything other than complementary PWM mode/reset-synchronized PWM mode.

Table 16.50 Output Level Select Function

Bit 2	Bit 1	Bit 0	Function					
WF	VF	UF	TIOC3B	TIOC4A	TIOC4B	TIOC3D	TIOC4C	TIOC4D
			U Phase	V Phase	W Phase	U Phase	V Phase	W Phase
0	0	0	OFF	OFF	OFF	OFF	OFF	OFF
0	0	1	ON	OFF	OFF	OFF	OFF	ON
0	1	0	OFF	ON	OFF	ON	OFF	OFF
0	1	1	OFF	ON	OFF	OFF	OFF	ON
1	0	0	OFF	OFF	ON	OFF	ON	OFF
1	0	1	ON	OFF	OFF	OFF	ON	OFF
1	1	0	OFF	OFF	ON	ON	OFF	OFF
1	1	1	OFF	OFF	OFF	OFF	OFF	OFF

16.2.22 MT34 Subcounter (MT34SCNT), MT67 Subcounter (MT67SCNT)

Address MT34SCNT: H'FFFF 8220, MT67SCNT: H'FFFF 9220

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Note: * The MTSCNT counter must not be accessed in eight bits; they should always be accessed in 16 bits.

The MTSCNT counters are 16-bit read-only counters used only in complementary PWM mode.

16.2.23 MT34 Cycle Data Register (MT34CD), MT67 Cycle Data Register (MT67CD)

Address MT34CD: H'FFFF 8214, MT67CD: H'FFFF 9214

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Note: * MTCD must not be accessed in eight bits; they should always be accessed in 16 bits.

The MTCD registers are 16-bit registers used only in complementary PWM mode. Set half the PWM carrier cycle value as the MTCD register value.

16.2.24 MT34 Cycle Buffer Register (MT34CB), MT67 Cycle Buffer Register (MT67CB)

Address MT34CB: H'FFFF 8222, MT67CB: H'FFFF 9222

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Note: * MTCB must not be accessed in eight bits; they should always be accessed in 16 bits.

The MTCB registers are 16-bit registers used only in complementary PWM mode. They function as buffer registers for the MTCD registers. The MTCB register values are transferred to the MTCD registers with the transfer timing set in the MTMD0 register.

16.2.25 MT34 Dead Time Data Register (MT34DTD), MT67 Dead Time Data Register (MT67DTD)

Address MT34DTD: H'FFFF 8216, MT67DTD: H'FFFF 9216

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Note: * MTDTD must not be accessed in eight bits; they should always be accessed in 16 bits.

The MTDTD registers are 16-bit registers used only in complementary PWM mode, which specifies the MT3CNT (MT6CNT) and MT4CNT (MT7CNT) counter offset values. In complementary PWM mode, when the MT3CNT (MT6CNT) and MT4CNT (MT7CNT) counters are cleared and then restarted, the MTDTD register value is loaded into the MT3CNT (MT6CNT) counter and the count operation starts.

16.2.26 MT3 Dead Time Enable Register (MT3DTEN), MT6 Dead Time Enable Register (MT6DTEN)

Address MT3DTEN: H'FFFF 8234, MT6DTEN: H'FFFF 9234

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	TDER
After Reset	0	0	0	0	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b7 to b1	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b0	TDER	Timer Dead Time Enable	Specifies whether to generate a dead time. 0: Does not generate a dead time 1: Generates a dead time* [Condition of being set to 0] • When 0 is written to the TDER flag after reading TDER = 1	R/W

Note: * MT34DTD and MT67DTD must be set to 1 or a larger value.

The MTU-III has one MTDTEN each for channel 3 (MT3DTEN) and channel 6 (MT6DTEN) to control dead time generation for complementary PWM mode. MTDTEN register settings should be conducted only when MTCNT counter operation is stopped.

16.2.27 MT34 Buffer Transfer Setting Register (MT34BTSE), MT67 Buffer Transfer Setting Register (MT67BTSE)

Address MT34BTSE: H'FFFF 8232, MT67BTSE: H'FFFF 9232

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	BTE [1:0]	
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b2	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b1, b0	BTE [1:0]		These bits enable or disable transfer from the buffer registers* used in complementary PWM mode to the temporary registers, and specify whether to link the transfer with interrupt skipping 1 operation. See table 16.51.	R/W

Note: * Applicable buffer registers (MT34BTSE):
MT3GRC, MT3GRD, MT4GRC, MT4GRD, and MT34CB
Applicable buffer registers (MT67BTSE):
MT6GRC, MT6GRD, MT7GRC, MT7GRD, and MT67CB

The MTBTSE registers are 8-bit registers that enable or disable transfer from the buffer registers* used in complementary PWM mode to the temporary registers, and specify whether to link the transfer with interrupt skipping 1 operation.

Table 16.51 BTE Bits Setting

Bit 1	Bit 0	Description
BTE1	BTE0	
0	0	Enables transfer from the buffer registers to the temporary registers*1 and does not link the transfer with interrupt skipping 1 operation.
0	1	Disables transfer from the buffer registers to the temporary registers.
1	0	Links transfer from the buffer registers to the temporary registers with interrupt skipping 1 operation*2
1	1	Setting prohibited

Notes: 1. Data is transferred according to the MD bits setting in MTMD0. For details, refer to section section 16.3.8, Complementary PWM Mode.
2. When interrupt skipping is disabled (the T3AEN and T4VEN (T6AEN and T7VEN) bits are cleared to 0 in the MT interrupt skipping set register 0 (MT34ISSE0 (MT67ISSE0), or the skipping count set bits (T3ACOR and T4VCOR (T6ACOR and T7VCOR)) in (MT34ISSE0 (MT67ISSE0) are cleared to 0), be sure to disable link of buffer transfer with interrupt skipping. (Clear the BTE1 bit in MTBTSE to 0).
If link with interrupt skipping is enabled while interrupt skipping is disabled, buffer transfer will not be performed.

16.2.28 MT34 Waveform Control Register (MT34WCR), MT67 Waveform Control Register (MT67WCR)

Address MT34WCR: H'FFFF 8260, MT67WCR: H'FFFF 9260

	b7	b6	b5	b4	b3	b2	b1	b0
	CCE	—	—	—	—	—	SCC	WRE
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Function	R/W
b7	CCE	Compare Match Clear Enable*	Specifies whether to clear counters at MT3GRA (MT6GRA) compare match in complementary PWM mode. 0: Does not clear counters 1: Clears counters [Condition of being set to 1] • When 1 is written to CCE after reading CCE = 0	R/W
b6 to b2	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b1	SCC	Synchronous Clearing Control (only for MT67WCR)	Specifies whether to clear MT6CNT and MT7CNT when the synchronous counter clearing for channels 0 to 2 and channels 6 and 7 occurs in complementary PWM mode. When using this control, place channels 6 and 7 in complementary PWM mode. When modifying the SCC bit in the counter operation, do not modify the CCE or WRE bit. The SCC bit setting disables the counter clearing synchronized with the MTU3 only when the synchronous clearing occurs outside the Tb interval at the trough. When the synchronous clearing occurs in the Tb interval at the trough including the time period immediately after MT6CNT and MT7CNT start operations, MT6CNT and MT7CNT are cleared. As for the Tb interval at the trough in complementary PWM mode, see Figure 16.41. 0: Enables the clearing of MT6CNT and MT7CNT 1: Disables the clearing of MT6CNT and MT7CNT [Condition of being set to 1] • When 1 is written to SCC after reading SCC = 0, this bit is reserved in MT34WCR and read as 0. The write value should be 0.	R/W
b0	WRE	Initial Output Suppression Enable	Selects the waveform output when a synchronous counter clearing occurs in complementary PWM mode. The output waveform is retained only when the synchronous clearing occurs within the Tb interval at the trough in complementary PWM mode. When the synchronous clearing occurs outside this interval, the initial value specified in registers MT34OCR0 and MT34OCR1 (registers MT67OCR0 and MT67OCR1) is output regardless of the WRE bit setting. Also the specified initial value is output when the synchronous clearing occurs in the Tb interval at the trough immediately after MT3CNT and MT4CNT (MT6CNT and MT7CNT) start operations. As for the Tb interval at the trough in complementary PWM mode, see Figure 16.41. 0: Outputs the initial value specified in registers MT34OCR0 and MT34OCR1 (registers MT67OCR0 and MT67OCR1) 1: Retains the waveform output immediately before the synchronous clearing [Condition of being set to 1] • When 1 is written to WRE after reading WRE = 0	R/W

Note: * Do not set to 1 without selecting complementary PWM mode 1.

The MTWCR registers are 8-bit registers that control the waveform when synchronous counter clearing occurs in MT3CNT and MT4CNT (MT6CNT and MT7CNT) in complementary PWM mode. The MTWCR registers also specify whether to clear the counters at MT3GRA (MT6GRA) compare match. Bits CCE and WRE in MTWCR must be modified only while the MTCNT counter stops.

16.2.29 MT4 A/D Converter Start Request Control Register (MT4ADSRCR), MT7 A/D Converter Start Request Control Register (MT7ADSRCR)

Address MT4ADSRCR: H'FFFF 8240

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	BF [1:0]		—	—	—	—	—	—	UT4AE	DT4AE	UT4BE	DT4BE	ITA3AE	ITA4VE	ITB3AE	ITB4VE
After Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Function	R/W
b15 to14	BF [1:0]	MT4ADSRCSEBA/ MT4ADSRCSEBB Transfer Timing Select	Select the timing for transferring from MT4ADSRCSEBA and MT4ADSRCSEBB to MT4ADSRCSEA and MT4ADSRCSEB. See table 16.52.	R/W
b13 to b8	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b7	UT4AE	Up-Count TRG4AN Enable	Enables or disables A/D converter start requests (TRG4AN) during MT4CNT counter up-count operation. 0: A/D converter start request (TRG4AN) disabled 1: A/D converter start request (TRG4AN) enabled	R/W
b6	DT4AE	Down-Count TRG4AN Enable*	Enables or disables A/D converter start requests (TRG4AN) during MT4CNT counter down-count operation. 0: A/D converter start request (TRG4AN) disabled 1: A/D converter start request (TRG4AN) enabled	R/W
b5	UT4BE	Up-Count TRG4BN Enable	Enables or disables A/D converter start requests (TRG4BN) during MT4CNT counter up-count operation. 0: A/D converter start request (TRG4BN) disabled 1: A/D converter start request (TRG4BN) enabled	R/W
b4	DT4BE	Down-Count TRG4BN Enable*	Enables or disables A/D converter start requests (TRG4BN) during MT4CNT counter down-count operation. 0: A/D converter start request (TRG4BN) disabled 1: A/D converter start request (TRG4BN) enabled	R/W
b3	ITA3AE	TGIA_3 Interrupt Skipping Link Enable*	Select whether to link A/D converter start requests (TRG4AN) with TGIA_3 interrupt skipping 1 operation. 0: Does not link with TGIA_3 interrupt skipping 1 1: Links with TGIA_3 interrupt skipping 1	R/W
b2	ITA4VE	TCIV_4 Interrupt Skipping Link Enable*	Select whether to link A/D converter start requests (TRG4AN) with TCIV_4 interrupt skipping 1 operation. 0: Does not link with TCIV_4 interrupt skipping 1 1: Links with TCIV_4 interrupt skipping 1	R/W
b1	ITB3AE	TGIA_3 Interrupt Skipping Link Enable*	Select whether to link A/D converter start requests (TRG4BN) with TGIA_3 interrupt skipping 1 operation. 0: Does not link with TGIA_3 interrupt skipping 1 1: Links with TGIA_3 interrupt skipping 1	R/W
b0	ITB4VE	TCIV_4 Interrupt Skipping Link Enable*	Select whether to link A/D converter start requests (TRG4BN) with TCIV_4 interrupt skipping 1 operation. 0: Does not link with TCIV_4 interrupt skipping 1 1: Links with TCIV_4 interrupt skipping 1	R/W

Notes: * Do not set to 1 without selecting complementary PWM mode.

1. MT4ADSRCR must not be accessed in eight bits; it should always be accessed in 16 bits.
2. When interrupt skipping is disabled (the T3AEN and T4VEN bits in MT34 interrupt skipping set register 0 (MT34ISSE0) are cleared to 0, or the skipping count set bits (T3ACOR and T4VCOR) in MT34ISSE0 are cleared to 0), do not link A/D converter start requests with interrupt skipping 1 operation (clear the ITA3AE, ITA4VE, ITB3AE, and ITB4VE bits in MT4ADSRCR to 0). If interrupt skipping is enabled, A/D converter start requests will not be issued.

Table 16.52 Setting of Transfer Timing by Bits BF

Bit 15	Bit 14	Description
BF1	BF0	
0	0	Does not transfer data from the cycle set buffer registers (MT4ADSRCSEBA and MT4ADSRCSEBB) to the cycle set registers (MT4ADSRCSEA and MT4ADSRCSEB).
0	1	Transfers data from the cycle set buffer registers (MT4ADSRCSEBA and MT4ADSRCSEBB) to the cycle set registers (MT4ADSRCSEA and MT4ADSRCSEB) at the crest of the MT4CNT counter.*1
1	0	Transfers data from the cycle set buffer registers (MT4ADSRCSEBA and MT4ADSRCSEBB) to the cycle set registers (MT4ADSRCSEA and MT4ADSRCSEB) at the trough of the MT4CNT counter.*2
1	1	Transfers data from the cycle set buffer registers (MT4ADSRCSEBA and MT4ADSRCSEBB) to the cycle set registers (MT4ADSRCSEA and MT4ADSRCSEB) at the crest and trough of the MT4CNT counter.*2

Notes: 1. When MT4CNT counter reaches the crest or MT4GRD is written in complementary PWM mode, or when compare match occurs between the MT3CNT counter and MT3GRA in reset-synchronized PWM mode, or when compare match occurs between the MT4CNT counter and MT4GRA in PWM mode 1 or normal operation mode, data is transferred from the cycle set buffer registers (MT4ADSRCSEBA and MT4ADSRCSEBB) to the cycle set registers (MT4ADSRCSEA and MT4ADSRCSEB).
2. These settings are prohibited without selecting complementary PWM mode.

Address MT7ADSRCR: H'FFFF 9240

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	BF [1:0]		—	—	—	—	—	—	UT7AE	DT7AE	UT7BE	DT7BE	ITA6AE	ITA7VE	ITB6AE	ITB7VE
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 b14	BF [1:0]	MT7ADSRCSEBA/ MT7ADSRCSEBB Transfer Timing Select	Select the timing for transferring from MT7ADSRCSEBA and MT7ADSRCSEBB to MT7ADSRCSEA and MT7ADSRCSEB. See table 16.53.	R/W
b13 to b8	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b7	UT7AE	Up-Count TRG7AN Enable	Enables or disables A/D converter start requests (TRG7AN) during MT7CNT counter up-count operation. 0: A/D converter start request (TRG7AN) disabled 1: A/D converter start request (TRG7AN) enabled	R/W
b6	DT7AE	Down-Count TRG7AN Enable*	Enables or disables A/D converter start requests (TRG7AN) during MT7CNT counter down-count operation. 0: A/D converter start request (TRG7AN) disabled 1: A/D converter start request (TRG7AN) enabled	R/W
b5	UT7BE	Up-Count TRG7BN Enable	Enables or disables A/D converter start requests (TRG7BN) during MT7CNT counter up-count operation. 0: A/D converter start request (TRG7BN) disabled 1: A/D converter start request (TRG7BN) enabled	R/W
b4	DT7BE	Down-Count TRG7BN Enable*	Enables or disables A/D converter start requests (TRG7BN) during MT7CNT counter down-count operation. 0: A/D converter start request (TRG7BN) disabled 1: A/D converter start request (TRG7BN) enabled	R/W
b3	ITA6AE	TGIA_6 Interrupt Skipping Link Enable*	Select whether to link A/D converter start requests (TRG7AN) with TGIA_6 interrupt skipping 1 operation. 0: Does not link with TGIA_6 interrupt skipping 1 1: Links with TGIA_6 interrupt skipping 1	R/W
b2	ITA7VE	TCIV_7 Interrupt Skipping Link Enable*	Select whether to link A/D converter start requests (TRG7AN) with TCIV_7 interrupt skipping 1 operation. 0: Does not link with TCIV_7 interrupt skipping 1 1: Links with TCIV_7 interrupt skipping 1	R/W

Bit	Symbol	Bit Name	Description	R/W
b1	ITB6AE	TGIA_6 Interrupt Skipping Link Enable*	Select whether to link A/D converter start requests (TRG7BN) with TGIA_6 interrupt skipping 1 operation. 0: Does not link with TGIA_6 interrupt skipping 1 1: Links with TGIA_6 interrupt skipping 1	R/W
b0	ITB7VE	TCIV_7 Interrupt Skipping Link Enable*	Select whether to link A/D converter start requests (TRG7BN) with TCIV_7 interrupt skipping 1 operation. 0: Does not link with TCIV_7 interrupt skipping 1 1: Links with TCIV_7 interrupt skipping 1	R/W

Notes: * Do not set to 1 without selecting complementary PWM mode.
1. MT7ADSRCR must not be accessed in eight bits; it should always be accessed in 16 bits.
2. When interrupt skipping is disabled (the T6AEN and T7VEN bits in MT67 interrupt skipping set register 0 (MT67ISSE0) are cleared to 0, or the skipping count set bits (T6ACOR and T7VCOR) in MT67ISSE0 are cleared to 0), do not link A/D converter start requests with interrupt skipping 1 operation (clear the ITA6AE, ITA7VE, ITB6AE, and ITB7VE bits in MT7ADSRCR to 0). If interrupt skipping is enabled, A/D converter start requests will not be issued.

Table 16.53 Setting of Transfer Timing by Bits BF

Bit 15	Bit 14	Description
BF1	BF0	
0	0	Does not transfer data from the cycle set buffer registers (MT7ADSRCSEBA and MT7ADSRCSEBB) to the cycle set registers (MT7ADSRCSEA and MT7ADSRCSEB)
0	1	Transfers data from the cycle set buffer registers (MT7ADSRCSEBA and MT7ADSRCSEBB) to the cycle set registers (MT7ADSRCSEA and MT7ADSRCSEB) at the crest of the MT7CNT counter.*1
1	0	Transfers data from the cycle set buffer registers (MT7ADSRCSEBA and MT7ADSRCSEBB) to the cycle set registers (MT7ADSRCSEA and MT7ADSRCSEB) at the trough of the MT7CNT counter.*2
1	1	Transfers data from the cycle set buffer registers (MT7ADSRCSEBA and MT7ADSRCSEBB) to the cycle set registers (MT7ADSRCSEA and MT7ADSRCSEB) at the crest and trough of the MT7CNT counter.*2

Notes: 1. When MT7CNT counter reaches the crest or MT7GRD is written in complementary PWM mode, or when compare match occurs between the MT6CNT counter and MT6GRA in reset-synchronized PWM mode, or when compare match occurs between the MT7CNT counter and MT7GRA in PWM mode 1 or normal operation mode, data is transferred from the cycle set buffer registers (MT7ADSRCSEBA and MT7ADSRCSEBB) to the cycle set registers (MT7ADSRCSEA and MT7ADSRCSEB).
2. These settings are prohibited without selecting complementary PWM mode.

The MTADSRCR registers are 8-bit registers that enable or disable A/D converter start requests and specify whether to link A/D converter start requests with interrupt skipping operation.

16.2.30 MT4 A/D Converter Start Request Cycle Setting Register A (MT4ADSRCSEA), MT7 A/D Converter Start Request Cycle Setting Register A (MT7ADSRCSEA), MT4 A/D Converter Start Request Cycle Setting Register B (MT4ADSRCSEB), MT7 A/D Converter Start Request Cycle Setting Register B (MT7ADSRCSEB)

Address: MT4ADSRCSEA: H'FFFF 8244, MT4ADSRCSEB: H'FFFF 8246,
MT7ADSRCSEA: H'FFFF 9244, MT7ADSRCSEB: H'FFFF 9246

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Note: • MTADSRCSE must not be accessed in eight bits; it should always be accessed in 16 bits.
When the A/D converter start request delaying function linked with skipping function (for details, see section 16.3.9 (4) A/D Converter Start Request Delaying Function Linked with Interrupt Skipping Function 1) is used, the value of this register should be within the following range:
- H'0002 to the value of MT34CD - 2 (for channel 4)
- H'0002 to the value of MT67CD - 2 (for channel 7)

When interrupt skipping function 2 is in use and the interval between updating of the values in MT4/7ADSRCSEA and MT4/7ADSRCSEB is short, correct counting of the interval to be skipped becomes impossible, and requests for A/D conversion may not be generated with the expected timing.
Use this function under the following conditions.

(1) Skipping function 2 is used and the skipping count is zero.

- The interval between updating of the MT4/7ADSRCSEA and MT4/7ADSRCSEB registers is at least four.
- The interval for comparison with MT4/7ADSRCSEA is at least 4PMϕ. The updated value of MT4/7ADSRCSEA is the previous value plus at least four or the previous value minus at least four.
- The interval for comparison with MT4/7ADSRCSEB is at least 4PMϕ. The updated value of MT4/7ADSRCSEB is the previous value plus at least four or the previous value minus at least four.

(2) Skipping function 2 is used and the skipping count is one or more.

- The interval between updating of the MT4/7ADSRCSEA and MT4/7ADSRCSEB registers is at least two.
- The interval for comparison with MT4/7ADSRCSEB is at least 2PMϕ. The updated value of MT4/7ADSRCSEB is the previous value plus at least two or the previous value minus at least two.

The MTADSRCSE registers (MT4ADSRCSEA and MT4ADSRCSEB in channel 4, and MT7ADSRCSEA and MT7ADSRCSEB in channel 7) are 16-bit readable/writable registers. When the MT4CNT and MT7CNT counter values reach the respective values, a corresponding A/D converter start request will be issued.

16.2.31 MT4 A/D Converter Start Request Cycle Setting Buffer Register A (MT4ADSRCSEBA), MT7 A/D Converter Start Request Cycle Setting Buffer Register A (MT7ADSRCSEBA), MT4 A/D Converter Start Request Cycle Setting Buffer Register B (MT4ADSRCSEBB), MT7 A/D Converter Start Request Cycle Setting Buffer Register B (MT7ADSRCSEBB)

Address MT4ADSRCSEBA: H'FFFF 8248, MT4ADSRCSEBB: H'FFFF 824A,
MT7ADSRCSEBA: H'FFFF 9248, MT7ADSRCSEBB: H'FFFF 924A

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Note: • MTADSRCSEB must not be accessed in eight bits; it should always be accessed in 16 bits.

The MTADSRCSEB registers (MT4ADSRCSEBA and MT4ADSRCSEBB in channel 4, and MT7ADSRCSEBA and MT7ADSRCSEBB in channel 7) are 16-bit readable/writable registers. Data is transferred from buffer register MTADSRCSEB to MTADSRCSE with the timing set in the BF bits in MTADSRCR.

16.2.32 MT34 Interrupt Skipping Mode Register (MT34ISMD), MT67 Interrupt Skipping Mode Register (MT67ISMD)

Address MT34ISMD: H'FFFF 823A, MT67ISMD: H'FFFF 923A

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	TITM
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b1	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b0	TITM	Interrupt Skipping Function Select	Selects one of the two types of interrupt skipping functions shown in Table 16.54.	R/W

The MTISMD registers are 8-bit registers that select the interrupt skipping function from two types.

Table 16.54 Interrupt Skipping Function Selected through TITM Bit

Bit 0	Description
TITM	
0	Selects interrupt skipping function 1*1
1	Selects interrupt skipping function 2*2

Notes: 1. The MT interrupt skipping set register 0 (MT34ISSE0 or MT67ISSE0) enables the interrupt skipping function 1.
2. The MT interrupt skipping set register 1 (MT4ISSE1 or MT7ISSE1) enables the interrupt skipping function 2.

16.2.33 MT34 Interrupt Skipping Setting Register 0 (MT34ISSE0), MT67 Interrupt Skipping Setting Register 0 (MT67ISSE0)

Address MT34ISSE0: H'FFFF 8230

	b7	b6	b5	b4	b3	b2	b1	b0
	T3AEN	T3ACOR [2:0]			T4VEN	T4VCOR [2:0]		
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	T3AEN	T3AEN	Enables or disables TGIA_3 interrupt skipping. 0: TGIA_3 interrupt skipping disabled 1: TGIA_3 interrupt skipping enabled	R/W
b6 to b4	T3ACOR [2:0]		These bits specify the TGIA_3 interrupt skipping count within the range from 0 to 7.* See table 16.55.	R/W
b3	T4VEN	T4VEN	Enables or disables TCIV_4 interrupt skipping. 0: TCIV_4 interrupt skipping disabled 1: TCIV_4 interrupt skipping enabled	R/W
b2 to b0	T4VCOR [2:0]		These bits specify the TCIV_4 interrupt skipping count within the range from 0 to 7.* See table 16.56.	R/W

Note: * When 0 is specified for the interrupt skipping count, no interrupt skipping will be performed. Before changing the interrupt skipping count, be sure to clear the T3AEN and T4VEN bits to 0 to clear the skipping count counter (MT34ISCNT0).

Address MT67ISSE0: H'FFFF 9230

	b7	b6	b5	b4	b3	b2	b1	b0
	T6AEN	T6ACOR [2:0]			T7VEN	T7VCOR [2:0]		
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	T6AEN	T6AEN	Enables or disables TGIA_6 interrupt skipping. 0: TGIA_6 interrupt skipping disabled 1: TGIA_6 interrupt skipping enable	R/W
b6 to b4	T6ACOR [2:0]		These bits specify the TGIA_6 interrupt skipping count within the range from 0 to 7.* See table 16.57.	R/W
b3	T7VEN	T7VEN	Enables or disables TCIV_7 interrupt skipping. 0: TCIV_7 interrupt skipping disabled 1: TCIV_7 interrupt skipping enabled	R/W
b2 to b0	T7VCOR [2:0]		These bits specify the TCIV_7 interrupt skipping count within the range from 0 to 7.* See table 16.58.	R/W

Note: * When 0 is specified for the interrupt skipping count, no interrupt skipping will be performed. Before changing the interrupt skipping count, be sure to clear the T6AEN and T7VEN bits to 0 to clear the skipping count counter (MT67ISCNT0).

The MTISSE0 registers are 8-bit registers that enable or disable interrupt skipping and specify the interrupt skipping count. This setting is valid only while the MT interrupt skipping mode register (MT34ISMD and MT67ISMD) is set to 0; when the MT interrupt skipping mode register (MT34ISMD and MT67ISMD) is set to 1, the setting in the corresponding register is cleared.

Table 16.55 Setting of Interrupt Skipping Count by T3ACOR Bits

Bit 6	Bit 5	Bit 4	Description
T3ACOR2	T3ACOR1	T3ACOR0	
0	0	0	Does not skip TGIA_3 interrupts.
0	0	1	Sets the TGIA_3 interrupt skipping count to 1.
0	1	0	Sets the TGIA_3 interrupt skipping count to 2.
0	1	1	Sets the TGIA_3 interrupt skipping count to 3.
1	0	0	Sets the TGIA_3 interrupt skipping count to 4.
1	0	1	Sets the TGIA_3 interrupt skipping count to 5.
1	1	0	Sets the TGIA_3 interrupt skipping count to 6.
1	1	1	Sets the TGIA_3 interrupt skipping count to 7.

Table 16.56 Setting of Interrupt Skipping Count by T4VCOR Bits

Bit 2	Bit 1	Bit 0	Description
T4VCOR2	T4VCOR1	T4VCOR0	
0	0	0	Does not skip TCIV_4 interrupts.
0	0	1	Sets the TCIV_4 interrupt skipping count to 1.
0	1	0	Sets the TCIV_4 interrupt skipping count to 2.
0	1	1	Sets the TCIV_4 interrupt skipping count to 3.
1	0	0	Sets the TCIV_4 interrupt skipping count to 4.
1	0	1	Sets the TCIV_4 interrupt skipping count to 5.
1	1	0	Sets the TCIV_4 interrupt skipping count to 6.
1	1	1	Sets the TCIV_4 interrupt skipping count to 7.

Table 16.57 Setting of Interrupt Skipping Count by T6ACOR Bits

Bit 6	Bit 5	Bit 4	Description
T6ACOR2	T6ACOR1	T6ACOR0	
0	0	0	Does not skip TGIA_6 interrupts.
0	0	1	Sets the TGIA_6 interrupt skipping count to 1.
0	1	0	Sets the TGIA_6 interrupt skipping count to 2.
0	1	1	Sets the TGIA_6 interrupt skipping count to 3.
1	0	0	Sets the TGIA_6 interrupt skipping count to 4.
1	0	1	Sets the TGIA_6 interrupt skipping count to 5.
1	1	0	Sets the TGIA_6 interrupt skipping count to 6.
1	1	1	Sets the TGIA_6 interrupt skipping count to 7.

Table 16.58 Setting of Interrupt Skipping Count by T7VCOR Bits

Bit 2	Bit 1	Bit 0	Description
T7VCOR2	T7VCOR1	T7VCOR0	
0	0	0	Does not skip TCIV_7 interrupts.
0	0	1	Sets the TCIV_7 interrupt skipping count to 1.
0	1	0	Sets the TCIV_7 interrupt skipping count to 2.
0	1	1	Sets the TCIV_7 interrupt skipping count to 3.
1	0	0	Sets the TCIV_7 interrupt skipping count to 4.
1	0	1	Sets the TCIV_7 interrupt skipping count to 5.
1	1	0	Sets the TCIV_7 interrupt skipping count to 6.
1	1	1	Sets the TCIV_7 interrupt skipping count to 7.

16.2.34 MT34 Interrupt Skipping Count Counter 0 (MT34ISCNT0), MT67 Interrupt Skipping Count Counter 0 (MT67ISCNT0)

Address MT34ISCNT0: H'FFFF 8231

	b7	b6	b5	b4	b3	b2	b1	b0
	—	T3ACNT [2:0]			—	T4VCNT [2:0]		
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	—	Reserved	This bit is always read as 0 and the write value should always be 0.	R
b6 to b4	T3ACNT [2:0]	TGIA_3 Interrupt Counter	While the T3AEN bit in MT34ISSE0 is set to 1, the count in these bits is incremented every time a TGIA_3 interrupt occurs. [Conditions of being set to 0] • When the TITM bit in MT34ISMD is 1 • When the T3AEN bit in MT34ISSE0 is 0 • When the T3ACOR bit in MT34ISSE0 is 0 • When the T3ACOR bit value in MT34ISSE0 matches T3ACNT2 to T3ACNT0 in MT34ISCNT0.	R
b3	—	Reserved	This bit is always read as 0 and the write value should always be 0.	R
b2 to b0	T4VCNT [2:0]	TCIV_4 Interrupt Counter	While the T4VEN bit in MT34ISSE0 is set to 1, the count in these bits is incremented every time a TCIV_4 interrupt occurs. [Conditions of being set to 0] • When the TITM bit in MT34ISMD is 1 • When the T4VEN bit in MT34ISSE0 is 0 • When the T4VCOR bit in MT34ISSE0 is 0 • When the T4VCOR bit value in MT34ISSE0 matches T4VCNT2 to T4VCNT0 in MT34ISCNT0.	R

Note: To clear the MT34ISCNT0 counter, clear the T3AEN and T4VEN bits in MT34ISSE0 to 0.

Address MT67ISCNT0: H'FFFF 9231

	b7	b6	b5	b4	b3	b2	b1	b0
	—	T6ACNT [2:0]			—	T7VCNT [2:0]		
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	—	Reserved	This bit is always read as 0 and the write value should always be 0.	R
b6 to b4	T6ACNT [2:0]	TGIA_6 Interrupt Counter	While the T6AEN bit in MT67ISSE0 is set to 1, the count in these bits is incremented every time a TGIA_6 interrupt occurs. [Conditions of being set to 0] • When the TITM bit in MT67ISMD is 1 • When the T6AEN bit in MT67ISSE0 is 0 • When the T6ACOR bit in MT67ISSE0 is 0 • When the T6ACOR bit value in MT67ISSE0 matches T6ACNT2 to T6ACNT0 in MT67ISCNT0.	R
b3	—	Reserved	This bit is always read as 0 and the write value should always be 0.	R
b2 to b0	T7VCNT [2:0]	TCIV_7 Interrupt Counter	While the T7VEN bit in MT67ISSE0 is set to 1, the count in these bits is incremented every time a TCIV_7 interrupt occurs. [Conditions of being set to 0] • When the TITM bit in MT67ISMD is 1 • When the T7VEN bit in MT67ISSE0 is 0 • When the T7VCOR bit in MT67ISSE0 is 0 • When the T7VCOR bit value in MT67ISSE0 matches T7VCNT2 to T7VCNT0 in MT67ISCNT0.	R

Note: To clear the MT67ISCNT0 counter, clear the T6AEN and T7VEN bits in MT67ISSE0 to 0.

The MTISCNT0 registers are 8-bit read-only registers that retain their values even after stopping the count operation of counters MT3CNT and MT4CNT (MT6CNT and MT7CNT).

16.2.35 MT4 Interrupt Skipping Setting Register 1 (MT4ISSE1), MT7 Interrupt Skipping Setting Register 1 (MT7ISSE1)

Address MT4ISSE1: H'FFFF 823B

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	TRG4COR [2:0]		
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b3	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b2 to b0	TRG4COR [2:0]		These bits specify the TRG4AN and TRG4BN interrupt skipping count within the range from 0 to 7. See table 16.59.	R/W

Table 16.59 Setting of Interrupt Skipping Count by TRG4COR Bits

Bit 2	Bit 1	Bit 0	Description
TRG4COR2	TRG4COR1	TRG4COR0	
0	0	0	Does not skip TRG4AN and TRG4BN interrupts.
0	0	1	Sets the TRG4AN and TRG4BN interrupt skipping count to 1.
0	1	0	Sets the TRG4AN and TRG4BN interrupt skipping count to 2.
0	1	1	Sets the TRG4AN and TRG4BN interrupt skipping count to 3.
1	0	0	Sets the TRG4AN and TRG4BN interrupt skipping count to 4.
1	0	1	Sets the TRG4AN and TRG4BN interrupt skipping count to 5.
1	1	0	Sets the TRG4AN and TRG4BN interrupt skipping count to 6.
1	1	1	Sets the TRG4AN and TRG4BN interrupt skipping count to 7.

Address MT7ISSE1: H'FFFF 923B

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	TRG7COR [2:0]		
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b3	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b2 to b0	TRG7COR [2:0]		These bits specify the TRG7AN and TRG7BN interrupt skipping count within the range from 0 to 7. See table 16.60.	R/W

Table 16.60 Setting of Interrupt Skipping Count by TRG7COR Bits

Bit 2	Bit 1	Bit 0	Description
TRG7COR2	TRG7COR1	TRG7COR0	
0	0	0	Does not skip TRG7AN and TRG7BN interrupts.
0	0	1	Sets the TRG7AN and TRG7BN interrupt skipping count to 1.
0	1	0	Sets the TRG7AN and TRG7BN interrupt skipping count to 2.
0	1	1	Sets the TRG7AN and TRG7BN interrupt skipping count to 3.
1	0	0	Sets the TRG7AN and TRG7BN interrupt skipping count to 4.
1	0	1	Sets the TRG7AN and TRG7BN interrupt skipping count to 5.
1	1	0	Sets the TRG7AN and TRG7BN interrupt skipping count to 6.
1	1	1	Sets the TRG7AN and TRG7BN interrupt skipping count to 7.

The MTISSE1 registers are 8-bit registers that specify the interrupt skipping count for TRG4AN and TRG4BN (TRG7AN and TRG7BN).

This setting is valid only while the MT interrupt skipping mode register (MT34ISMD and MT67ISMD) is set to 1.

16.2.36 MT4 Interrupt Skipping Count Counter 1 (MT4ISCNT1), MT7 Interrupt Skipping Count Counter 1 (MT7ISCNT1)

Address MT4ISCNT1: H'FFFF 823C

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	TRG4CNT [2:0]		
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b3	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b2 to b0	TRG4CNT [2:0]		These bits start counting from the value set in TRG4COR and the count decrements every time TRG4AN or TRG4BN is generated. When the count reaches 0 and is reloaded, the TRG4AN and TRG4BN interrupts become valid. [Conditions of being set to 0] • When the TITM bit in MT34ISMD is cleared to 0 • When the TRG4COR2 to TRG4COR0 bits in MT4ISSE1 are cleared to 0 • When the count of TRG4AN and TRG4BN occurrence matches the TRG4COR2 to TRG4COR0 value in MT4ISSE1	R

Address MT7ISCNT1: H'FFFF 923C

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	TRG7CNT [2:0]		
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b3	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b2 to b0	TRG7CNT [2:0]		These bits start counting from the value set in TRG7COR and the count decrements every time TRG7AN or TRG7BN is generated. When the count reaches 0 and is reloaded, the TRG7AN and TRG7BN interrupts become valid. [Conditions of being set to 0] • When the TITM bit in MT67ISMD is cleared to 0 • When the TRG7COR2 to TRG7COR0 bits in MT7ISSE1 are cleared to 0 • When the count of TRG7AN and TRG7BN occurrence matches the TRG7COR2 to TRG7COR0 value in MT7ISSE1	R

MTISCNT1 is an 8-bit read-only counter. The counter starts counting from the value set in TRG4COR and TRG7COR and the count decrements every time TRG4AN or TRG4BN (MT4ISCNT1), or TRG7AN or TRG7BN (MT7ISCNT1) is generated. When the count reaches 0 and is reloaded, the TRG4AN, TRG4BN, TRG7AN, and TRG7BN interrupts become valid.

16.2.37 MT3467 Waveform Switchover Register (MT3467WSW)

Address H'FFFF 9500

	b7	b6	b5	b4	b3	b2	b1	b0
	—	TWSW 67 [2]	TWSW 67 [1]	TWSW 67 [0]	—	TWSW 34 [2]	TWSW 34 [1]	TWSW 34 [0]
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	—	Reserved	This bit is always read as 0 and the write value should always be 0.	R
b6	TWSW67 [2]	Software Switch between MTU-III PWM Output and Square Wave Output	TWSW67[2] switches the waveform output through external pin TIOC7B/D. 0: MTU-III PWM output 1: Square wave output	R/W
b5	TWSW67 [1]	Software Switch between MTU-III PWM Output and Square Wave Output	TWSW67[1] switches the waveform output through external pin TIOC7A/C. 0: MTU-III PWM output 1: Square wave output	R/W
b4	TWSW67 [0]	Software Switch between MTU-III PWM Output and Square Wave Output	TWSW67[0] switches the waveform output through external pin TIOC6B/D. 0: MTU-III PWM output 1: Square wave output	R/W
b3	—	Reserved	This bit is always read as 0 and the write value should always be 0.	R
b2	TWSW34 [2]	Software Switch between MTU-III PWM Output and Square Wave Output	TWSW34[2] switches the waveform output through external pin TIOC4B/D. 0: MTU-III PWM output 1: Square wave output	R/W
b1	TWSW34 [1]	Software Switch between MTU-III PWM Output and Square Wave Output	TWSW34[1] switches the waveform output through external pin TIOC4A/C. 0: MTU-III PWM output 1: Square wave output	R/W
b0	TWSW34 [0]	Software Switch between MTU-III PWM Output and Square Wave Output	TWSW34[0] switches the waveform output through external pin TIOC3B/D. 0: MTU-III PWM output 1: Square wave output	R/W

Writing 1 to the corresponding bit in MT3467WSW forcibly sets the waveform output to MTU-III PWM output or square wave output through external pin (TIOC3B/D, TIOC4A/C, TIOC4B/D, TIOC6B/D, TIOC7A/C, or TIOC7B/D) by software.

16.2.38 MT1 Waveform Input Capture/Output Compare Switchover Enable Register A (MT1WIOSWENA)

Address H'FFFF 9510

	b7	b6	b5	b4	b3	b2	b1	b0
	—	ICOC1A67 [2]	ICOC1A67 [1]	ICOC1A67 [0]	—	ICOC1A34 [2]	ICOC1A34 [1]	ICOC1A34 [0]
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	—	Reserved	This bit is always read as 0. The write value should always be 0.	R
b6	ICOC1A67 [2]		Switches the output at MTGRA input capture/output compare occurrence in channel 1 between the MTU-III PWM output and square wave output. ICOC1A67[2] switches the waveform output through external pin TIOC7B/D. 0: MTU-III PWM output 1: Square wave output	R/W
b5	ICOC1A67 [1]		Switches the output at MTGRA input capture/output compare occurrence in channel 1 between the MTU-III PWM output and square wave output. ICOC1A67[1] switches the waveform output through external pin TIOC7A/C. 0: MTU-III PWM output 1: Square wave output	R/W
b4	ICOC1A67 [0]		Switches the output at MTGRA input capture/output compare occurrence in channel 1 between the MTU-III PWM output and square wave output. ICOC1A67[0] switches the waveform output through external pin TIOC6B/D. 0: MTU-III PWM output 1: Square wave output	R/W
b3	—	Reserved	This bit is always read as 0. The write value should always be 0.	R
b2	ICOC1A34 [2]		Switches the output at MTGRA input capture/output compare occurrence in channel 1 between the MTU-III PWM output and square wave output. ICOC1A34[2] switches the waveform output through external pin TIOC4B/D. 0: MTU-III PWM output 1: Square wave output	R/W
b1	ICOC1A34 [1]		Switches the output at MTGRA input capture/output compare occurrence in channel 1 between the MTU-III PWM output and square wave output. ICOC1A34[1] switches the waveform output through external pin TIOC4A/C. 0: MTU-III PWM output 1: Square wave output	R/W
b0	ICOC1A34 [0]		Switches the output at MTGRA input capture/output compare occurrence in channel 1 between the MTU-III PWM output and square wave output. ICOC1A34[0] switches the waveform output through external pin TIOC3B/D. 0: MTU-III PWM output 1: Square wave output	R/W

Writing 1 to the corresponding bit in MT1WIOSWENA switches the output from the corresponding external pin (TIOC3B/D, TIOC4A/C, TIOC4B/D, TIOC6B/D, TIOC7A/C, or TIOC7B/D) between the MTU-III PWM output and the square wave output at MTGRA input capture or compare match occurrence in channel 1.

When MTGRA input capture or compare match occurrence in channel 1 is used as waveform switching timing, the interrupt signal is used to switch the waveform. Enable the interrupt by MT interrupt enable register. Also, clear the TGFA bit of MTSR in channel 1 beforehand.

16.2.39 MT1 Waveform Input Capture/Output Compare Switchover Enable Register B (MT1WIOSWENB)

Address H'FFFF 9512

	b7	b6	b5	b4	b3	b2	b1	b0
	—	ICOC1B67 [2]	ICOC1B67 [1]	ICOC1B67 [0]	—	ICOC1B34 [2]	ICOC1B34 [1]	ICOC1B34 [0]
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	—	Reserved	This bit is always read as 0. The write value should always be 0.	R
b6	ICOC1B67 [2]		Switches the output at MTGRB input capture/output compare occurrence in channel 1 between the MTU-III PWM output and square wave output. ICOC1B67[2] switches the waveform output through external pin TIOC7B/D. 0: MTU-III PWM output 1: Square wave output	R/W
b5	ICOC1B67 [1]		Switches the output at MTGRB input capture/output compare occurrence in channel 1 between the MTU-III PWM output and square wave output. ICOC1B67[1] switches the waveform output through external pin TIOC7A/C. 0: MTU-III PWM output 1: Square wave output	R/W
b4	ICOC1B67 [0]		Switches the output at MTGRB input capture/output compare occurrence in channel 1 between the MTU-III PWM output and square wave output. ICOC1B67[0] switches the waveform output through external pin TIOC6B/D. 0: MTU-III PWM output 1: Square wave output	R/W
b3	—	Reserved	This bit is always read as 0. The write value should always be 0.	R
b2	ICOC1B34 [2]		Switches the output at MTGRB input capture/output compare occurrence in channel 1 between the MTU-III PWM output and square wave output. ICOC1B34[2] switches the waveform output through external pin TIOC4B/D. 0: MTU-III PWM output 1: Square wave output	R/W
b1	ICOC1B34 [1]		Switches the output at MTGRB input capture/output compare occurrence in channel 1 between the MTU-III PWM output and square wave output. ICOC1B34[1] switches the waveform output through external pin TIOC4A/C. 0: MTU-III PWM output 1: Square wave output	R/W
b0	ICOC1B34 [0]		Switches the output at MTGRB input capture/output compare occurrence in channel 1 between the MTU-III PWM output and square wave output. ICOC1B34[0] switches the waveform output through external pin TIOC3B/D. 0: MTU-III PWM output 1: Square wave output	R/W

Writing 1 to the corresponding bit in MT1WIOSWENB switches the output from the corresponding external pin (TIOC3B/D, TIOC4A/C, TIOC4B/D, TIOC6B/D, TIOC7A/C, or TIOC7B/D) between the MTU-III PWM output and the square wave output at MTGRB input capture or compare match occurrence in channel 1.

When MTGRB input capture or compare match occurrence in channel 1 is used as waveform switching timing, the interrupt signal is used to switch the waveform. Enable the interrupt by MT interrupt enable register. Also, clear the TGFB bit of MTSR in channel 1 beforehand.

16.2.40 MT2 Waveform Input Capture/Output Compare Switchover Enable Register A (MT2WIOSWENA)

Address H'FFFF 9520

	b7	b6	b5	b4	b3	b2	b1	b0
	—	ICOC2A67 [2]	ICOC2A67 [1]	ICOC2A67 [0]	—	ICOC2A34 [2]	ICOC2A34 [1]	ICOC2A34 [0]
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	—	Reserved	This bit is always read as 0. The write value should always be 0.	R
b6	ICOC2A67 [2]		Switches the output at MTGRA input capture/output compare occurrence in channel 2 between the MTU-III PWM output and square wave output. ICOC2A67[2] switches the waveform output through external pin TIOC7B/D. 0: MTU-III PWM output 1: Square wave output	R/W
b5	ICOC2A67 [1]		Switches the output at MTGRA input capture/output compare occurrence in channel 2 between the MTU-III PWM output and square wave output. ICOC2A67[1] switches the waveform output through external pin TIOC7A/C. 0: MTU-III PWM output 1: Square wave output	R/W
b4	ICOC2A67 [0]		Switches the output at MTGRA input capture/output compare occurrence in channel 2 between the MTU-III PWM output and square wave output. ICOC2A67[0] switches the waveform output through external pin TIOC6B/D. 0: MTU-III PWM output 1: Square wave output	R/W
b3	—	Reserved	This bit is always read as 0. The write value should always be 0.	R
b2	ICOC2A34 [2]		Switches the output at MTGRA input capture/output compare occurrence in channel 2 between the MTU-III PWM output and square wave output. ICOC2A34[2] switches the waveform output through external pin TIOC4B/D. 0: MTU-III PWM output 1: Square wave output	R/W
b1	ICOC2A34 [1]		Switches the output at MTGRA input capture/output compare occurrence in channel 2 between the MTU-III PWM output and square wave output. ICOC2A34[1] switches the waveform output through external pin TIOC4A/C. 0: MTU-III PWM output 1: Square wave output	R/W
b0	ICOC2A34 [0]		Switches the output at MTGRA input capture/output compare occurrence in channel 2 between the MTU-III PWM output and square wave output. ICOC2A34[0] switches the waveform output through external pin TIOC3B/D. 0: MTU-III PWM output 1: Square wave output	R/W

Writing 1 to the corresponding bit in MT2WIOSWENA switches the output from the corresponding external pin (TIOC3B/D, TIOC4A/C, TIOC4B/D, TIOC6B/D, TIOC7A/C, or TIOC7B/D) between the MTU-III PWM output and the square wave output at MTGRA input capture or compare match occurrence in channel 2.

When MTGRA input capture or compare match occurrence in channel 2 is used as waveform switching timing, the interrupt signal is used to switch the waveform. Enable the interrupt by MT interrupt enable register. Also, clear the TGFA bit of MTSR in channel 2 beforehand.

16.2.41 MT2 Waveform Input Capture/Output Compare Switchover Enable Register B (MT2WIOSWENB)

Address H'FFFF 9522

	b7	b6	b5	b4	b3	b2	b1	b0
	—	ICOC2 B67 [2]	ICOC2 B67 [1]	ICOC2 B67 [0]	—	ICOC2 B34 [2]	ICOC2 B34 [1]	ICOC2 B34 [0]
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	—	Reserved	This bit is always read as 0. The write value should always be 0.	R
b6	ICOC2B67 [2]		Switches the output at MTGRB input capture/output compare occurrence in channel 2 between the MTU-III PWM output and square wave output. ICOC2B67[2] switches the waveform output through external pin TIOC7B/D. 0: MTU-III PWM output 1: Square wave output	R/W
b5	ICOC2B67 [1]		Switches the output at MTGRB input capture/output compare occurrence in channel 2 between the MTU-III PWM output and square wave output. ICOC2B67[1] switches the waveform output through external pin TIOC7A/C. 0: MTU-III PWM output 1: Square wave output	R/W
b4	ICOC2B67 [0]		Switches the output at MTGRB input capture/output compare occurrence in channel 2 between the MTU-III PWM output and square wave output. ICOC2B67[0] switches the waveform output through external pin TIOC6B/D. 0: MTU-III PWM output 1: Square wave output	R/W
b3	—	Reserved	This bit is always read as 0. The write value should always be 0.	R
b2	ICOC2B34 [2]		Switches the output at MTGRB input capture/output compare occurrence in channel 2 between the MTU-III PWM output and square wave output. ICOC2B34[2] switches the waveform output through external pin TIOC4B/D. 0: MTU-III PWM output 1: Square wave output	R/W
b1	ICOC2B34 [1]		Switches the output at MTGRB input capture/output compare occurrence in channel 2 between the MTU-III PWM output and square wave output. ICOC2B34[1] switches the waveform output through external pin TIOC4A/C. 0: MTU-III PWM output 1: Square wave output	R/W
b0	ICOC2B34 [0]		Switches the output at MTGRB input capture/output compare occurrence in channel 2 between the MTU-III PWM output and square wave output. ICOC21B34 [0] switches the waveform output through external pin TIOC3B/D. 0: MTU-III PWM output 1: Square wave output	R/W

Writing 1 to the corresponding bit in MT2WIOSWENB switches the output from the corresponding external pin (TIOC3B/D, TIOC4A/C, TIOC4B/D, TIOC6B/D, TIOC7A/C, or TIOC7B/D) between the MTU-III PWM output and the square wave output at MTGRB input capture or compare match occurrence in channel 2.

When MTGRB input capture or compare match occurrence in channel 2 is used as waveform switching timing, the interrupt signal is used to switch the waveform. Enable the interrupt by MT interrupt enable register. Also, clear the TGFB bit of MTSR in channel 2 beforehand.

16.3 Operation

16.3.1 Basic Functions

Each channel has an MTCNT counter and MTGR registers. The MTCNT counter performs up-counting, and is also capable of free-running operation, periodic counting, and external event counting.
Each MTGR can be used as an input capture register or output compare register.
For functional settings for MTU-III external pins, see section 13, I/O Ports.

(1) Counter Operation

When one of bits CST0 to CST4 in MT01234STR, bits CST6 and CST7 in MT67STR, or bits CSTU5, CSTV5, and CSTW5 in MT5STR is set to 1, the MTCNT counter for the corresponding channel begins counting. MTCNT can operate as a free-running counter, periodic counter, for example.

(a) Example of Count Operation Setting Procedure

Figure 16.5 shows an example of the count operation setting procedure.

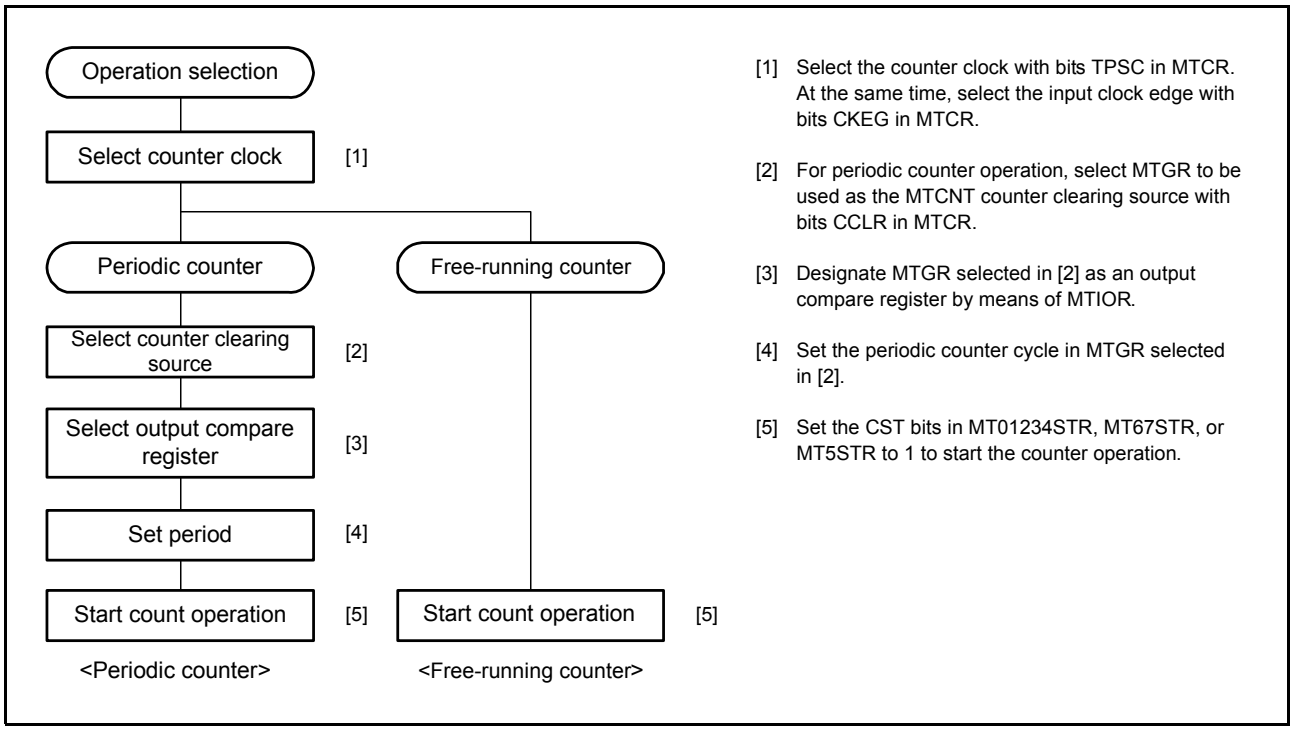


Figure 16.5 Example of Counter Operation Setting Procedure

(b) Free-Running Count Operation and Periodic Count Operation

Immediately after a reset, the MTU-III's MTCNT counters are all designated as free-running counters. When the relevant bit in MTSTR is set to 1, the corresponding MTCNT counter starts up-count operation as a free-running counter. When the MTCNT counter overflows (from H'FFFF to H'0000), the TCFV bit in MTSR is set to 1. If the value of the corresponding TCIEV bit in MTIEN is 1 at this point, the MTU-III requests an interrupt. After overflow, the MTCNT counter starts counting up again from H'0000.

Figure 16.6 illustrates free-running counter operation.

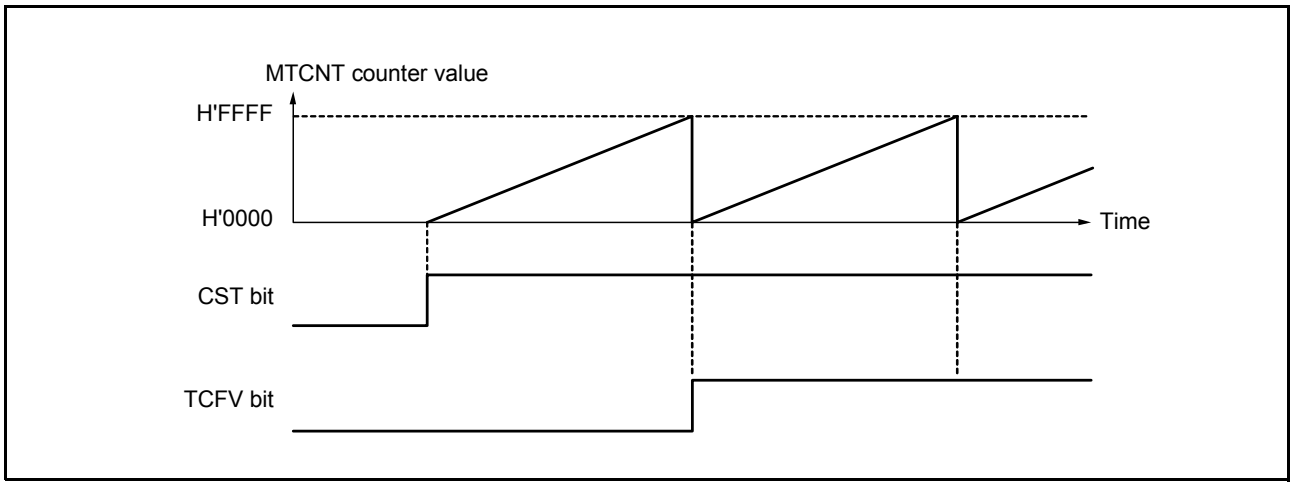


Figure 16.6 Free-Running Counter Operation

When compare match is selected as the MTCNT counter clearing source, the MTCNT counter for the relevant channel performs periodic count operation. The MTGR register for setting the period is designated as an output compare register, and counter clearing by compare match is selected by means of the CCLR bit in MTiCR (i = 0 to 4, 6, and 7). After the settings have been made, MTCNT starts up-count operation as a periodic counter when the corresponding bit in MTSTR is set to 1. When the count value matches the value in MTGR, the TGF flag in MTSR is set to 1 and the MTCNT counter is cleared to H'0000.

If the value of the corresponding TGIE bit in MTIEN is 1 at this point, the MTU-III requests an interrupt. After a compare match, the MTCNT counter starts counting up again from H'0000.

Figure 16.7 illustrates periodic counter operation.

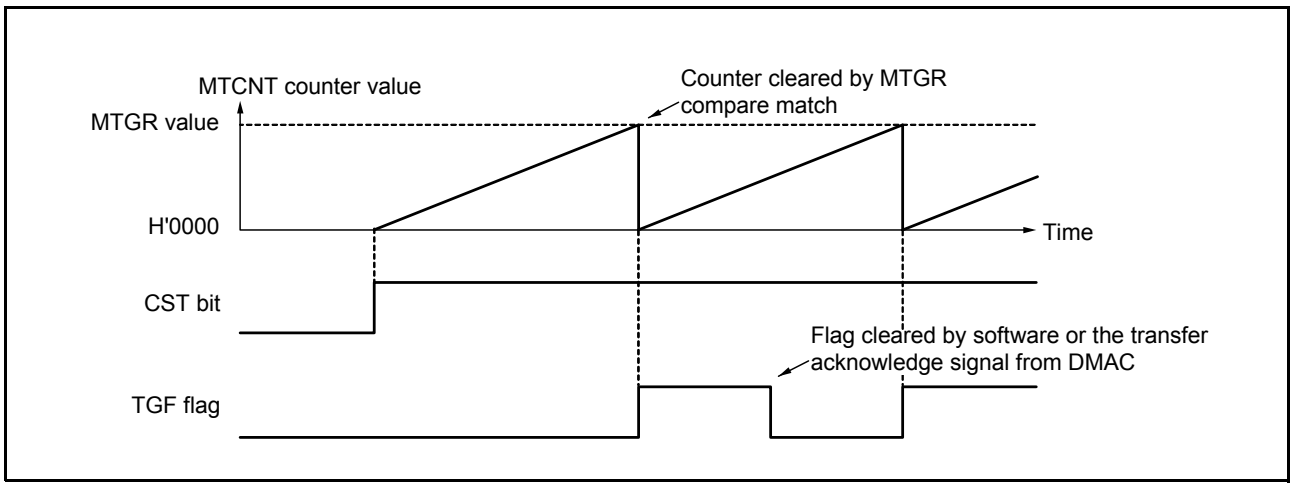


Figure 16.7 Periodic Counter Operation

(2) Waveform Output by Compare Match

The MTU-III can perform 0, 1, or toggle output from the corresponding output pin using compare match.

(a) Example of Setting Procedure for Waveform Output by Compare Match

Figure 16.8 shows an example of the setting procedure for waveform output by compare match.

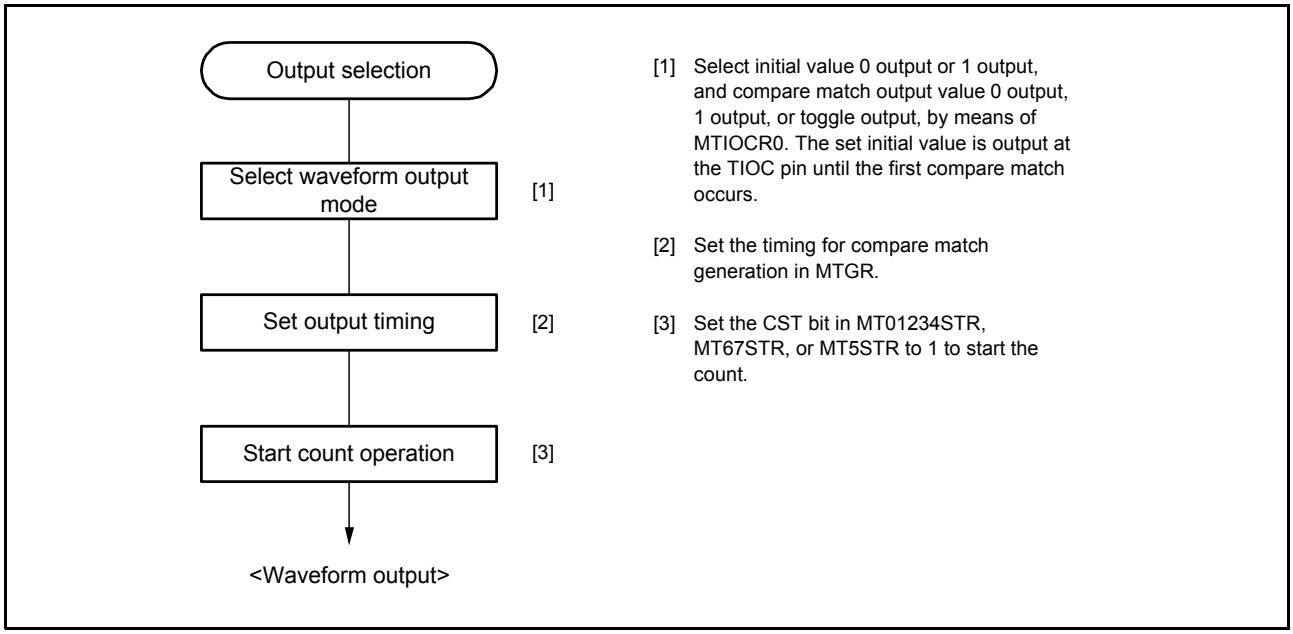


Figure 16.8 Example of Setting Procedure for Waveform Output by Compare Match

(b) Examples of Waveform Output Operation

Figure 16.9 illustrates an example of 0 output/1 output operation.

In this example the MTCNT counter has been designated as a free-running counter, and settings have been made such that 1 is output by compare match A, and 0 is output by compare match B. When the set level and the pin level coincide, the pin level does not change.

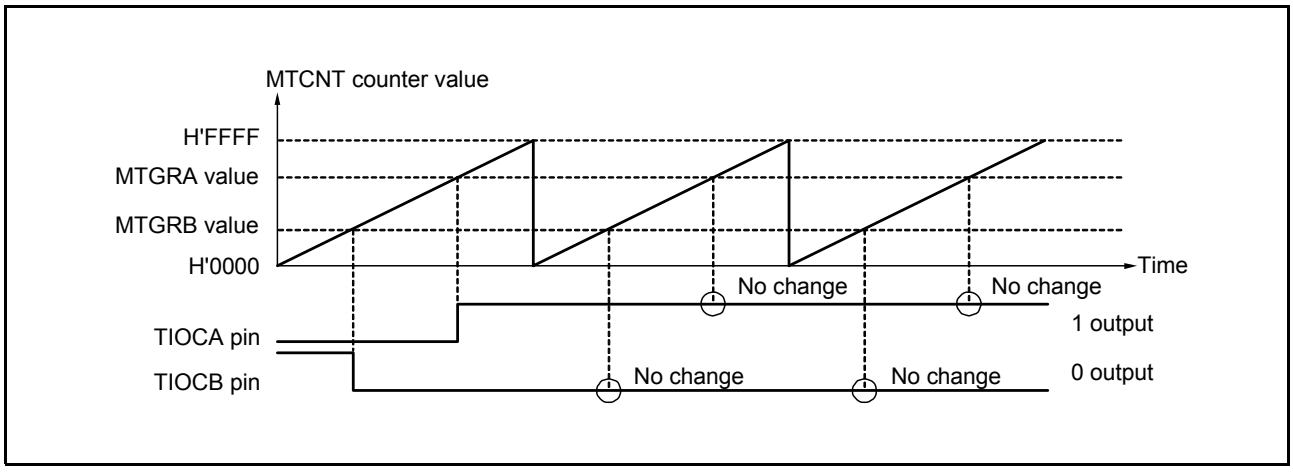


Figure 16.9 Example of 0 Output/1 Output Operation

Figure 16.10 shows an example of toggle output operation.
In this example, the MTCNT counter has been designated as a periodic counter (with counter clearing on compare match B), and settings have been made such that both compare matches A and B toggle the output.

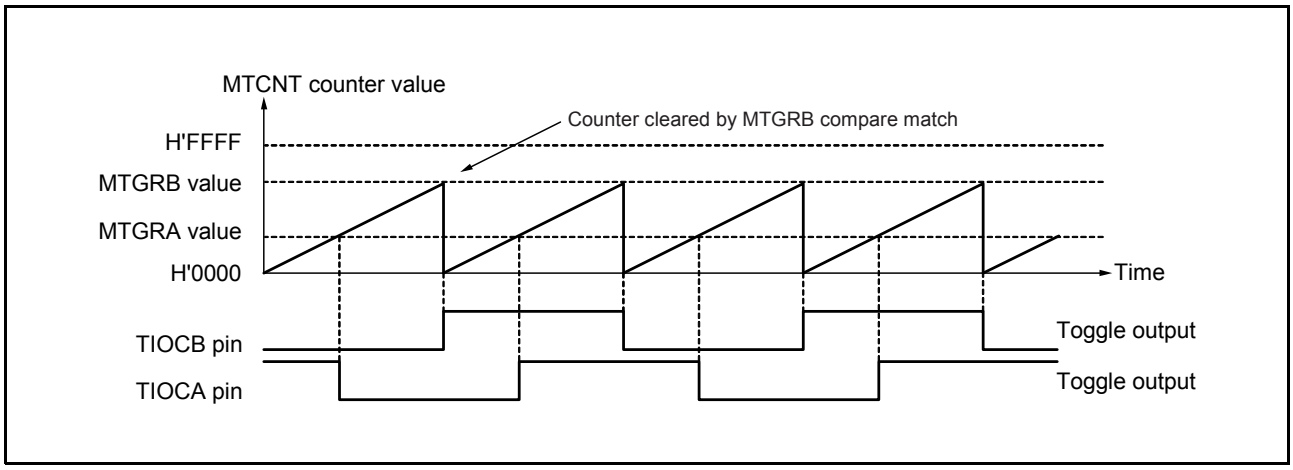


Figure 16.10 Example of Toggle Output Operation

(3) Input Capture Function

The MTCNT counter value can be transferred to MTGR on detection of the TIOC pin input edge.
Rising edge, falling edge, or both edges can be selected as the detected edge. It is also possible to specify channel 0's compare match signal or channel 0's count clock as the input capture source for channel 1.

Note: When channel 0 counter input clock is used as the input capture input for channels 0 and 1, MTU operating clock without frequency division should not be selected as the counter input clock used for input capture input. Input capture will not be generated if MTU operating clock without frequency division is selected.

(a) Example of Input Capture Operation Setting Procedure

Figure 16.11 shows an example of the input capture operation setting procedure.

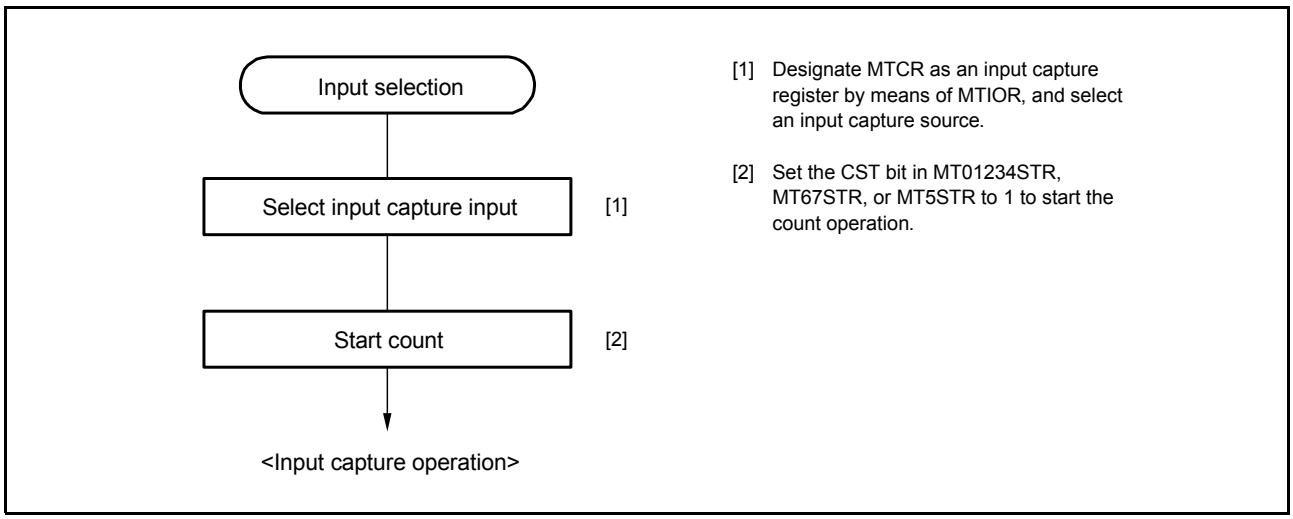


Figure 16.11 Example of Input Capture Operation Setting Procedure

(b) Example of Input Capture Operation

Figure 16.12 shows an example of input capture operation.

In this example both rising and falling edges have been selected as the TIOCA pin input capture input edge, the falling edge has been selected as the TIOCB pin input capture input edge, and counter clearing by MTGRB input capture has been designated for the MTCNT counter.

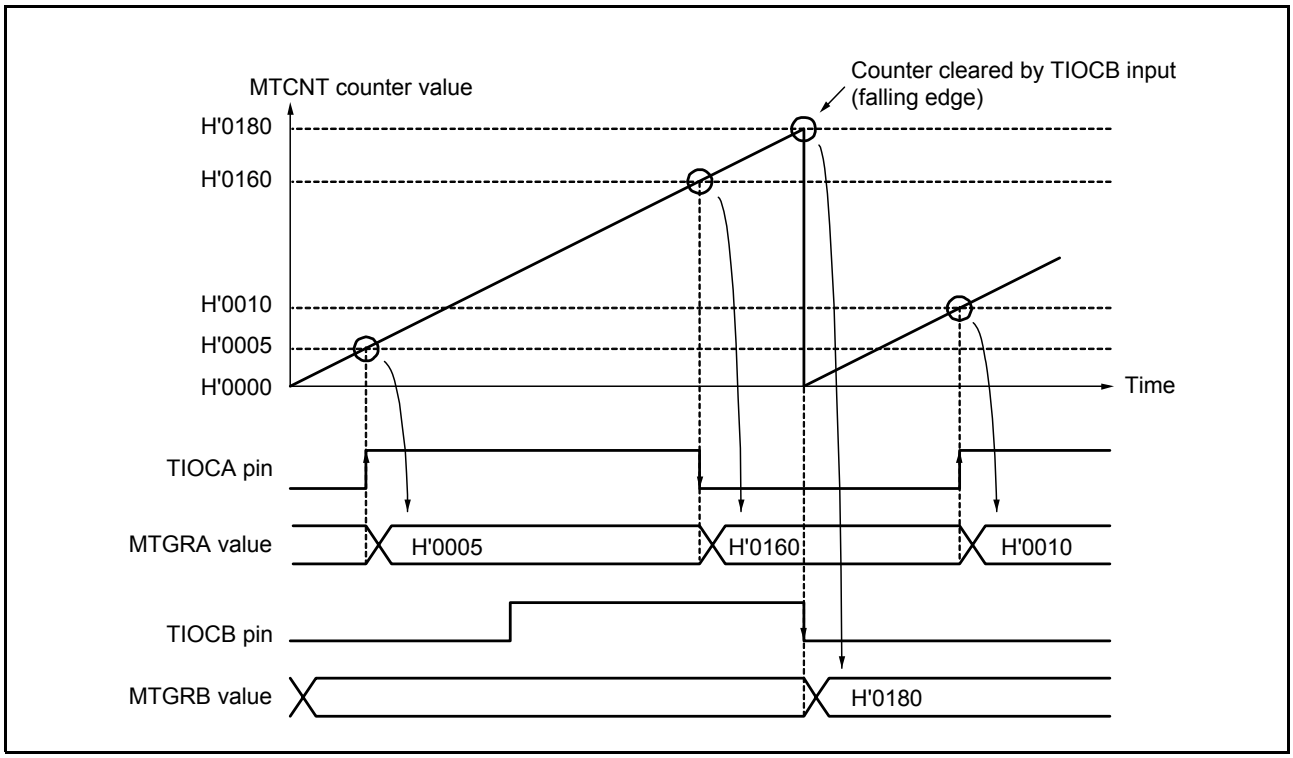


Figure 16.12 Example of Input Capture Operation

16.3.2 Synchronous Operation

In synchronous operation, multiple MTCNT counters can be rewritten simultaneously (synchronous presetting). Also, a number of MTCNT counters can be cleared simultaneously by making the appropriate setting in MTCR (synchronous clearing).

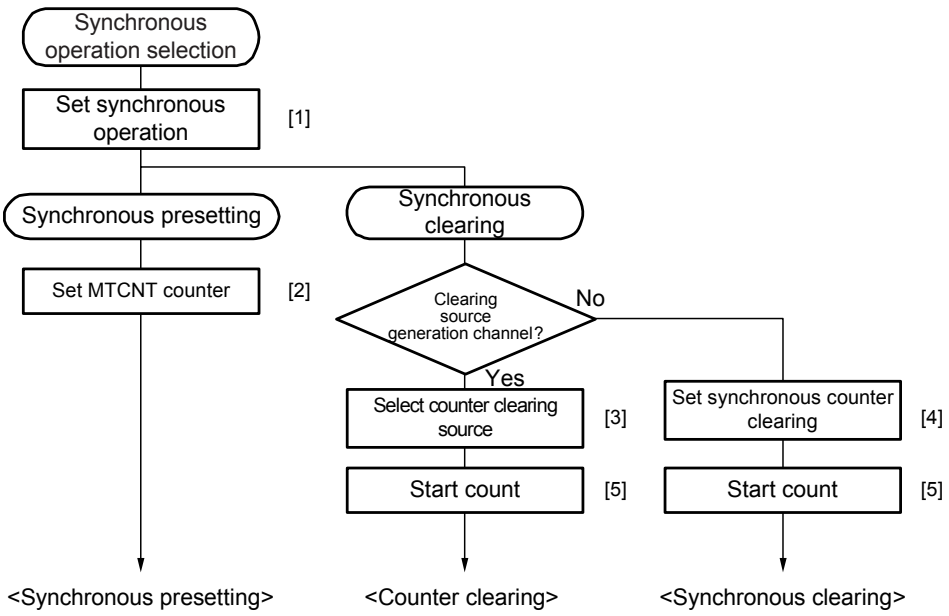
Synchronous operation enables the number of MTGRs to be increased with respect to a single time base.

Channels 0 to 4 and channels 6 and 7 can be separately designated for synchronous operation.

Channel 5 cannot be used for synchronous operation.

(1) Example of Synchronous Operation Setting Procedure

Figure 16.13 shows an example of the synchronous operation setting procedure.



[1] Set to 1 the SYNC bits in MT01234SY or MT67SY corresponding to the channels to be designated for synchronous operation.

[2] When the MTCNT counter of any of the channels designated for synchronous operation is written to, the same value is simultaneously written to the other MTCNT counters.

[3] Use bits CCLR in MTCR to specify MTCNT counter clearing by input capture/output compare, etc.

[4] Use bits CCLR in MTCR to designate synchronous clearing for the counter clearing source.

[5] Set to 1 the CST bits in MT01234STR or MT67STR for the relevant channels, to start the count operation.

Figure 16.13 Example of Synchronous Operation Setting Procedure

(2) Example of Synchronous Operation

Figure 16.14 shows an example of synchronous operation.

In this example, synchronous operation and PWM mode 1 have been designated for channels 0 to 2, MT0GRB compare match has been set for the channel 0 counter clearing source, and synchronous clearing has been set for the channel 1 and 2 counter clearing source.

Three-phase PWM waveforms are output from pins TIOC0A, TIOC1A, and TIOC2A. At this time, synchronous presetting, and synchronous clearing by MT0GRB compare match, are performed for channel-0 to -2 MTCNT counters, and the data set in MT0GRB is used as the PWM cycle.

For details of PWM modes, see section section 16.3.5, PWM Mode.

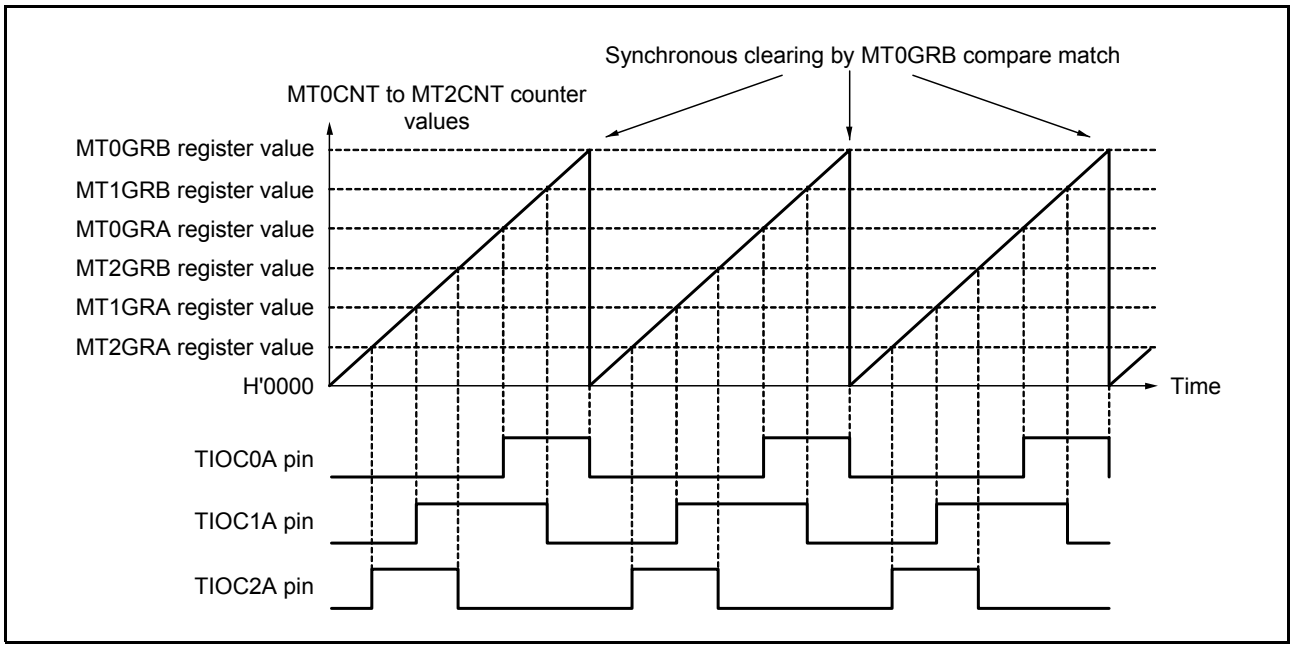


Figure 16.14 Example of Synchronous Operation

16.3.3 Buffer Operation

Buffer operation, provided for channels 0, 3, 4, 6, and 7, enables MTGRC and MTGRD to be used as buffer registers. In channel 0, MTGRF can also be used as a buffer register.

Buffer operation differs depending on whether MTGR has been designated as an input capture register or as an compare match register.

Note: MT0GRE cannot be designated as an input capture register and can only operate as a compare match register.

Table 16.61 shows the register combinations used in buffer operation.

Table 16.61 Register Combinations in Buffer Operation

Channel	MTGR Register	Buffer Register
0	MT0GRA	MT0GRC
	MT0GRB	MT0GRD
	MT0GRE	MT0GRF
3	MT3GRA	MT3GRC
	MT3GRB	MT3GRD
4	MT4GRA	MT4GRC
	MT4GRB	MT4GRD
6	MT6GRA	MT6GRC
	MT6GRB	MT6GRD
7	MT7GRA	MT7GRC
	MT7GRB	MT7GRD

- When MTGR is an output compare register

When a compare match occurs, the value in the buffer register for the corresponding channel is transferred to MTGR. Figure 16.15 illustrates compare match buffer operation.

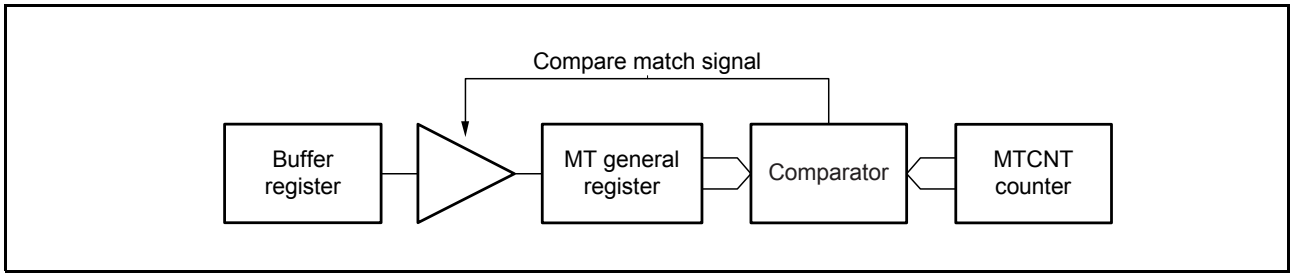


Figure 16.15 Compare Match Buffer Operation

- When MTGR is an input capture register

When input capture occurs, the value in the MTCNT counter is transferred to MTGR and the value previously held in MTGR is transferred to the buffer register.

Figure 16.16 shows input capture buffer operation.

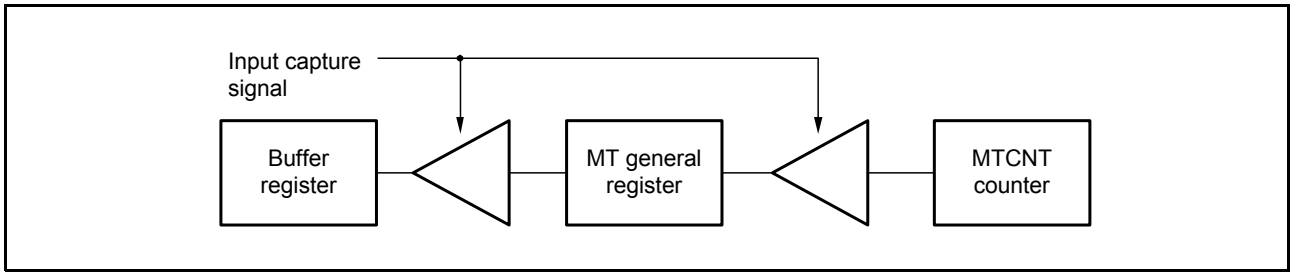


Figure 16.16 Input Capture Buffer Operation

(1) Example of Buffer Operation Setting Procedure

Figure 16.17 shows an example of the buffer operation setting procedure.

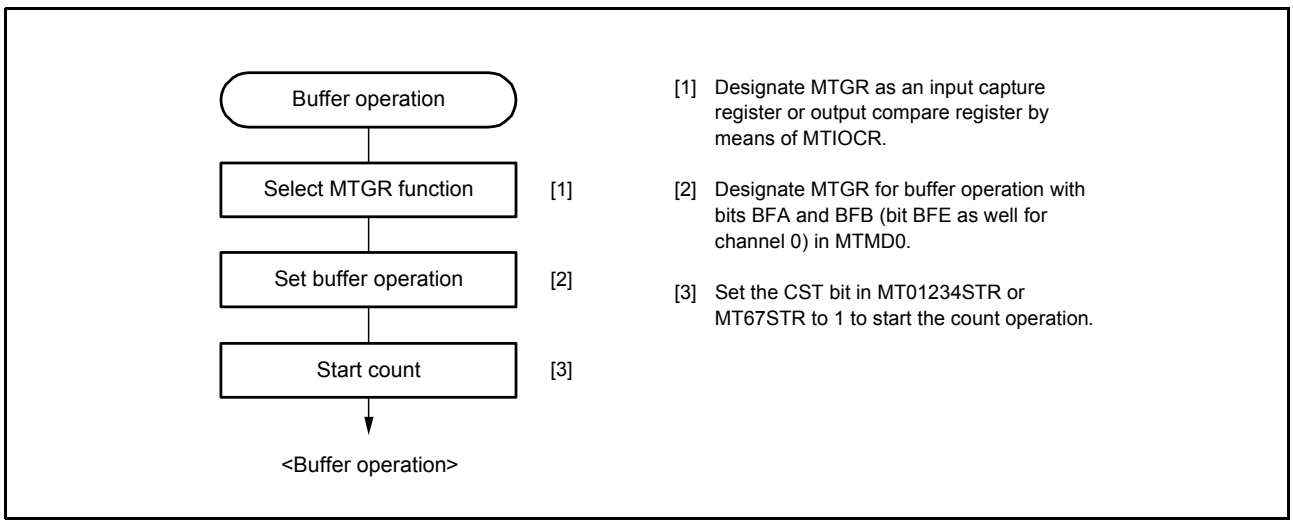


Figure 16.17 Example of Buffer Operation Setting Procedure

(2) Examples of Buffer Operation

(a) When MTGR is an output compare register

Figure 16.18 shows an operation example in which PWM mode 1 has been designated for channel 0, and the buffer operation has been designated for registers MTGRA and MTGRC. The settings used in this example are MTCNT counter clearing by compare match B, 1 output at compare match A, and 0 output at compare match B. In this example, the TTSA bit in MTBTM is cleared to 0.

As buffer operation has been set, when compare match A occurs, the output changes and the value in the buffer register (MTGRC) is simultaneously transferred to MT general register A (MTGRA). This operation is repeated each time that compare match A occurs.

For details of PWM modes, see section section 16.3.5, PWM Mode.

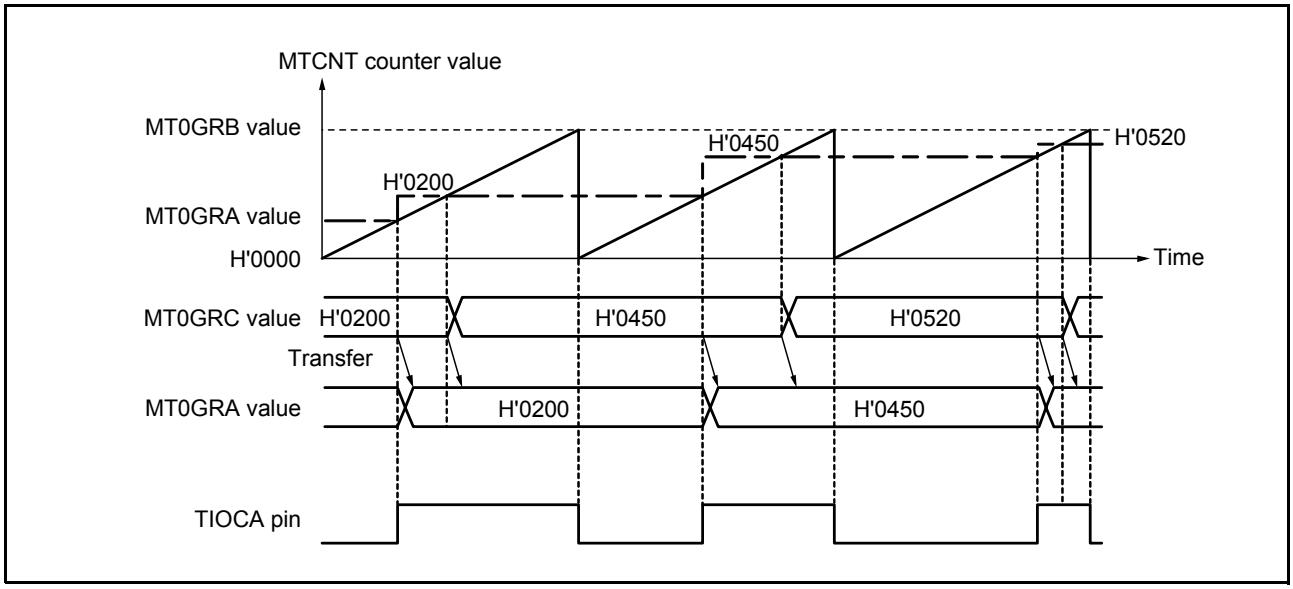


Figure 16.18 Example of Buffer Operation (1)

(b) When MTGR is an input capture register

Figure 16.19 shows an operation example in which MTGRA has been designated as an input capture register, and the buffer operation has been designated for registers MTGRA and MTGRC.

Counter clearing by MTGRA input capture has been set for the MTCNT counter, and both rising and falling edges have been selected as the TIOCA pin input capture input edge.

As buffer operation has been set, when the MTCNT counter value is stored in MTGRA upon the occurrence of input capture A, the value previously stored in MTGRA is simultaneously transferred to MTGRC.

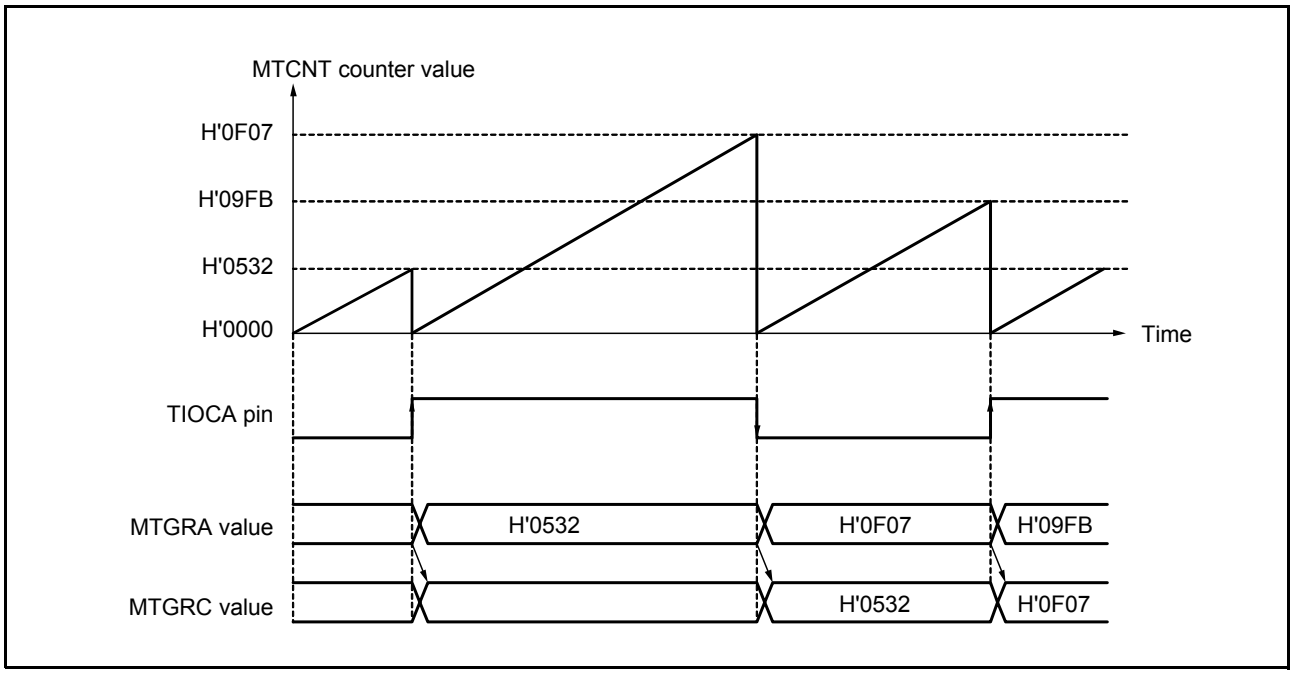


Figure 16.19 Example of Buffer Operation (2)

(3) Selecting Timing for Transfer from Buffer Registers to MTGR in Buffer Operation

The timing for transfer from buffer registers to MTGR can be selected in PWM mode 1 or 2 for channel 0 or in PWM mode 1 for channels 3, 4, 6, and 7 by setting MTBTM. Either compare match (initial setting) or the MTCNT counter clearing can be selected for the transfer timing. The MTCNT counter clearing as transfer timing is one of the following cases.

- When the MTCNT counter overflows (changes from H'FFFF to H'0000)
- When H'0000 is written to the MTCNT counter during counting
- When the MTCNT counter is cleared to H'0000 under the condition specified in the CCLR bits in MTCR

Note: MTBTM register settings should be conducted only when MTCNT counter operation is stopped.

Figure 16.20 shows an operation example in which PWM mode 1 has been designated for channel 0, and the buffer operation has been designated for registers MTGRA and MTGRC. The settings used in this example are MT0CNT counter clearing by compare match B, 1 output at compare match A, and 0 output at compare match B. The TTSA bit in MT0BTM is set to 1.

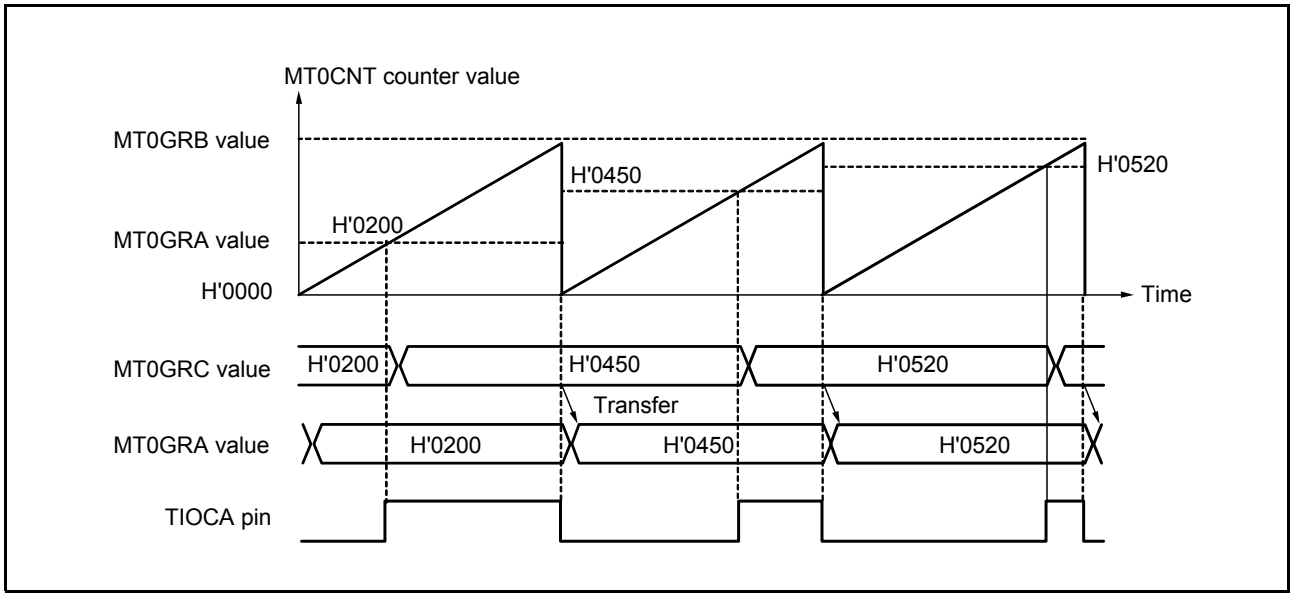


Figure 16.20 Example of Buffer Operation When MT0CNT Counter Clearing is Selected for MT0GRC to MT0GRA Transfer Timing

16.3.4 Cascaded Operation

In cascaded operation, two 16-bit counters for different channels are used together as a 32-bit counter.
This function works by using overflow/underflow of the MT2CNT counter for counting channel 1 counter clock as set in the TPSC bits in MTCR.
Underflow occurs only when the lower 16-bit MTCNT is in phase-counting mode.
Table 16.62 shows the register combinations used in cascaded operation.

Note: When phase counting mode is set for channel 1, the counter clock setting is invalid and the counters operates independently in phase counting mode.

Table 16.62 Cascaded Combinations

Combination	Upper 16 Bits	Lower 16 Bits
Channels 1 and 2	MT1CNT	MT2CNT

For simultaneous input captures of counters MT1CNT and MT2CNT during cascaded operation, additional input capture input pins can be specified by the input capture control register (TICCR). The edge used for input capture conditions is detected on the signal that has ORed the input pin and the additional input pin. For details, see section 16.3.4, (4) Cascaded Operation Example (c). For input capture in cascade connection, refer to section section 16.6.20, Simultaneous Capture of Counters MT1CNT and MT2CNT in Cascade Connection.
Table 16.63 shows the MT1ICCR setting and input capture input pins.

Table 16.63 MT1ICCR Setting and Input Capture Input Pins

Target Input Capture	MT1ICCR Setting	Input Capture Input Pins
Input capture from MT1CNT to MT1GRA	I2AE bit = 0 (initial value)	TIOC1A
	I2AE bit = 1	TIOC1A, TIOC2A
Input capture from MT1CNT to MT1GRB	I2BE bit = 0 (initial value)	TIOC1B
	I2BE bit = 1	TIOC1B, TIOC2B
Input capture from MT2CNT to MT2GRA	I1AE bit = 0 (initial value)	TIOC2A
	I1AE bit = 1	TIOC2A, TIOC1A
Input capture from MT2CNT to MT2GRB	I1BE bit = 0 (initial value)	TIOC2B
	I1BE bit = 1	TIOC2B, TIOC1B

(1) Example of Cascaded Operation Setting Procedure

Figure 16.21 shows an example of the cascaded operation setting procedure.

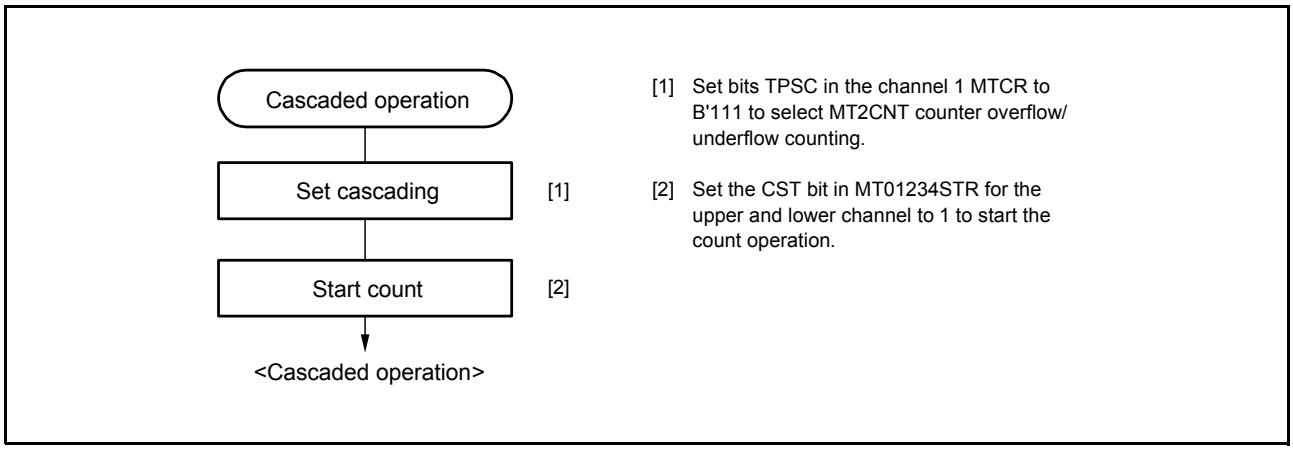


Figure 16.21 Cascaded Operation Setting Procedure

(2) Cascaded Operation Example (a)

Figure 16.22 illustrates the operation where the relevant register has been set so that the MT1CNT counter increments/ decrements when the MC2CNT counter overflow/underflow occurs and phase counting mode has been designated for channel 2. The MT1CNT counter is incremented by MT2CNT counter overflow and decremented by MT2CNT counter underflow.

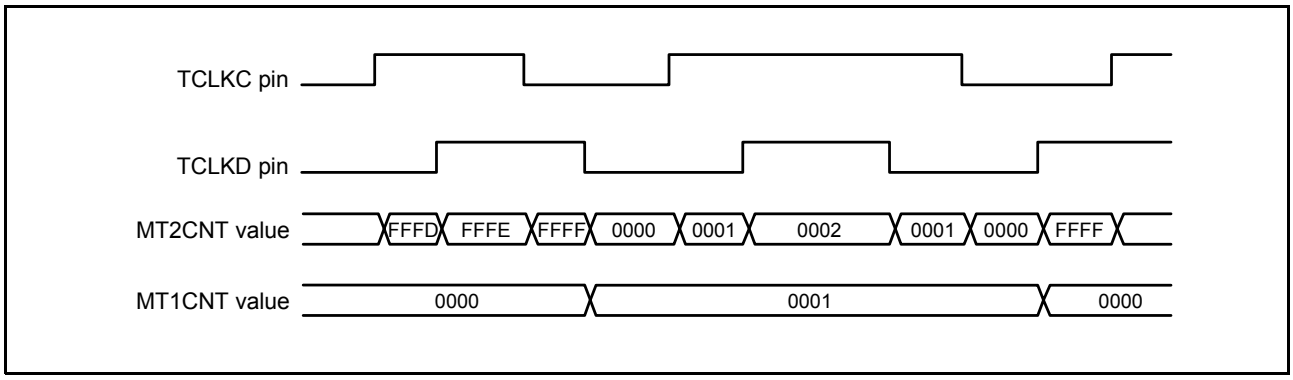


Figure 16.22 Cascaded Operation Example (a)

(3) Cascaded Operation Example (b)

Figure 16.23 illustrates the operation when the MT1CNT and MT2CNT counters have been cascaded and the I2AE bit in MT1ICCR has been set to 1 to include the TIOC2A pin in the MT1GRA input capture conditions. In this example, the IOA bits in MT1IOCR0 has selected the TIOC1A rising edge for the input capture timing, while MT2IOCR0 has selected the TIOC2A rising edge for the input capture timing.

Under these conditions, the rising edges of both TIOC1A and TIOC2A pins are used for the MT1GRA input capture condition. For the MT2GRA input capture condition, the TIOC2A pin rising edge is used.

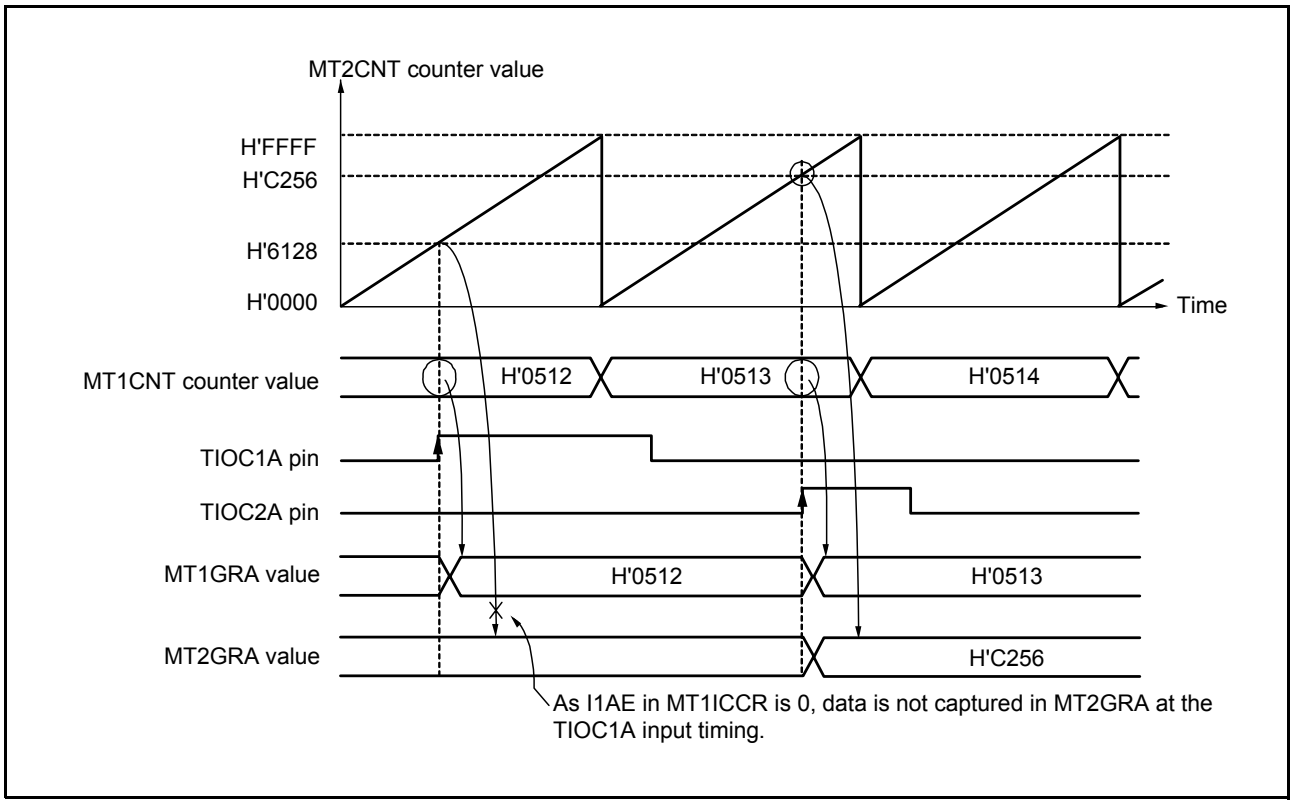


Figure 16.23 Cascaded Operation Example (b)

(4) Cascaded Operation Example (c)

Figure 16.24 illustrates the operation when the MT1CNT and MT2CNT counters have been cascaded and the I2AE and I1AE bits in MT1ICCR have been set to 1 to include the TIOC2A and TIOC1A pins in the MT1GRA and MT2GRA input capture conditions, respectively. In this example, the IOA bits in both MT1IOCR0 and MT2IOCR0 have selected both the rising and falling edges for the input capture timing. Under these conditions, the ORed result of TIOC1A and TIOC2A pin input is used for the MT1GRA and MT2GRA input capture conditions.

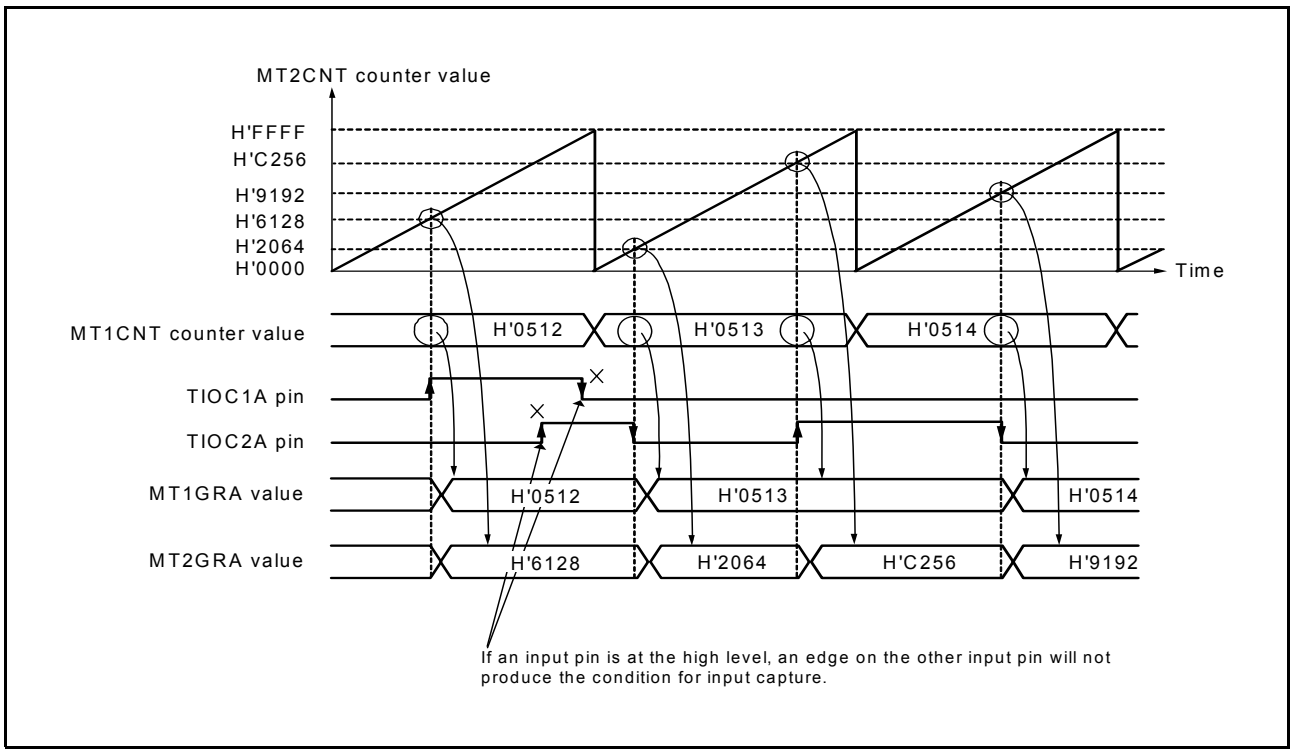


Figure 16.24 Cascaded Operation Example (c)

(5) Cascaded Operation Example (d)

Figure 16.25 illustrates the operation when the MT1CNT and MT2CNT counters have been cascaded and the I2AE bit in MT1ICCR has been set to 1 to include the TIOC2A pin in the MT1GRA input capture conditions. In this example, the IOA bits in MT1IOCR0 has selected MT0GRA compare match or input capture occurrence for the input capture timing, while the IOA bits in MT2IOCR0 has selected the TIOC2A rising edge for the input capture timing.

Under these conditions, as MT1IOCR0 has selected MT0GRA compare match or input capture occurrence for the input capture timing, the TIOC2A pin edge is not used for MT1GRA input capture condition even when the I2AE bit in MT1ICCR has been set to 1.

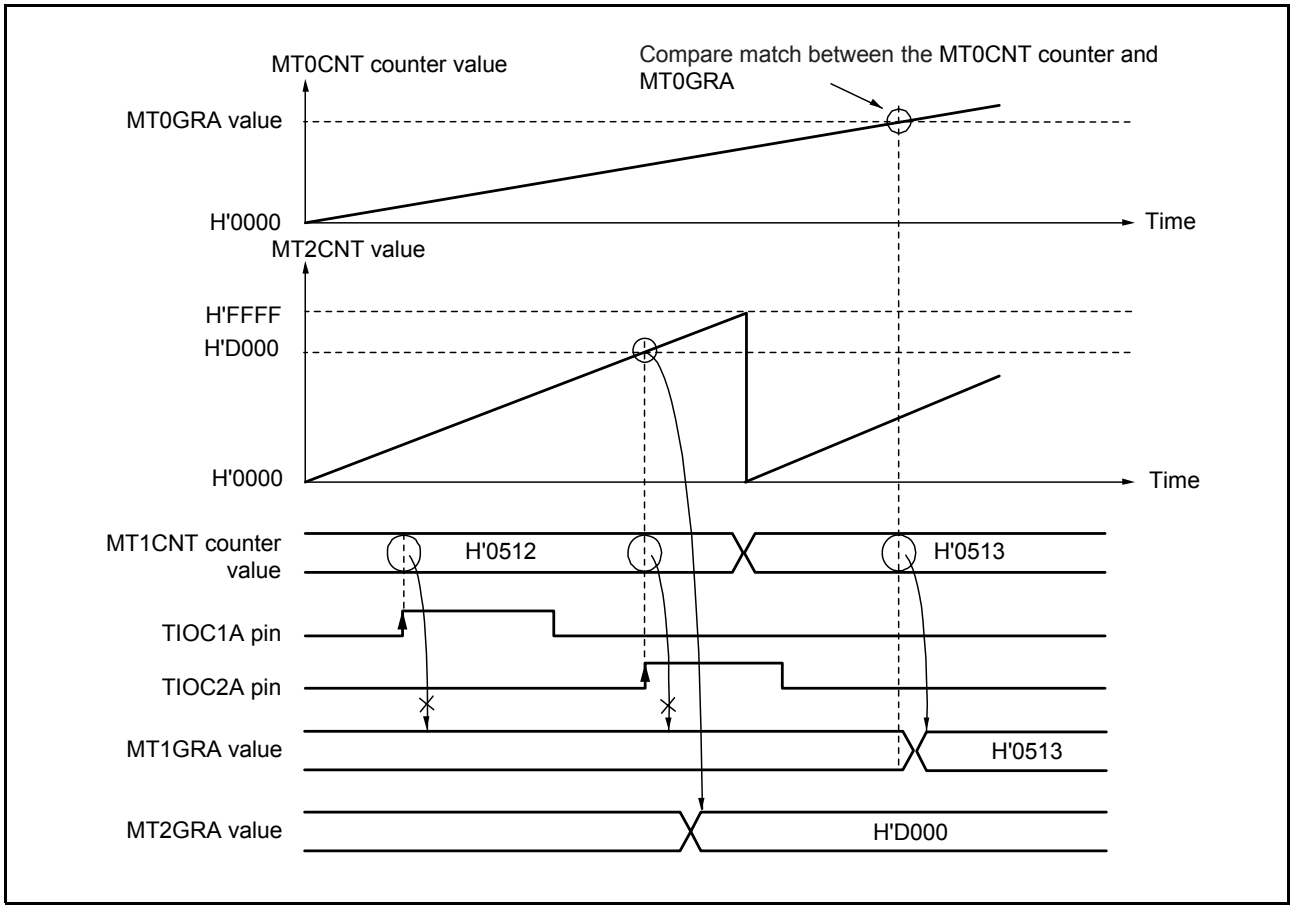


Figure 16.25 Cascaded Operation Example (d)

16.3.5 PWM Mode

In PWM mode, PWM waveforms are output from the output pins. The output level can be selected as 0, 1, or toggle output in response to a compare match of each MTGR.

MTGR register settings can be used to output a PWM waveform in the range of 0% to 100% duty.

Designating MTGR compare match as the counter-clearing source enables the period to be set in that register. All channels can be designated for PWM mode independently. Synchronous operation is also possible.

There are two PWM modes, as described below.

(a) PWM mode 1

PWM output is generated from the TIOCA and TIOCC pins by pairing MTGRA with MTGRB and MTGRC with MTGRD. The output specified by bits IOA0 to IOA3 and IOC0 to IOC3 in MTIOCR is output from the TIOCA and TIOCC pins at compare matches A and C, and the output specified by bits IOB0 to IOB3 and IOD0 to IOD3 in MTIOCR is output at compare matches B and D. The initial output value is the value set in MTGRA or MTGRC. If the set values of paired MTGRs are identical, the output value does not change when a compare match occurs.

In PWM mode 1, a maximum 12-phase PWM output is possible.

(b) PWM mode 2

PWM output is generated using one MTGR as the cycle register and the others as duty registers. The output specified in MTIOCR is performed by means of compare matches. Upon counter clearing by a synchronization register compare match, the output value of each pin is the initial value set in MTIOCR. If the set values of the cycle and duty registers are identical, the output value does not change when a compare match occurs.

In PWM mode 2, a maximum 8-phase PWM output is possible in combination use with synchronous operation.

Table 16.64 shows the correspondence between PWM output pins and registers.

Table 16.64 PWM Output Registers and Output Pins

Channel	Registers	Output Pins	
		PWM Mode 1	PWM Mode 2
0	MT0GRA	TIOC0A	TIOC0A
	MT0GRB		TIOC0B
	MT0GRC	TIOC0C	TIOC0C
	MT0GRD		TIOC0D*
1	MT1GRA	TIOC1A*	TIOC1A*
	MT1GRB		TIOC1B
2	MT2GRA	TIOC2A*	TIOC2A*
	MT2GRB		TIOC2B*
3	MT3GRA	TIOC3A	Cannot be set
	MT3GRB		
	MT3GRC	TIOC3C	
	MT3GRD		
4	MT4GRA	TIOC4A	
	MT4GRB		
	MT4GRC	TIOC4C	
	MT4GRD		
6	MT6GRA	TIOC6A*	
	MT6GRB		
	MT6GRC	TIOC6C*	
	MT6GRD		
7	MT7GRA	TIOC7A	
	MT7GRB		
	MT7GRC	TIOC7C	
	MT7GRD		

Notes: * This function cannot be used in the SH72A0 Group.
- In PWM mode 2, the MTGR register for setting the period is not used for PWM output.

(1) Example of PWM Mode Setting Procedure

Figure 16.26 shows an example of the PWM mode setting procedure.

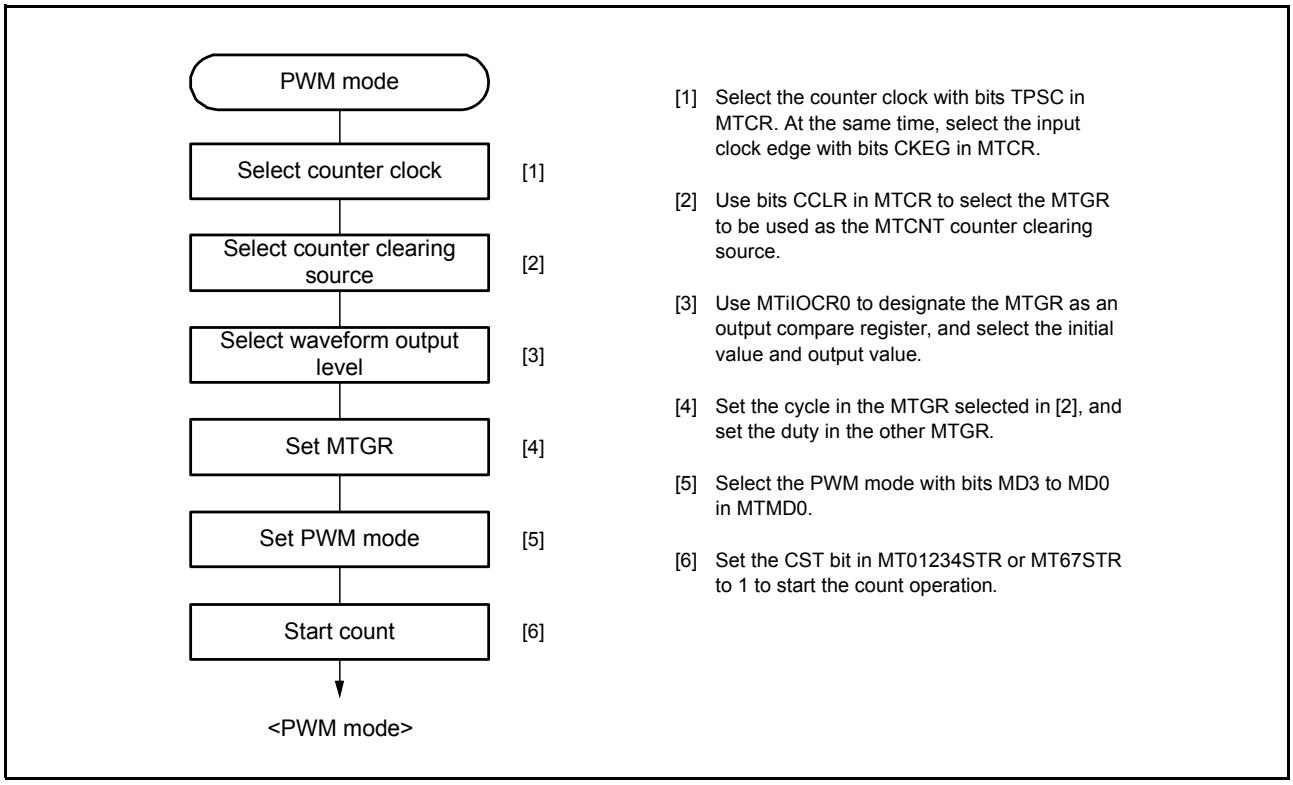


Figure 16.26 Example of PWM Mode Setting Procedure

(2) Examples of PWM Mode Operation

Figure 16.27 shows an example of the PWM mode 1 operation.

In this example, MTGRA compare match is set as the MTCNT counter clearing source, 0 is set for the MTGRA initial output value and output value, and 1 is set as the MTGRB output value.

In this case, the value set in MTGRA is used as the period, and the values set in the MTGRB registers are used as the duty levels.

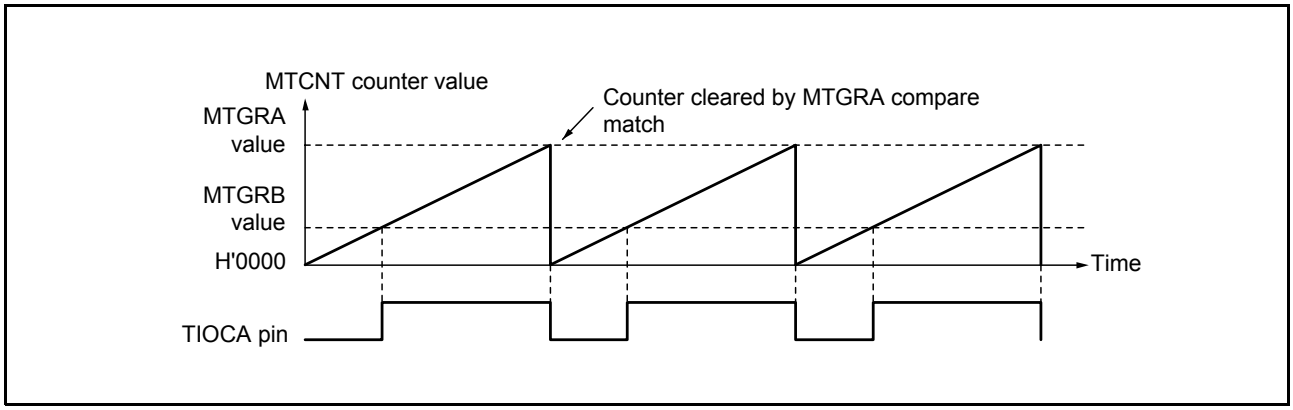


Figure 16.27 Examples of PWM Mode 1 Operation

Figure 16.28 shows an example of the PWM mode 2 operation.

In this example, synchronous operation is designated for channels 0 and 1, MT1GRB compare match is set as the MTCNT counter clearing source, and 0 is set for the initial output value and 1 for the output value of other registers (MT0GRA to MT0GRD, MT1GRA), outputting a 5-phase PWM waveform.

In this case, the value set in MT1GRB is used as the period, and the values set in other MTGRs are used as the duty levels.

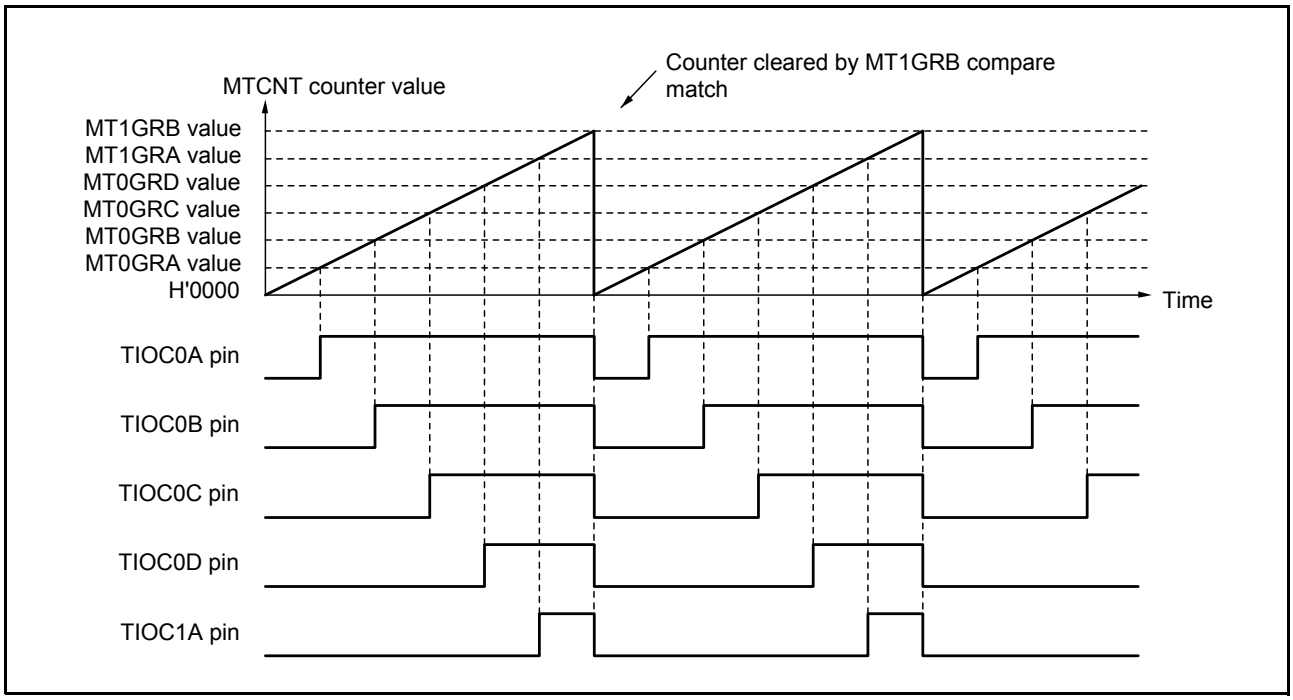


Figure 16.28 Example of PWM Mode 2 Operation

Figure 16.29 shows examples of PWM waveform output with 0% duty and 100% duty in PWM mode.

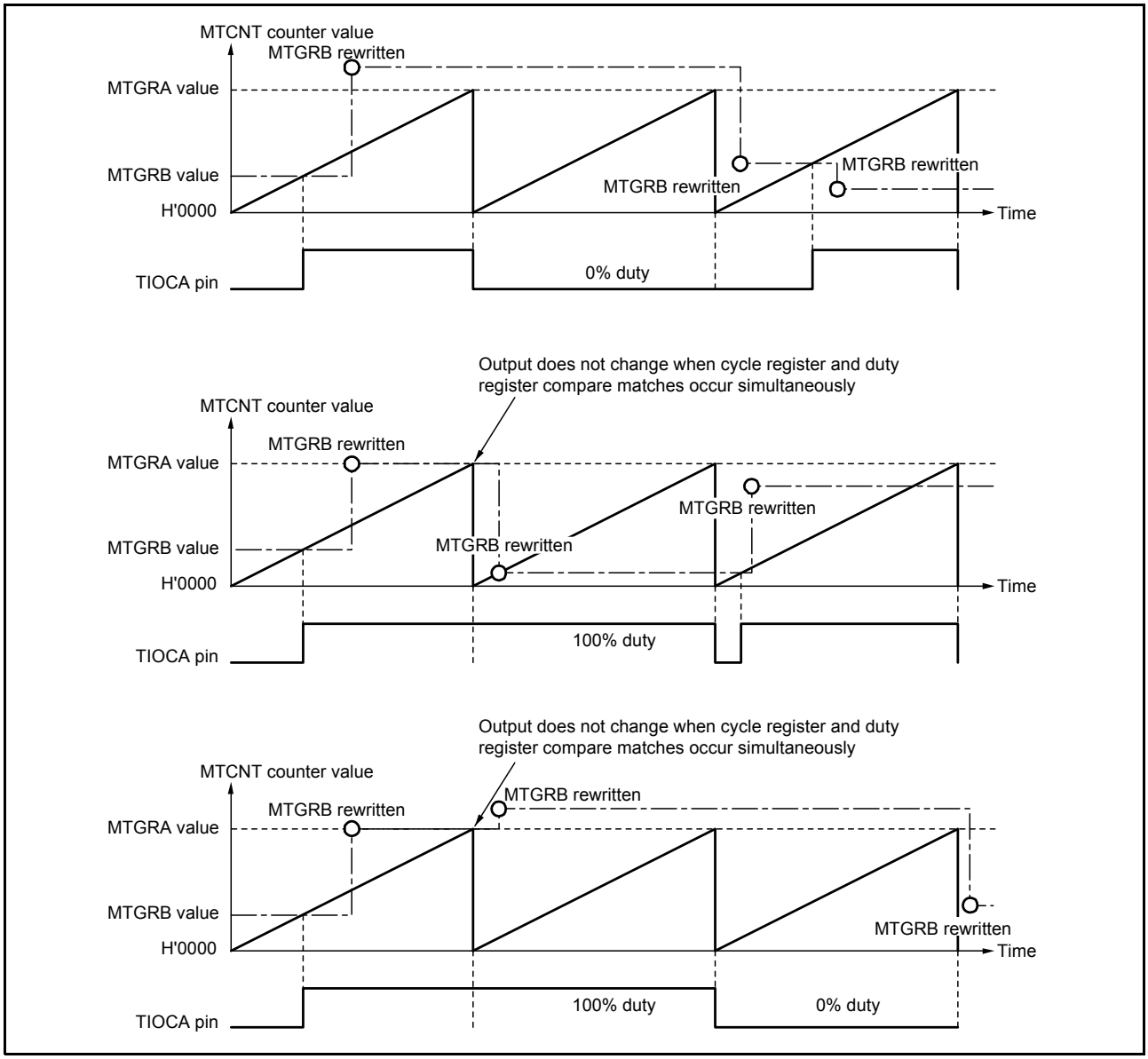


Figure 16.29 Examples of PWM Mode Operation

16.3.6 Phase Counting Mode

In phase counting mode, the phase difference between two external clock inputs is detected and the MTCNT counters incremented/decremented accordingly. This mode can be set for channels 1 and 2.

When phase counting mode is set, external clocks are selected as the counter input clock and the MTCNT counter operates as an up/down-counter regardless of the setting of bits TPSC and CKEG in MTCR. However, the functions of the CCLR bits in MTCR, and MTIOCR, MTIEN, and MTGR, are valid, and input capture/compare match and interrupt functions can be used.

This can be used for two-phase encoder pulse input.

If overflow occurs when the MTCNT counter is counting up, the TCFV flag in MTSR is set; if underflow occurs when MTCNT is counting down, the TCFU flag is set.

The TCFD bit in MTSR is the count direction flag. Reading the TCFD flag reveals whether the MTCNT counter is counting up or down.

Table 16.65 shows the correspondence between external clock pins and channels.

Table 16.65 Phase Counting Mode Clock Input Pins

Channels	External Clock Pins	
	A-Phase	B-Phase
When channel 1 is set to phase counting mode	TCLKA	TCLKB
When channel 2 is set to phase counting mode	TCLKC	TCLKD

(1) Example of Phase Counting Mode Setting Procedure

Figure 16.30 shows an example of the phase counting mode setting procedure.

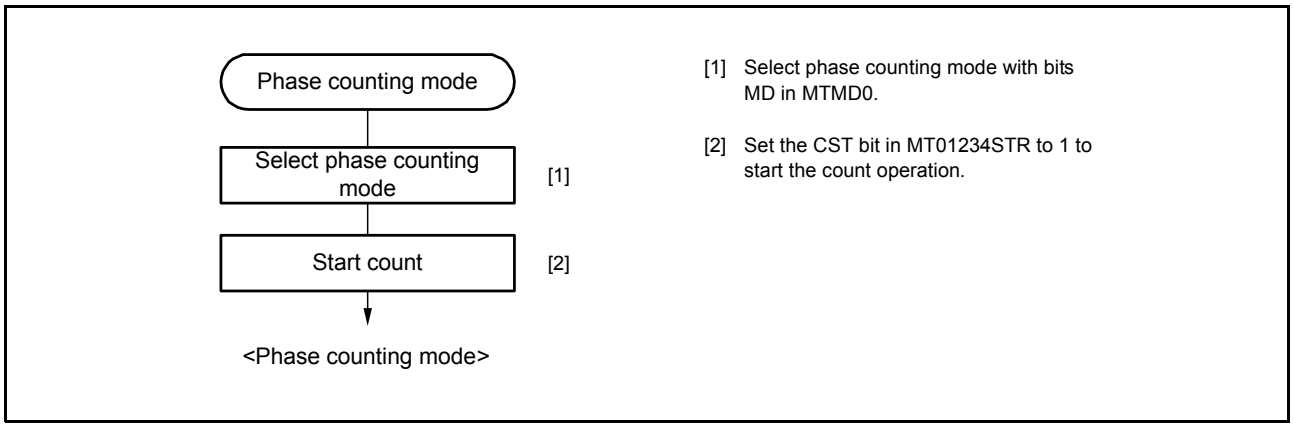


Figure 16.30 Example of Phase Counting Mode Setting Procedure

(2) Examples of Phase Counting Mode Operation

In phase counting mode, the MTCNT counter 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 16.31 shows an example of phase counting mode 1 operation, and table 16.66 summarizes the MTCNT counter up/down-count conditions.

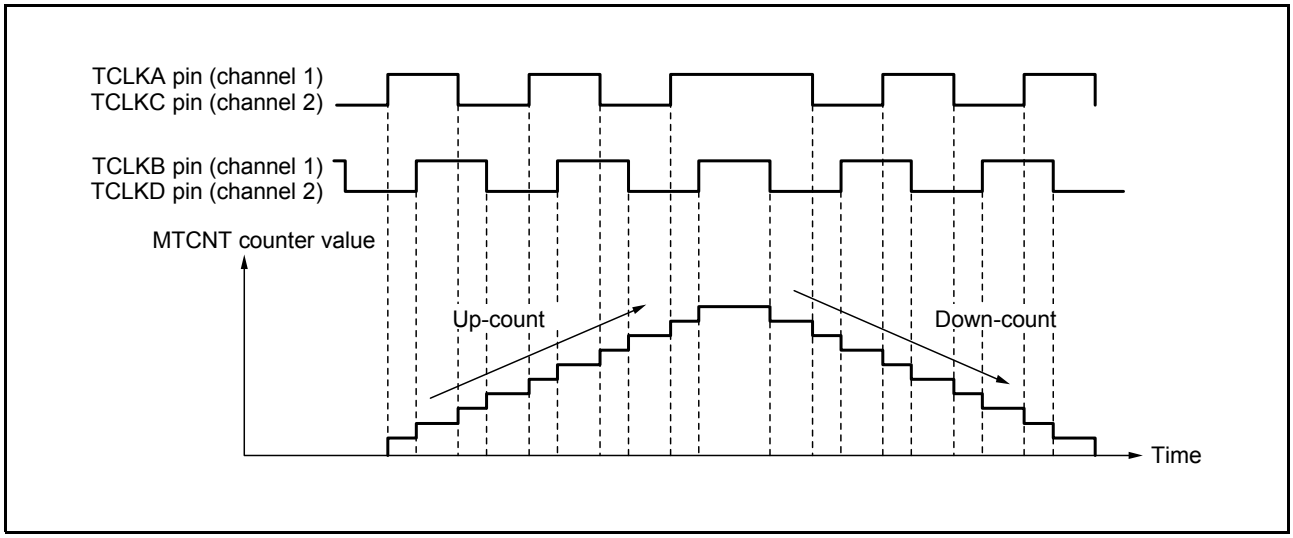


Figure 16.31 Example of Phase Counting Mode 1 Operation

Table 16.66 Up/Down-Count Conditions in Phase Counting Mode 1

TCLKA Pin (Channel 1)	TCLKB Pin (Channel 1)	Operation
TCLKC Pin (Channel 2)*	TCLKD Pin (Channel 2)*	
High level	Rising edge	Up-count
Low level	Falling edge	
Rising edge	Low level	
Falling edge	High level	
High level	Falling edge	Down-count
Low level	Rising edge	
Rising edge	High level	
Falling edge	Low level	

Note: * This function cannot be used in the SH72A0 Group.

(b) Phase counting mode 2

Figure 16.32 shows an example of phase counting mode 2 operation, and table 16.67 summarizes the MTCNT counter up/down-count conditions.

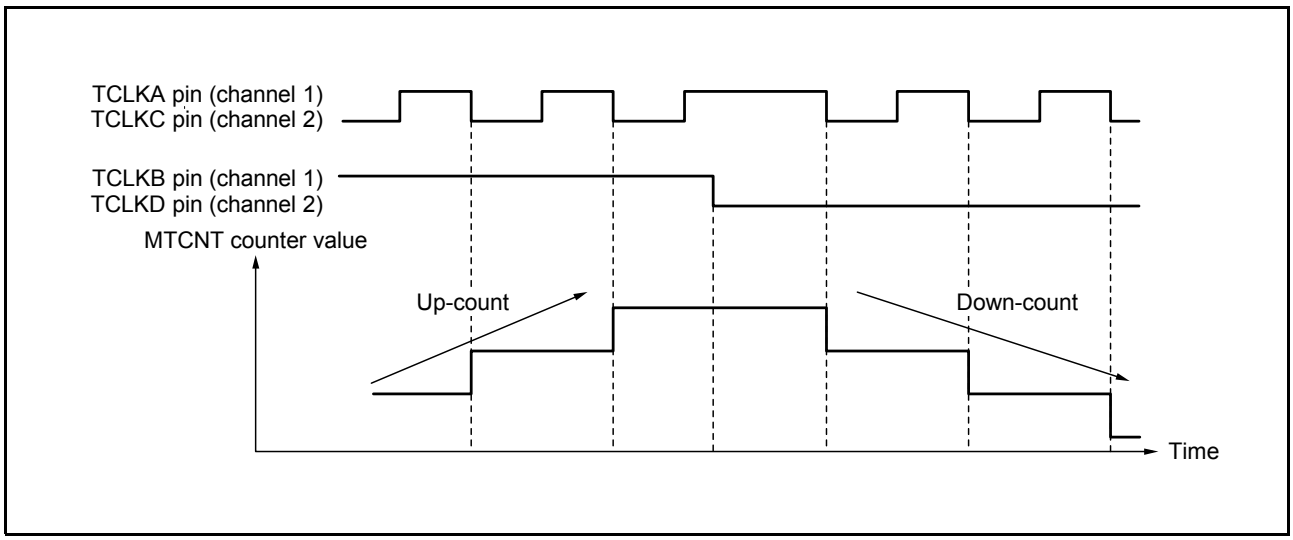


Figure 16.32 Example of Phase Counting Mode 2 Operation

Table 16.67 Up/Down-Count Conditions in Phase Counting Mode 2

TCLKA Pin (Channel 1) / TCLKC Pin (Channel 2)*	TCLKB Pin (Channel 1) / TCLKD Pin (Channel 2)*	Operation
High level	Rising edge	Don't care
Low level	Falling edge	Don't care
Rising edge	Low level	Don't care
Falling edge	High level	Up-count
High level	Falling edge	Don't care
Low level	Rising edge	Don't care
Rising edge	High level	Don't care
Falling edge	Low level	Down-count

Note: * This function cannot be used in the SH72A0 Group.

(c) Phase counting mode 3

Figure 16.33 shows an example of phase counting mode 3 operation, and table 16.68 summarizes the MTCNT counter up/down-count conditions.

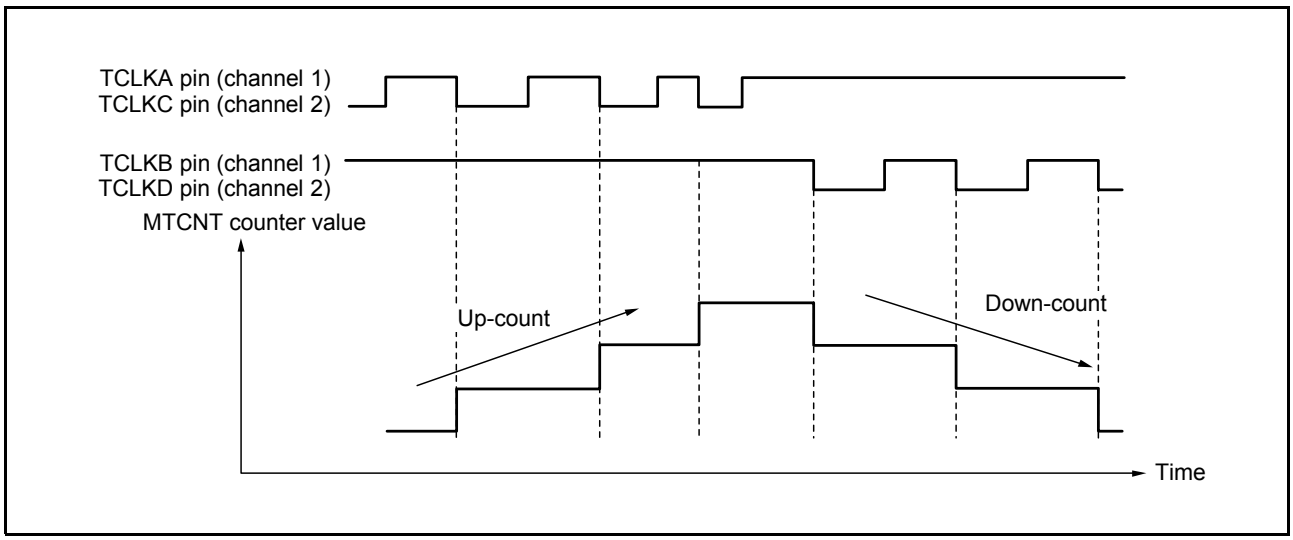


Figure 16.33 Example of Phase Counting Mode 3 Operation

Table 16.68 Up/Down-Count Conditions in Phase Counting Mode 3

TCLKA Pin (Channel 1) TCLKC Pin (Channel 2)*	TCLKB Pin (Channel 1) TCLKD Pin (Channel 2)*	Operation
High level	Rising edge	Don't care
Low level	Falling edge	Don't care
Rising edge	Low level	Don't care
Falling edge	High level	Up-count
High level	Falling edge	Down-count
Low level	Rising edge	Don't care
Rising edge	High level	Don't care
Falling edge	Low level	Don't care

Note: * This function cannot be used in the SH72A0 Group.

(d) Phase counting mode 4

Figure 16.34 shows an example of phase counting mode 4 operation, and table 16.69 summarizes the MTCNT counter up/down-count conditions.

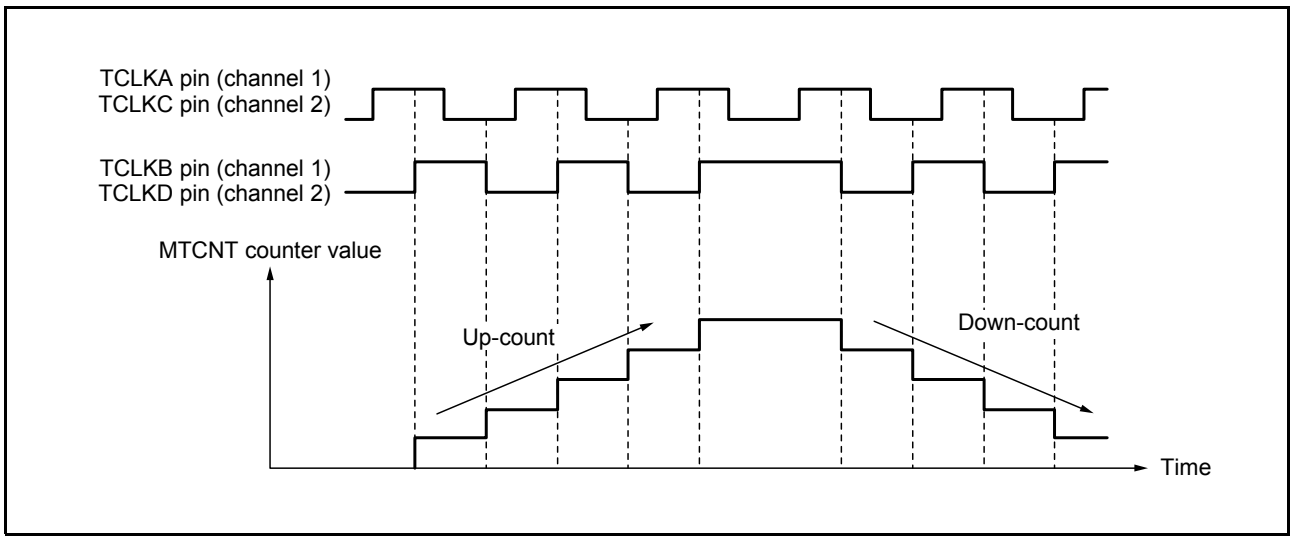


Figure 16.34 Example of Phase Counting Mode 4 Operation

Table 16.69 Up/Down-Count Conditions in Phase Counting Mode 4

TCLKA Pin (Channel 1)	TCLKB Pin (Channel 1)	Operation
TCLKC Pin (Channel 2)*	TCLKD Pin (Channel 2)*	
High level	Rising edge	Up-count
Low level	Falling edge	
Rising edge	Low level	Don't care
Falling edge	High level	
High level	Falling edge	Down-count
Low level	Rising edge	
Rising edge	High level	Don't care
Falling edge	Low level	

Note: * This function cannot be used in the SH72A0 Group.

(3) Phase Counting Mode Application Example

Figure 16.35 shows an example in which channel 1 is in phase counting mode, and channel 1 is coupled with channel 0 to input servo motor 2-phase encoder panel in order to detect position or speed.

Channel 1 is set to phase counting mode 1, and the encoder pulse A-phase and B-phase are input to the TCLKA and TCLKB pins.

Channel 0 operates with the MTCNT counter clearing by MT0GRC compare match; MT0GRA and MT0GRC are used for the compare match function and are set with the speed control period and position control period. MT0GRB is used for input capture, with MT0GRB and MT0GRD operating in buffer mode. The channel 1 counter input clock is designated as the MT0GRB input capture source, and the pulse widths of 2-phase encoder 4-multiplication pulses are detected.

MT1GRA and MT1GRB for channel 1 are designated for input capture, and channel-0 MT0GRA and MT0GRC compare matches are selected as the input capture source and store the up/down-counter values for the control periods. This procedure enables the accurate detection of position and speed.

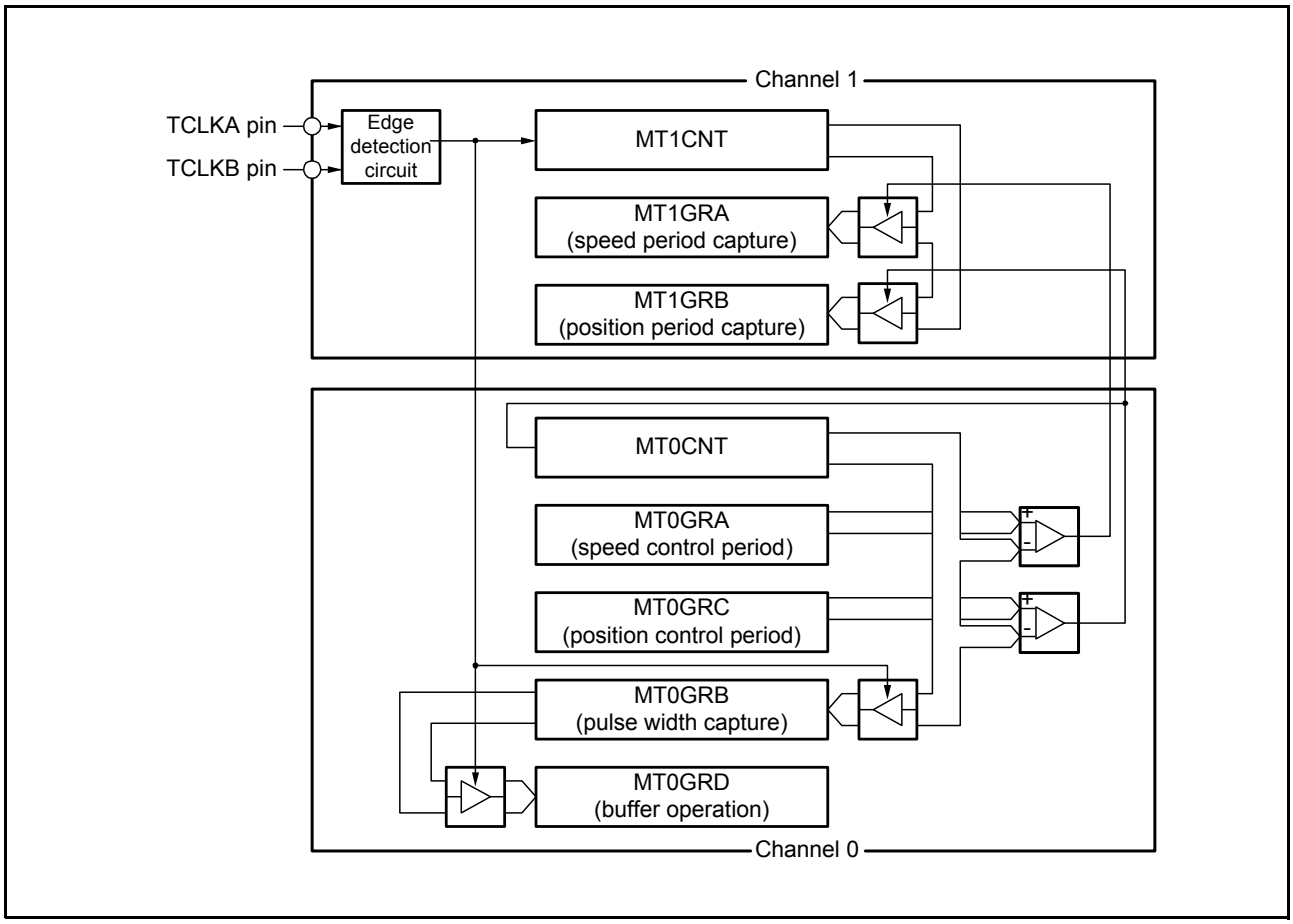


Figure 16.35 Phase Counting Mode Application Example

16.3.7 Reset-Synchronized PWM mode

In the reset-synchronized PWM mode, positive and negative PWM waveforms that share a common wave transition point can be output by combining channels 3 and 4 or channels 6 and 7. Three-phase PWM waveforms can be obtained for each channel combination; that is, six phases in total.

When set for reset-synchronized PWM mode, the TIOC3B, TIOC3D, TIOC4A, TIOC4C, TIOC4B, TIOC4D, TIOC6B, TIOC6D, TIOC7A, TIOC7C, TIOC7B, and TIOC7D pins function as PWM output pins and the MT3CNT and MT6CNT counters function as up-counters.

Table 16.70 shows the PWM output pins used. Table 16.71 shows the settings of the registers.

Table 16.70 Output Pins for Reset-Synchronized PWM Mode

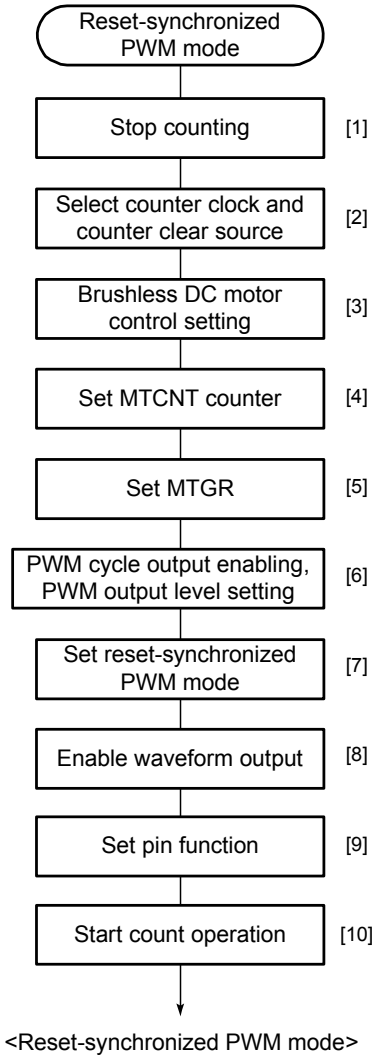
Channel	Output Pin	Description
3	TIOC3B	PWM output pin 1
	TIOC3D	PWM output pin 1' (negative-phase waveform of PWM output 1)
4	TIOC4A	PWM output pin 2
	TIOC4C	PWM output pin 2' (negative-phase waveform of PWM output 2)
	TIOC4B	PWM output pin 3
	TIOC4D	PWM output pin 3' (negative-phase waveform of PWM output 3)
6	TIOC6B	PWM output pin 4
	TIOC6D	PWM output pin 4' (negative-phase waveform of PWM output 4)
7	TIOC7A	PWM output pin 5
	TIOC7C	PWM output pin 5' (negative-phase waveform of PWM output 5)
	TIOC7B	PWM output pin 6
	TIOC7D	PWM output pin 6' (negative-phase waveform of PWM output 6)

Table 16.71 Register Settings for Reset-Synchronized PWM Mode

Register	Description of Setting
MT3CNT	Initial setting of H'0000
MT4CNT	Initial setting of H'0000
MT3GRA	Set count cycle for MT3CNT
MT3GRB	Sets the transition timing for PWM waveform output by the TIOC3B and TIOC3D pins
MT4GRA	Sets the transition timing for PWM waveform output by the TIOC4A and TIOC4C pins
MT4GRB	Sets the transition timing for PWM waveform output by the TIOC4B and TIOC4D pins
MT6CNT	Initial setting of H'0000
MT7CNT	Initial setting of H'0000
MT6GRA	Set count cycle for MT6CNT
MT6GRB	Sets the transition timing for PWM waveform output by the TIOC6B and TIOC6D pins
MT7GRA	Sets the transition timing for PWM waveform output by the TIOC7A and TIOC7C pins
MT7GRB	Sets the transition timing for PWM waveform output by the TIOC7B and TIOC7D pins

(1) Procedure for Setting Reset-Synchronized PWM Mode

Figure 16.36 shows an example of procedure for selecting the reset synchronized PWM mode.



[1] Clear the CST3 and CST4 bits in the MT01234 start register (MT01234STR) to 0 to halt the counting of the MT counter (MTCNT). The reset-synchronized PWM mode must be set up while counters MT3CNT and MT4CNT are halted.

[2] Set bits TPSC and CKEG in the MT3 control register (MT3CR) to select the counter clock and clock edge for channel 3. Set bit CCLR to select MTGRA compare-match as a counter clear source.

[3] When performing brushless DC motor control, set bit BDC in the MT34 gate control register (MT34GCR) and set the feedback signal input source and output chopping or gate signal direct output.

[4] Reset counters MT3CNT and MT4CNT to H'0000.

[5] MT3GRA is the cycle register. Set the waveform period value in MT3GRA. Set the transition timing of the PWM output waveforms in MT3GRB, MT4GRA, and MT4GRB. Choose timings within the compare-match range of the MT3CNT counter.
$X \leq$ MT3GRA (X: transition timings).

[6] Select enabling/disabling of toggle output synchronized with the PWM cycle using bit PSYE in the MT34 output control register 0 (MT34OCR0), and set the PWM output level with bits OLSP and OLSN. When specifying the PWM output level by using MT34OLB as a buffer for MT34OCR1 and MT34OLB, see figure 16.4, PWM Output Level Setting Procedure in Buffer Operation.

[7] Set bits MD in the MT3 mode register 0 (MT3MD0) to B'1000 to select the reset-synchronized PWM mode. Do not set to MT4MD0.

[8] Set the enabling/disabling of the PWM waveform output pin in the MT34 output enable register (MT34OEN).

[9] Set the port function select register and the port function enable register.

[10] Set the CST3 bit in the MT01234STR to 1 to start the count operation.

Note: If X is equal to MT3GRA, i.e., cycle is equal to duty, the output waveform toggles when MT3CNT matches MT3GRA.

Figure 16.36 Procedure for Selecting Reset-Synchronized PWM Mode

(2) Reset-Synchronized PWM Mode Operation

Figure 16.37 shows an example of operation in the reset-synchronized PWM mode.

MT3CNT and MT4CNT (MT6CNT and MT7CNT) operate as up-counters. The counter is cleared when an MT3CNT (MT6CNT) and MT3GRA (MT6GRA) compare-match occurs, and then begins incrementing from H'0000. The PWM output pin outputs toggles with each occurrence of an MT3GRB (MT6GRB), MT4GRA (MT7GRA), MT4GRB (MT7GRB) compare-match, and upon counter clears.

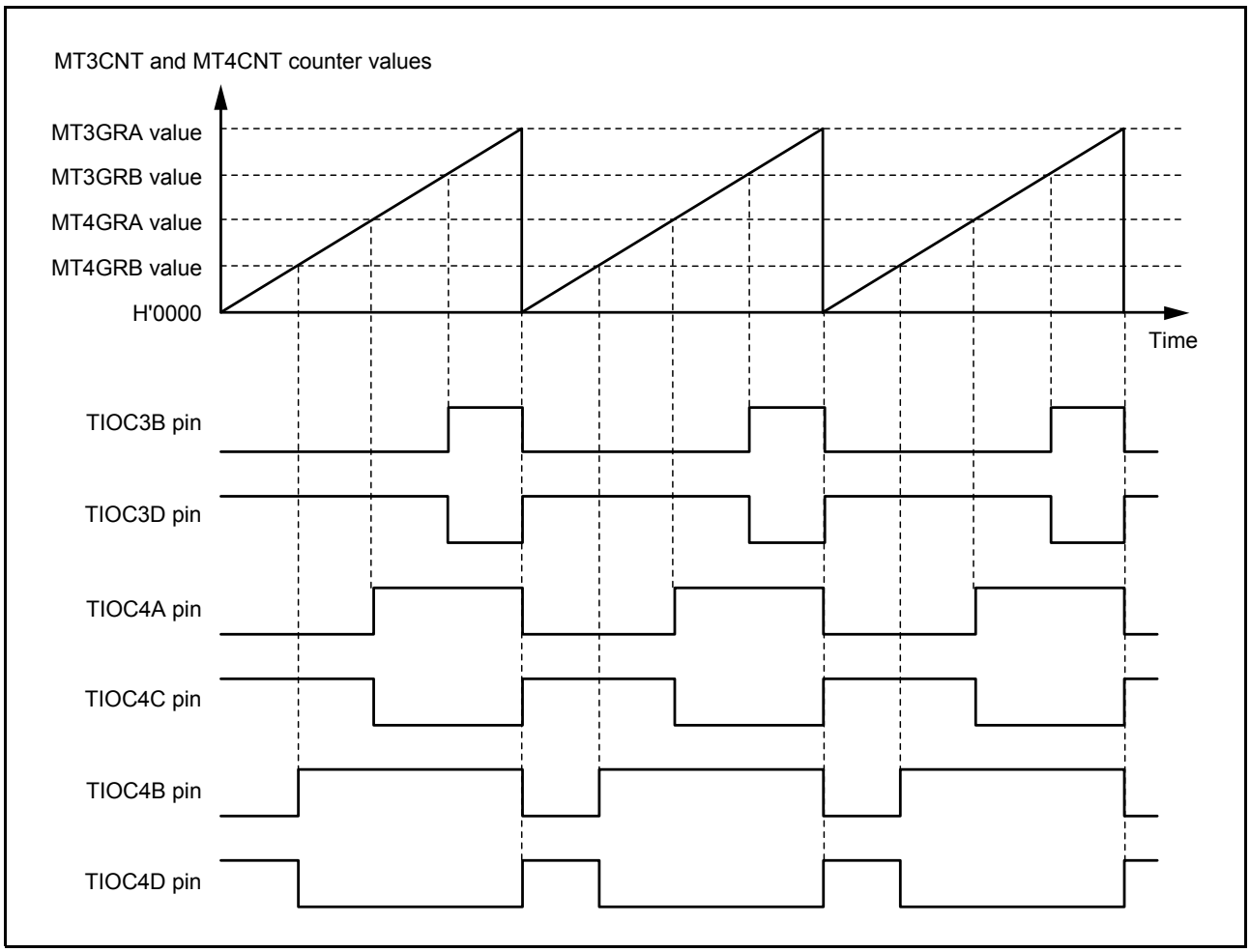


Figure 16.37 Example of Reset-Synchronized PWM Mode Operation (Channels 3 and 4) (When OLSN = 1 and OLSP = 1 in MT34OCR0 (MT67OCR0))

16.3.8 Complementary PWM Mode

In the complementary PWM mode, combining channels 3 and 4 or channels 6 and 7 can output non-overlapping positive and negative PWM waveforms. Three-phase PWM waveforms can be obtained for each channel combination; that is, six phases in total. PWM waveforms without non-overlapping interval are also available.

In complementary PWM mode, TIOC3B, TIOC3D, TIOC4A, TIOC4B, TIOC4C, TIOC4D, TIOC6B, TIOC6D, TIOC7A, TIOC7B, TIOC7C, and TIOC7D pins function as PWM output pins, the TIOC3A and TIOC6A pins can be set for toggle output synchronized with the PWM period.

Counters MT3CNT, MT4CNT, MT6CNT, and MT7CNT function as up/down counters.

Table 16.72 shows the PWM output pins used. Table 16.73 and table 16.74 show the settings of the registers.

A function to directly cut off the PWM output by using an external signal is supported as a port function (POE).

Table 16.72 Output Pins for Complementary PWM Mode

Channel	Output Pin	Description
3	TIOC3A	Toggle output synchronized with PWM period (or I/O port)
	TIOC3B	PWM output pin 1
	TIOC3C	I/O port*1
	TIOC3D	PWM output pin 1' (non-overlapping negative-phase waveform of PWM output 1; PWM output without non-overlapping interval is also available)
4	TIOC4A	PWM output pin 2
	TIOC4C	PWM output pin 2' (non-overlapping negative-phase waveform of PWM output 2; PWM output without non-overlapping interval is also available)
	TIOC4B	PWM output pin 3
	TIOC4D	PWM output pin 3' (non-overlapping negative-phase waveform of PWM output 3; PWM output without non-overlapping interval is also available)
6	TIOC6A*2	Toggle output synchronized with PWM period (or I/O port)
	TIOC6B	PWM output pin 4
	TIOC6C*2	I/O port*1
	TIOC6D	PWM output pin 4' (non-overlapping negative-phase waveform of PWM output 4; PWM output without non-overlapping interval is also available)
7	TIOC7A	PWM output pin 5
	TIOC7C	PWM output pin 5' (non-overlapping negative-phase waveform of PWM output 5; PWM output without non-overlapping interval is also available)
	TIOC7B	PWM output pin 6
	TIOC7D	PWM output pin 6' (non-overlapping negative-phase waveform of PWM output 6; PWM output without non-overlapping interval is also available)

Notes: 1. Avoid setting the TIOC3C and TIOC6C pins as timer I/O pins in the complementary PWM mode.
2. This pin cannot be used in the SH72A0 Group.

Table 16.73 Register Settings for Complementary PWM Mode (1)

Channel	Counter/ Register	Description	Read/Write from CPU
3	MT3CNT	Start of up-count from value set in dead time register	Maskable by MT34RWEN setting*1
	MT3GRA	Set MT3CNT counter upper limit value (1/2 carrier cycle + dead time)	Maskable by MT34RWEN setting*1
	MT3GRB	PWM output 1 compare register	Maskable by MT34RWEN setting*1
	MT3GRC	MT3GRA buffer register	Always readable/writable
	MT3GRD	PWM output 1/MT3GRB buffer register	Always readable/writable
	MT3GRE	MT3GRB buffer register B (when double buffer function is used)	Always readable/writable
4	MT4CNT	Up-count from H'0000	Maskable by MT34RWEN setting*1
	MT4GRA	PWM output 2 compare register	Maskable by MT34RWEN setting*1
	MT4GRB	PWM output 3 compare register	Maskable by MT34RWEN setting*1
	MT4GRC	PWM output 2/MT4GRA buffer register	Always readable/writable
	MT4GRD	PWM output 3/MT4GRB buffer register	Always readable/writable
	MT4GRE	MT4GRA buffer register B (when double buffer function is used)	Always readable/writable
	MT4GRF	MT4GRB buffer register B (when double buffer function is used)	Always readable/writable
6	MT6CNT	Start of up-count from value set in dead time register	Maskable by MT67RWEN setting*2
	MT6GRA	Set MT6CNT counter upper limit value (1/2 carrier cycle + dead time)	Maskable by MT67RWEN setting*2
	MT6GRB	PWM output 4 compare register	Maskable by MT67RWEN setting*2
	MT6GRC	MT6GRA buffer register	Always readable/writable
	MT6GRD	PWM output 4/MT6GRB buffer register	Always readable/writable
	MT6GRE	MT6GRB buffer register B (when double buffer function is used)	Always readable/writable
7	MT7CNT	Up-count from H'0000	Maskable by MT67RWEN setting*2
	MT7GRA	PWM output 5 compare register	Maskable by MT67RWEN setting*2
	MT7GRB	PWM output 6 compare register	Maskable by MT67RWEN setting*2
	MT7GRC	PWM output 5/MT7GRA buffer register	Always readable/writable
	MT7GRD	PWM output 6/MT7GRB buffer register	Always readable/writable
	MT7GRE	MT7GRA buffer register B (when double buffer function is used)	Always readable/writable
	MT7GRF	MT7GRB buffer register B (when double buffer function is used)	Always readable/writable

Notes: 1. Access can be enabled or disabled according to the setting of MT34RWEN.
2. Access can be enabled or disabled according to the setting of MT67RWEN.

Table 16.74 Register Settings for Complementary PWM Mode (2)

Channel	Counter/ Register	Description	Read/Write from CPU
MT34DTD		Set MT4CNT and MT3CNT offset value (dead time value)	Maskable by MT34RWEN setting*1
MT67DTD		Set MT7CNT and MT6CNT offset value (dead time value)	Maskable by MT67RWEN setting*2
MT34SCNT		Set MT4CNT upper limit value (1/2 carrier cycle)	Maskable by MT34RWEN setting*1
MT67SCNT		Set MT7CNT upper limit value (1/2 carrier cycle)	Maskable by MT67RWEN setting*2
MT34CB		MT34CD buffer register	Always readable/writable
MT67CB		MT67CD buffer register	Always readable/writable
Subcounter_A (MT34SCNT)		Subcounter A for dead time generation	Read-only
Subcounter_B (MT67SCNT)		Subcounter B for dead time generation	Read-only
Temporary register 1A (TEMP1A)		PWM output 1/MT3GRB temporary register A	Not readable/writable
Temporary register 1B (TEMP1B)		PWM output 1/MT3GRB temporary register B (when double buffer function is used)	Not readable/writable
Temporary register 2A (TEMP2A)		PWM output 2/MT4GRA temporary register A	Not readable/writable
Temporary register 2B (TEMP2B)		PWM output 2/MT4GRA temporary register B (when double buffer function is used)	Not readable/writable
Temporary register 3A (TEMP3A)		PWM output 3/MT4GRB temporary register A	Not readable/writable
Temporary register 3B (TEMP3B)		PWM output 3/MT4GRB temporary register B (when double buffer function is used)	Not readable/writable
Temporary register 4A (TEMP4A)		PWM output 4/MT6GRB temporary register A	Not readable/writable
Temporary register 4B (TEMP4B)		PWM output 4/MT6GRB temporary register B (when double buffer function is used)	Not readable/writable
Temporary register 5A (TEMP5A)		PWM output 5/MT7GRA temporary register A	Not readable/writable
Temporary register 5B (TEMP5B)		PWM output 5/MT7GRB temporary register B (when double buffer function is used)	Not readable/writable
Temporary register 6A (TEMP6A)		PWM output 6/MT7GRB temporary register A	Not readable/writable
Temporary register 6B (TEMP6B)		PWM output 6/MT7GRB temporary register B (when double buffer function is used)	Not readable/writable

Notes: 1. Access can be enabled or disabled according to the setting of MT34RWEN.
2. Access can be enabled or disabled according to the setting of MT67RWEN.

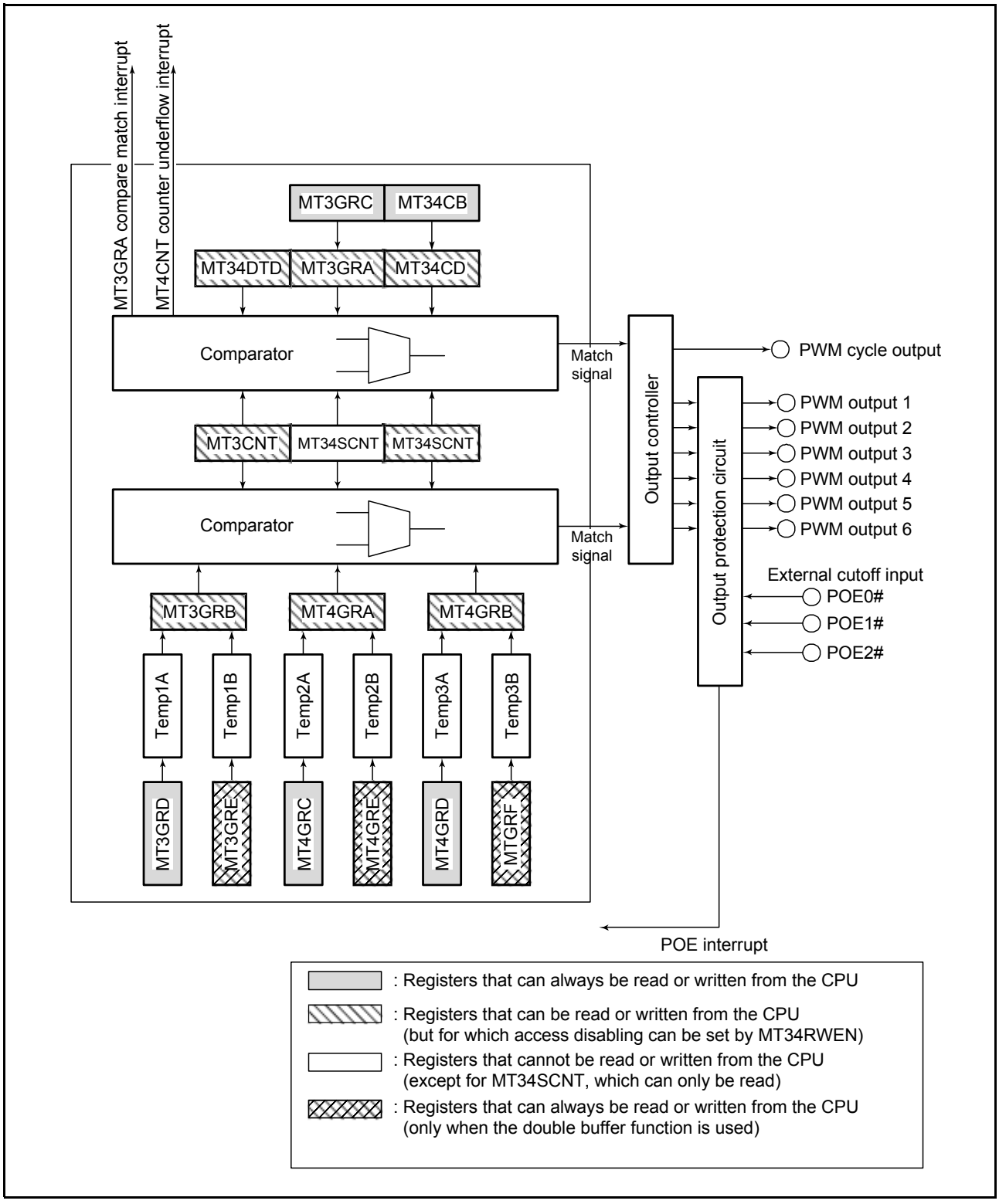
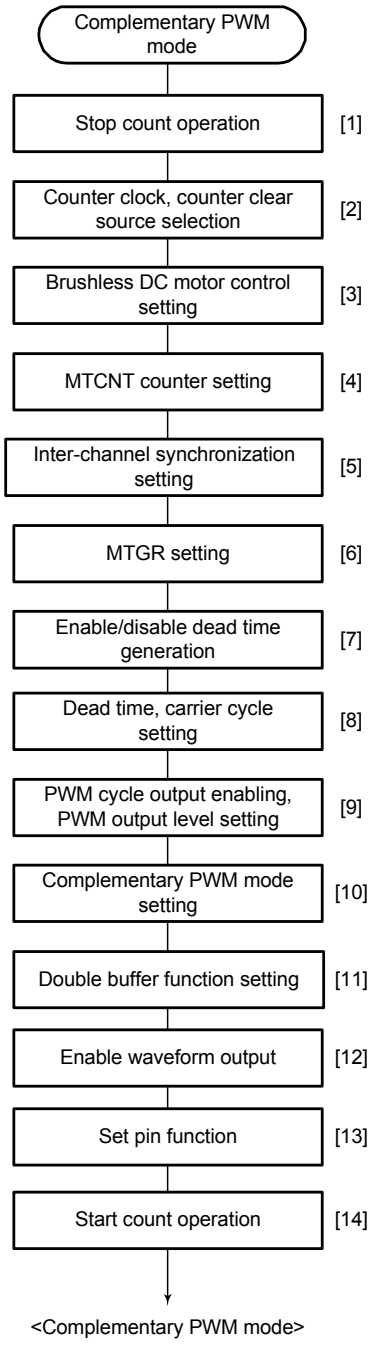


Figure 16.38 Block Diagram of Channels 3 and 4 in Complementary PWM Mode

(1) Example of Complementary PWM Mode Setting Procedure

An example of the complementary PWM mode setting procedure (channels 3 and 4) is shown in figure 16.39.



[1] Clear bits CST3 and CST4 in the MT01234 start register (MT01234STR) to 0, and halt the MT counter (MTCNT) operation. Perform complementary PWM mode setting when MT3CNT and MT4CNT are stopped.

[2] Set the same counter clock and clock edge for channels 3 and 4 with bits TPSC and CKEG in the MT control register (MTCR). Use bits CCLR to set synchronous clearing only when restarting by a synchronous clear from another channel during complementary PWM mode operation.

[3] When performing brushless DC motor control, set bit BDC in the MT34 gate control register (MT34GCR) and set the feedback signal input source and output chopping or gate signal direct output.

[4] Set the dead time in MT3CNT counter. Set the MT4CNT counter to H'0000.

[5] Set only when restarting by a synchronous clear from another channel during complementary PWM mode operation. In this case, synchronize the channel generating the synchronous clear with channels 3 and 4 using the MT01234 synchronous register (MT01234SY).

[6] Set the output PWM duty in the duty registers (MT3GRB, MT4GRA, MT4GRB) and buffer registers (MT3GRD, MT4GRC, MT4GRD). Set the same initial value in each corresponding MTGR. Only when using the double buffer function, set the output PWM duty value - 1 in the buffer registers (MT3GRD, MT4GRC, MT4GRD).

[7] This setting is necessary only when no dead time should be generated. Make appropriate settings in the MT3 dead time enable register (MT3DTEN) so that no dead time is generated.

[8] Set the dead time in the MT34 dead time register (MT34DTD), 1/2 the carrier cycle in the MT34 cycle data register (MT34CD) and the MT34 cycle buffer register (MT34CB), and 1/2 the carrier cycle plus the dead time in MT3GRA and MT3GRC. When no dead time generation is selected, set 1 in MT34DTD and 1/2 the carrier cycle + 1 in MT3GRA and MT3GRC.

[9] Select enabling/disabling of toggle output synchronized with the PWM cycle using bit PSYE in the MT34 output control register 0 (MT34OCR0), and set the PWM output level with bits OLSP and OLSN. When specifying the PWM output level by setting MT34OCR1 and MT34OLB to buffer operation, see figure 16.4, PWM Output Level Setting Procedure in Buffer Operation.

[10] Select complementary PWM mode in the MT3 mode register 0 (MT3MD0). Do not set in MT4MD0.

[11] Only when using the double buffer function, enable the double buffer function in the MT3 mode register 1 (MT3MD1).

[12] Set enabling/disabling of PWM waveform output pin output in the MT34 output enable register (MT34OEN).

[13] Set the port function select register and the port function enable register.

[14] Set bits CST3 and CST4 in MT01234STR to 1 simultaneously to start the count operation.

Figure 16.39 Example of Complementary PWM Mode Setting Procedure (Channels 3 and 4)

(2) Outline of Complementary PWM Mode Operation

In complementary PWM mode, 6-phase PWM output is possible. Figure 16.40 illustrates counter operation in complementary PWM mode (channels 3 and 4), and figure 16.41 illustrates an operation example in complementary PWM mode (channels 3 and 4).

(a) Counter Operation

In complementary PWM mode, three counters—MT3CNT, MT4CNT, and MT34SCNT (MT6CNT, MT7CNT, and MT67SCNT)—perform up/down-count operations.

The MT3CNT (MT6CNT) counter is automatically initialized to the value set in MT34DTD (MT67DTD) when complementary PWM mode is selected and the CST bit in MT01234STR (MT67STR) is 0.

When the CST bit is set to 1, the MT3CNT (MT6CNT) counter counts up to the value set in MT3GRA (MT6GRA), then switches to down-counting when it matches MT3GRA (MT6GRA). When the MT4CNT (MT7CNT) counter value matches H'0000, the MT3CNT (MT6CNT) counter switches to up-counting, and the operation is repeated in this way.

The MT4CNT (MT7CNT) counter is initialized to H'0000. When the CST bit is set to 1, the MT4CNT (MT7CNT) counter counts up in synchronization with the MT3CNT (MT6CNT) counter, and switches to down-counting when the MT3CNT (MT6CNT) counter matches MT3GRA (MT6GRA). After matching H'0000, the MT4CNT (MT7CNT) counter switches to up-counting, and the operation is repeated in this way.

The MT34SCNT (MT67SCNT) counter is a read-only counter. It need not be initialized.

When the MT3CNT (MT6CNT) counter matches MT34CD (MT67CD) during MT3CNT and MT4CNT (MT6CNT and MT7CNT) up-counting, down-counting is started, and when the MT3CNT (MT6CNT) counter matches MT3GRA (MT6GRA), the operation switches to up-counting. When the MT4CNT (MT7CNT) counter matches MT34DTD (MT67DTD), the MT34SCNT (MT67SCNT) counter is set with the value in MT3GRA (MT6GRA) and the counter is stopped.

When the MT4CNT (MT7CNT) counter matches MT34DTD (MT67DTD) during MT3CNT and MT4CNT (MT6CNT and MT7CNT) down-counting, up-counting is started. When the MT4CNT (MT7CNT) counter matches H'0000, it switches to down-counting. In addition, when the MT3CNT (MT6CNT) counter matches MT34CD (MT67CD), the MT34SCNT (MT67SCNT) counter is cleared to H'0000 and counter is stopped.

The MT34SCNT (MT67SCNT) counter is compared with the compare register and temporary register in which the PWM duty is set during the count operation only.

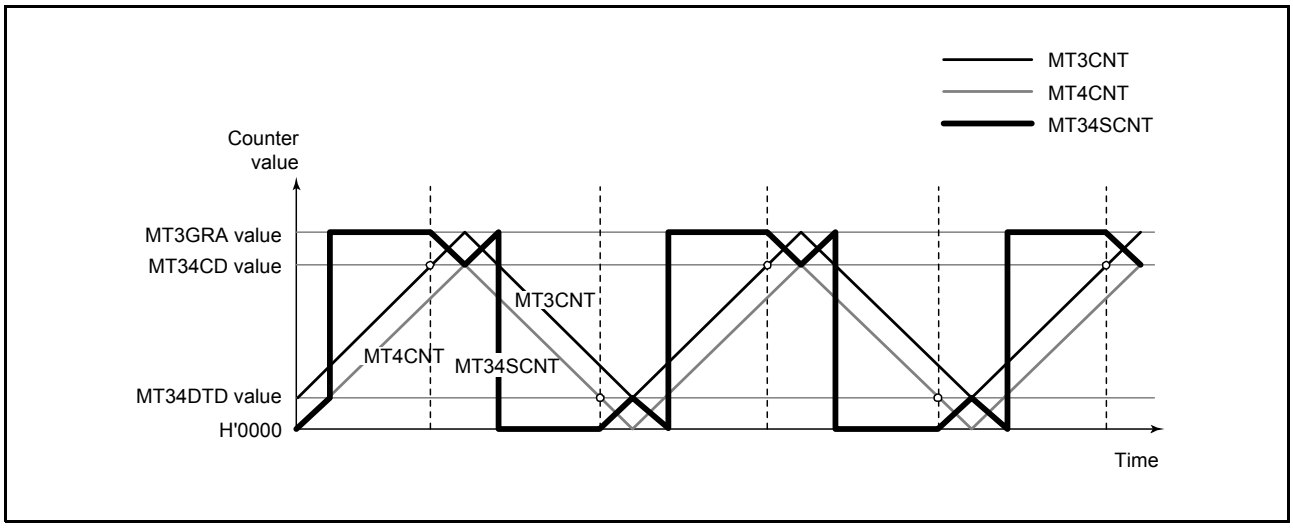


Figure 16.40 Complementary PWM Mode Counter Operation (Channels 3 and 4)

(b) Register Operation

In complementary PWM mode, nine registers are used, comprising compare registers, buffer registers, and temporary registers. Figure 16.41 shows an example of complementary PWM mode operation (channels 3 and 4).

The registers that are constantly compared with the counters to perform PWM output are the compare registers MT3GRB, MT4GRA, and MT4GRB (MT6GRB, MT7GRA, and MT7GRB). When these registers match the counter, the value set in bits OLSN and OLSP in MTOCR0 is output.

The buffer registers for these compare registers are MT3GRD, MT4GRC, and MT4GRD (MT6GRD, MT7GRC, and MT7GRD). Between a buffer register and compare register there is a temporary register. The temporary registers cannot be accessed by the CPU.

When the double buffer function is used, MT3GRE, MT4GRE, and MT4GRF (MT6GRE, MT7GRE, and MT7GRF) are also used as buffer registers B. For details of double buffer operation, refer to (s) Double Buffer Function in Complementary PWM Mode.

Between a buffer register and a compare match register, there is a temporary register, the temporary register cannot be accessed by the CPU.

Data in a compare register is changed by writing the new data to the corresponding buffer register. The buffer registers can be read or written at any time.

The data written to a buffer register is constantly transferred to the temporary register in the Ta interval. Data is not transferred to the temporary register in the Tb interval. Data written to a buffer register in this interval is transferred to the temporary register at the end of the Tb interval.

When the MT34SCNT (MT67SCNT) counter matches MT3GRA (MT6GRA) by counting up, or it becomes H'0000 by counting down, the Tb interval ends and the value transferred to a temporary register is transferred to the compare register. The timing for transfer from the temporary register to the compare register can be selected with the MD bit in MTMD0. Figure 16.41 shows an example in which the mode is selected in which the change is made in the trough.

In the Tb interval (Tb1 in figure 16.41) in which data transfer to the temporary register is not performed, the temporary register has the same function as the compare register, and is compared with the counter. In this interval, therefore, there are two compare match registers for one-phase output, with the compare register containing the pre-change data, and the temporary register containing the new data. In this interval, the three counters—MT3CNT, MT4CNT, and MT34SCNT (MT6CNT, MT7CNT, and MT67SCNT)—and two registers—compare register and temporary register—are compared, and PWM output controlled accordingly.

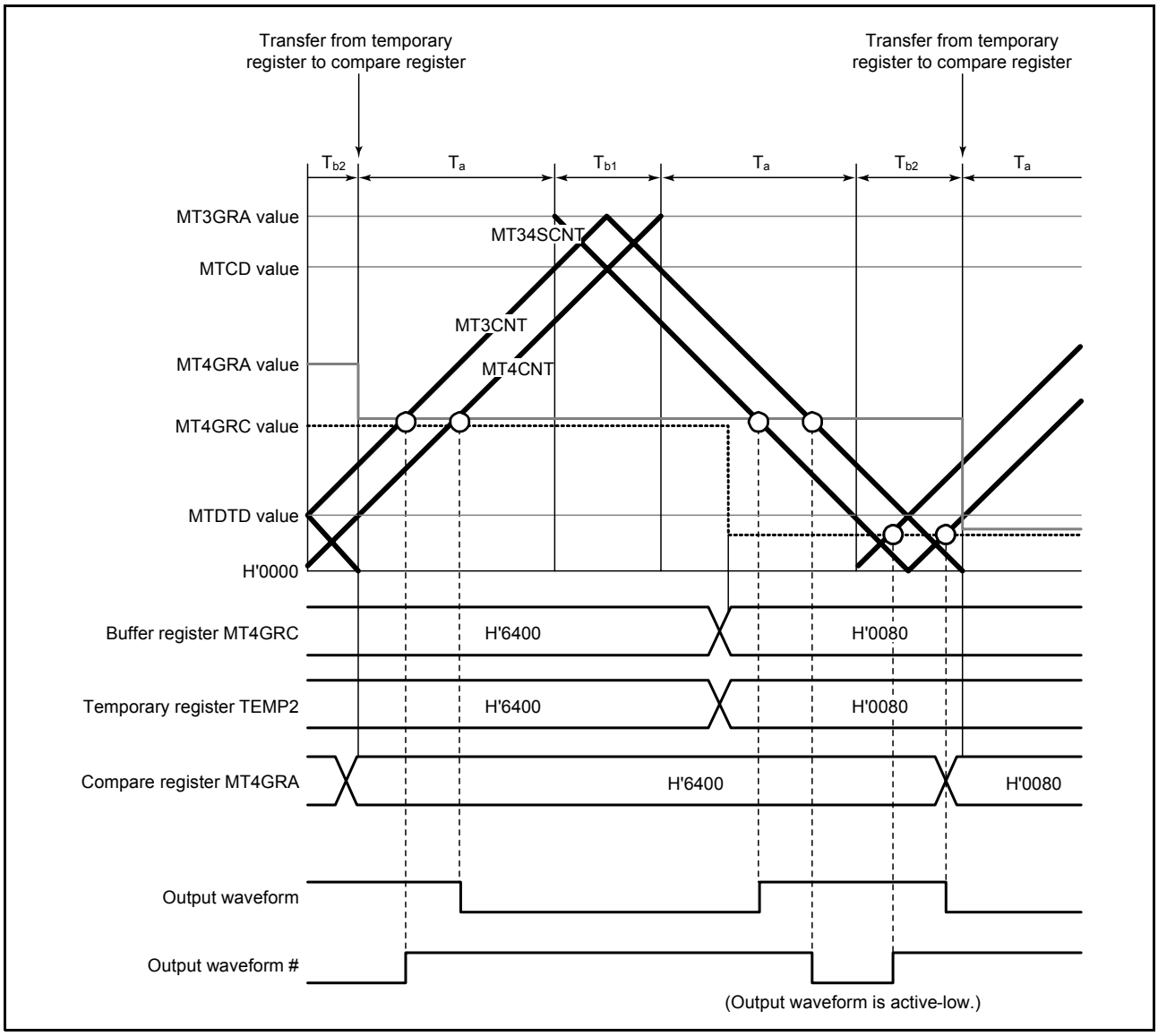


Figure 16.41 Example of Complementary Mode Operation (Channels 3 and 4)

(c) Initialization

In complementary PWM mode, there are six registers that must be initialized. In addition, there is a register that specifies whether to generate dead time (it should be used only when dead time generation should be disabled).

Before setting complementary PWM mode with the MD bits in MTMD0, the following initial register values must be set.

MT3GRC (MT6GRC) operates as the buffer register for MT3GRA (MT6GRA), and should be set with 1/2 the PWM carrier cycle + dead time Td. MTCB operates as the buffer register for MTCD, and should be set with 1/2 the PWM carrier cycle. Set dead time Td in MTDTD.

When no dead time is needed, the TDER bit in MTDTEN should be cleared to 0, MT3GRC and MT3GRA (MT6GRC and MT6GRA) should be set to 1/2 the PWM carrier cycle + 1, and MT34DTD (MT67DTD) should be set to 1.

Set the respective initial PWM duty values in buffer registers A (MT3GRD, MT4GRC, and MT4GRD (MT6GRD, MT7GRC, and MT7GRD)).

Set the respective (initial PWM duty - 1) values in buffer registers B (MT3GRE, MT4GRE, and MT4GRF (MT6GRE, MT7GRE, and MT7GRF)) only when the double buffer function is used.

The values set in the five buffer registers excluding MT34DTD (MT67DTD) are transferred simultaneously to the corresponding compare registers when complementary PWM mode is set.

Set the MT4CNT (MT7CNT) counter to H'0000 before setting complementary PWM mode.

Table 16.75 Registers and Counters Requiring Initialization

Register/Counter	Set value
MT3GRC, MT6GRC	1/2 PWM carrier cycle + dead time Td (1/2 PWM carrier cycle + 1 when dead time generation is disabled by MT3DTEN (MT6DTEN))
MT34DTD, MT67DTD	Dead time Td (1 when dead time generation is disabled by MT3DTEN (MT6DTEN))
MT34CB, MT67CB	1/2 PWM carrier cycle
MT3GRD, MT4GRC, MT4GRD MT6GRD, MT7GRC, MT7GRD	Initial PWM duty value for each phase
MT3GRE, MT4GRE, MT4GRF MT6GRE, MT7GRE, MT7GRF	Initial PWM duty - 1 value for each phase (only when double buffer function is used)
MT4CNT, MT7CNT	H'0000

Note: The MT3GRC (MT6GRC) set value must be the sum of 1/2 the PWM carrier cycle set in MT34CB (MT67CB) and dead time Td set in MT34DTD (MT67DTD). When dead time generation is disabled by MT3DTEN (MT6DTEN), 1/2 the PWM carrier cycle + 1 must be set.

(d) PWM Output Level Setting

In complementary PWM mode, the PWM pulse output level is set with bits OLSN and OLSP in MTOCR0 or bits OLS1P to OLS3P and OLS1N to OLS3N in MTOCR1.

The output level can be set for each of the three positive phases and three negative phases of 6-phase output.

Complementary PWM mode should be cleared before setting or changing output levels.

(e) Dead Time Setting

In complementary PWM mode, PWM pulses are output with a non-overlapping relationship between the positive and negative phases. This non-overlap time is called the dead time.

The non-overlap time is set in MTDTD. The value set in MT34DTD (MT67DTD) is used as the MT3CNT (MT6CNT) counter start value, and creates non-overlap between counters MT3CNT (MT6CNT) and MT4CNT (MT7CNT). Complementary PWM mode should be cleared before changing the contents of MT34DTD (MT67DTD).

(f) Dead Time Suppressing

Dead time generation is suppressed by clearing the MT3DTEN (MT6DTEN) bit in MTDTD to 0. MT3DTEN (MT6DTEN) can be cleared to 0 only when 0 is written to it after reading MT3DTEN (MT6DTEN) = 1.

MT3GRA and MT3GRC (MT6GRA and MT6GRC) should be set to 1/2 PWM carrier cycle + 1 and MTDTD should be set to 1.

By the above settings, PWM waveforms without dead time can be obtained.

Figure 16.42 shows an example of operation without dead time (channels 3 and 4).

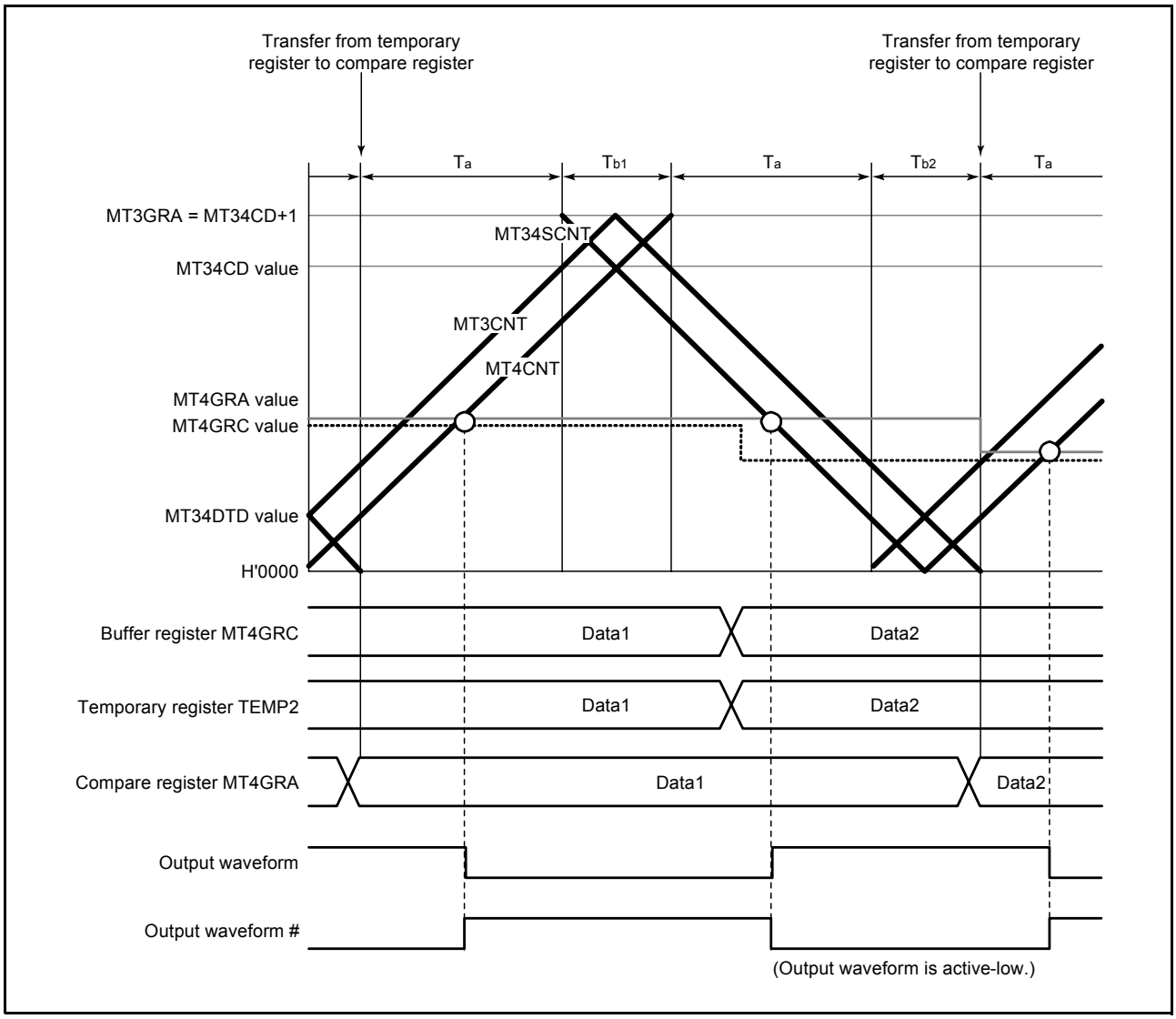


Figure 16.42 Example of Operation without Dead Time (Channels 3 and 4)

(g) PWM Cycle Setting

In complementary PWM mode, the PWM pulse cycle is set in two registers—MT3GRA (MT6GRA), in which the MTCNT3 (MT6CNT) counter upper limit value is set, and MT34CD (MT67CD), in which the MT4CNT (MT7CNT) counter upper limit value is set. The settings should be made so as to achieve the following relationship between these two registers:

With dead time: MT3GRA (MT6GRA) set value = MT34CD (MT67CD) set value + MT34DTD (MT67DTD) set value

Without dead time: MT3GRA (MT6GRA) set value = MT34CD (MT67CD) set value + 1

The MT3GRA and MT34CD (MT6GRA and MT67CD) settings are made by setting the values in buffer registers MT3GRC and MT34CB (MT6GRC and MT67CB). The values set in MT3GRC and MT34CB (MT6GRC and MT67CB) are transferred simultaneously to MT3GRA and MT34CD (MT6GRA an MT67CD) in accordance with the transfer timing selected with the MD bits in MTMD0.

The updated PWM cycle is reflected from the next cycle when the data update is performed at the crest, and from the current cycle when performed in the trough.

Figure 16.43 illustrates the operation when the PWM cycle is updated at the crest. See the following section, (h) Register Data Updating, for the method of updating the data in each buffer register.

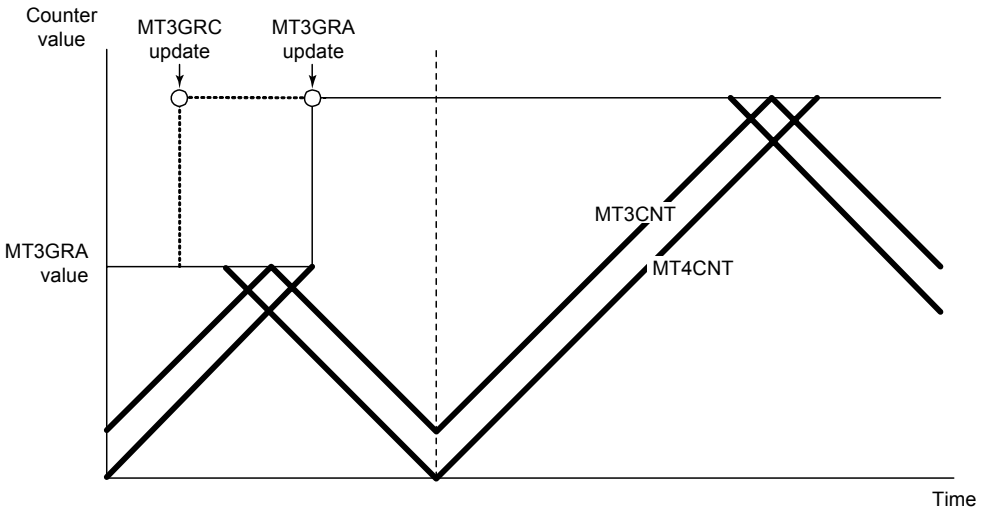


Figure 16.43 Example of PWM Cycle Updating (Channels 3 and 4)

(h) Register Data Updating

In complementary PWM mode, the buffer register is used to update the data in a compare register. The update data can be written to the buffer register at any time. There are five PWM duty and carrier cycle registers that have buffer registers and can be updated during operation.

There is a temporary register between each of these registers and its buffer register. When sub-counter MT34SCNT (MT67SCNT) is not counting, if buffer register data is updated, the temporary register value is also rewritten. Transfer is not performed from buffer registers to temporary registers when the MT34SCNT (MT67SCNT) counter is counting; in this case, the value written to a buffer register is transferred after the MT34SCNT (MT67SCNT) counter halts.

The temporary register value is transferred to the compare register at the data update timing set with the MD bits in MTMD0.

Figure 16.44 shows an example of data updating in complementary PWM mode (channels 3 and 4). This example shows the mode in which data updating is performed at both the counter crest and trough.

When rewriting buffer register data, a write to MT4GRD (MT7GRD) must be performed at the end of the update. Data transfer from the buffer registers to the temporary registers is performed simultaneously for all five registers after the write to MT4GRD (MT7GRD).

A write to MT4GRD (MT7GRD) must be performed after writing data to the registers to be updated, even when not updating all five registers, or when not updating the MT4GRD (MT7GRD) data. In this case, the data written to MT4GRD (MT7GRD) should be the same as the data prior to the write operation.

For data updating when the double buffer function is used, refer to (s) Double Buffer Function in Complementary PWM Mode.

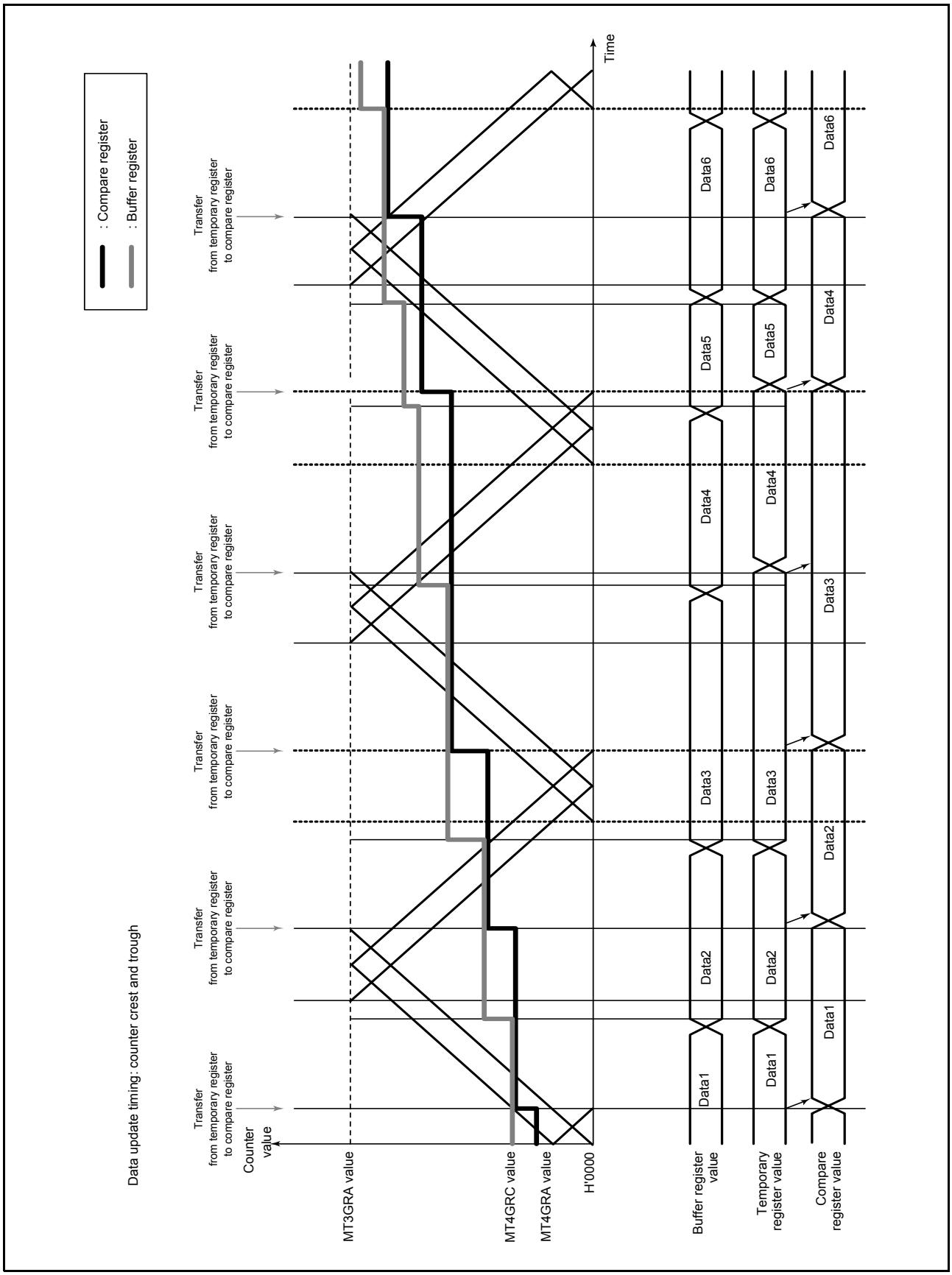


Figure 16.44 Example of Data Update in Complementary PWM Mode (Channels 3 and 4)

(i) Initial Output in Complementary PWM Mode

In complementary PWM mode, the initial output is determined by the setting of bits OLSN and OLSP in MTOCR0 or bits OLS1N to OLS3N and OLS1P to OLS3P in MTOCR1.

This initial output is the PWM pulse non-active level, and is output from when complementary PWM mode is set with MTMD0 until the MT4CNT (MT7CNT) counter exceeds the value set in MTDTD.

Figure 16.45 shows an example of the initial output in complementary PWM mode, and figure 16.46 shows an example of the waveform when the initial PWM duty value is smaller than the MT34DTD (MT67DTD) value.

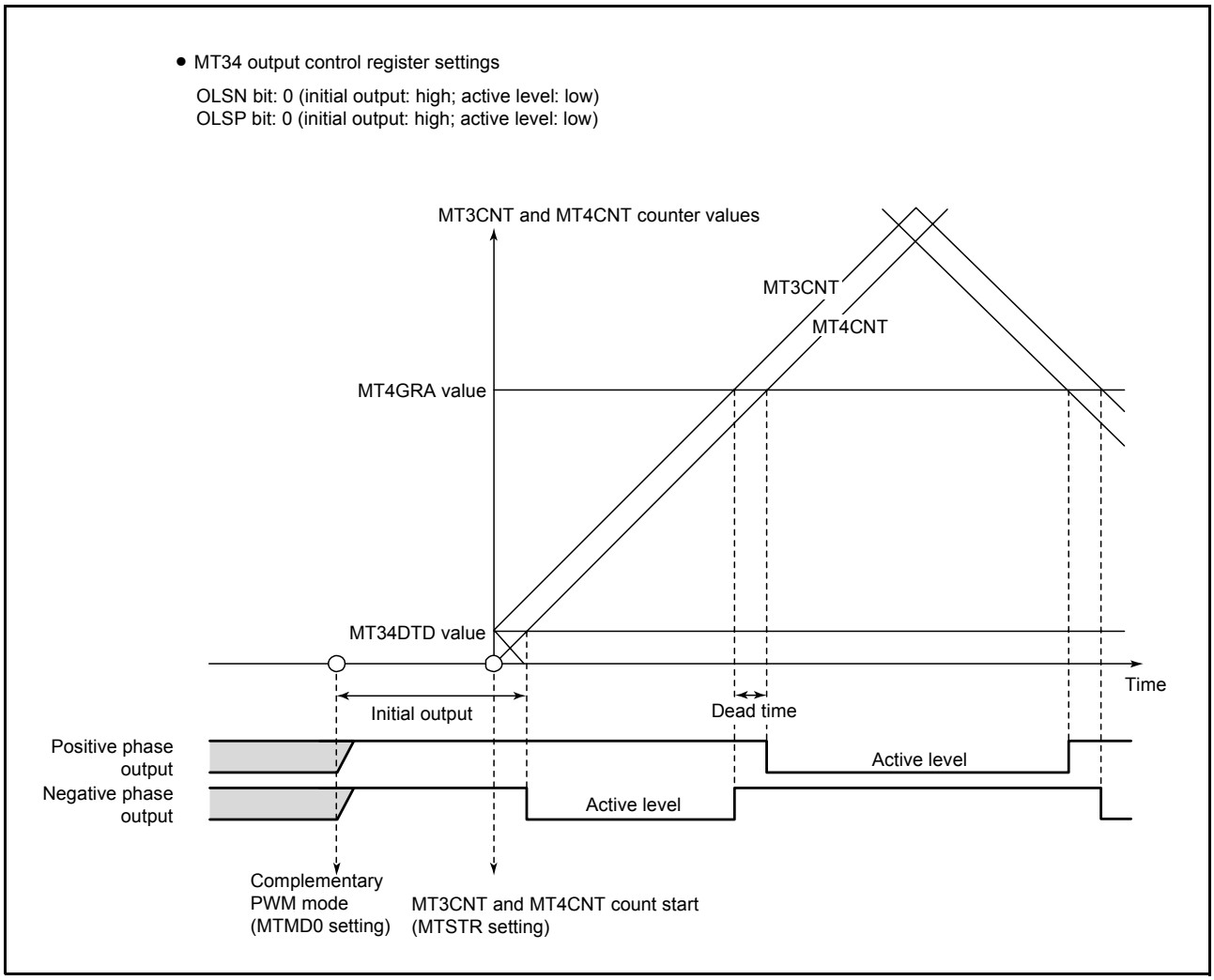


Figure 16.45 Example of Initial Output in Complementary PWM Mode (Channels 3 and 4) (1)

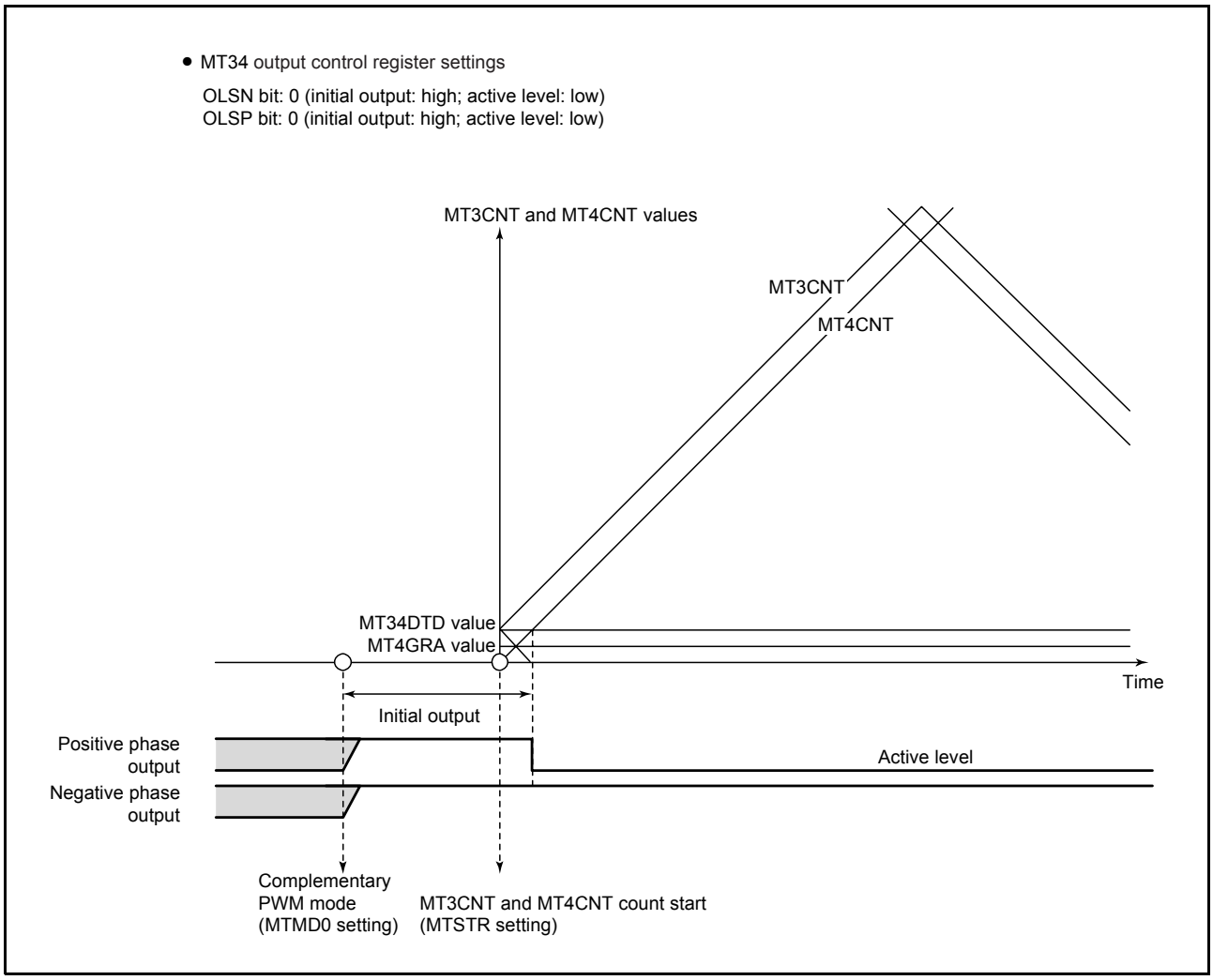


Figure 16.46 Example of Initial Output in Complementary PWM Mode (Channels 3 and 4) (2)

(j) Complementary PWM Mode PWM Output Generation Method

In complementary PWM mode, 3-phase output is performed of PWM waveforms with a non-overlap time between the positive and negative phases. This non-overlap time is called the dead time.

A PWM waveform is generated by output of the output level selected in the MT output control register in the event of a compare-match between a counter and data register. While MT34SCNT (MT67SCNT) is counting, data register and temporary register values are simultaneously compared to generate PWM pulses the duty of which is continuous from 0 to 100%. The relative timing of on and off compare-match occurrence may vary, but the compare-match that turns off each phase takes precedence to secure the dead time and ensure that the positive phase and negative phase on times do not overlap. Figure 16.47 to figure 16.49 show examples of waveform generation in complementary PWM mode.

The positive phase/negative phase off timing is generated by a compare-match with the solid-line counter, and the on timing by a compare-match with the dotted-line counter operating with a delay of the dead time behind the solid-line counter. In the T1 period, compare-match a that turns off the negative phase has the highest priority, and compare-matches occurring prior to a are ignored. In the T2 period, compare-match c that turns off the positive phase has the highest priority, and compare-matches occurring prior to c are ignored.

In normal cases, compare-matches occur in the order a → b → c → d (or c → d → a' → b'), as shown in figure 16.47. If compare-matches deviate from the a → b → c → d order, since the time for which the negative phase is off is less than twice the dead time, the figure shows the positive phase is not being turned on. If compare-matches deviate from the c → d → a' → b' order, since the time for which the positive phase is off is less than twice the dead time, the figure shows the negative phase is not being turned on.

If compare-match c occurs first following compare-match a, as shown in figure 16.48, compare-match b is ignored, and the negative phase is turned off by compare-match d. This is because turning off of the positive phase has priority due to the occurrence of compare-match c (positive phase off timing) before compare-match b (positive phase on timing) (consequently, the waveform does not change since the positive phase goes from off to off).

Similarly, in the example in figure 16.49, compare-match a' with the new data in the temporary register occurs before compare-match c, but other compare-matches occurring up to c, which turns off the positive phase, are ignored. As a result, the negative phase is not turned on.

Thus, in complementary PWM mode, compare-matches at turn-off timings take precedence, and turn-on timing compare-matches that occur before a turn-off timing compare-match are ignored.

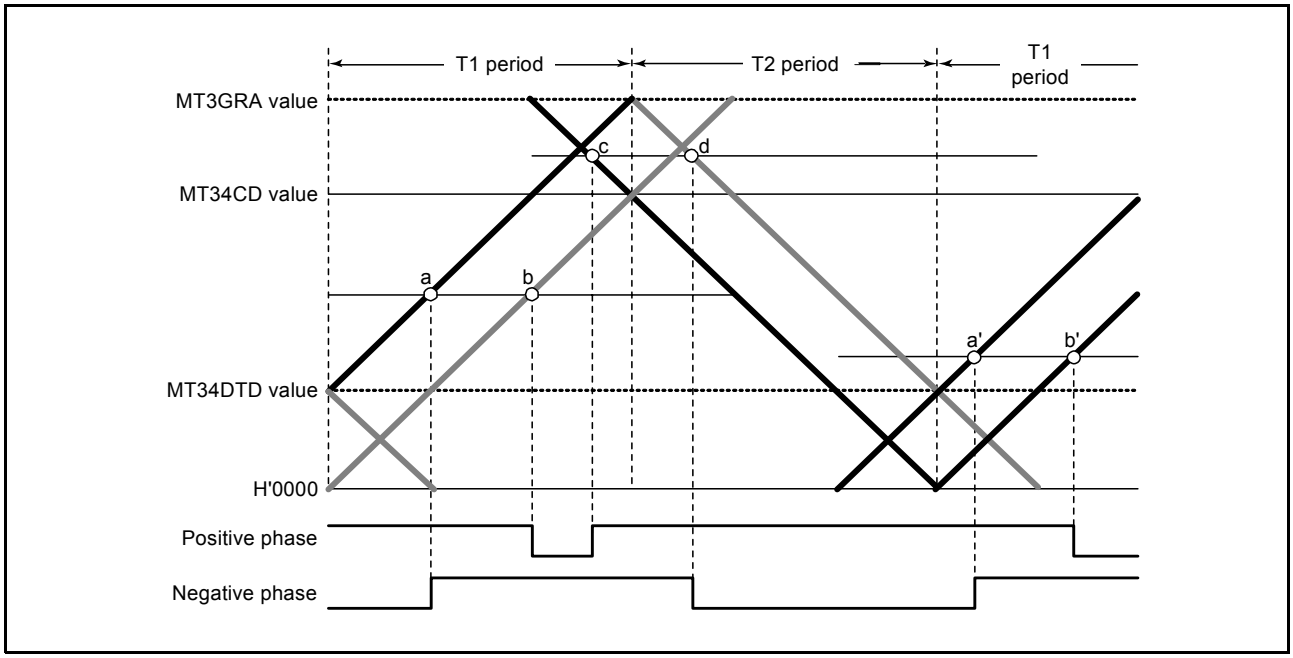


Figure 16.47 Example of Complementary PWM Mode Waveform Output (Channels 3 and 4) (1)

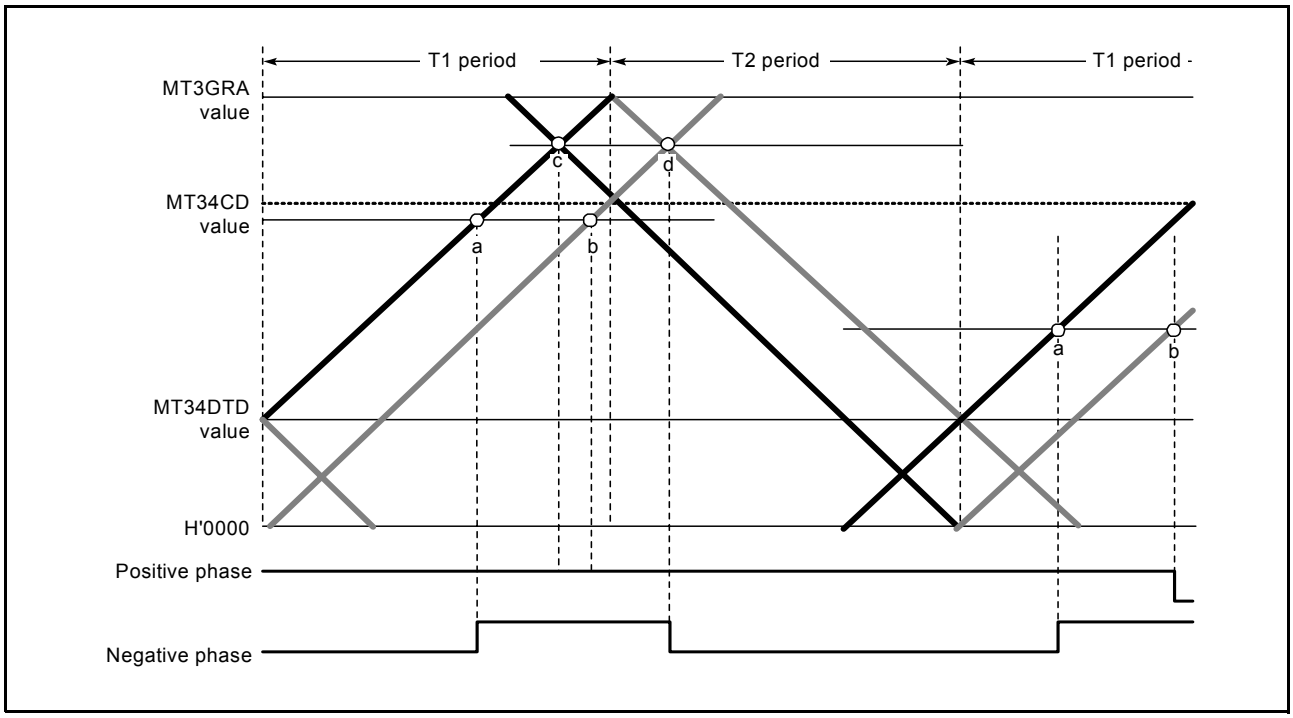


Figure 16.48 Example of Complementary PWM Mode Waveform Output (Channels 3 and 4) (2)

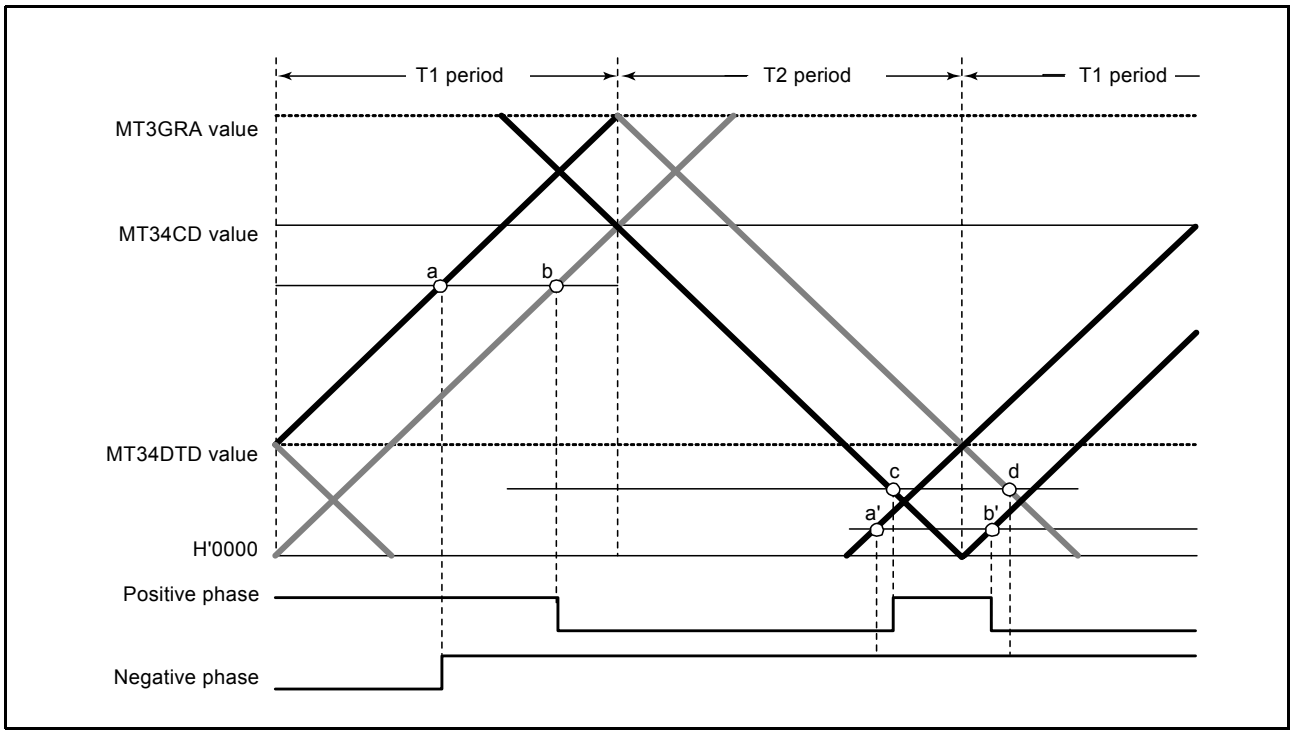


Figure 16.49 Example of Complementary PWM Mode Waveform Output (Channels 3 and 4) (3)

(k) Complementary PWM Mode 0% and 100% Duty Output

In complementary PWM mode, 0% and 100% duty cycles can be output as required. Figure 16.50 to figure 16.54 show output examples.

100% duty output is performed when the data register value is set to H'0000. The waveform in this case has a positive phase with a 100% on-state. 0% duty output is performed when the data register value is set to the same value as MT3GRA (MT6GRA). The waveform in this case has a positive phase with a 100% off-state.

On and off compare-matches occur simultaneously, but if a turn-on compare-match and turn-off compare-match for the same phase occur simultaneously, both compare-matches are ignored and the waveform does not change.

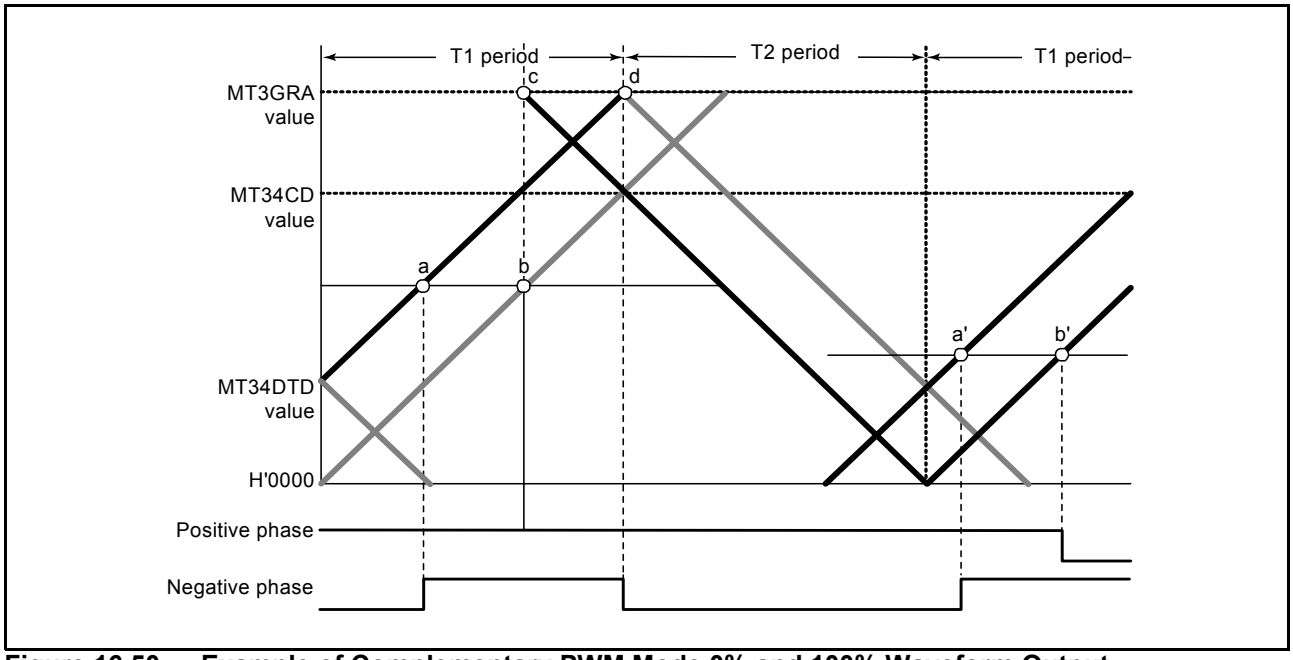


Figure 16.50 Example of Complementary PWM Mode 0% and 100% Waveform Output (Channels 3 and 4) (1)

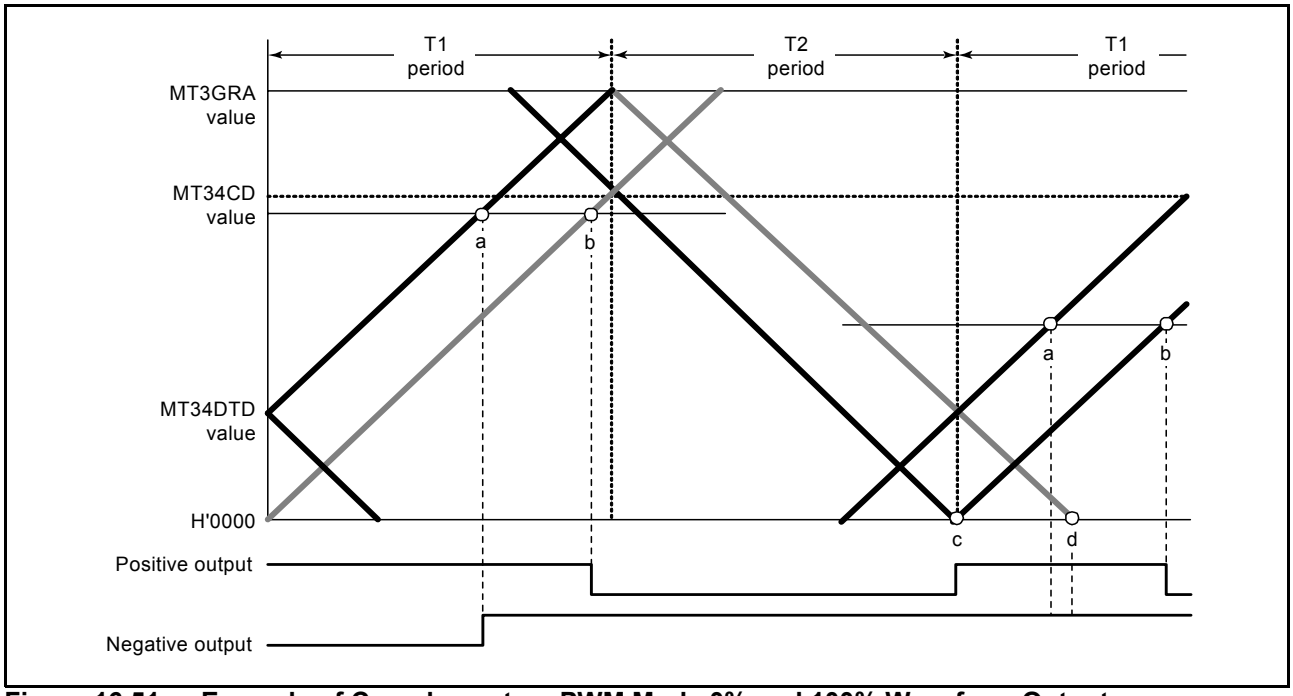


Figure 16.51 Example of Complementary PWM Mode 0% and 100% Waveform Output (Channels 3 and 4) (2)

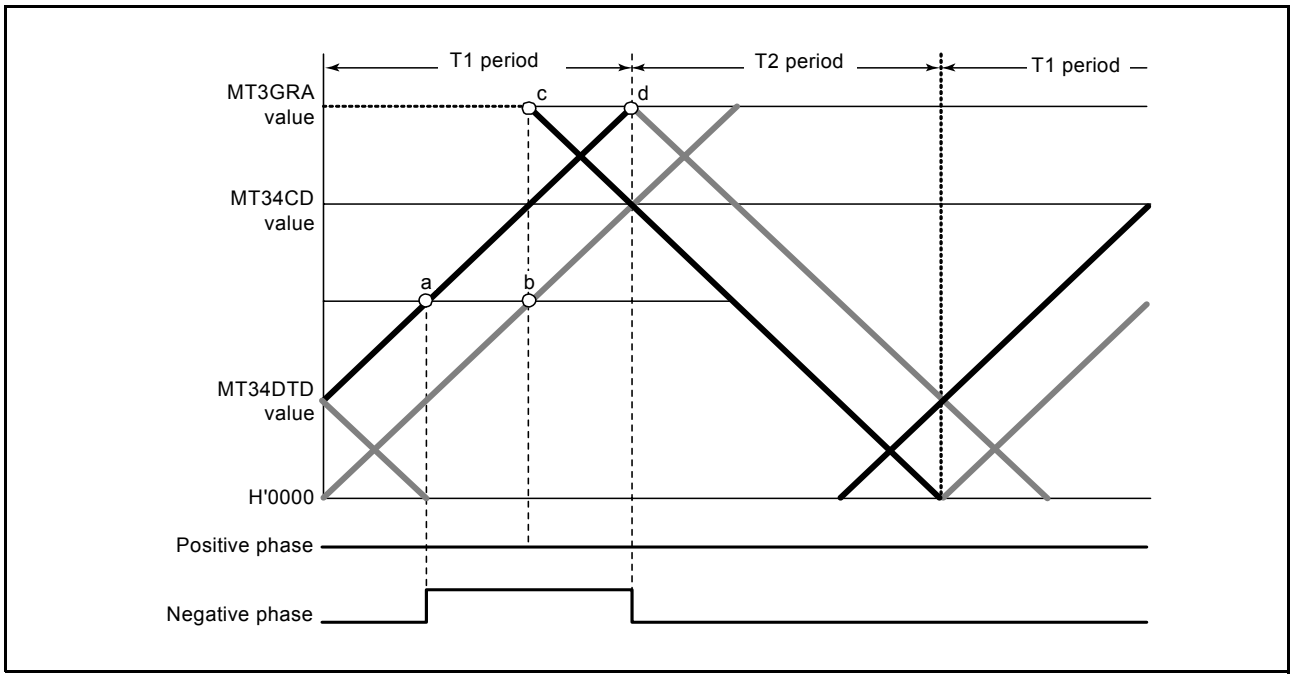


Figure 16.52 Example of Complementary PWM Mode 0% and 100% Waveform Output (Channels 3 and 4) (3)

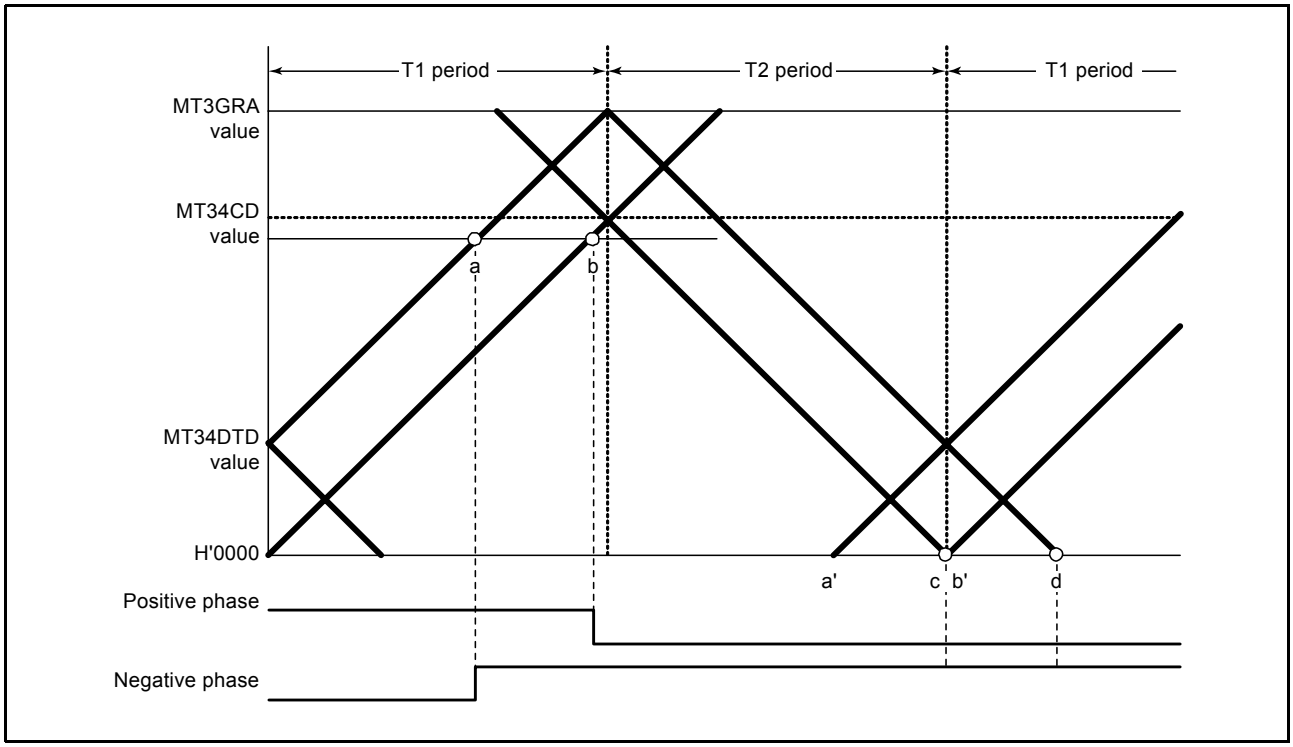


Figure 16.53 Example of Complementary PWM Mode 0% and 100% Waveform Output (Channels 3 and 4) (4)

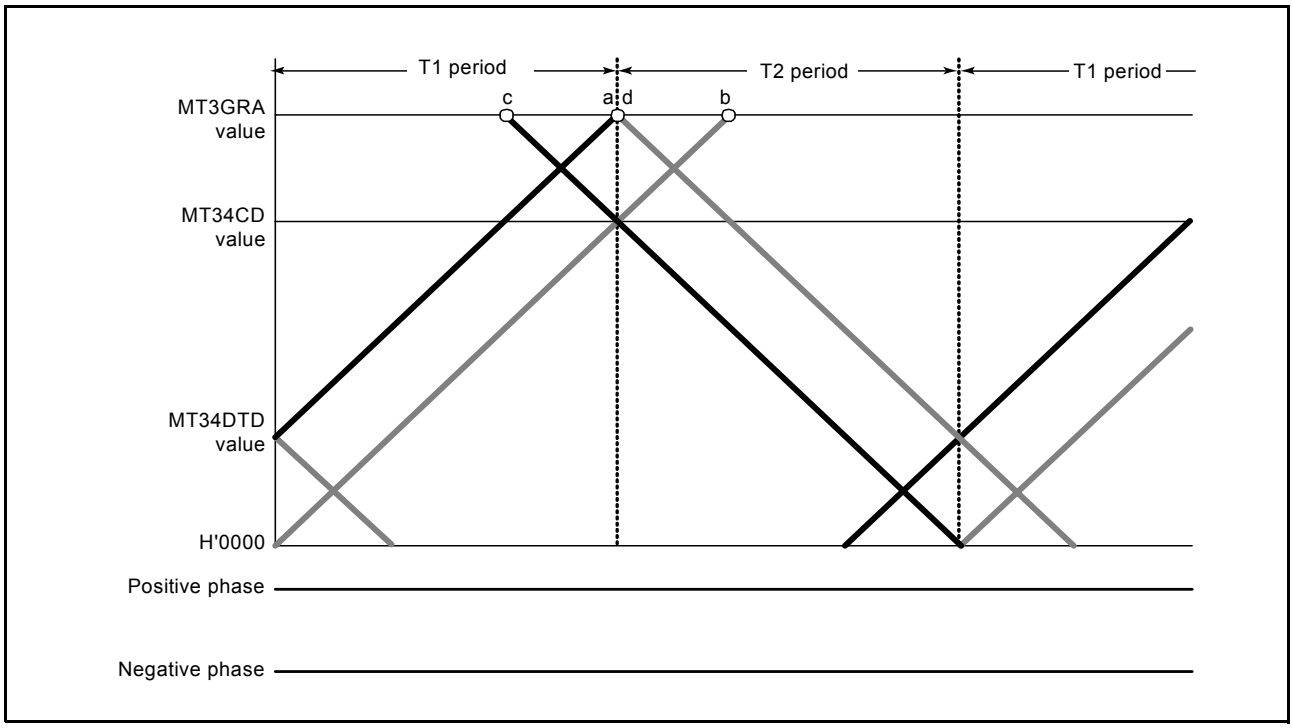


Figure 16.54 Example of Complementary PWM Mode 0% and 100% Waveform Output (Channels 3 and 4) (5)

(l) Toggle Output Synchronized with PWM Cycle

In complementary PWM mode, toggle output can be performed in synchronization with the PWM carrier cycle by setting the PSYE bit to 1 in MTOCR0. Figure 16.55 shows an example of a toggle output waveform. This output is toggled by a compare-match between the MT3CNT counter and MT3GRA (MT6CNT and MT6GRA) and a compare-match between the MT4CNT (MT7CNT) counter and H'0000.

The output pin for this toggle output is the TIOC3A (TIOC6A) pin. The initial output is 1.

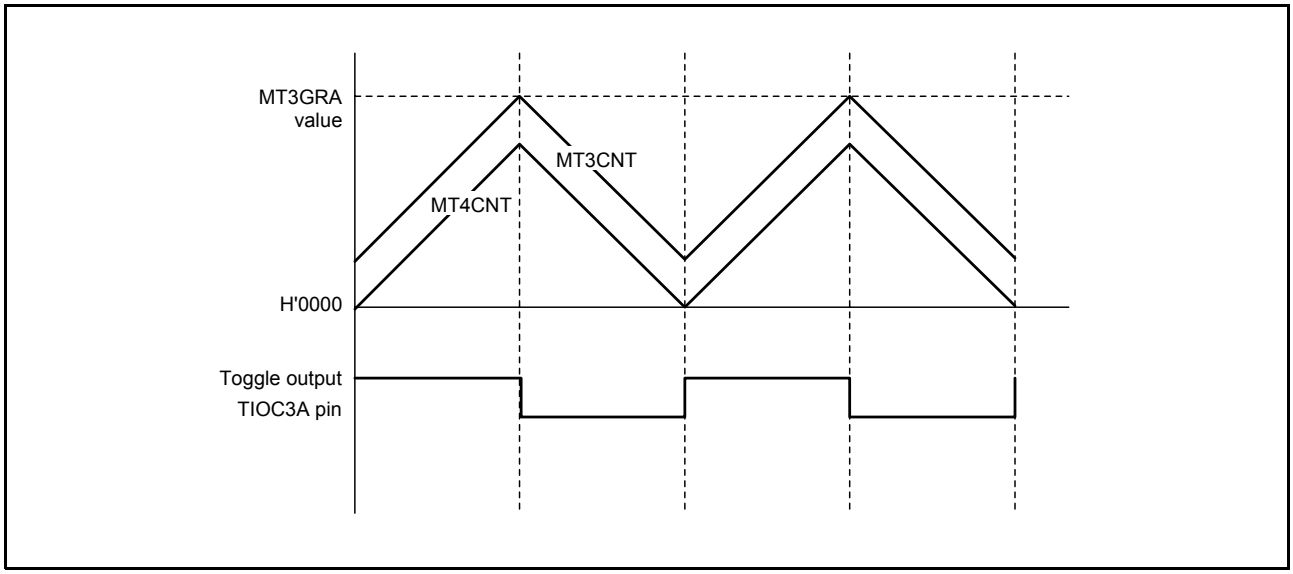


Figure 16.55 Example of Toggle Output Waveform Synchronized with PWM Output (Channels 3 and 4)

(m) Counter Clearing by Another Channel

In complementary PWM mode, by setting a mode for synchronization with another channel by means of MTSY, and selecting synchronous clearing with the CCLR bits in MTCR, it is possible to have the counters MT3CNT and MT4CNT, and MT34SCNT (MT6CNT, MT7CNT, and MT67SCNT) cleared by another channel.

Figure 16.56 illustrates the operation.

Use of this function enables counter clearing and restarting to be performed by means of an external signal.

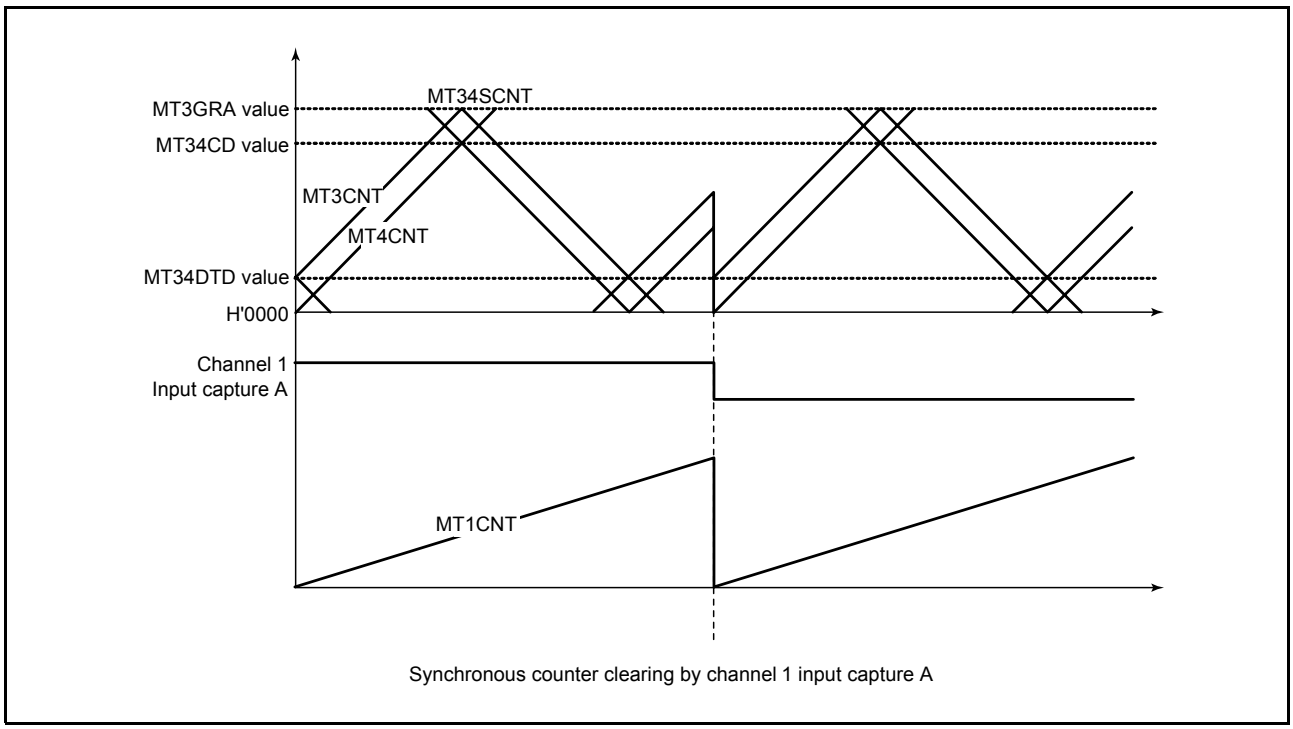


Figure 16.56 Counter Clearing Synchronized with Another Channel (Channels 3 and 4)

(n) Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode

Setting the WRE bit in MT34WCR (MT67WCR) to 1 suppresses initial output when synchronous counter clearing occurs in the Tb interval at the trough in complementary PWM mode and controls abrupt change in duty cycle at synchronous counter clearing.

Initial output suppression is applicable only when synchronous clearing occurs in the Tb interval at the trough as indicated by (10) or (11) in figure 16.57. When synchronous clearing occurs outside that interval, the initial value specified by the OLS bits in MTOCR0 is output. Even in the Tb interval at the trough, if synchronous clearing occurs in the initial value output period (indicated by (1) in figure 16.57) immediately after the counters start operation, initial value output is not suppressed.

This function can be used in both channel combinations (channels 3 and 4, and channels 6 and 7). The counter clearing source of this function for channels 3 and 4 is synchronous clearing from channels 0 to 2. The counter clearing source for channels 6 and 7 is flag setting (compare match/input capture) of channels 0 to 2.

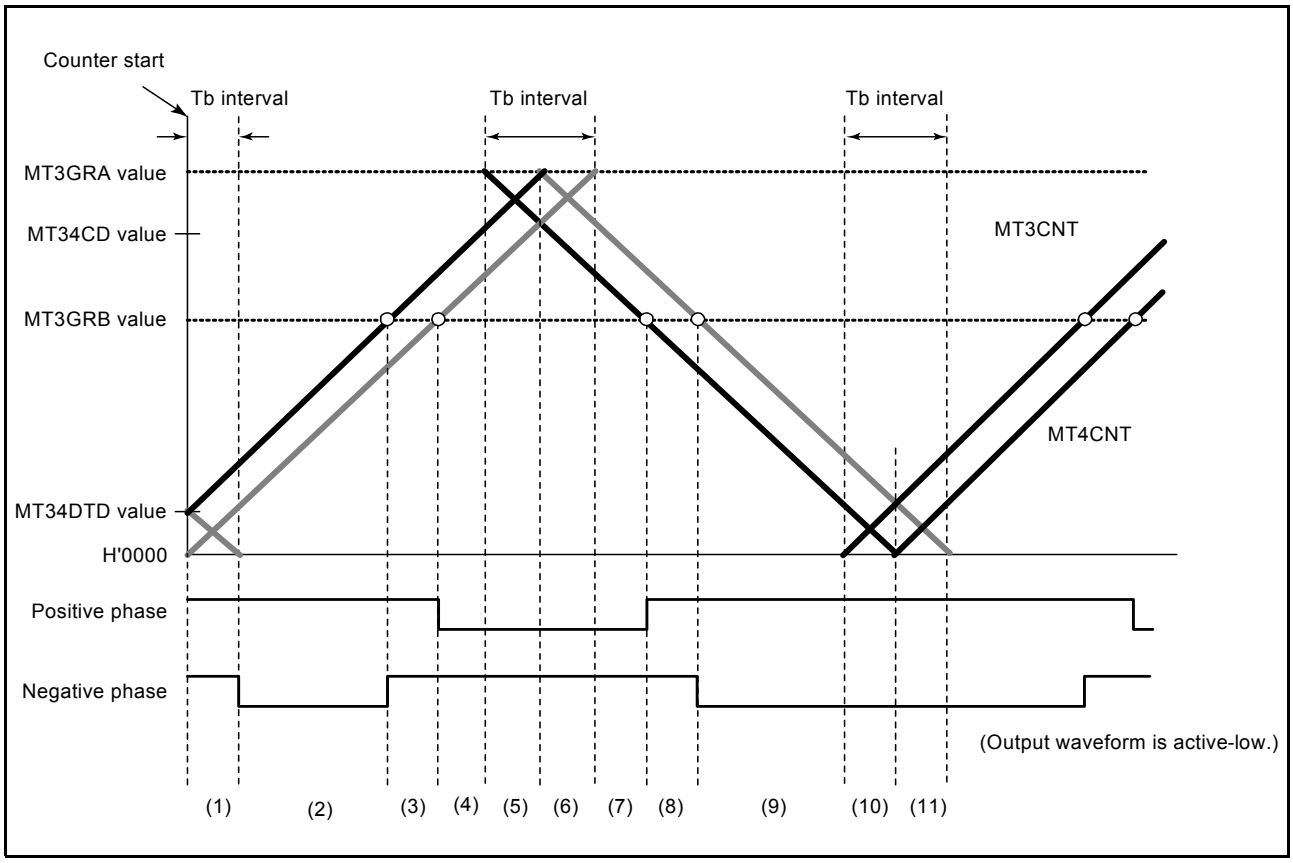


Figure 16.57 Timing for Synchronous Counter Clearing (Channels 3 and 4)

- Example of Procedure for Setting Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode

Figure 16.58 shows an example of the procedure for setting output waveform control at synchronous counter clearing in complementary PWM mode.

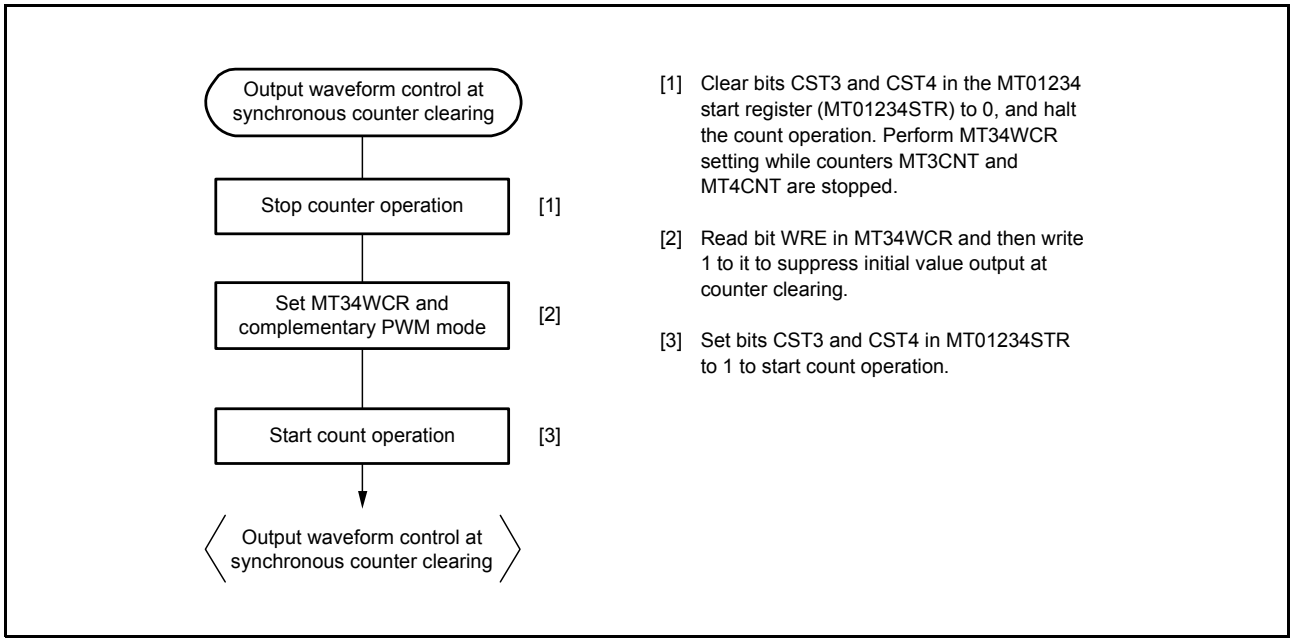


Figure 16.58 Example of Procedure for Setting Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode (Channels 3 and 4)

- Examples of Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode

Figure 16.59 to figure 16.62 show examples of output waveform control in which channels 3 and 4 operate in complementary PWM mode and synchronous counter clearing is generated while the WRE bit in MT34WCR is set to 1. In the examples shown in figure 16.59 to figure 16.62, synchronous counter clearing occurs at timing (3), (6), (8), and (11) shown in figure 16.57, respectively.

In channels 6 and 7, these examples are equivalent to the cases when channels 6 and 7 operate in complementary PWM mode and synchronous counter clearing is generated while the SCC bit is cleared to 0 and the WRE bit is set to 1 in MT67WCR.

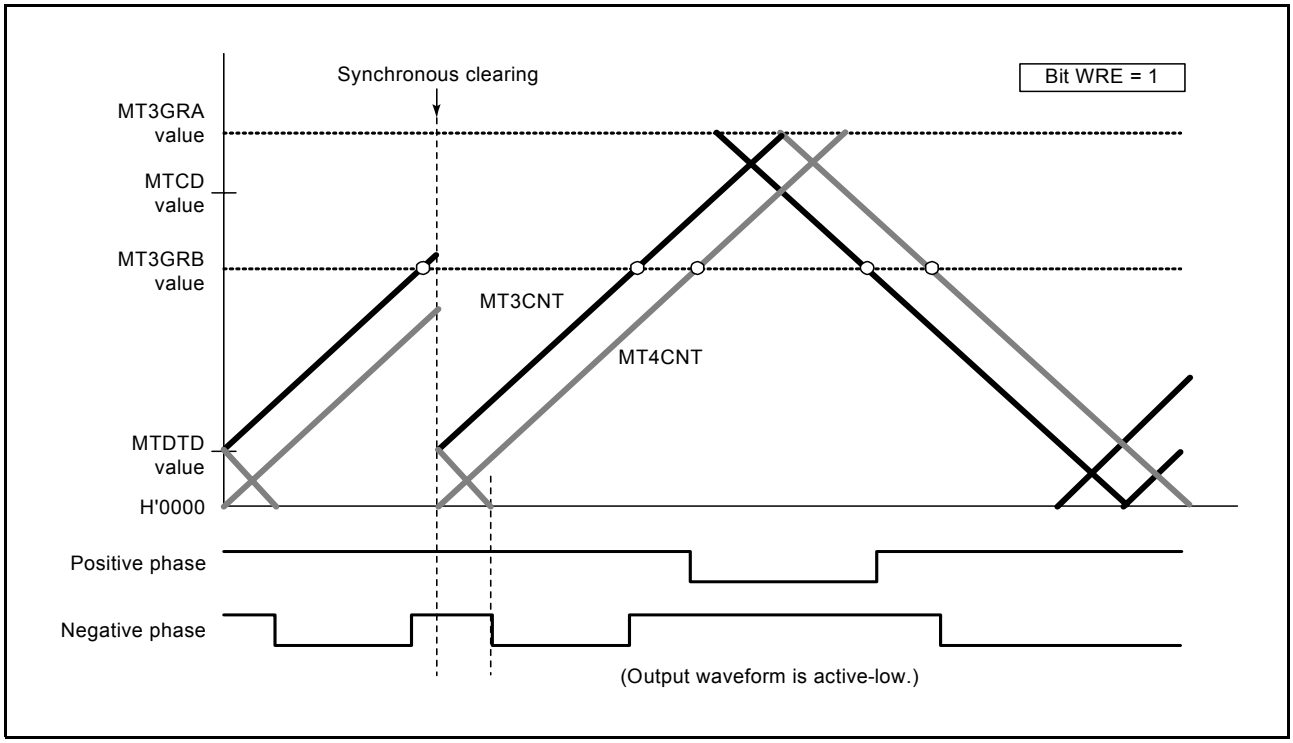


Figure 16.59 Example of Synchronous Clearing in Dead Time during Up-Counting (Timing (3) in Figure 16.57; Bit WRE of MT34WCR and MT67WCR is 1)

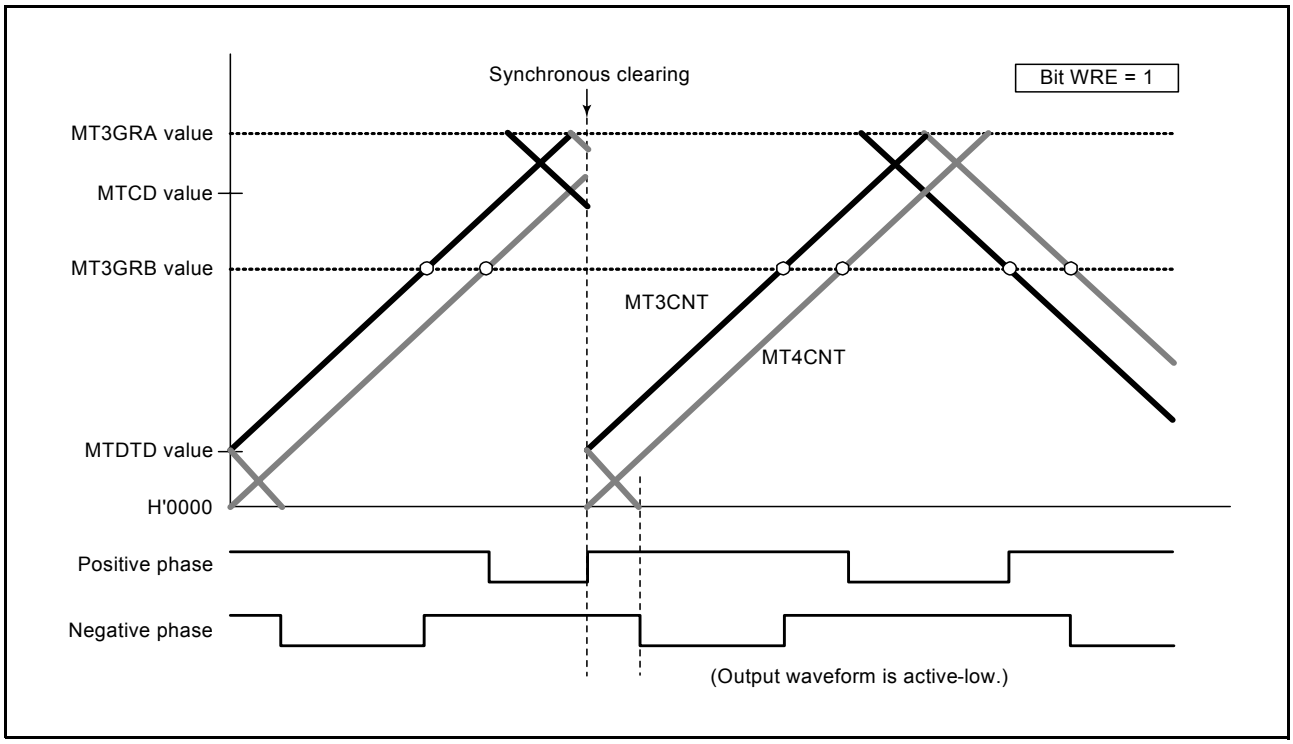


Figure 16.60 Example of Synchronous Clearing in Interval Tb at Crest (Timing (6) in Figure 16.57; Bit WRE of MT34WCR and MT67WCR is 1)

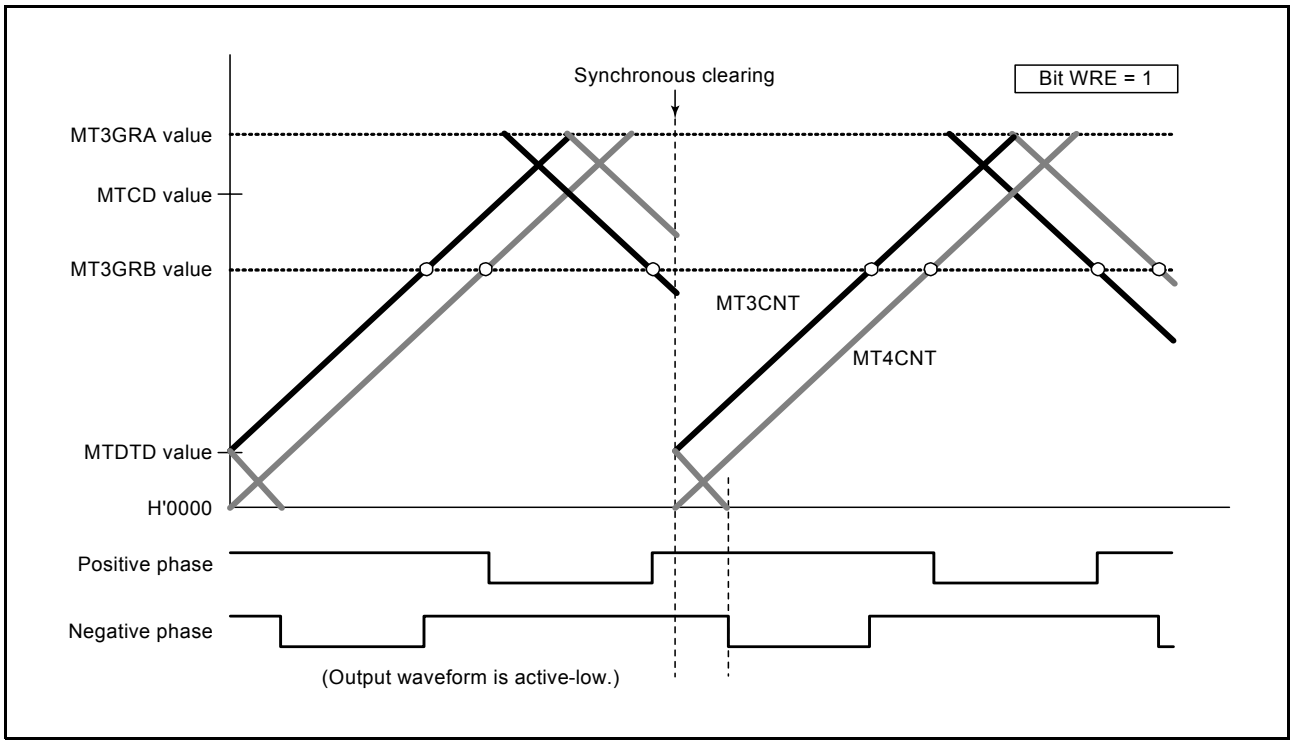


Figure 16.61 Example of Synchronous Clearing in Dead Time during Down-Counting (Timing (8) in Figure 16.57; Bit WRE of MT34WCR and MT67WCR is 1)

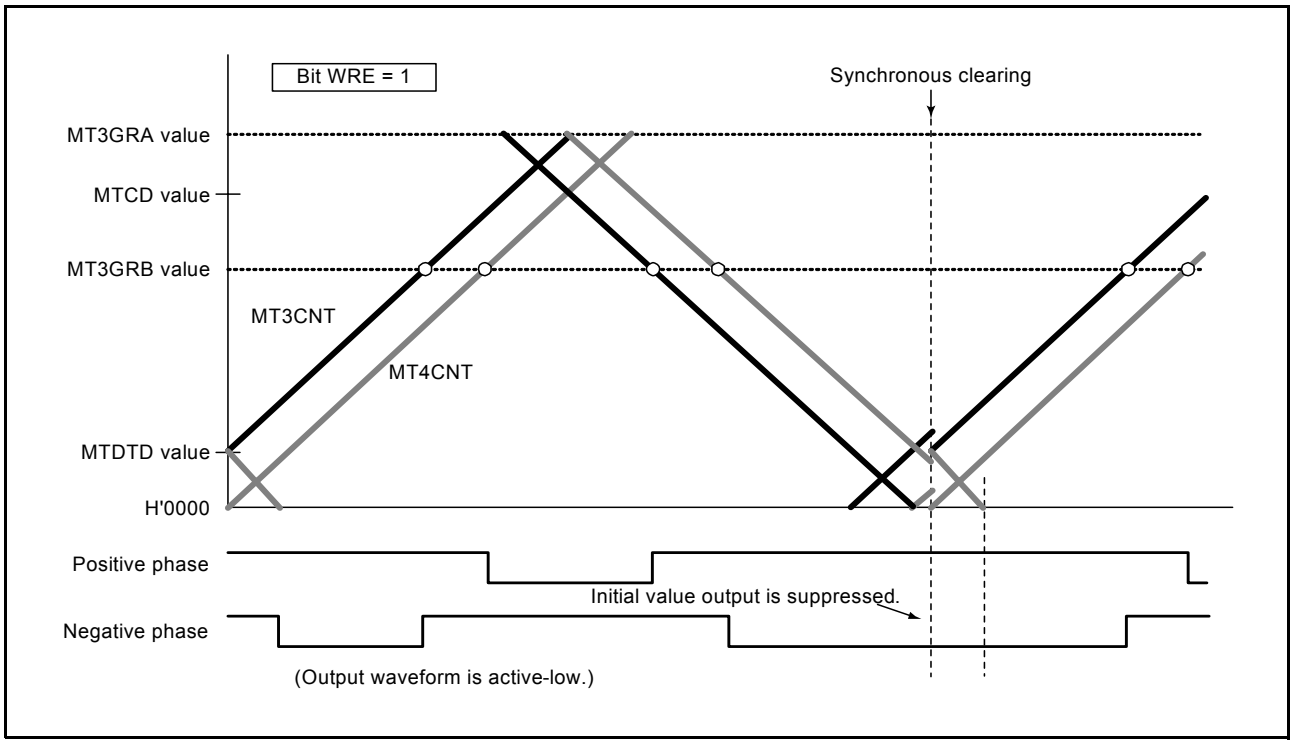


Figure 16.62 Example of Synchronous Clearing in Interval Tb at Trough (Timing (11) in Figure 16.57; Bit WRE of MT34WCR and MT67WCR is 1)

(o) Suppressing Synchronous Counter Clearing for Channels 0, 1, and 2, and Channels 6 and 7

In channels 6 and 7, setting the SCC bit in MT67WCR to 1 suppresses synchronous counter clearing caused by channels 0 to 2.

Synchronous counter clearing is suppressed only within the interval shown in figure 16.63.

When using this function, channels 6 and 7 should be set to complementary PWM mode.

For details of synchronous clearing caused by channels 0 to 2, refer to the description in section 16.3.10 (2) Clearing Counters of Channels 6 and 7 by Flag Setting Sources (Synchronous Counter Clearing for Channels 6 and 7).

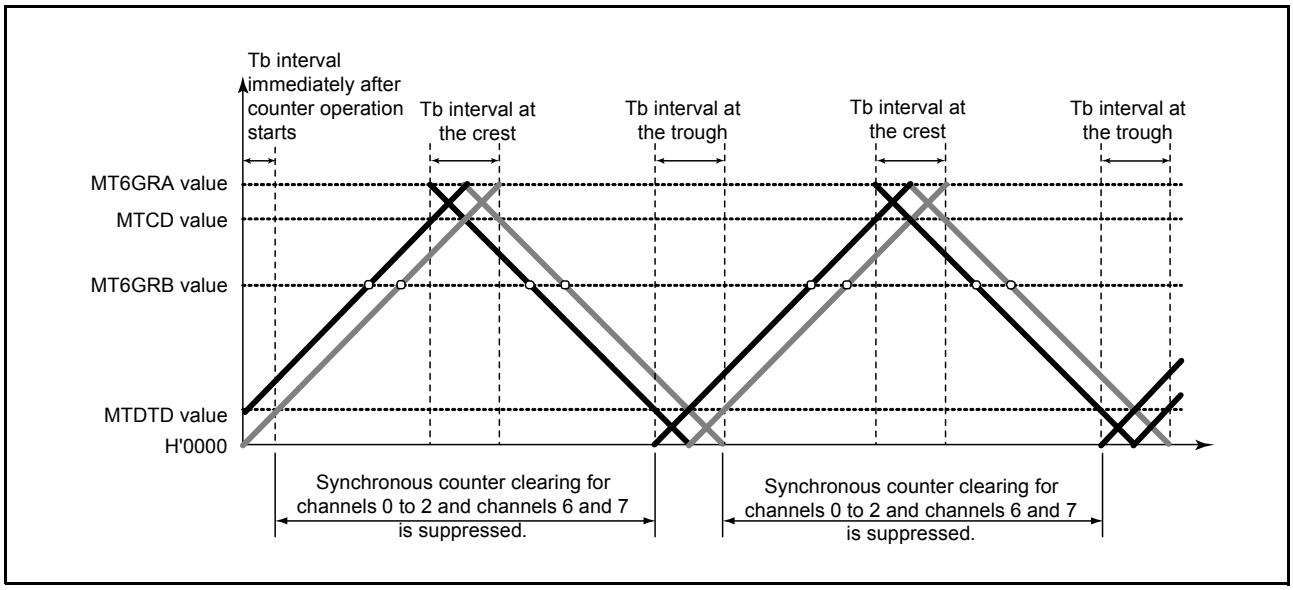


Figure 16.63 Synchronous Clearing-Suppressed Interval Specified by SCC Bit in MT67WCR (for Channels 0, 1, and 2, and Channels 6 and 7)

- Example of Procedure for Suppressing Synchronous Counter Clearing for Channels 0, 1, and 2, and Channels 6 and 7

Figure 16.64 shows an example of the procedure for suppressing synchronous counter clearing for channels 0, 1, and 2, and channels 6 and 7.

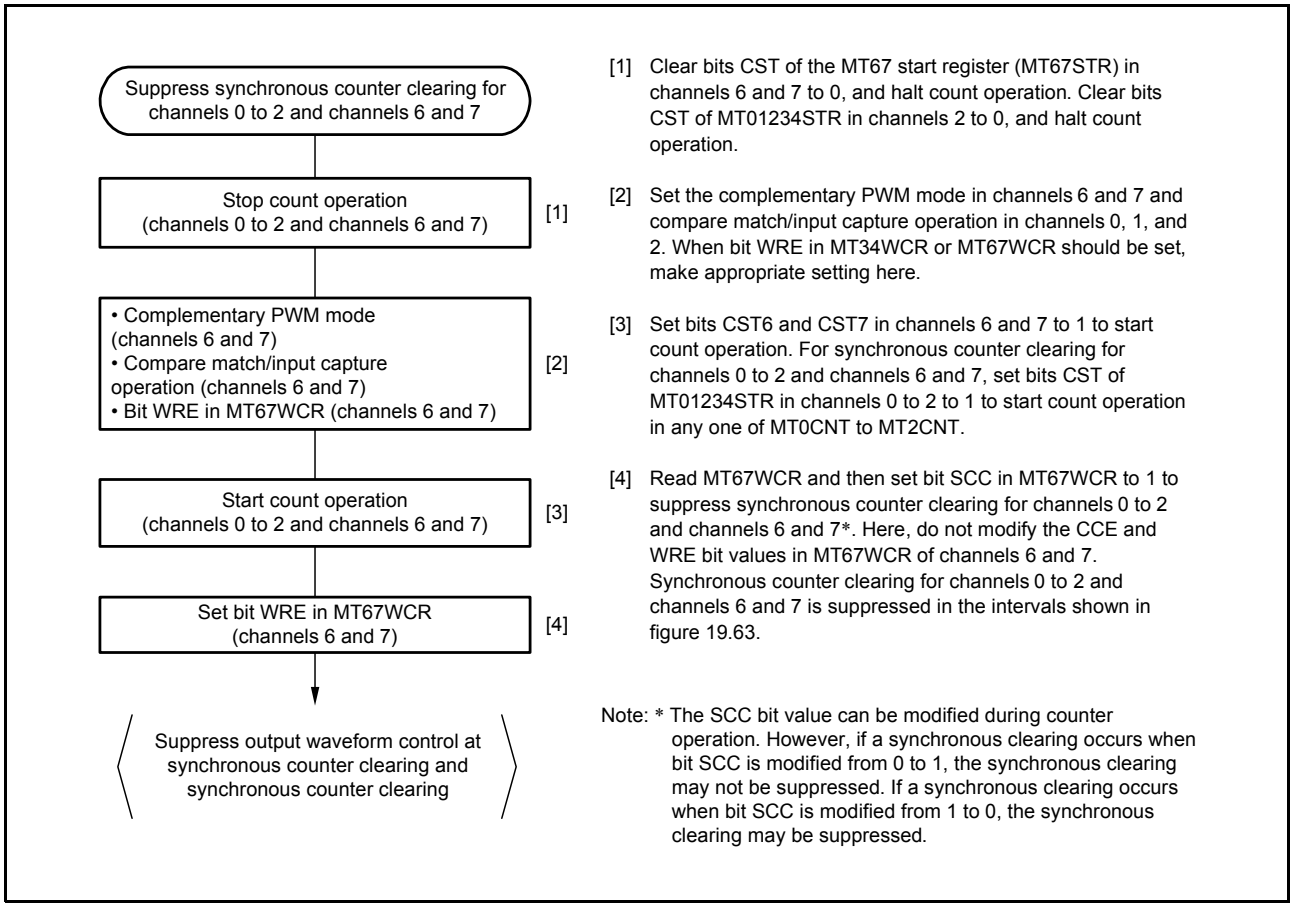


Figure 16.64 Example of Procedure for Suppressing Synchronous Counter Clearing for Channels 0 to 2 and Channels 6 and 7

- Examples of Suppression of Synchronous Counter Clearing for Channels 0, 1, and 2, and Channels 6 and 7

Figure 16.65 to figure 16.68 show examples of operation in which channels 6 and 7 operate in complementary PWM mode and synchronous counter clearing for channels 0, 1, and 2, and channels 6 and 7 is suppressed by setting the SCC bit in MT67WCR in channels 6 and 7 to 1.

In the examples shown in figure 16.65 to figure 16.68, synchronous counter clearing occurs at timing (3), (6), (8), and (11) shown in figure 16.57, respectively. In these examples, the WRE bit in MT67WCR in channels 6 and 7 is set to 1.

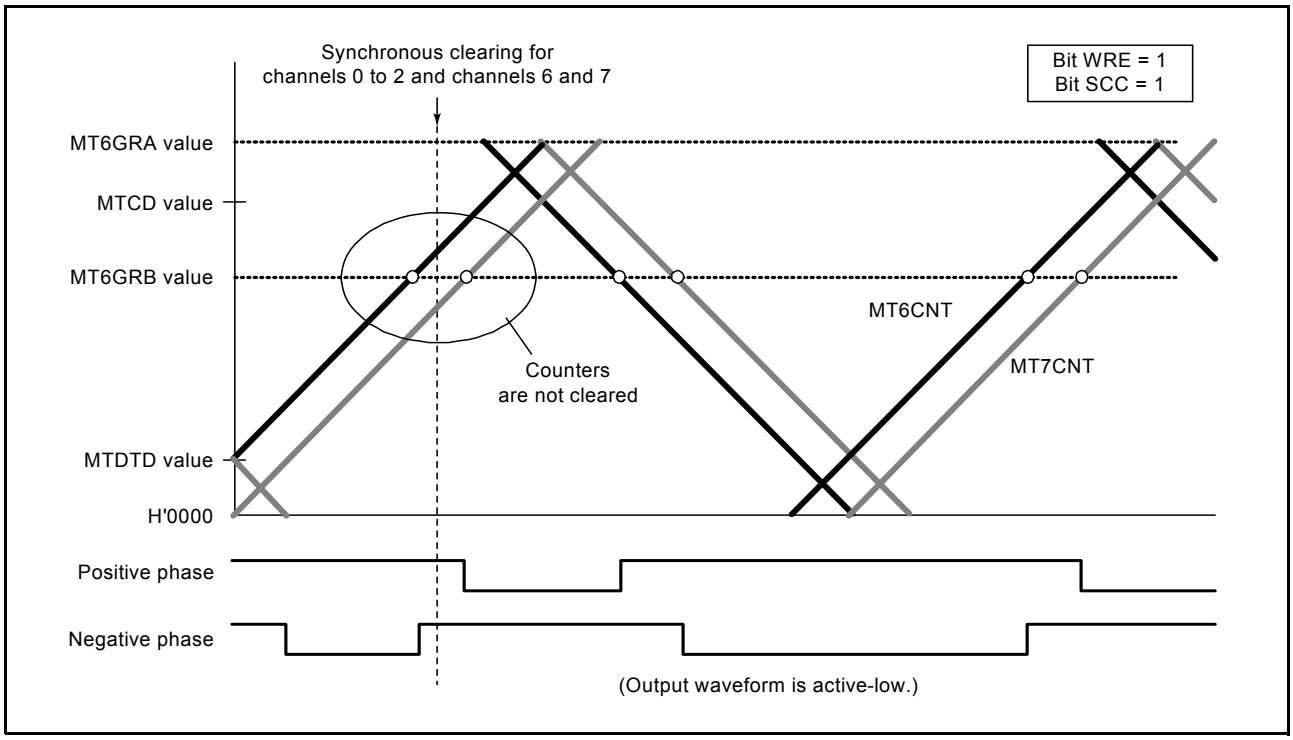


Figure 16.65 Example of Synchronous Clearing in Dead Time during Up-Counting (Timing (3) in Figure 16.57; Bit WRE is 1 and Bit SCC is 1 in MT67WCR in Channels 6 and 7))

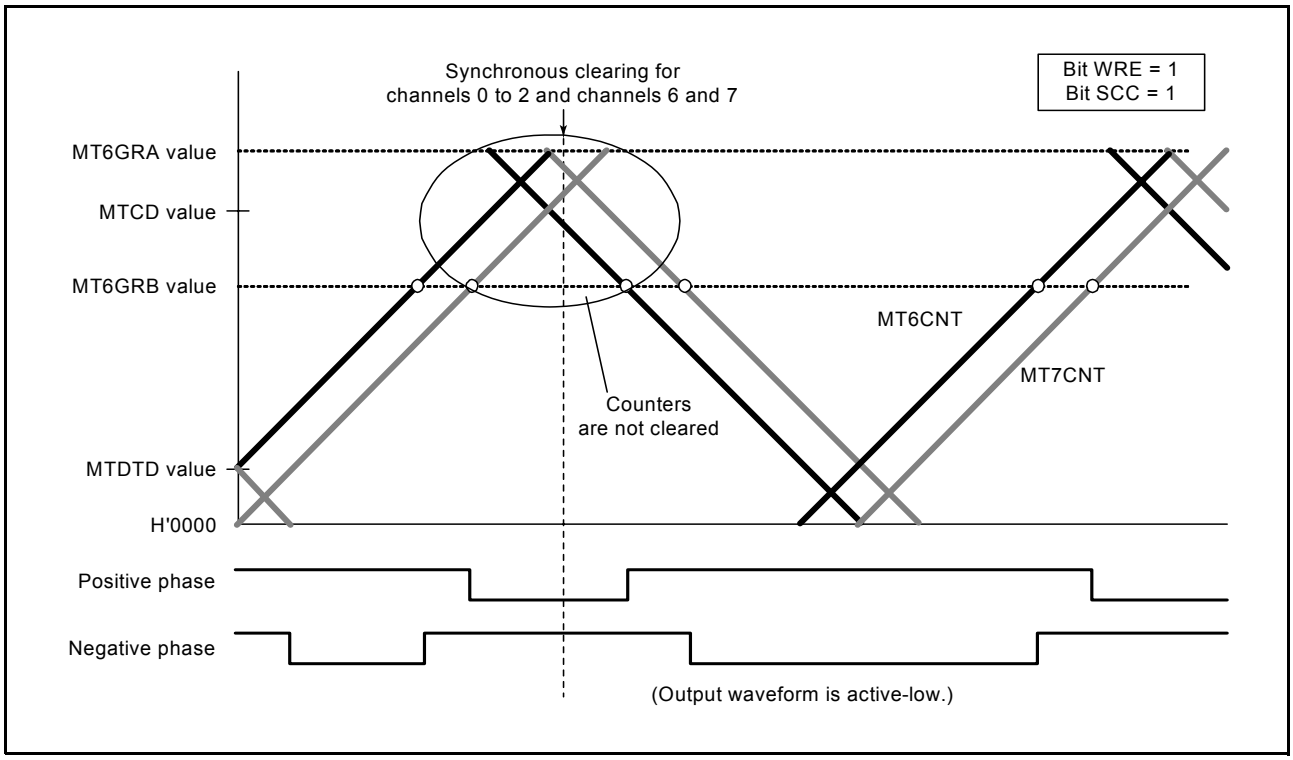


Figure 16.66 Example of Synchronous Clearing in Interval Tb at Crest (Timing (6) in Figure 16.57; Bit WRE is 1 and Bit SCC is 1 in MT67WCR in Channels 6 and 7))

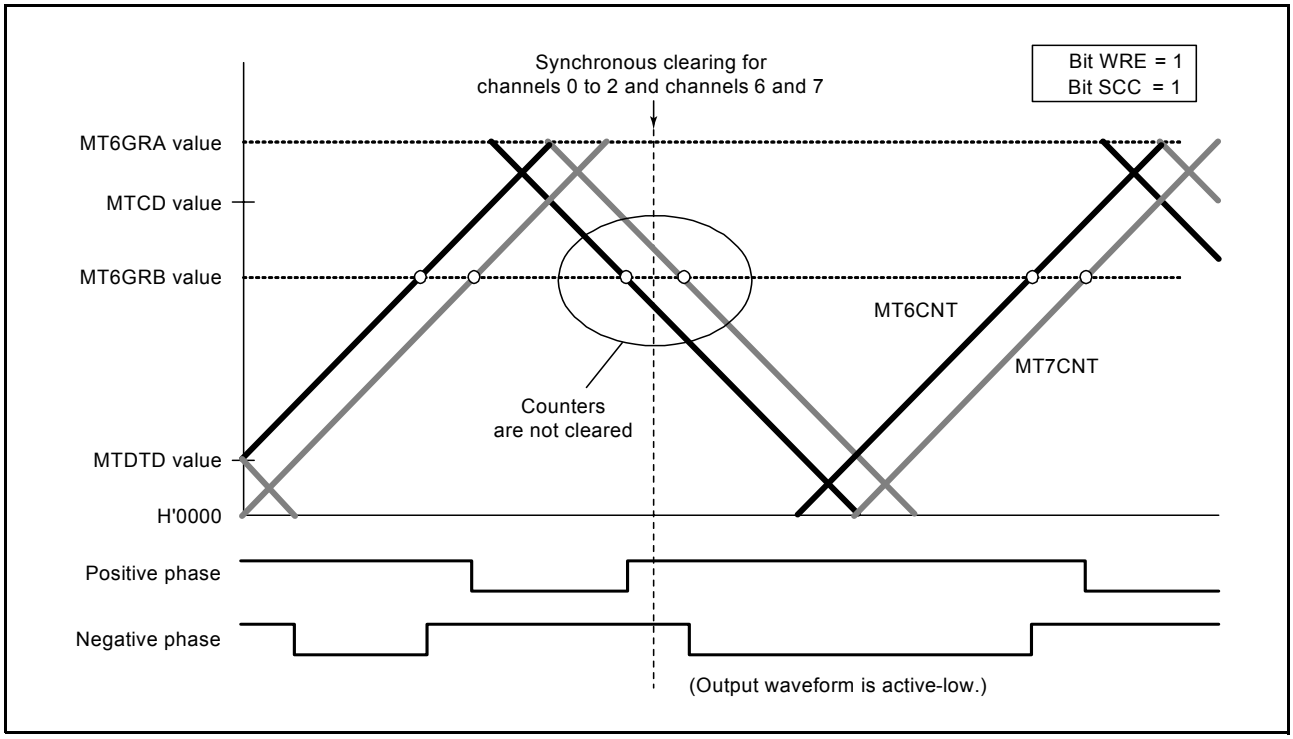


Figure 16.67 Example of Synchronous Clearing in Dead Time during Down-Counting (Timing (8) in Figure 16.57; Bit WRE is 1 and Bit SCC is 1 in MT67WCR in Channels 6 and 7))

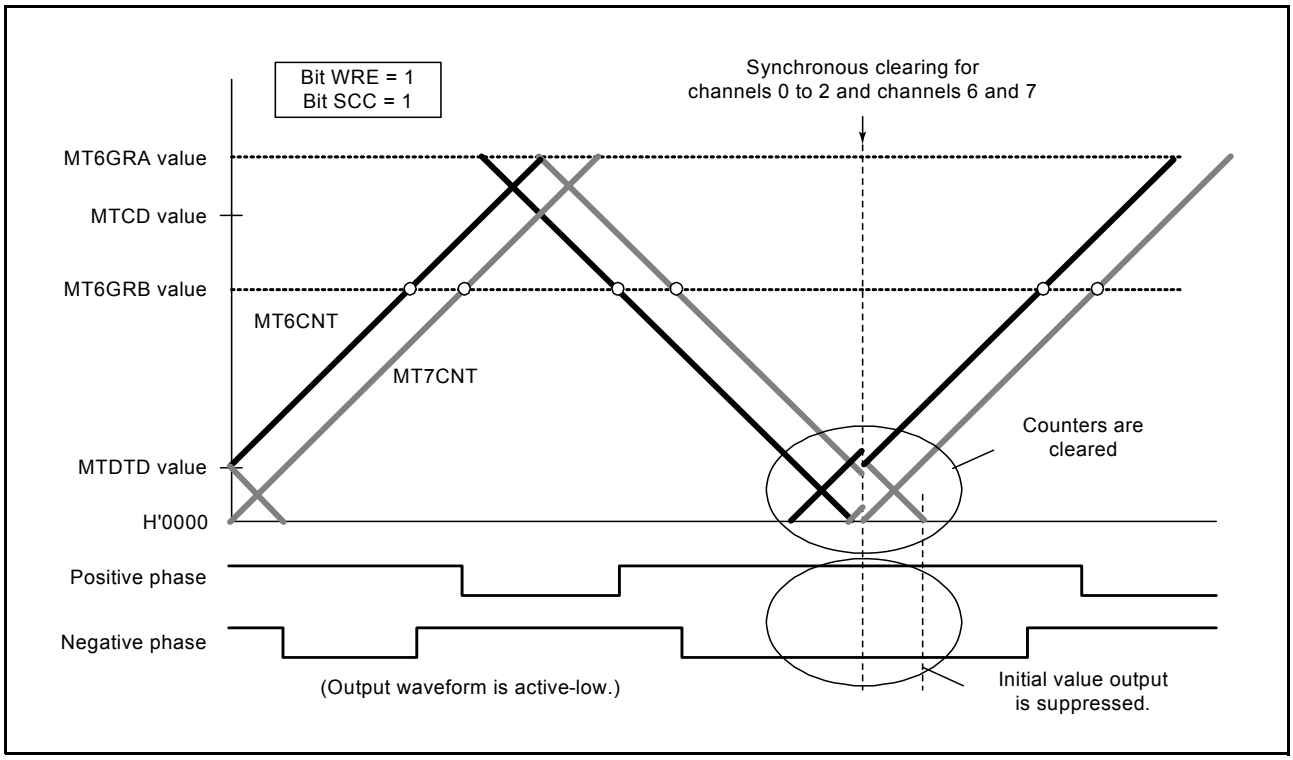


Figure 16.68 Example of Synchronous Clearing in Interval Tb at Trough (Timing (11) in Figure 16.57; Bit WRE is 1 and Bit SCC is 1 in MT67WCR in Channels 6 and 7)

(p) Counter Clearing by MT3GRA (MT6GRA) Compare Match

In complementary PWM mode, by setting the CCE bit in MTWCR, it is possible to have counters MT3CNT (MT6CNT), MT4CNT (MT7CNT), and MT34SCNT (MT67SCNT) cleared by MT3GRA (MT6GRA) compare match.

Figure 16.69 illustrates an operation example.

Notes.
- Use this function only in complementary PWM mode 1 (transfer at crest)
- Do not specify synchronous clearing by another channel. (Do not set the SYNC0 to SYNC4, SYNC6, and SYNC7 bits in MTSY to 1 or the CE0A, CE0B, CE0C, CE0D, CE1A, CE1B, CE1C, and CE1D bits in MT6SYC to 1.)
- Do not set the PWM duty value to H'0000.
- Do not set the PSYE bit of MTOCR0 to 1.

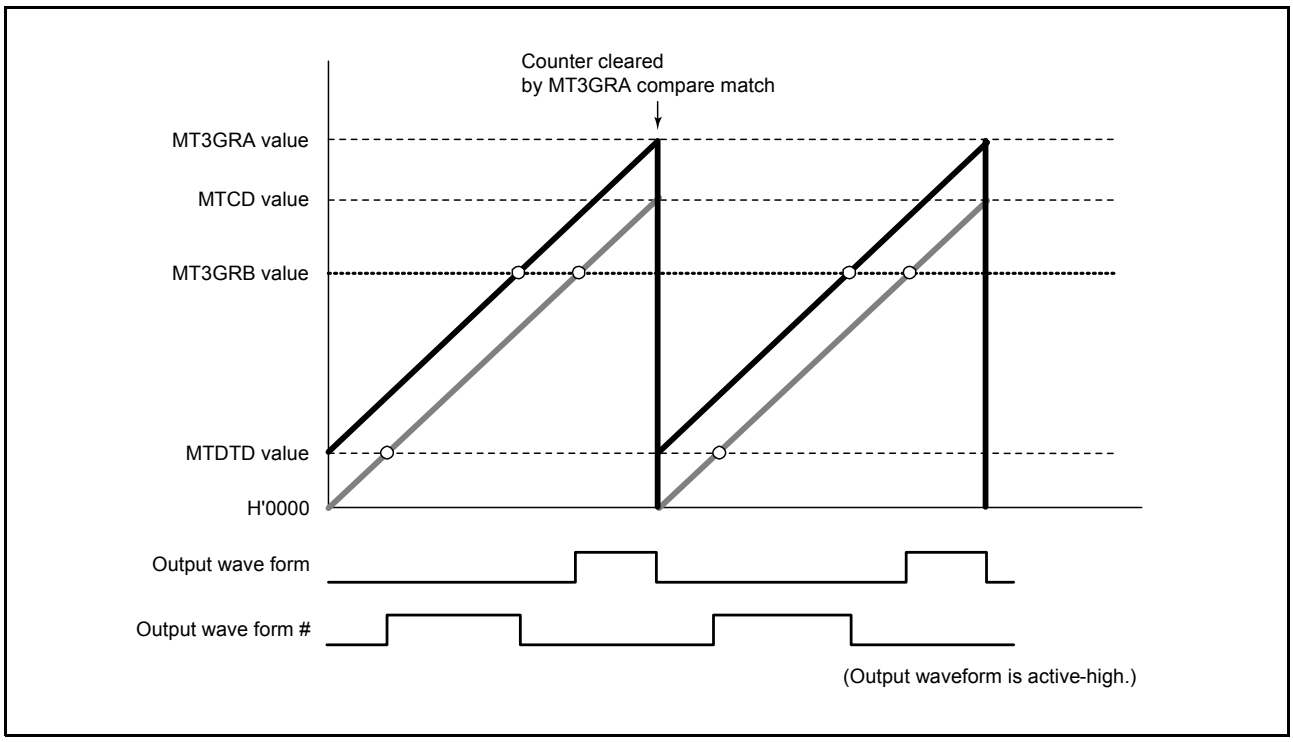


Figure 16.69 Example of Counter Clearing Operation by MT3GRA Compare Match

(q) Example of AC Synchronous Motor (Brushless DC Motor) Drive Waveform Output

In complementary PWM mode, a brushless DC motor can easily be controlled using MT34GCR. Figure 16.70 to figure 16.73 show examples of brushless DC motor drive waveforms created using MT34GCR.

When output phase switching for a 3-phase brushless DC motor is performed by means of external signals detected with a Hall sensor, etc., clear the FB bit in MT34GCR to 0. In this case, the external signals indicating the polarity position are input to channel 0 timer input pins TIOC0A, TIOC0B, and TIOC0C. (For pin function settings, see section 13, I/O Ports.) When an edge is detected at pin TIOC0A, TIOC0B, or TIOC0C, the output on/off state is switched automatically.

When the FB bit is 1, the output on/off state is switched when the UF, VF, or WF bit in MT34GCR is cleared to 0 or set to 1.

The drive waveforms are output from the complementary PWM mode 6-phase output pins.

With this 6-phase output, in the case of on output, it is possible to use complementary PWM mode output and perform chopping output by setting the N bit or P bit to 1. When the N bit or P bit is 0, level output is selected.

The 6-phase output active level (on output level) can be set with the OLSN and OLSP bits in MT34OCR0 regardless of the setting of the N and P bits.

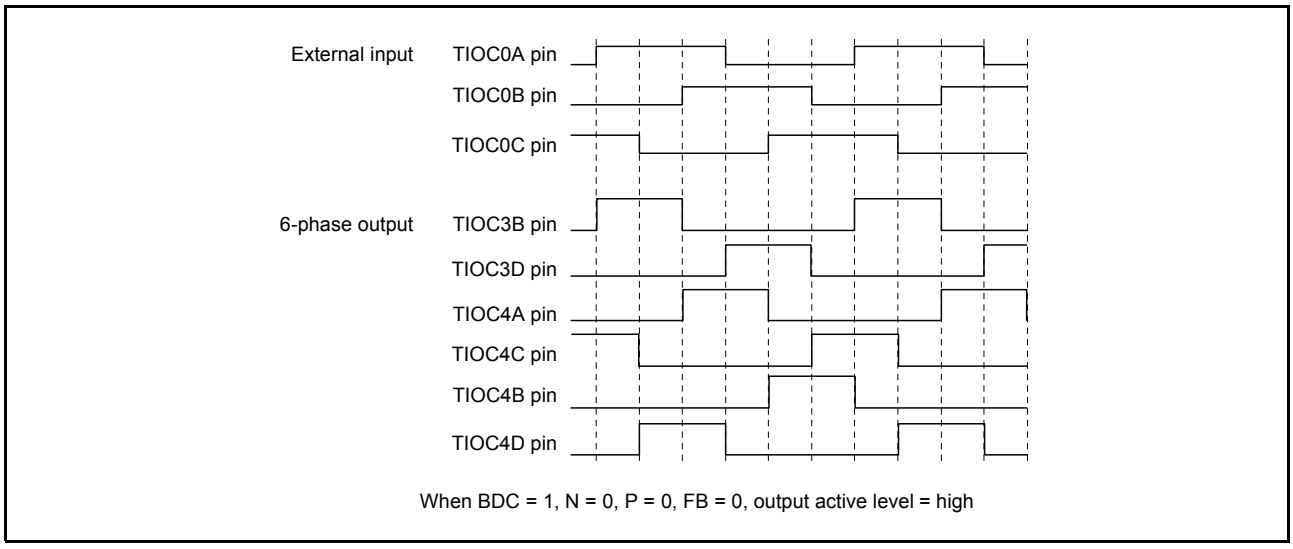


Figure 16.70 Example of Output Phase Switching by External Input (1)

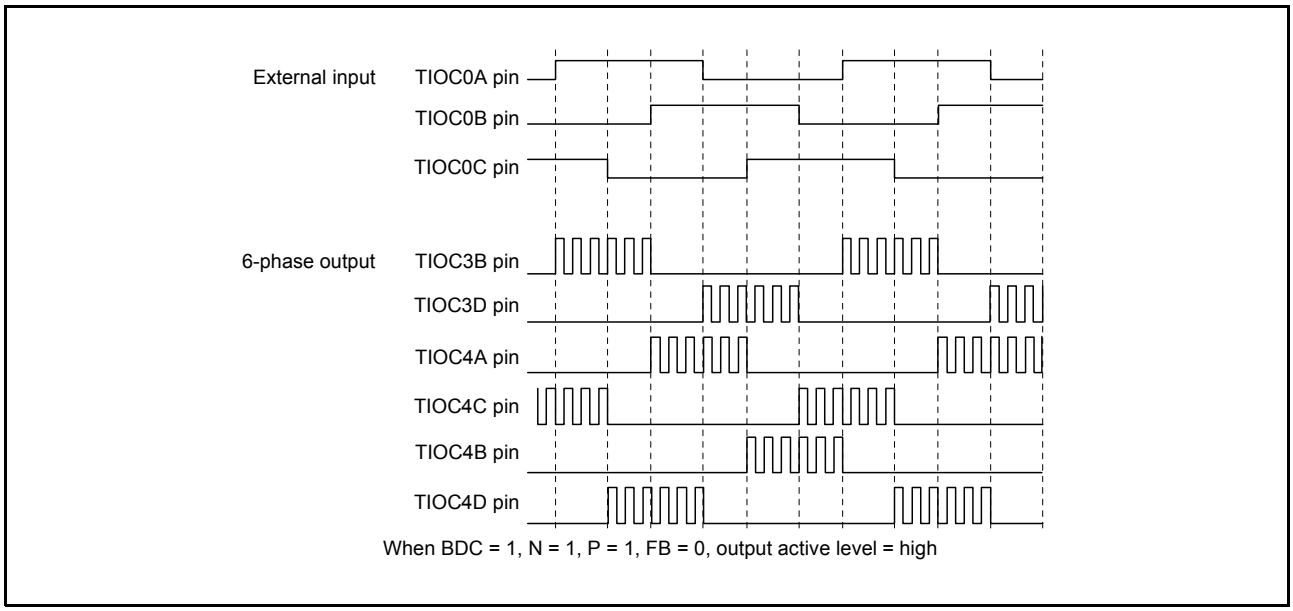


Figure 16.71 Example of Output Phase Switching by External Input (2)

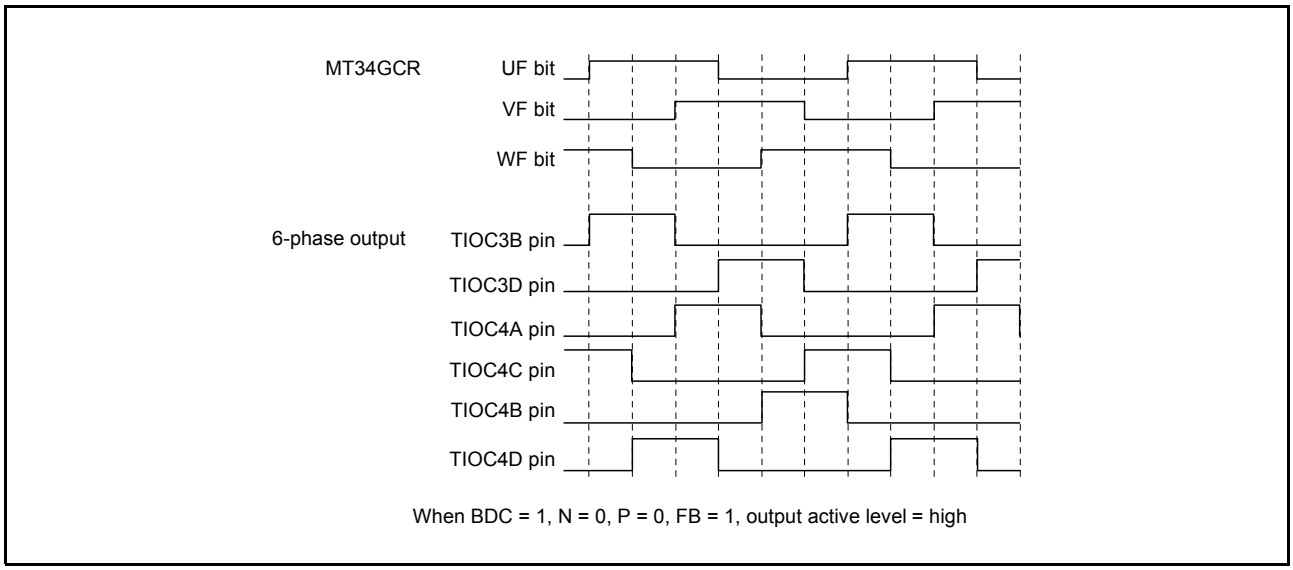


Figure 16.72 Example of Output Phase Switching by Means of UF, VF, WF Bit Settings (1)

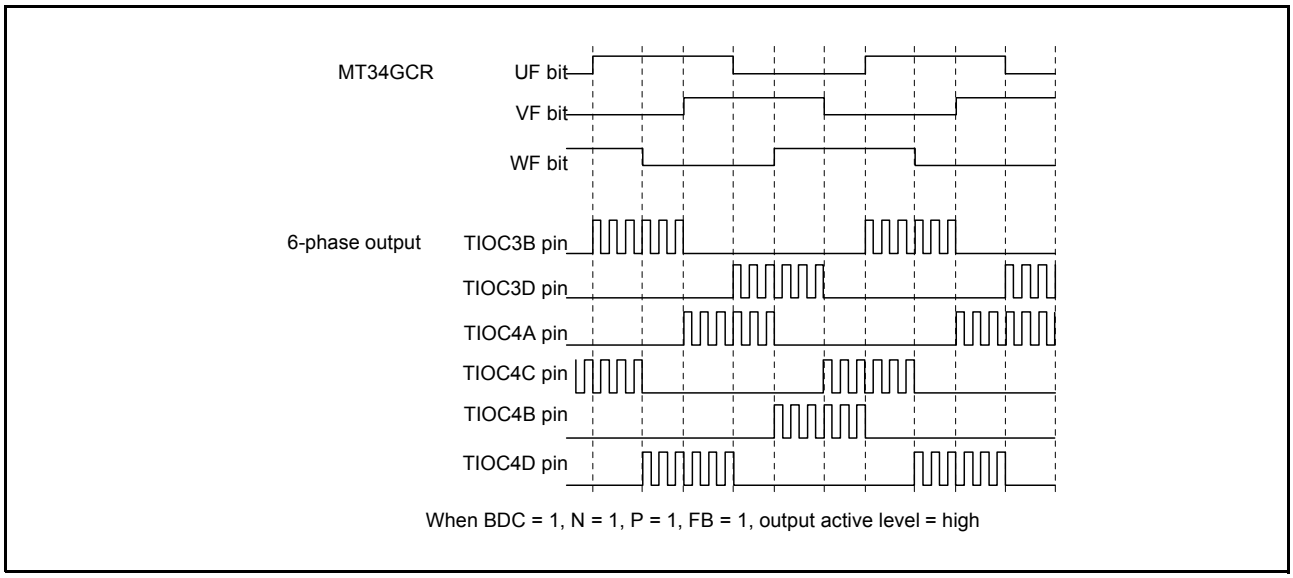


Figure 16.73 Example of Output Phase Switching by Means of UF, VF, WF Bit Settings (2)

(r) A/D Converter Start Request Setting

In complementary PWM mode, an A/D converter start request can be issued using a MT3GRA (MT6GRA) compare-match, MT4CNT (MT7CNT) counter underflow (trough), or compare-match on a channel other than channels 3 and 4 (channels 6 and 7).

When start requests using a MT3GRA (MT6GRA) compare-match are specified, A/D conversion can be started at the crest of the MT3CNT (MT6CNT) count.

A/D converter start requests can be set by setting the TTGE bit to 1 in MTIEN. To issue an A/D converter start request at an MT4CNT (MT7CNT) counter underflow (trough), set the TTGE2 bit in MT4IEN0 (MT7IEN0) to 1.

(s) Double Buffer Function in Complementary PWM Mode

In complementary PWM mode 3 (transfer at the crest and trough), the PWM output setting resolution can be improved from ±2 to ±1 by setting the DRS bit in MTMD1 to 1.

When setting buffer registers A (MT3GRD, MT4GRC, and MT4GRD), set also buffer registers B (MT3GRE, MT4GRE, and MT4GRF) at the same time. Buffer register B should be set to the buffer register A value or (buffer register A value – 1). For details of the setting procedure, refer to section 16.3.8 (1) Example of Complementary PWM Mode Setting Procedure.

Note: When buffer register B is set to the buffer register A value, symmetric PWM waveforms are output. When buffer register B is set to (buffer register A value – 1), asymmetric PWM waveforms are output.

Figure 16.74 shows an example of double buffer operation.

Each register data is transferred as follows.

- After MT4GRD (buffer A) is written to, data is transferred from MT4GRD (buffer A) to Temp3A (temporary A) and from MT4GRF (buffer B) to Temp3B (temporary B).
- With timing (1) in the figure, data is transferred from Temp3A (temporary A) to MT4GRB (compare).
- With timing (2) in the figure, data is transferred from Temp3B (temporary B) to MT4GRB (compare).

In the crest interval (Tb interval at crest), the compare register and temporary register A are valid; in the trough interval (Tb interval at trough), the compare register and temporary register B are valid.

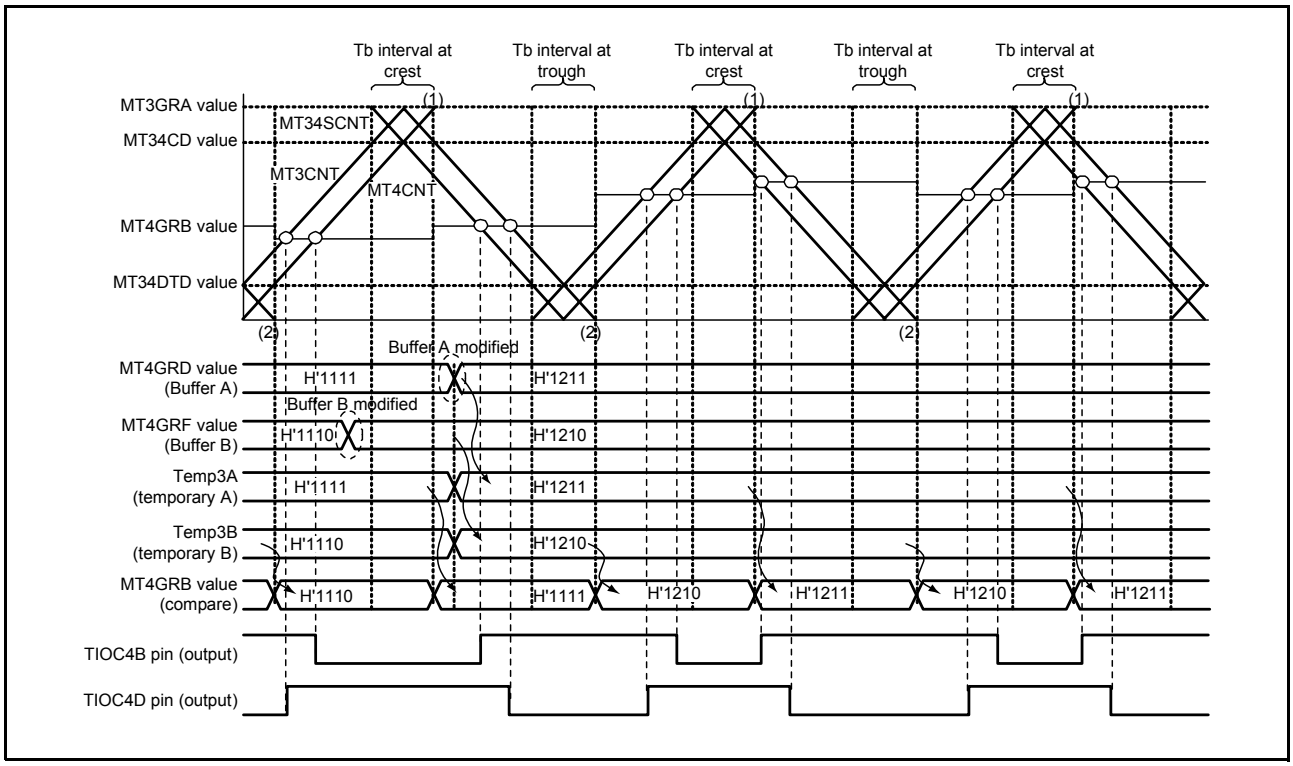


Figure 16.74 Example of Double Buffer Operation

Figure 16.75 shows an example when the buffer write value is smaller than the MT34DTD value, and figure 16.76 shows an example when the write value is greater than MT34CD.

In the crest interval, the output is controlled according to the compare match with the compare register or temporary register A; in the trough interval, the output is controlled according to the compare match with the compare register or temporary register B.

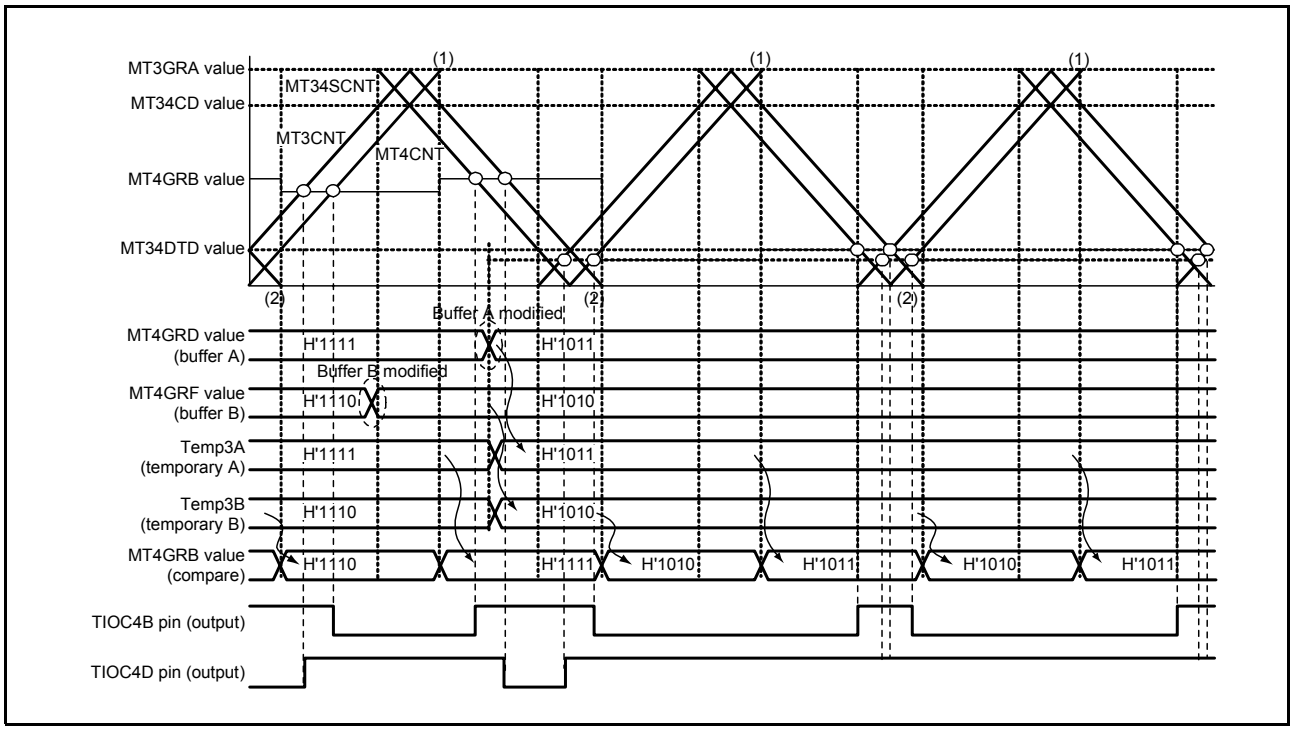


Figure 16.75 Example of Double Buffer Operation (Buffer Write Value is Smaller than MT34DTD)

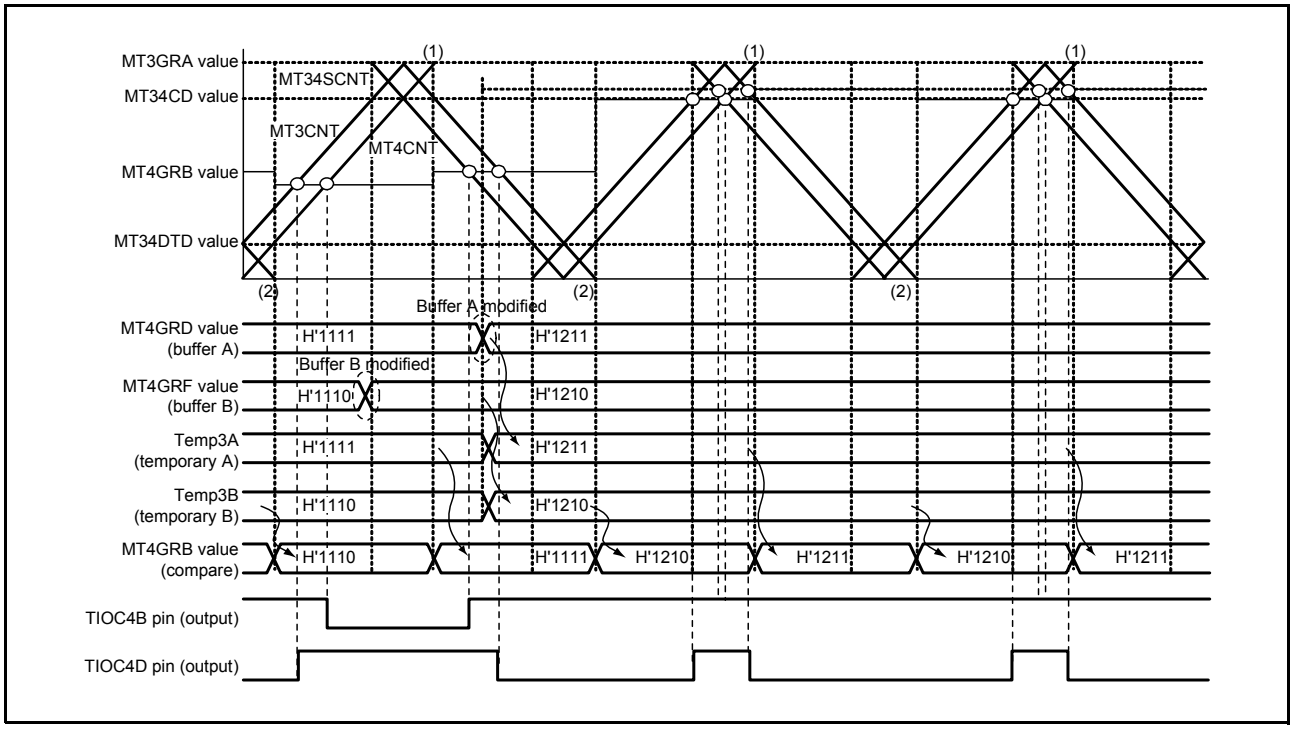


Figure 16.76 Example of Double Buffer Operation (Buffer Write Value is Greater than MT34CD)

(3) Interrupt Skipping Function 1 in Complementary PWM Mode

Interrupts TGIA_3 (TGIA_6) (at the crest) and TCIV_4 (TCIV_7) (at the trough) in channels 3 and 4 (channels 6 and 7) can be skipped up to seven times by making settings in the MT interrupt skipping set register 1 (MT34ISSE0 and MT67ISSE0).

Transfers from a buffer register to a temporary register or a compare register can be skipped in coordination with interrupt skipping by making settings in MTBTSE. For the linkage with buffer registers, refer to description (c) Buffer Transfer Control Linked with Interrupt Skipping, below.

A/D converter start requests generated by the A/D converter start request delaying function can also be skipped in coordination with interrupt skipping by making settings in MTADSRCR. For the linkage with the A/D converter start request delaying function, refer to section 16.3.9, A/D Converter Start Request Delaying Function.

The setting of the MT interrupt skipping set register 0 (MT34ISSE0 or MT67ISSE0) must be done while interrupt skipping function 1 is selected by setting the TITM bit to 0 in MT34 interrupt skipping mode register (MT34ISMD or MT67ISMD) and the TGIA_3 (TGIA_6) and TCIV_4 (TCIV_7) interrupt requests are disabled by the settings of MT3IEN0 and MT4IEN0 (MT6IEN0 and MT7IEN0) along with under the conditions in which TGFA_3 (TGFA_6) and TCFV_4 (TCFV_7) flag settings by compare match never occur. Before changing the skipping count, be sure to clear the T3AEN (T6AEN) and T4VEN (T7VEN) bits to 0 to clear the skipping count counter.

(a) Example of Interrupt Skipping 1 Operation Setting Procedure

Figure 16.77 shows an example of procedure for setting interrupt skipping function 1, and figure 16.78 shows the periods during which interrupt skipping count can be changed.

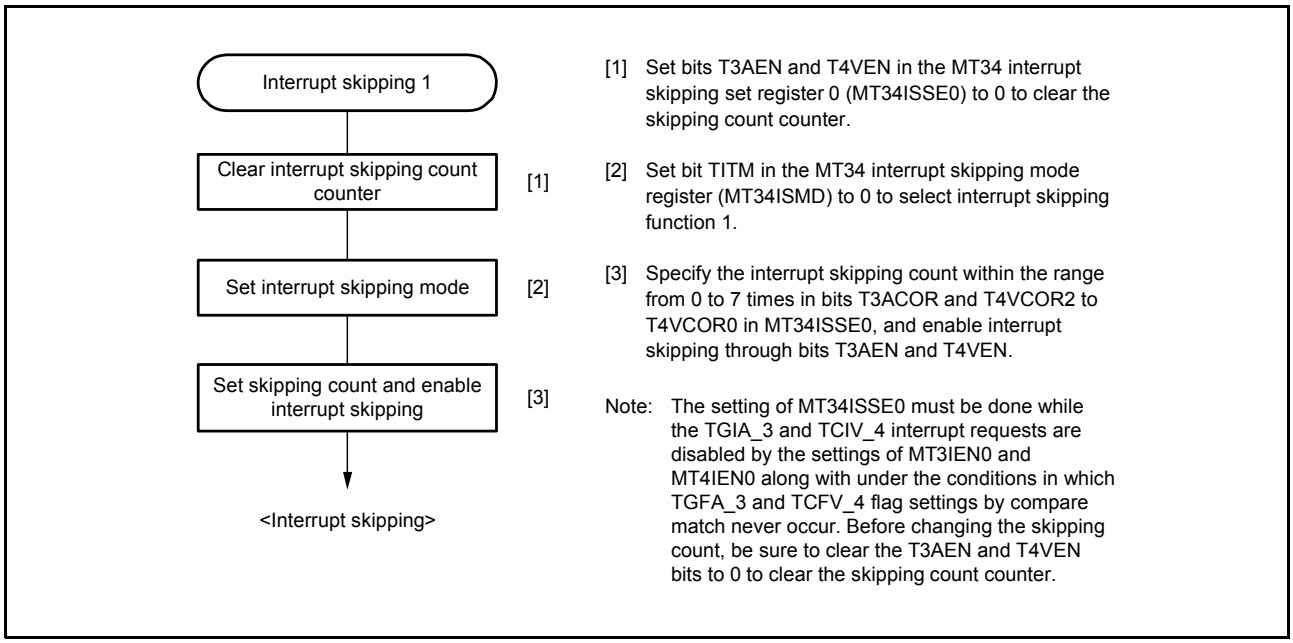


Figure 16.77 Example of Interrupt Skipping 1 Operation Setting Procedure

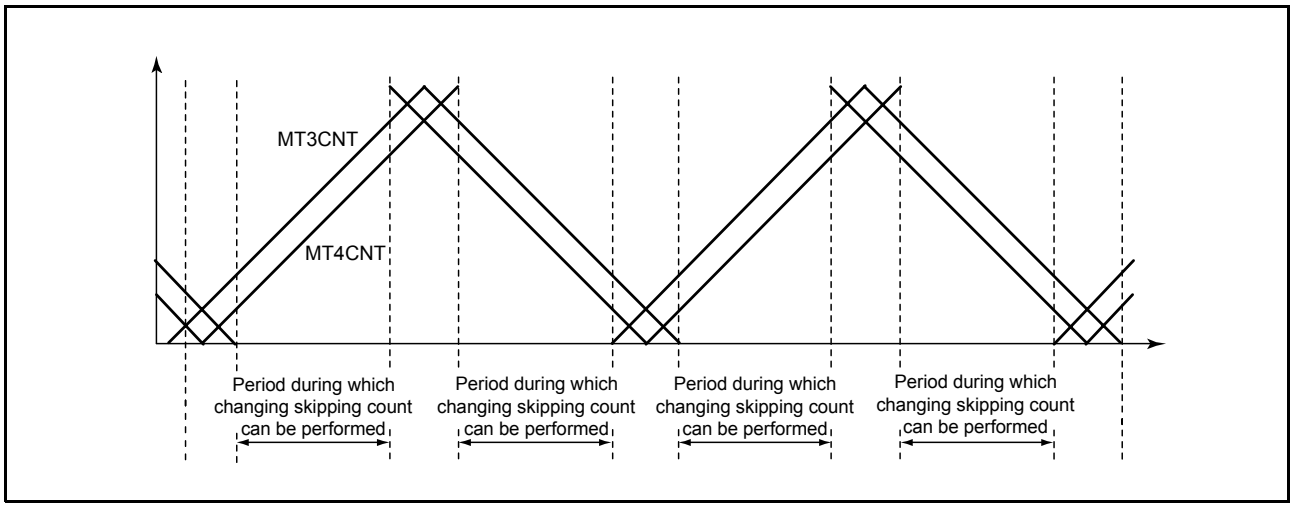


Figure 16.78 Periods during which Interrupt Skipping Count can be Changed

(b) Example of Interrupt Skipping 1 Operation

Figure 16.79 shows an example of TGIA_3 (TGIA_6) interrupt skipping in which the interrupt skipping count is set to three by the T3ACOR (T6ACOR) bit and the T3AEN (T6AEN) bit is set to 1 in MT interrupt skipping set register 0 (MT34ISSE0 or MT67ISSE0).

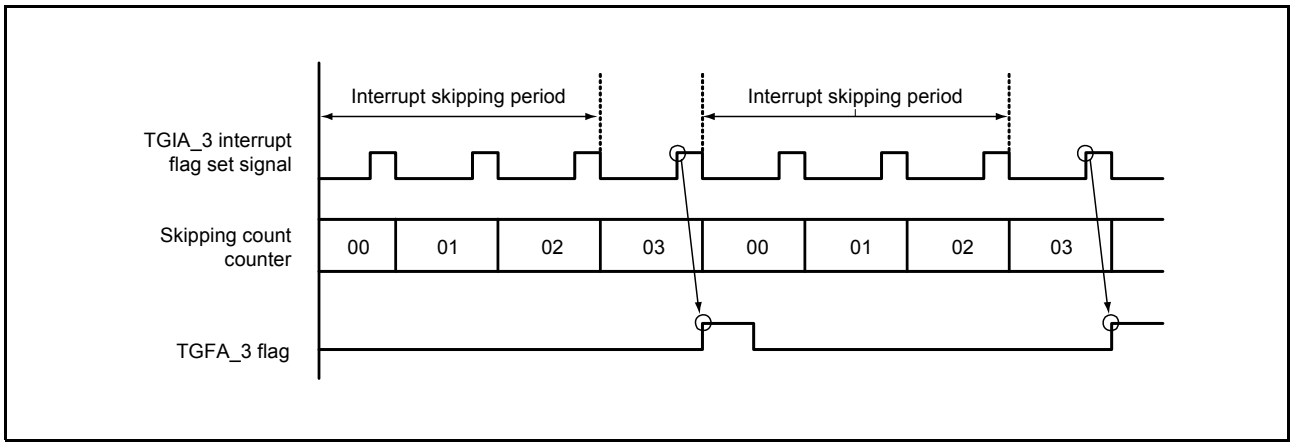


Figure 16.79 Example of Interrupt Skipping 1 Operation

(c) Buffer Transfer Control Linked with Interrupt Skipping

In complementary PWM mode, whether to transfer data from a buffer register to a temporary register and whether to link the transfer with interrupt skipping can be specified with the BTE1 and BTE0 bits in MTBTSE.

Figure 16.80 shows an example of operation when buffer transfer is suppressed (BTE1 = 0 and BTE0 = 1). While this setting is valid, data is not transferred from the buffer register to the temporary register.

Figure 16.81 shows an example of operation when buffer transfer is linked with interrupt skipping (BTE1 = 1 and BET0 = 0). While this setting is valid, data is not transferred from the buffer register outside the buffer transfer-enabled period.

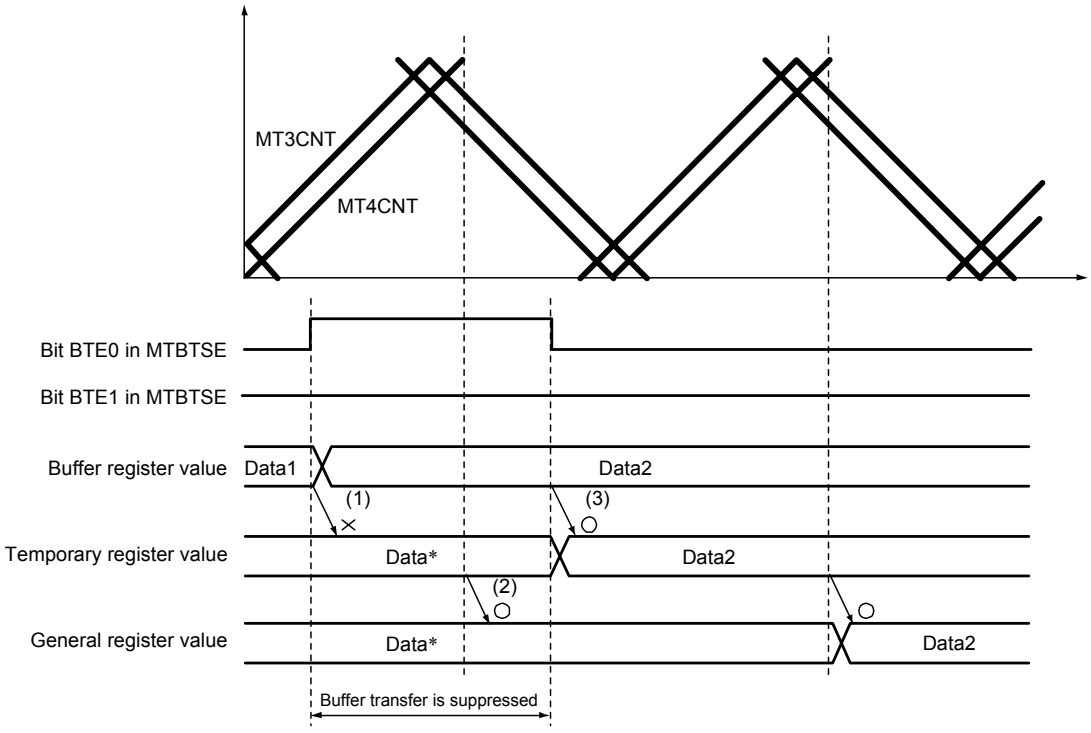
Note that the buffer transfer-enabled period depends on the T3AEN (T6AEN) and T4VEN (T7VEN) bit settings in MT34 interrupt skipping set register 0 (MT34ISSE0 or MT67ISSE0).

Figure 16.82 shows the relationship between the T3AEN and T4VEN (T6AEN and T7VEN) bit settings in MT34ISSE0 (MT67ISSE0) and buffer transfer-enabled period.

Note: This function must always be used in combination with interrupt skipping.

When interrupt skipping is disabled (the T3AEN and T4VEN (T6AEN and T7VEN) bits are cleared to 0 in MT interrupt skipping set register 0 (MT34ISSE0 (MT67ISSE0)), or the skipping count set bits (T3ACOR and T4COR (T6ACOR and T7VCOR)) in MT34ISSE0 (MT67ISSE0) are cleared to 0), be sure to disable link of buffer transfer with interrupt skipping. (Clear the BTE1 bit in MTBTSE to 0).

If buffer transfer is linked with interrupt skipping while interrupt skipping is disabled, buffer transfer is never performed.

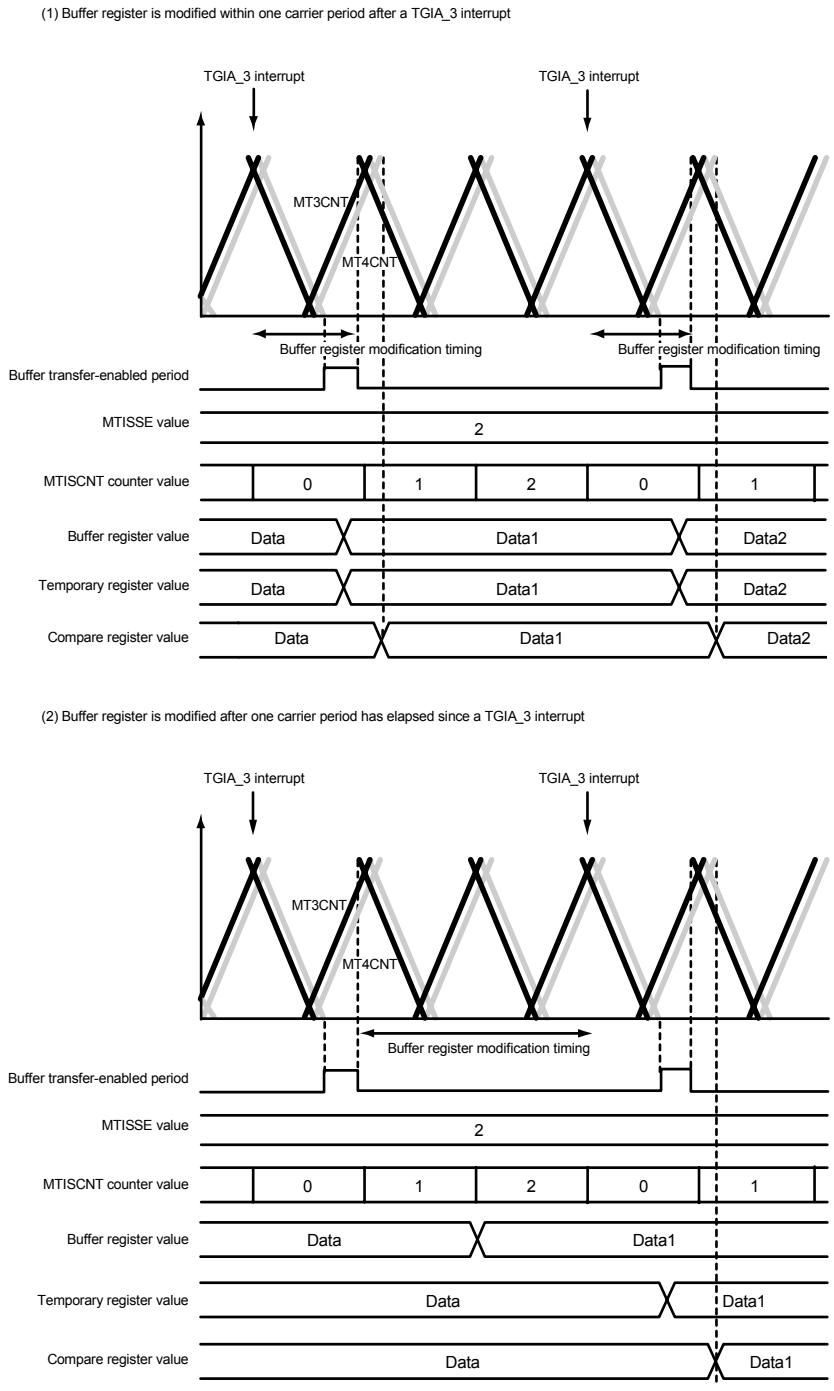


[Legend]

(1) No data is transferred from the buffer register to the temporary register in the buffer transfer-disabled period (bits BTE1 and BTE0 in MTBTSE are set to 0 and 1, respectively).
(2) Data is transferred from the temporary register to the general register even in the buffer transfer-disabled period.
(3) After buffer transfer is enabled, data is transferred from the buffer register to the temporary register.

Note: * When buffer transfer at the crest is selected.

Figure 16.80 Example of Operation when Buffer Transfer is Suppressed (BTE1 = 0 and BTE0 = 1)



Notes: Bit MD is set to 1101 in MT3MD0. (Buffer transfer at the crest is selected.)
The skipping count is set to two.
T3AEN in MT34ISSE0 is set to 1 and T4VEN in MT34ISSE0 is set to 0.

Figure 16.81 Example of Operation when Buffer Transfer is Linked with Interrupt Skipping (BTE1 = 1 and BTE0 = 0)

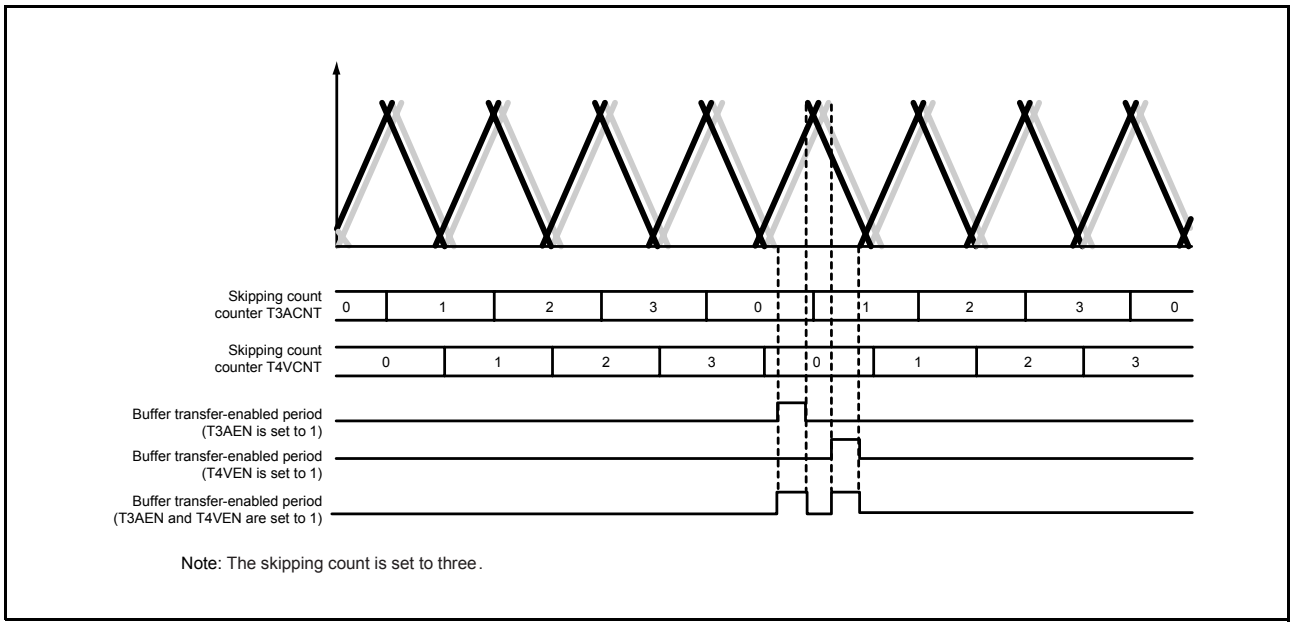


Figure 16.82 Relationship between Bits T3AEN and T4VEN in MT34 Interrupt Skipping Set Register 0 (MT34ISSE0) and Buffer Transfer-Enabled Period

(4) Complementary PWM Mode Output Protection Function

Complementary PWM mode output has the following protection functions.

(a) Register and counter miswrite prevention function

With the exception of the buffer registers, which can be rewritten at any time, access by the CPU can be enabled or disabled for the mode registers, control registers, compare registers, and counters used in complementary PWM mode by means of the RWE bit in MTRWEN. The applicable registers are some (43 in total) of the registers in channels 3, 4, 6, and 7 shown in the following:

MT3CR, MT4CR, MT3MD0, MT4MD0, MT3IOCR0, MT4IOCR0, MT3IOCR1, MT4IOCR1,
MT3IEN0, MT4IEN0, MT3CNT, MT4CNT, MT3GRA, MT4GRA, MT3GRB, MT4GRB,
MT34OEN, MT34OCR0, MT34OCR1, MT34GCR, MT34CD, MT34DTD
MT6CR, MT7CR, MT6MD0, MT7MD0, MT6IOCR0, MT7IOCR0, MT6IOCR1, MT7IOCR1,
MT6IEN0, MT7IEN0, MT6CNT, MT7CNT, MT6GRA, MT7GRA, MT6GRB, MT7GRB,
MT67OEN, MT67OCR0, MT67OCR1, MT67CD, MT67DTD

This function enables miswriting due to CPU runaway to be prevented by disabling CPU access to the mode registers, control registers, and counters. When the applicable registers are read in the access-disabled state, undefined values are returned. Writing to these registers is ignored.

(b) Halting of PWM output by external signal

The 6-phase PWM output pins can be set automatically to the high-impedance state by inputting specified external signals.

16.3.9 A/D Converter Start Request Delaying Function

A/D converter start requests can be issued in channel 4 (channel 7) by making settings in MTADSRCR, MTADSRCSE, and MTADSRCSEB.

The A/D converter start request delaying function compares the MT4CNT counter with MT4ADSRCSEA or MT4ADSRCSEB (the MT7CNT counter with MT7ADSRCSEA or MT7ADSRCSEB), and when their values match, the function issues a respective A/D converter start request (TRG4AN or TRG4BN (TRG7AN or TRG7BN)).

A/D converter start requests (TRG4AN and TRG4BN (TRG7AN and TRG7BN)) can be skipped in coordination with interrupt skipping 1 by making settings in the ITA3AE, ITA4VE, ITB3AE, and ITB4VE bits in MT4ADSRCR (ITA6AE, ITA7VE, ITB6AE, and ITB7VE bits in MT7ADSRCR).

(1) Example of Procedure for Specifying A/D Converter Start Request Delaying Function

Figure 16.83 shows an example of procedure for specifying the A/D converter start request delaying function.

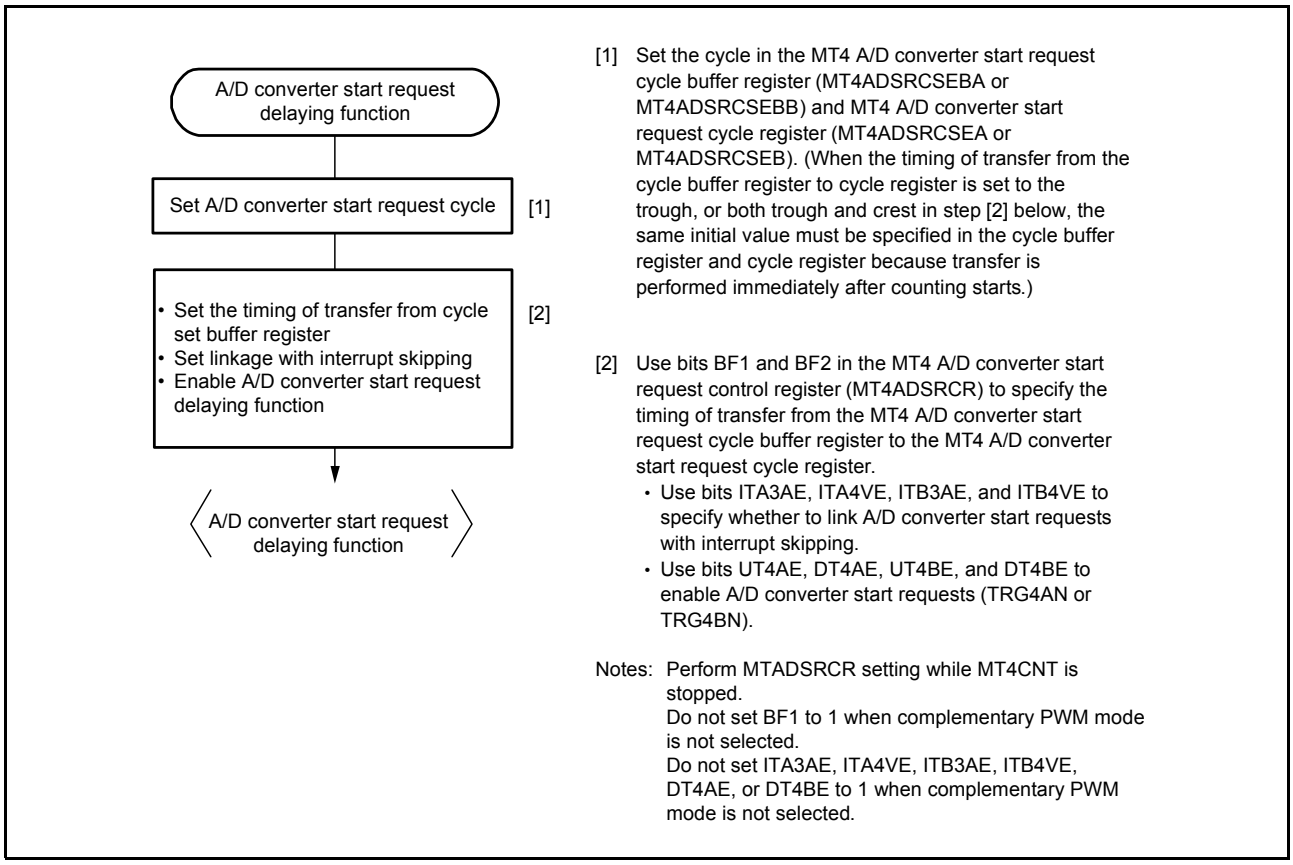


Figure 16.83 Example of Procedure for Specifying A/D Converter Start Request Delaying Function (Channels 3 and 4)

(2) Basic Operation Example of A/D Converter Start Request Delaying Function

Figure 16.84 shows a basic example of A/D converter start request signal (TRG4AN (TRG7AN)) operation when the trough of the MT4CNT (MT7CNT) counter is specified for the buffer transfer timing and an A/D converter start request signal is output during MT4CNT (MT7CNT) down-counting.

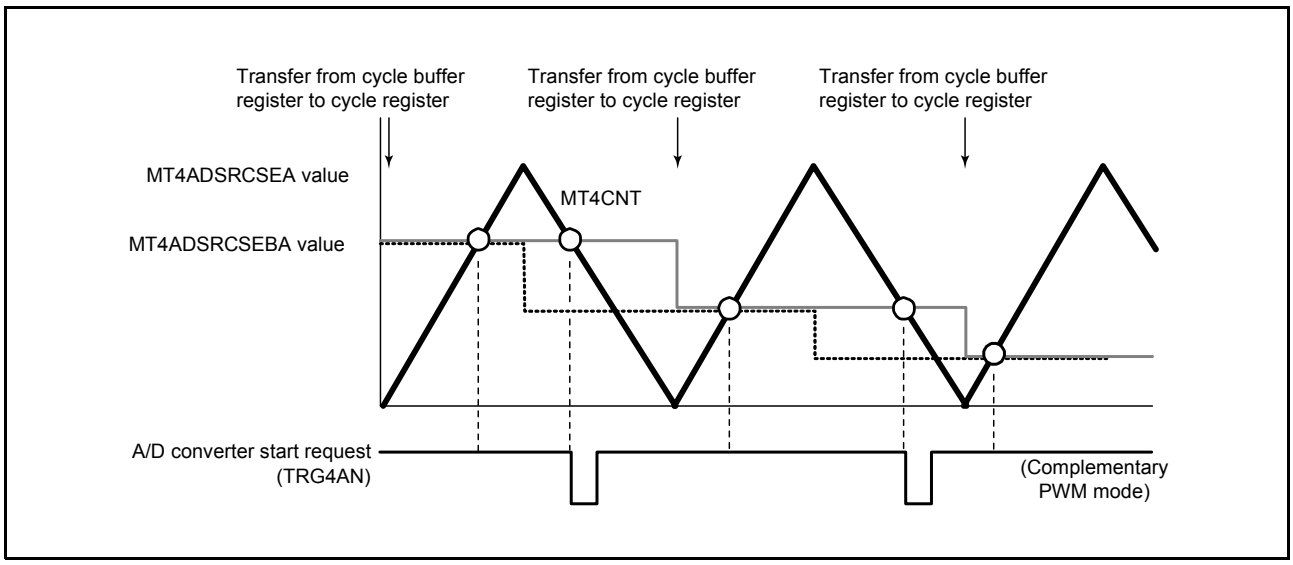


Figure 16.84 Basic Example of A/D Converter Start Request Signal (TRG4AN) Operation

(3) Buffer Transfer

The data in MTADSRCSE is updated by writing data to MTADSRCSEB. Data is transferred from the buffer registers to the respective cycle set registers at the timing selected with the BF bits in MTADSRCR.

In complementary PWM mode, data is also transferred from the MT A/D converter start request cycle set buffer registers to the MT A/D converter start request cycle set registers when MT4GRD and MT7GRD are updated.

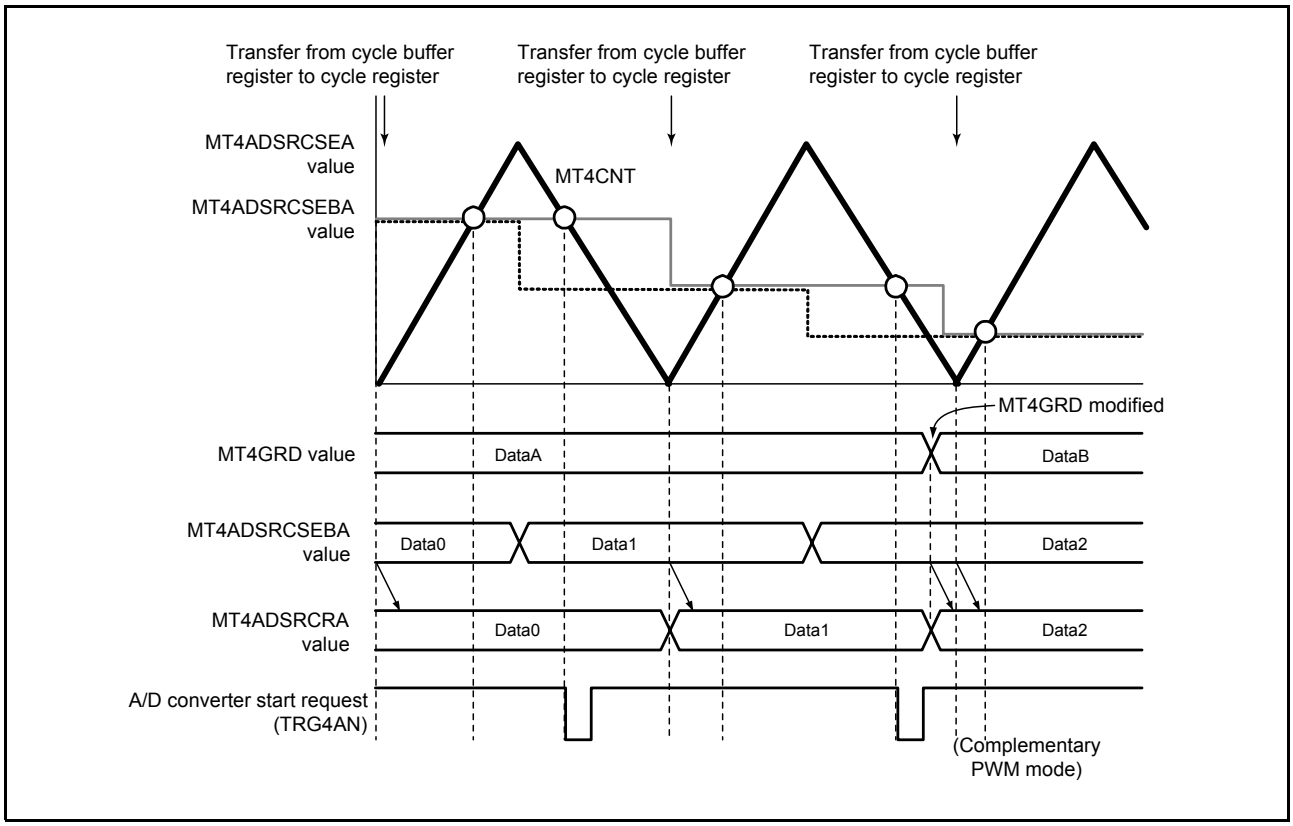


Figure 16.85 Example of A/D Converter Start Request Signal (TRG4AN) and Buffer Transfer Operation

(4) A/D Converter Start Request Delaying Function Linked with Interrupt Skipping Function 1

A/D converter start requests (TRG4AN and TRG4BN (TRG7AN and TRG7BN)) can be made in coordination with interrupt skipping 1 by making settings in the ITA3AE, ITA4VE, ITB3AE, and ITB4VE bits in MTADSRCR (ITA6AE, ITA7VE, ITB6AE, and ITB7VE bits).

Figure 16.86 shows an example of A/D converter start request signal (TRG4AN (TRG7AN)) operation when TRG4AN (TRG7AN) output is enabled during MT4CNT (MT7CNT) up-counting and down-counting and A/D converter start requests are linked with interrupt skipping function 1.

Figure 16.87 shows another example of A/D converter start request signal (TRG4AN (TRG7AN)) operation when TRG4AN (TRG7AN) output is enabled during MT4CNT (MT7CNT) up-counting and A/D converter start requests are linked with interrupt skipping function 1.

Note: This function must always be used in combination with interrupt skipping function 1.

When interrupt skipping is disabled (the T3AEN and T4VEN (T6AEN and T7VEN) bits in MT interrupt skipping set register 0 (MT34ISSE0 (MT67ISSE0)) are cleared to 0, or the skipping count set bits (T3ACOR and T4VCOR (T6ACOR and T7VCOR)) in MT34ISSE0 (MT67ISSE0) are cleared to 0), make sure that A/D converter start requests are not linked with interrupt skipping 1 operation (clear the ITA3AE, ITA4VE, ITB3AE, and ITB4VE (ITA6AE, ITA7VE, ITB6AE, and ITB7VE) bits in MTADSRCR to 0).

When this function is used, MT4ADSRCSEA and MT4ADSRCSEB (MT7ADSRCSEA and MT7ADSRCSEB) should be set with the value ranging H'0002 to the set value in MT34CD minus 2 (H'0002 to the set value in MT67CD minus 2).

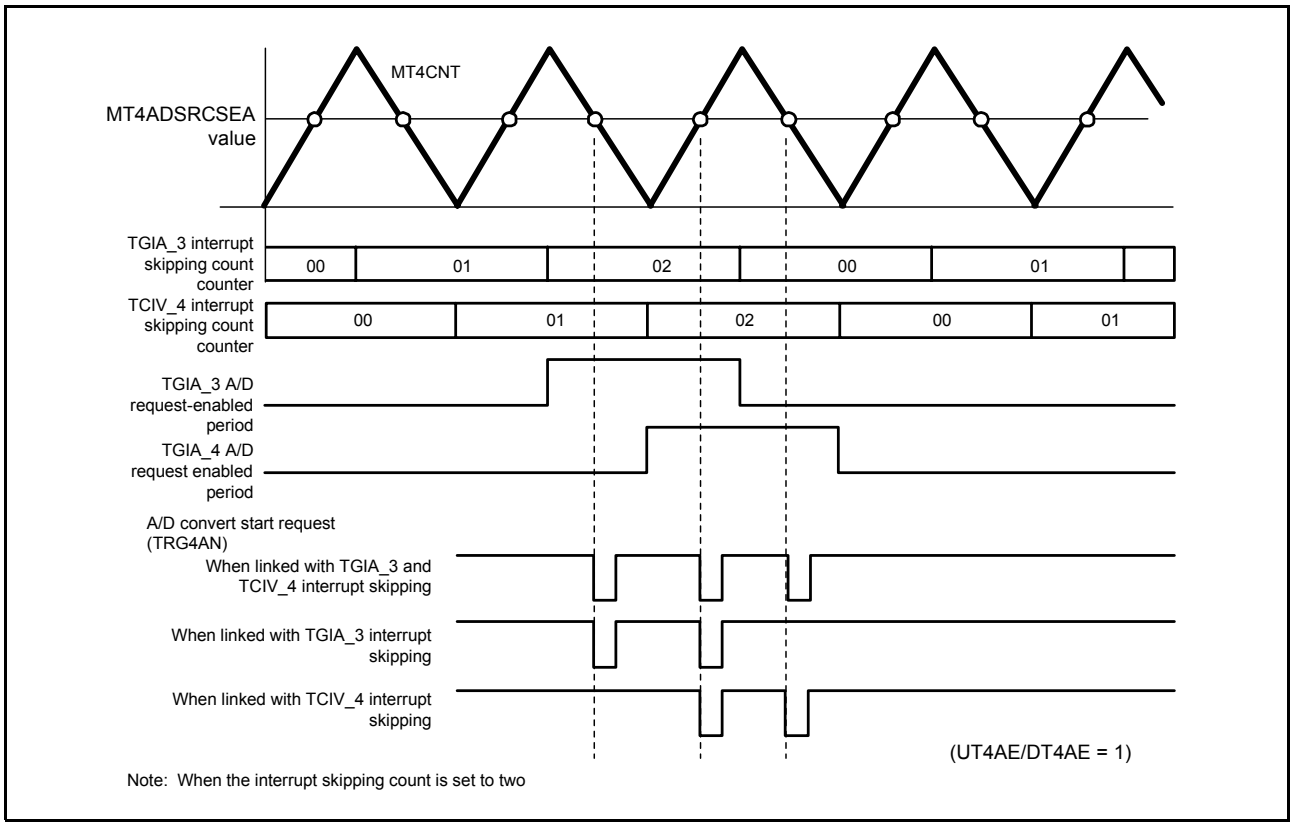


Figure 16.86 Example of A/D Converter Start Request Signal (TRG4AN) Operation Linked with Interrupt Skipping Function 1

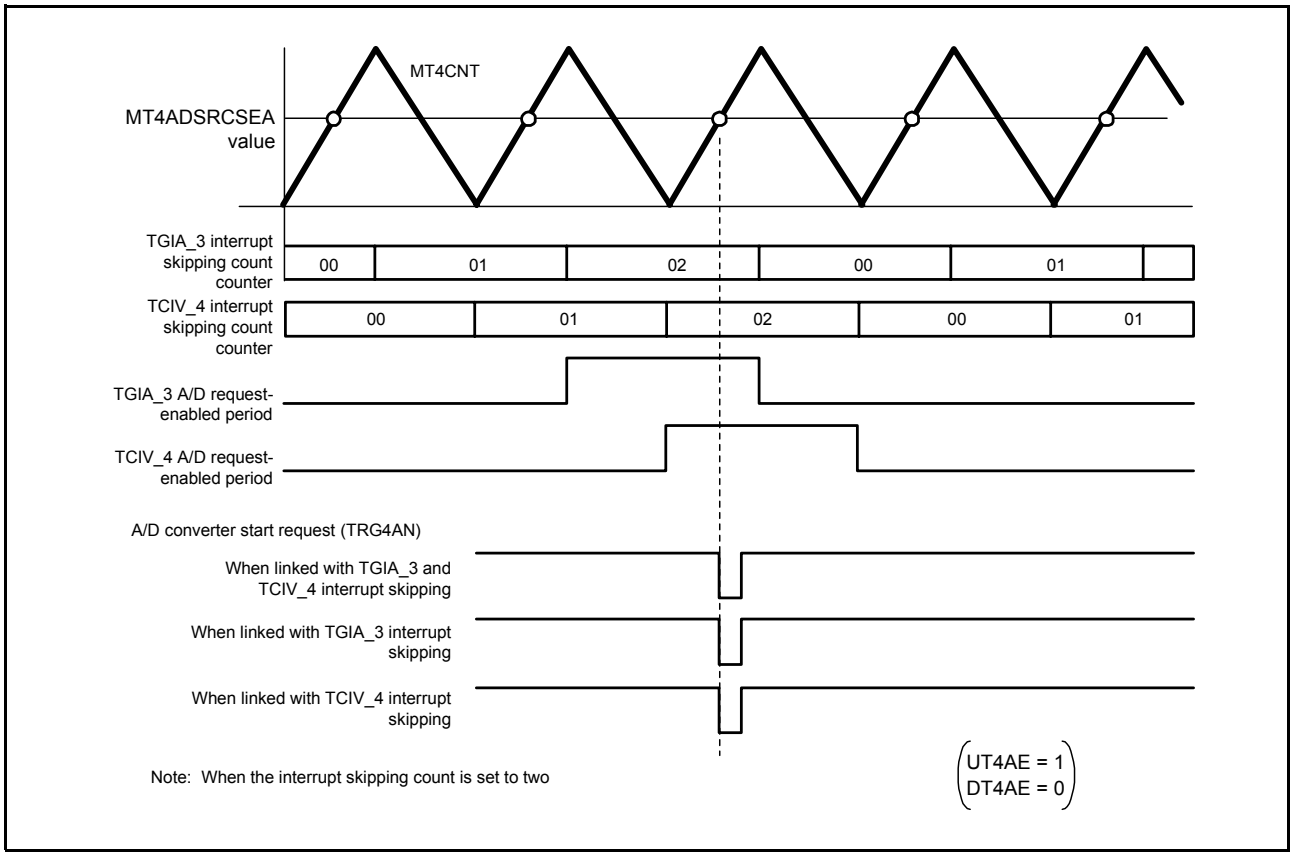


Figure 16.87 Example of A/D Converter Start Request Signal (TRG4AN) Operation Linked with Interrupt Skipping Function 1

(5) A/D Converter Start Request Delaying Function Linked with Interrupt Skipping Function 2

By setting the TITM bit to 1 in the MT34 interrupt skipping mode register (MT34ISMD or MT67ISMD), the counter starts down-counting from the value (0 to 7) set in the TRG4COR (TRG4COR) bits in MT4 interrupt skipping set register 1 (MT4ISSE1 (MT7ISSE1)) every time an A/D converter start trigger (TGR4AN or TRG4BN (TGR7AN or TRG7BN)) is generated. When the counter value reaches 0 and is reloaded, the TRG4AN and TRG4BN (TRG7AN and TRG7BN) interrupts become valid and an A/D converter start request signal (TRG4ABN (TRG7ABN)) is output. This function is valid only when the A/D converter request delaying function is enabled.

(a) Example of Procedure for Setting Interrupt Skipping Function 2

Figure 16.88 shows an example of procedure for setting interrupt skipping function 2.

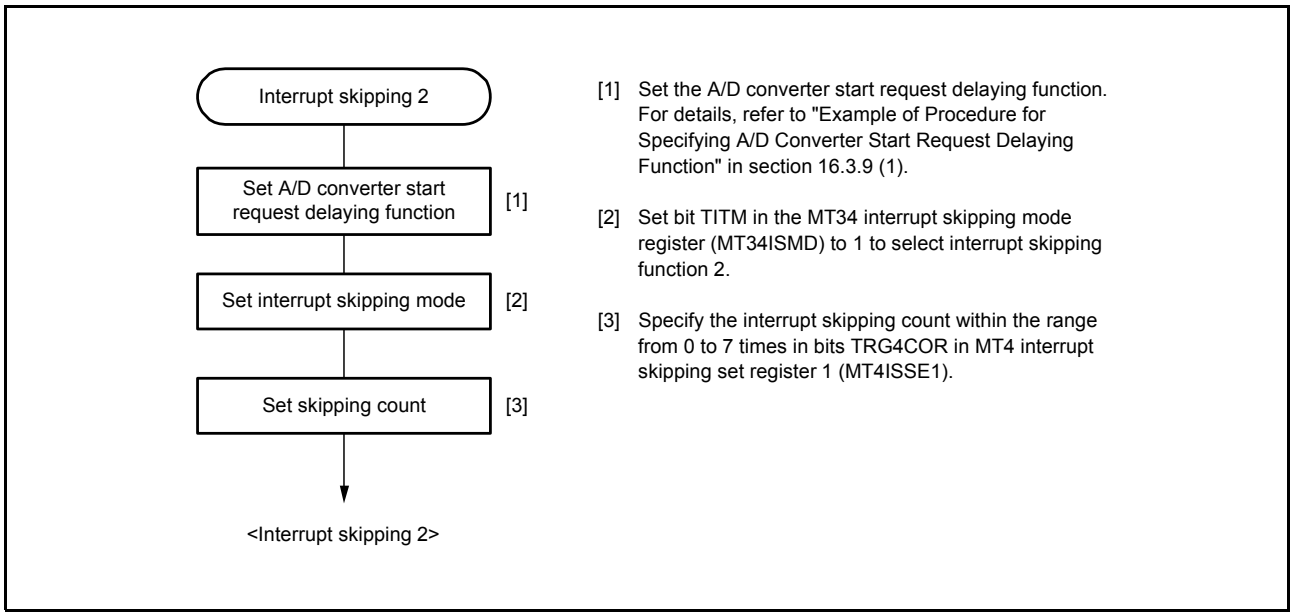


Figure 16.88 Example of Procedure for Setting Interrupt Skipping Function 2

(b) Example of Interrupt Skipping 2 Operation

Figure 16.89 shows an example of interrupt skipping 2 operation.

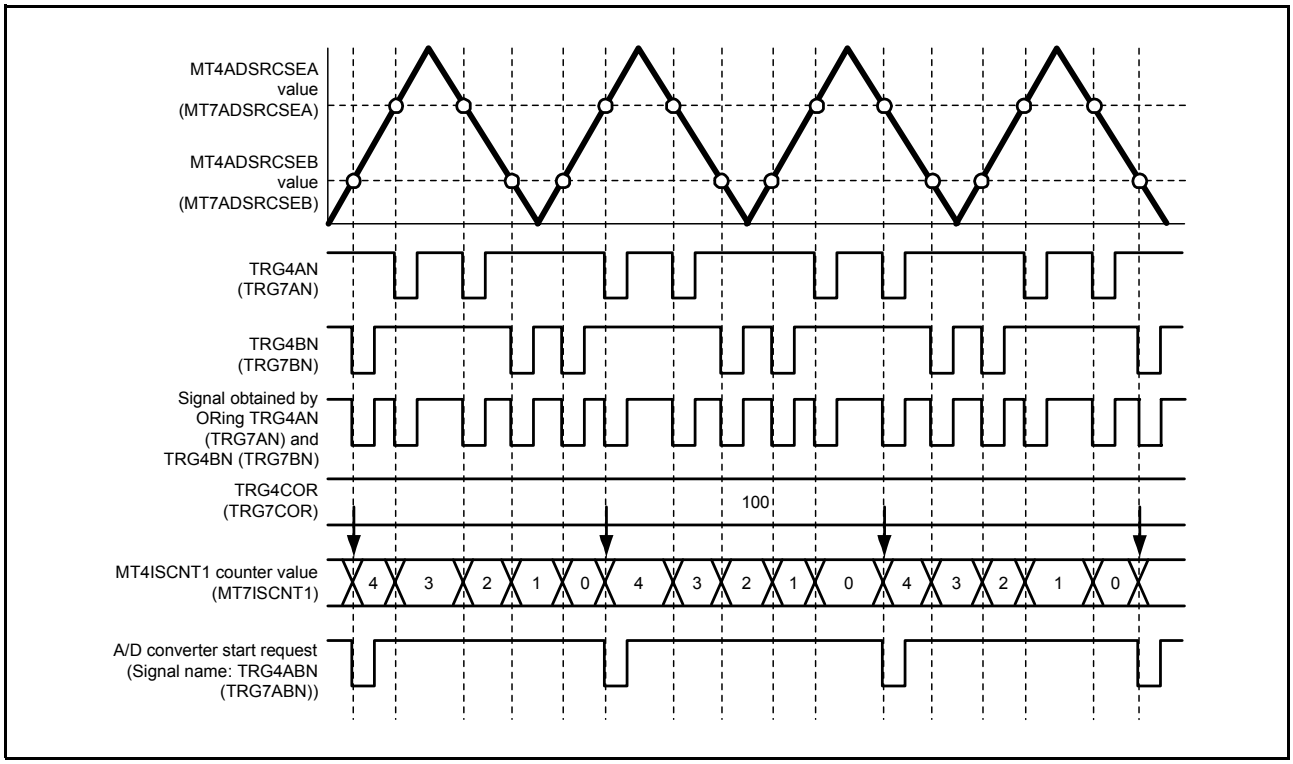


Figure 16.89 Example of Interrupt Skipping 2 Operation (Skipping Count is Set to Four)

16.3.10 Synchronous Operation between Channels 0 to 4 and Channels 6 and 7

(1) Synchronous Counter Start for Channels 0 to 4 and Channels 6 and 7

The counters in channels 0 to 4 and channels 6 and 7, which operate at different clock systems, can be started synchronously by making the MTCSYSTR settings.

(a) Example of Procedure for Setting Synchronous Counter Start for Channels 0 to 4 and Channels 6 and 7

Figure 16.90 shows an example of synchronous counter start setting procedure.

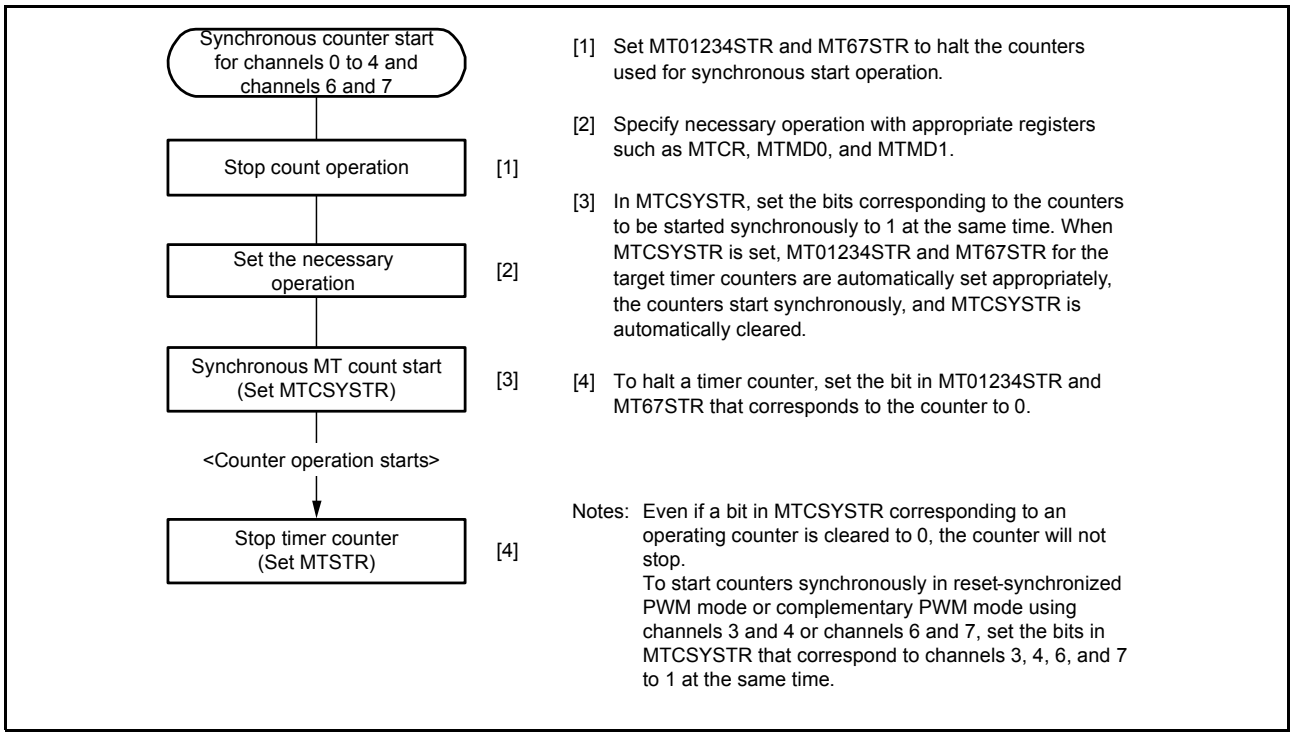


Figure 16.90 Example of Synchronous Counter Start Setting Procedure

(b) Examples of Synchronous Counter Start Operation

Figure 16.91 shows an example of synchronous counter start operation for channels 0 to 4 and channels 6 and 7.

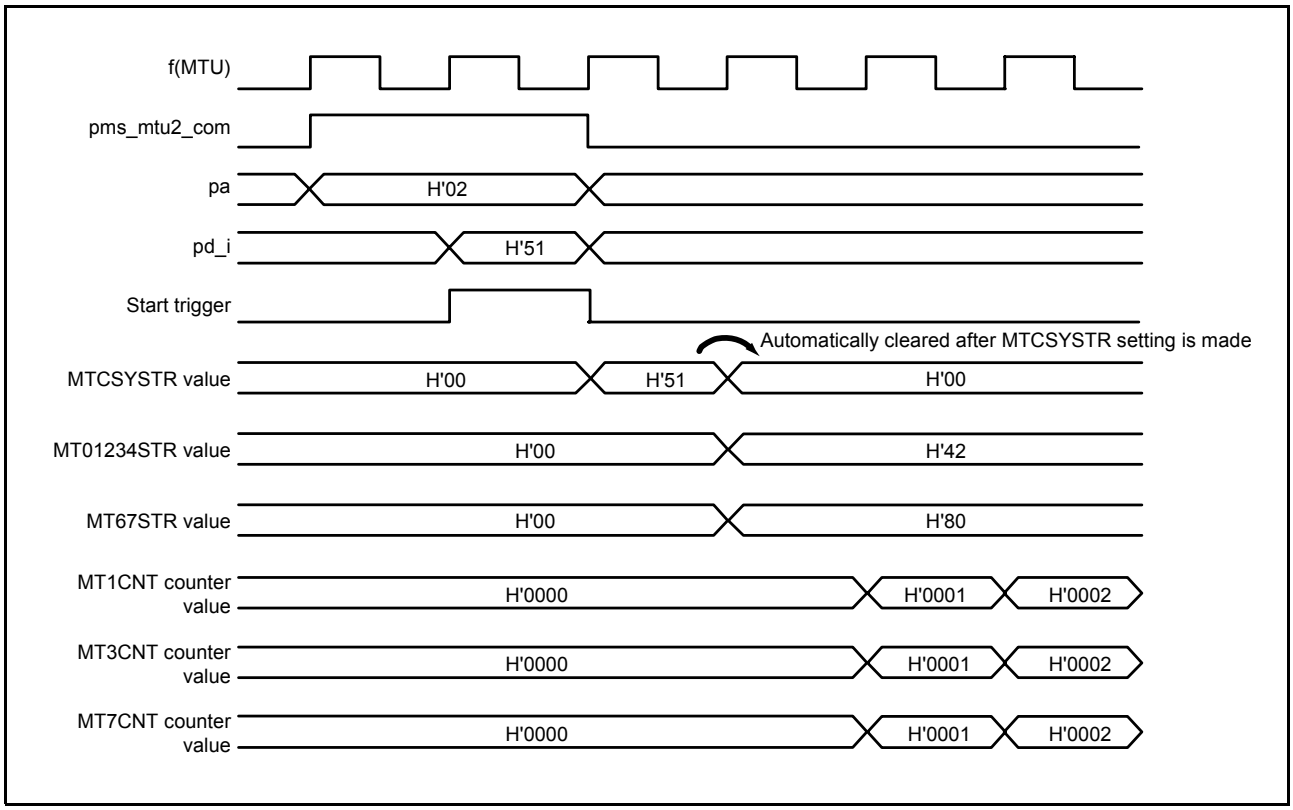


Figure 16.91 Example of Synchronous Counter Start Operation

(2) Clearing Counters of Channels 6 and 7 by Flag Setting Sources (Synchronous Counter Clearing for Channels 6 and 7)

The counters in channels 6 and 7 can be cleared by sources for setting the flags in MT0SR0 to MT2SR0 through the MT6SYC setting.

(a) Example of Procedure for Specifying Counter Clearing for Channels 6 and 7 by Flag Setting Sources

Figure 16.92 shows an example of procedure for specifying counter clearing for channels 6 and 7 by flag setting sources.

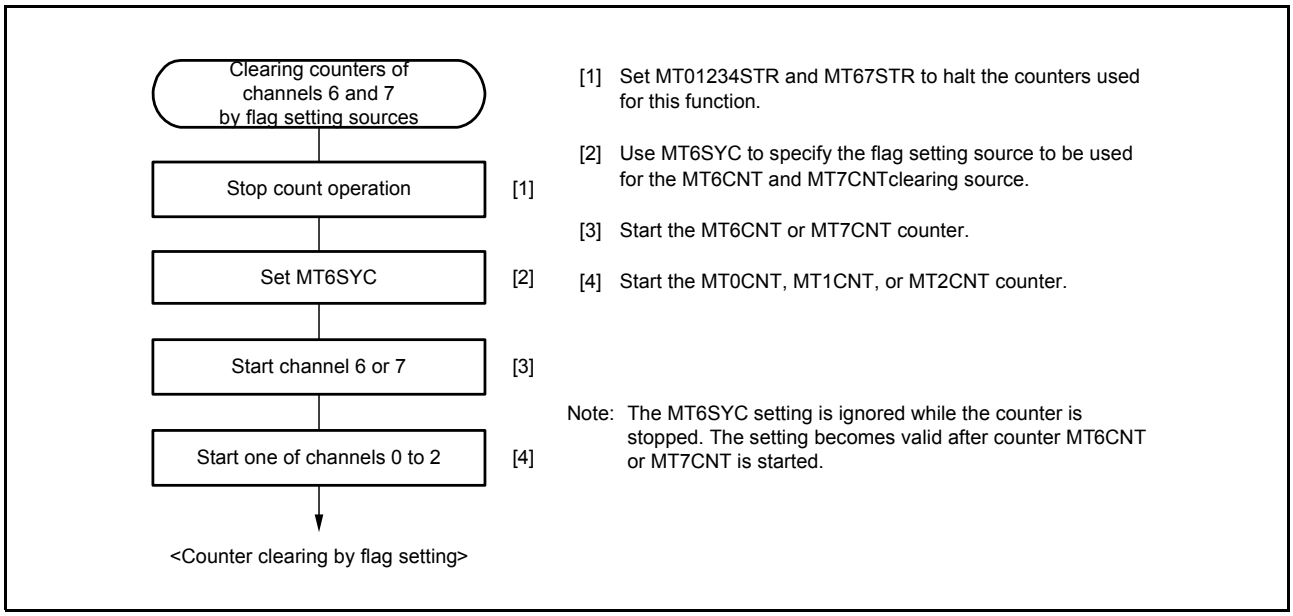


Figure 16.92 Example of Procedure for Specifying Counter Clearing for Channels 6 and 7 by Flag Setting Sources

(b) Examples of Counter Clearing for Channels 6 and 7 by Flag Setting Sources

Figure 16.93 and figure 16.94 shows examples of counter clearing for channels 6 and 7 by flag setting sources.

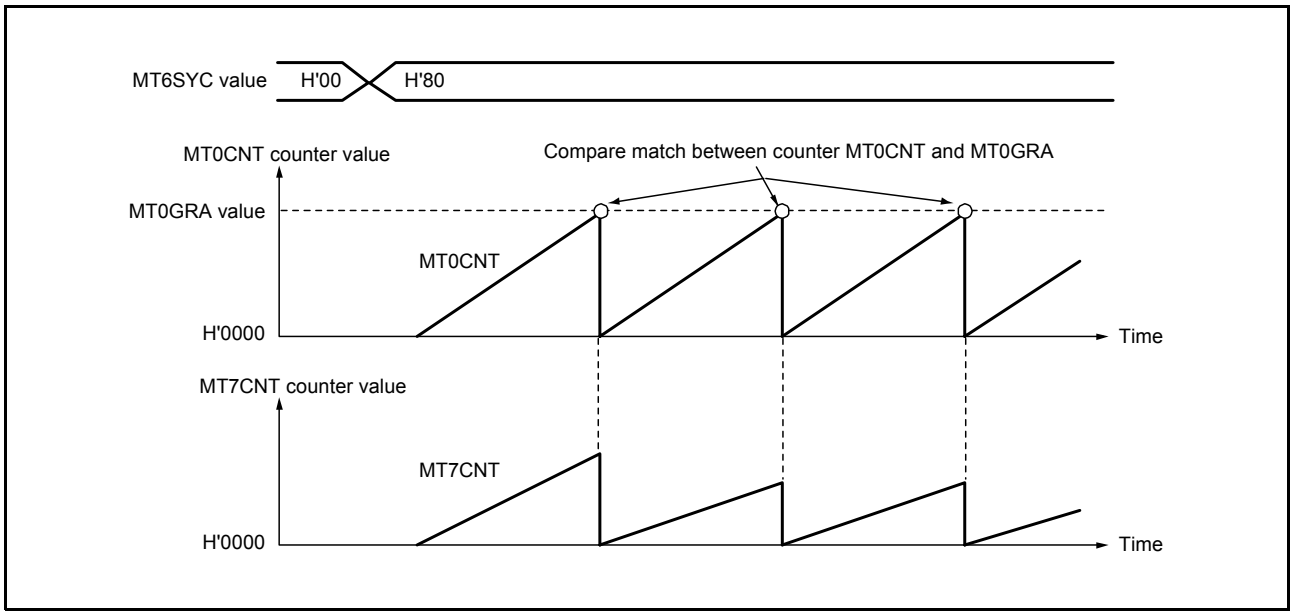


Figure 16.93 Example of Counter Clearing for Channels 6 and 7 by Flag Setting Sources (1)

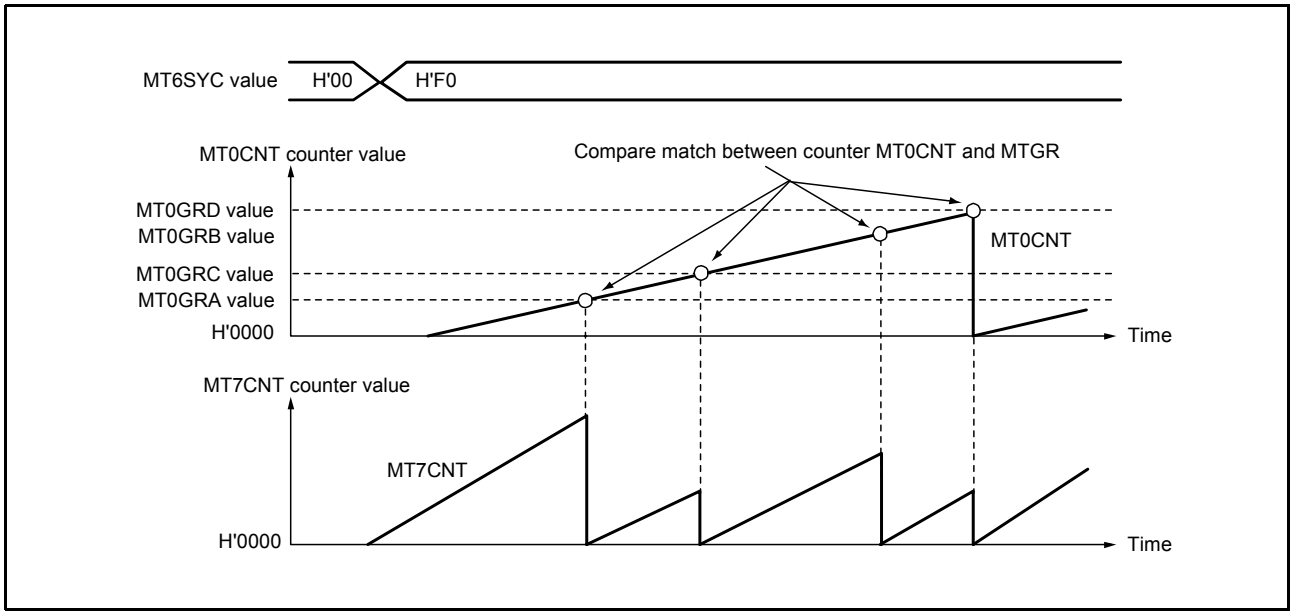


Figure 16.94 Example of Counter Clearing for Channels 6 and 7 by Flag Setting Sources (2)

16.3.11 External Pulse Width Measurement

The pulse widths of up to three external input signals can be measured in channel 5.

(1) Example of External Pulse Width Measurement Setting Procedure

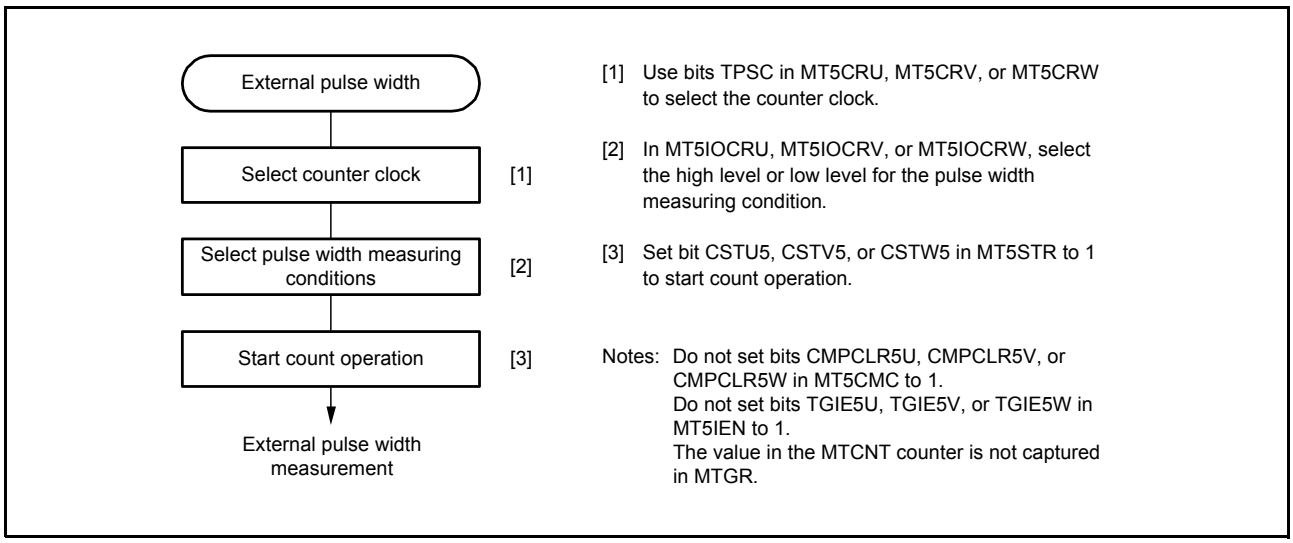


Figure 16.95 Example of External Pulse Width Measurement Setting Procedure

(2) Example of External Pulse Width Measurement

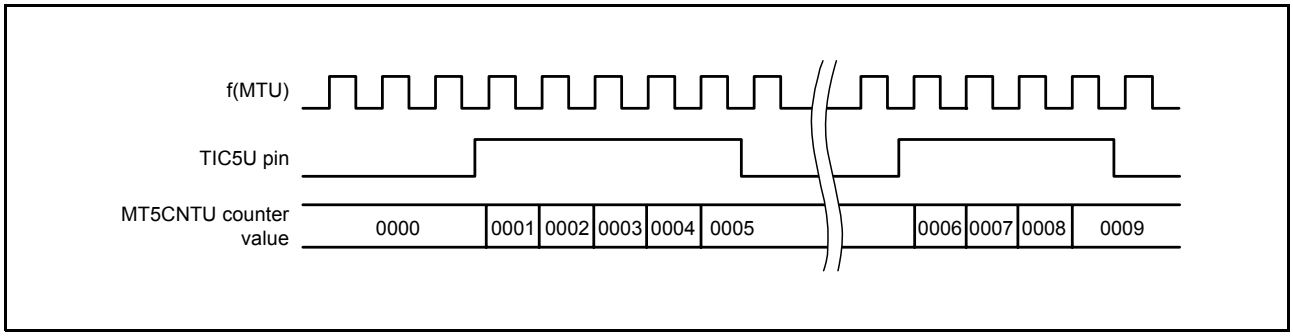


Figure 16.96 Example of External Pulse Width Measurement (Measuring High Pulse Width)

16.3.12 Dead Time Compensation

By measuring the delay of the output waveform and reflecting it to duty, the external pulse width measurement function can be used as the dead time compensation function for PWM output waveform while the complementary PWM is in operation.

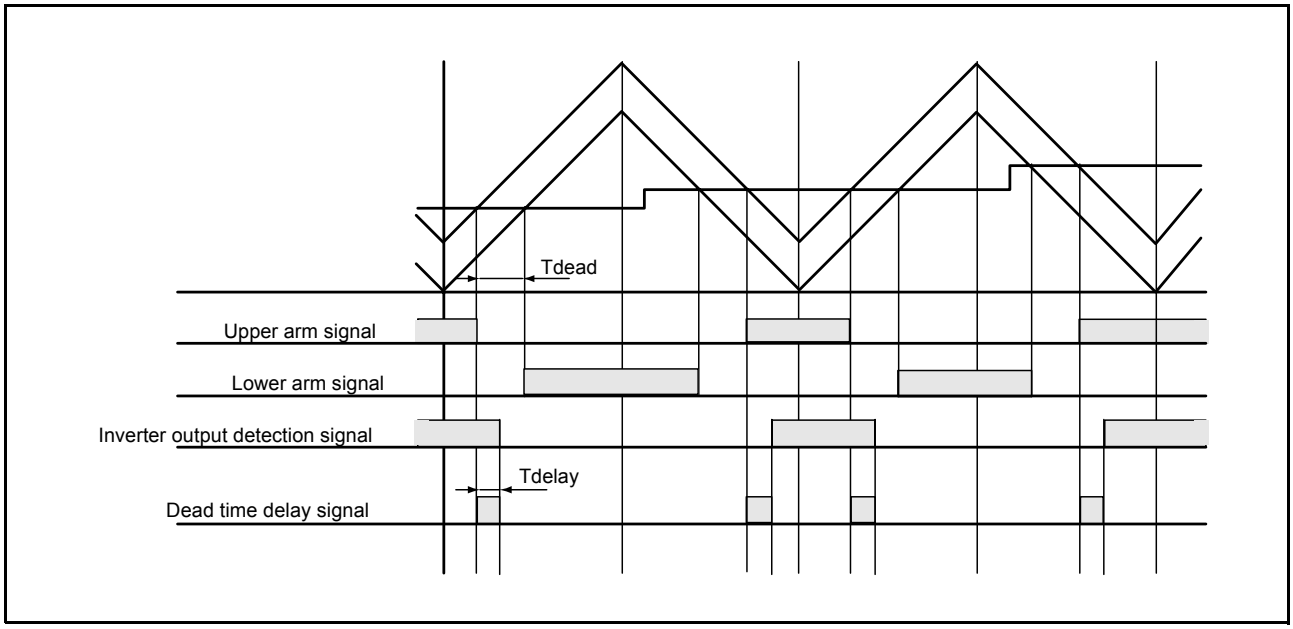
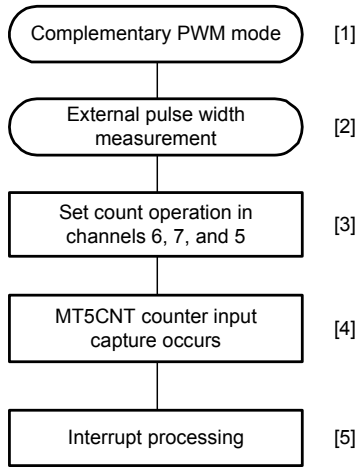


Figure 16.97 Delay in Dead Time in Complementary PWM Operation

(1) Example of Dead Time Compensation Setting Procedure

Figure 16.98 shows an example of dead time compensation setting procedure by using three counters in channel 5.



[1] Place channels 6 and 7 in complementary PWM mode. For details, refer to section 16.3.8, Complementary PWM Mode.

[2] Specify the external pulse width measurement function for the target MT5IOCRU/V/W in channel 5. For details, refer to section 16.3.11, External Pulse Width Measurement.

[3] Set bits CST6 and CST7 in MT67STR and bits CSTU5, CSTV5, and CSTW5 in MT5STR to 1 to start count operation.

[4] When the capture condition specified in MT5IOCRU/V/W is satisfied, the MT5CNT counter value is captured in MT5GR.

[5] For U-phase dead time compensation, when an interrupt is generated at the crest (TGIA_6) or trough (TCIV_7) in complementary PWM mode, read the MT5GRU value, calculate the difference in time in MT6GRB, and write the corrected value to MT6GRD in the interrupt processing. For the V phase and W phase, read the MT5GRV and MT5GR values and write the corrected values to MT7GRC and MT7GRD, respectively, in the same way as for U-phase compensation. The MT5CNT value should be cleared through the MT5CMC setting or by software.

Notes: The pin function settings must be completed in advance.
As an interrupt flag is set under the capture condition specified in MTIOR, do not enable interrupt requests in MT5IEN0.

Figure 16.98 Example of Dead Time Compensation Setting Procedure

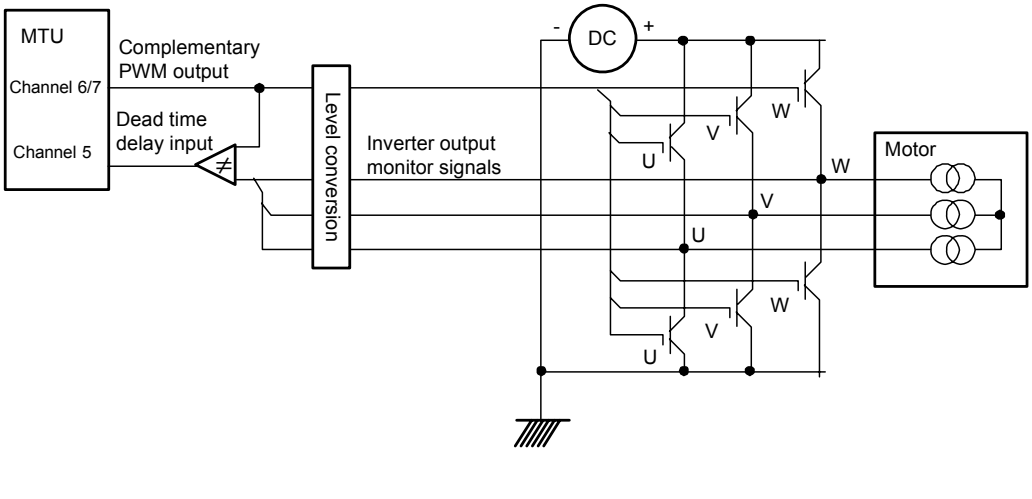


Figure 16.99 Example of Motor Control Circuit Configuration

16.3.13 MTCNT Counter Capture at Crest and/or Trough in Complementary PWM Operation

The MTCNT counter value is captured in MTGR at either the crest or trough or at both the crest and trough during complementary PWM operation. The timing for capturing in MTGR can be selected by MTIOCR.

Figure 16.100 shows an example in which the MTCNT counter is used as a free-running counter without being cleared, and the MTCNT counter value is captured in MTGR at the specified timing (either crest or trough, or both crest and trough).

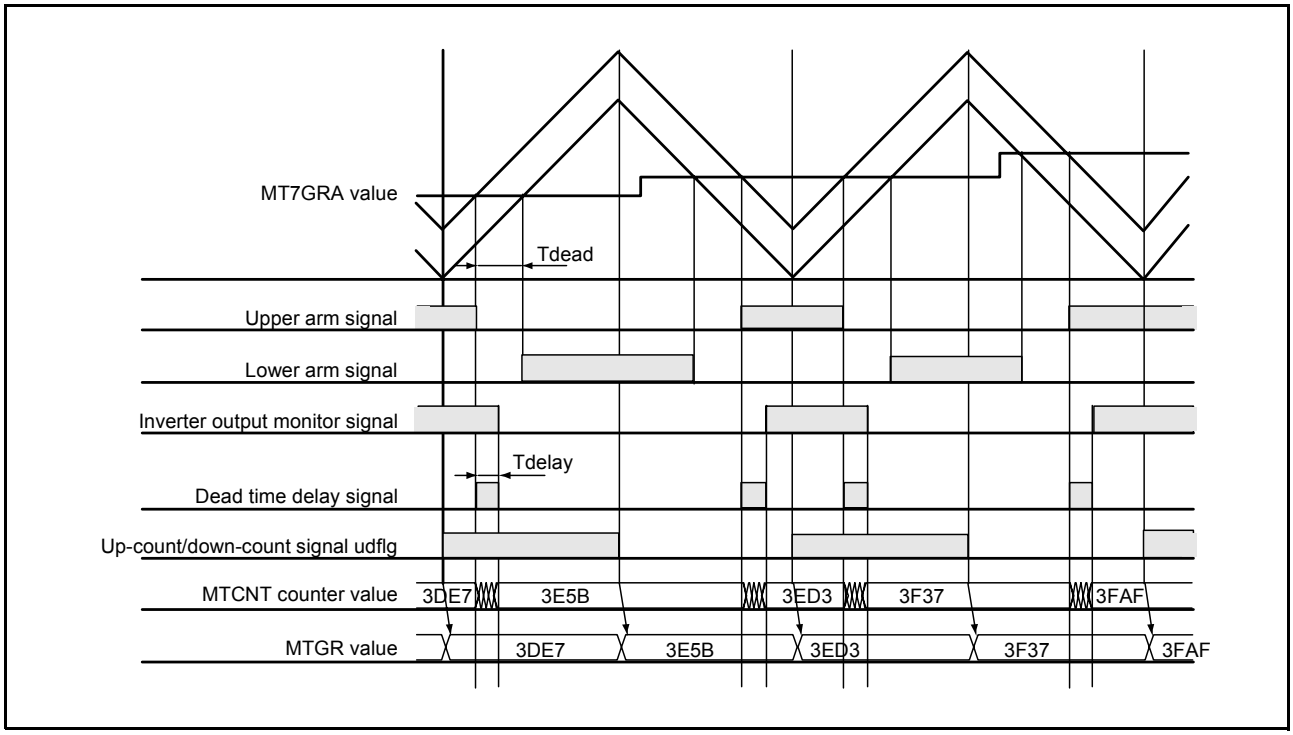


Figure 16.100 MTCNT Counter Capture at Crest and/or Trough in Complementary PWM Operation

16.3.14 Switch between PWM and Square Wave Output

(1) Overview

- The PWM output through the output pins (TIOC3B/D, TIOC4A/C, TIOC4B/D, TIOC6/D, TIOC7A/C, and TIOC7B/D) can be switched to square wave output with desired timing. The square wave output can also be switched back to the MTU-III PWM output with desired timing. The active level is determined by the settings in MTOCR0 and MTOCR1.
- After switching from the PWM output to square wave output, the non-active level is output through both the positive and negative phase output pins for the short-circuit preventing period. After the short-circuit preventing period has passed, a square waveform is output (100% duty output for the positive phase and 0% duty output for the negative phase).
  After switching from the square wave output to PWM output, the non-active level is output through both the positive and negative phase output pins for the short-circuit preventing period. After the short-circuit preventing period has passed, the PWM waveform is output.
- The output can be switched with desired timing by the following register or event settings.
  – Switching by software (MT3467 waveform switch register (MT3467WSW))
  – Input capture/output compare by MTGRA and MTGRB in channels 1 and 2 (enabled by the following register settings)
  (MT1 waveform input capture/output compare switchover enable register A (for MTGRA in channel 1) MT1WIOSWENA, MT1 waveform input capture/output compare switchover enable register B (for MTGRB in channel 1) MT1WIOSWENB, MT2 waveform input capture/output compare switchover enable register A (for MTGRA in channel 2) MT2WIOSWENA, MT2 waveform input capture/output compare switchover enable register B (for MTGRB in channel 2) MT2WIOSWENB)
- All three phases of output (phases U, V, and W) can be switched by the same register or event setting with the same timing.

They can also be switched independently by different settings for each phase.

Figure 16.101 shows an example of switch between the PWM and square wave output when the active level is high.

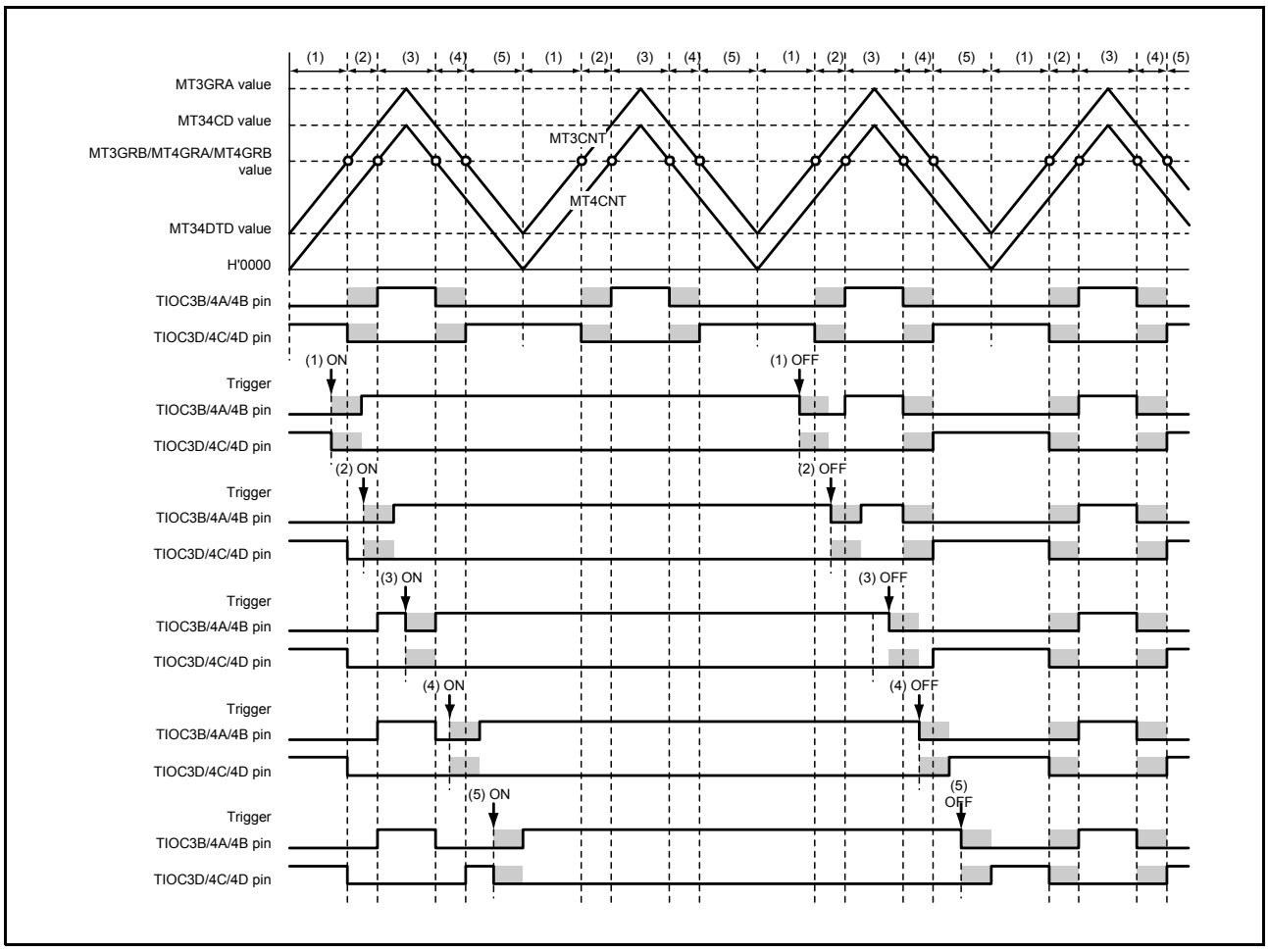


Figure 16.101 Example of Switch between PWM and Square Wave Output

(2) Operation

When the first trigger is generated by the specified event for switching between PWM and square wave output, the internal counter starts up-counting until its value reaches the short-circuit preventing period (MT34DTD or MT67DTD). During this period, the non-active level is output for both the positive and negative phases. When the counter value reaches the short-circuit preventing period, the active level is output for the positive phase and the non-active level output continues for the negative phase.

When the second trigger is generated, the internal counter starts down-counting until its value reaches 0. During this period, the non-active level is output for both the positive and negative phases. When the counter value reaches 0, the PWM waveform is output for the positive and negative phases.

Figure 16.102 shows the switch operation between the PWM and square wave output.

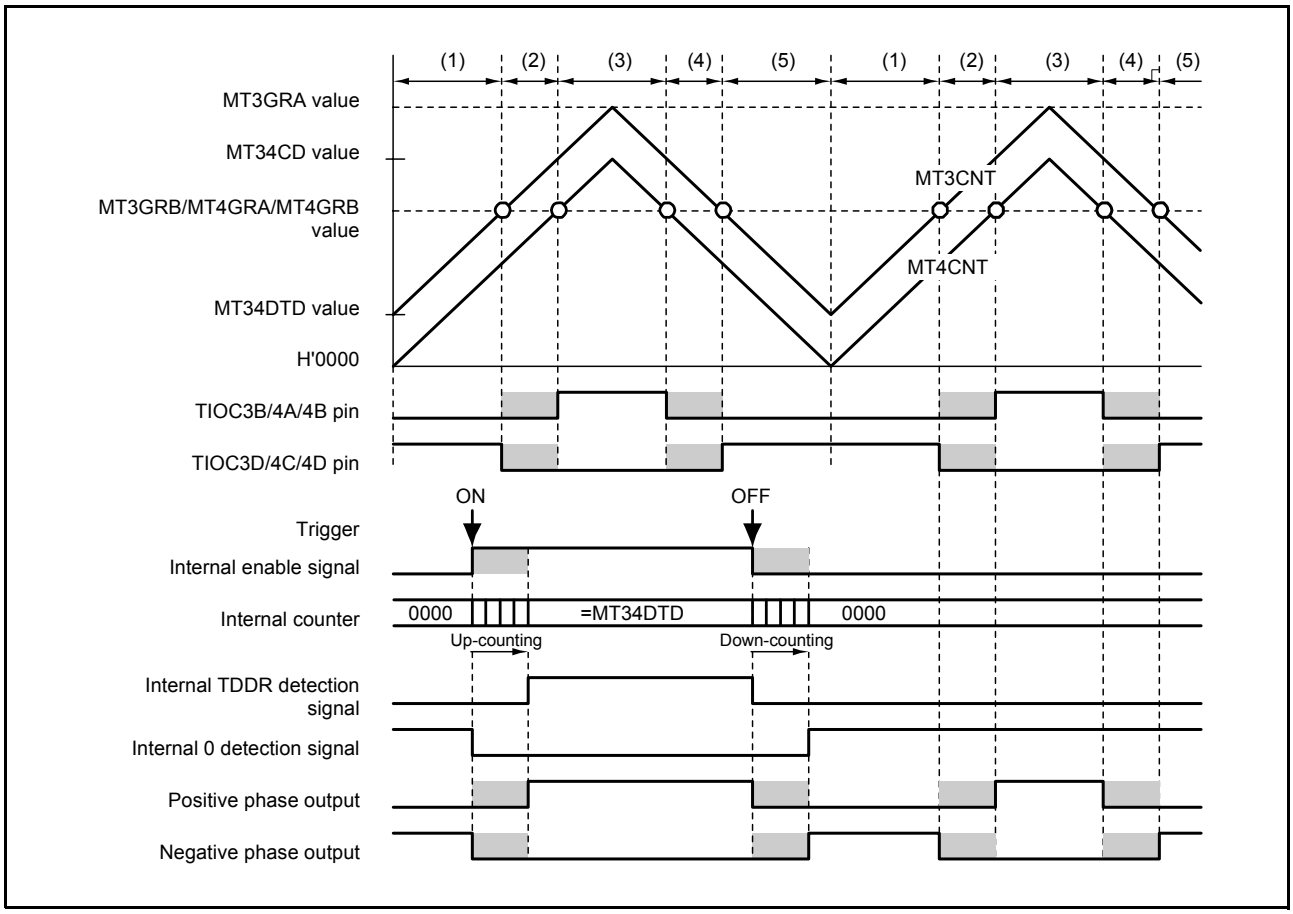


Figure 16.102 Switch Operation between PWM and Square Wave Output

16.4 Interrupt Sources

16.4.1 Interrupt Sources and Priorities

There are three kinds of MTU-III interrupt source; MTGR input capture/compare match, MTCNT counter overflow, and MTCNT counter underflow. Each interrupt source has its own status flag and enable/disabled bit, allowing the generation of interrupt request signals to be enabled or disabled individually.

When an interrupt request is generated, the corresponding status flag in MTSR is set to 1. If the corresponding enable/disable bit in MTIEN is set to 1 at this time, an interrupt is requested. Clearing the status flag to 0 clears the interrupt request.

Relative channel priorities can be changed by the interrupt controller, however the priority order within a channel is fixed. For details, see section 8, Interrupt Controller (INTC). Table 16.76 lists the MTU-III interrupt sources.

Table 16.76 MTU-III Interrupt Sources

Channel	Name	Interrupt Source	Interrupt Flag Flag	DMAC Activation	Priority
0	TGIA_0	MT0GRA interrupt	TGFA_0	Possible	High
	TGIB_0	MT0GRB interrupt	TGFB_0	Possible	
	TGIC_0	MT0GRC interrupt	TGFC_0	Possible	
	TGID_0	MT0GRD interrupt	TGFD_0	Possible	
	TCIV_0	MT0 counter overflow interrupt	TCFV_0	Not possible	
	TGIE_0	MT0GRE interrupt	TGFE_0	Not possible	
	TGIF_0	MT0GRF interrupt	TGFF_0	Not possible	
1	TGIA_1	MT1GRA interrupt	TGFA_1	Possible	
	TGIB_1	MT1GRB interrupt	TGFB_1	Possible	
	TCIV_1	MT1 counter overflow interrupt	TCFV_1	Not possible	
	TCIU_1	MT1 counter underflow interrupt	TCFU_1	Not possible	
2	TGIA_2	MT2GRA interrupt	TGFA_2	Possible	
	TGIB_2	MT2GRB interrupt	TGFB_2	Possible	
	TCIV_2	MT2 counter overflow interrupt	TCFV_2	Not possible	
	TCIU_2	MT2 counter underflow interrupt	TCFU_2	Not possible	
3	TGIA_3	MT3GRA interrupt	TGFA_3	Possible	
	TGIB_3	MT3GRB interrupt	TGFB_3	Possible	
	TGIC_3	MT3GRC interrupt	TGFC_3	Possible	
	TGID_3	MT3GRD interrupt	TGFD_3	Possible	
	TCIV_3	MT3 counter overflow interrupt	TCFV_3	Not possible	
4	TGIA_4	MT4GRA interrupt	TGFA_4	Possible	
	TGIB_4	MT4GRB interrupt	TGFB_4	Possible	
	TGIC_4	MT4GRC interrupt	TGFC_4	Possible	
	TGID_4	MT4GRD interrupt	TGFD_4	Possible	
	TCIV_4	MT4 counter overflow/underflow interrupt	TCFV_4	Not possible	
5	TGIU_5	MT5GRU interrupt	TGFU_5	Possible	
	TGIV_5	MT5GRV interrupt	TGFV_5	Possible	
	TGIW_5	MT5GRW interrupt	TGFW_5	Possible	
6	TGIA_6	MT6GRA interrupt	TGFA_6	Possible	
	TGIB_6	MT6GRB interrupt	TGFB_6	Possible	
	TGIC_6	MT6GRC interrupt	TGFC_6	Possible	
	TGID_6	MT6GRD interrupt	TGFD_6	Possible	
	TCIV_6	MT6 counter overflow interrupt	TCFV_6	Not possible	
7	TGIA_7	MT7GRA interrupt	TGFA_7	Possible	
	TGIB_7	MT7GRB interrupt	TGFB_7	Possible	
	TGIC_7	MT7GRC interrupt	TGFC_7	Possible	
	TGID_7	MT7GRD interrupt	TGFD_7	Possible	
	TCIV_7	MT7 counter overflow/underflow interrupt	TCFV_7	Not possible	Low

Note: This table shows the initial state immediately after a reset. The interrupt controller can change the relative channel priorities.

(1) Input Capture/Compare Match Interrupt

An interrupt is requested if the TGIE bit in MTIEN is set to 1 when the TGF flag in MTSR is set to 1 by the occurrence of a MTGR input capture/compare match on a particular channel. Clearing the TGF flag to 0 clears the interrupt request. The MTU-III has 29 input capture/compare match interrupts, six for channel 0, four each for channels 3, 4, 6, and 7, two each for channels 1 and 2, and three for channel 5. The TGFE_0 and TGFF_0 flags in channel 0 are not set by the occurrence of an input capture.

(2) Overflow Interrupt

An interrupt is requested if the TCIEV bit in MTIEN is set to 1 when the TCFV flag in MTSR is set to 1 by the occurrence of MTCNT counter overflow on a channel. Clearing the TCFV flag to 0 clears the interrupt request. The MTU-III has seven overflow interrupts, one for each channel (excluding channel 5).

(3) Underflow Interrupt

An interrupt is requested if the TCIEU bit in MTIEN is set to 1 when the TCFU flag in MTSR is set to 1 by the occurrence of the MTCNT counter underflow on a channel. Clearing the TCFU flag to 0 clears the interrupt request. The MTU-III has two underflow interrupts, one each for channels 1 and 2.

16.4.2 DMAC Activation

The DMAC can be activated by the MTGR input capture/compare match interrupt in each channel. For details, see section 12, DMAC.

A total of 27 MTU-III input capture/compare match interrupts can be used as DMAC activation sources, four each for channels 0, 3, 4, 6, and 7, two each for channels 1 and 2, and three for channel 5.

16.4.3 A/D Converter Activation

The A/D converter can be activated by one of the following three methods in the MTU-III.

Table 16.77 shows the relationship between interrupt sources and A/D converter start request signals.

(1) A/D Converter Activation by MTGRA Input Capture/Compare Match or at MT4CNT (MT7CNT) Counter Trough in Complementary PWM Mode

The A/D converter can be activated by the occurrence of a MTGRA input capture/compare match in each channel. In addition, if complementary PWM operation is performed while the TTGE2 bit in MT4IEN0 (MT7IEN0) is set to 1, the A/D converter can be activated at the trough of the MT4CNT (MT7CNT) counter (MT4CNT (MT7CNT) = H'0000).

A/D converter start request signal TRGAnN (n: channels 0 to 4, 6, and 7) is issued to the A/D converter under either one of the following conditions.

- When the TGFA flag in MTSR is set to 1 by the occurrence of a MTGRA input capture/compare match on a particular channel while the TTGE bit in MTIEN is set to 1
- When the MT4CNT (MT7CNT) count reaches the trough (MT4CNT (MT7CNT) = H'0000) during complementary PWM operation while the TTGE2 bit in MT4IEN0 (MT7IEN0) is set to 1

When either condition is satisfied, if A/D converter start signal TRGAnN from the MTU-III is selected as the trigger in the A/D converter, A/D conversion will start.

(2) A/D Converter Activation by Compare Match between the MT0CNT Counter and MT0GRE

The A/D converter can be activated by generating A/D converter start request signal TRG0N when a compare match occurs between the MT0CNT counter and MT0GRE in channel 0.

When the TGFE flag in MT0SR1 is set to 1 by the occurrence of a compare match between the MT0CNT counter and MT0GRE in channel 0 while the TTGE2 bit in MT0IEN1 is set to 1, A/D converter start request TRG0N is issued to the A/D converter. If A/D converter start signal TRG0N from the MTU-III is selected as the trigger in the A/D converter, A/D conversion will start.

(3) A/D Converter Activation by A/D Converter Start Request Delaying Function

The A/D converter can be activated by generating A/D converter start request signal TRG4AN or TRG4BN (TRG7AN or TRG7BN) when the MT4CNT counter matches the MT4ADSRCSEA or MT4ADSRCSEB value (when the MT7CNT counter matches the MT7ADSRCSEA or MT7ADSRCSEB value) if the UT4AE, DT4AE, UT4BE, DT4BE (UT7AE, DT7AE, UT7BE, DT7BE) bits in MTADSRCR are set to 1. Sending TRG4ABN (TRG7ABN) when TRG4AN (TRG7AN) or TRG4BN (TRG7BN) is generated can also activate the A/D converter. For details, refer to section section 16.3.9, A/D Converter Start Request Delaying Function.

A/D conversion will start if A/D converter start signal TRG4AN (TRG7AN) from the MTU-III is selected as the trigger in the A/D converter when TRG4AN (TRG7AN) is generated, if TRG4BN (TRG7BN) from the MTU-III is selected as the trigger in the A/D converter when TRG4BN (TRG7BN) is generated, or if TRG4ABN (TRG7ABN) from the MTU-III is selected as the trigger in the A/D converter when TRG4ABN (TRG7ABN) is generated.

Table 16.77 Interrupt Sources and A/D Converter Start Request Signals

Target Registers	Interrupt Source	A/D Converter Start Request Signal
MT0GRA and MT0CNT	Input capture/compare match	TRGA0N
MT1GRA and MT1CNT		TRGA1N
MT2GRA and MT2CNT		TRGA2N
MT3GRA and MT3CNT		TRGA3N
MT4GRA and MT4CNT*		TRGA4N
MT4CNT	MT4CNT trough in complementary PWM mode	
MT6GRA and MT6CNT	Input capture/compare match	TRGA6N
MT7GRA and MT7CNT*		TRGA7N
MT7CNT	MT7CNT trough in complementary PWM mode	
MT0GRE and MT0CNT	Compare match	TRG0N
MT4ADSRCSEA and MT4CNT	Compare match (interrupt skipping function 1)	TRG4AN
MT4ADSRCSEB and MT4CNT		TRG4BN
MT7ADSRCSEA and MT7CNT		TRG7AN
MT7ADSRCSEB and MT7CNT		TRG7BN
MT4ADSRCSEA and MT4CNT	Compare match (interrupt skipping function 2)	TRG4ABN
MT4ADSRCSEB and MT4CNT		
MT7ADSRCSEA and MT7CNT		TRG7ABN
MT7ADSRCSEB and MT7CNT		

Note: * Since PWM waveforms are generated in complementary PWM mode, MT4GRA (MT7GRA) compare match not only with the MT4CNT (MT7CNT) counter but also with the MT3CNT (MT6CNT) counter and the MT34SCNT (MT67SCNT) counter is detected. In response to this, when compare match with the MT3CNT (MT6CNT) counter and the MT34SCNT (MT67SCNT) counter occurs, TRGA4N (TRGA7N) is also generated.
When channel 3 and channel 4 (channel 6 and channel 7) are made to operate in complementary PWM mode for generating an A/D converter start request, use the A/D converter start request by compare match between the MT4CNT (MT7CNT) counter and MT4ADSRCSEA or MT4ADSRCSEB (MT7ADSRCSEA or MT7ADSRCSEB).

16.5 Operation Timing

16.5.1 Input/Output Timing

(1) MTCNT Count Timing

Figure 16.103 and figure 16.104 show MTCNT count timing in internal clock operation, and figure 16.105 shows MTCNT count timing in external clock operation (normal mode), and figure 16.106 shows MTCNT count timing in external clock operation (phase counting mode).

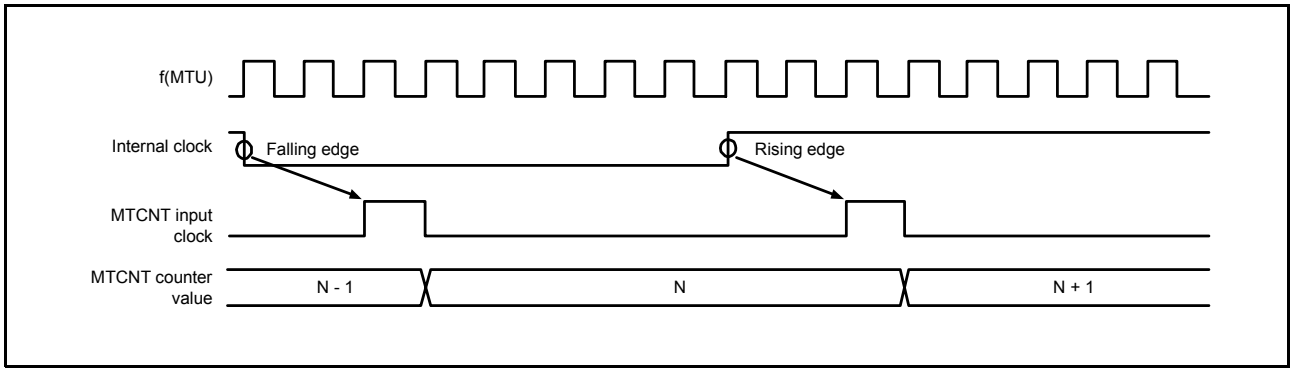


Figure 16.103 Count Timing in Internal Clock Operation (Channels 0 to 4, 6, and 7)

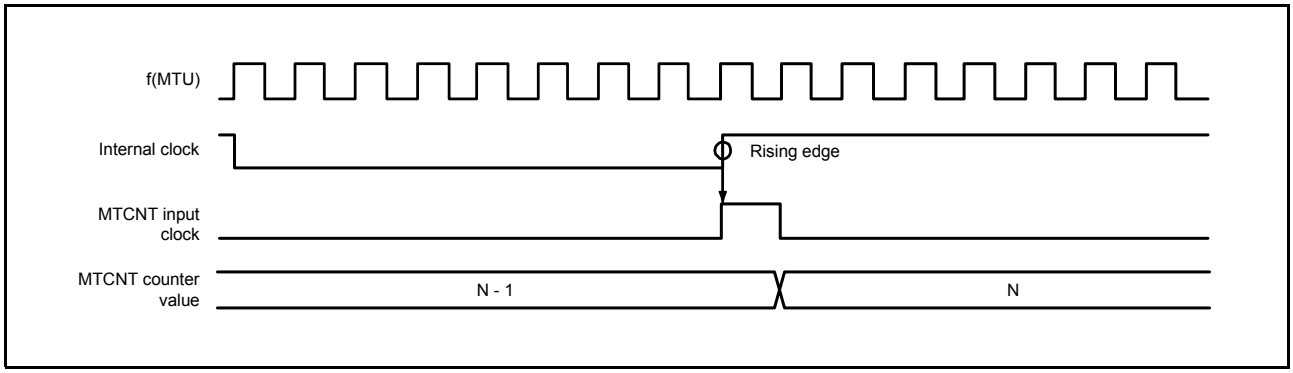


Figure 16.104 Count Timing in Internal Clock Operation (Channel 5)

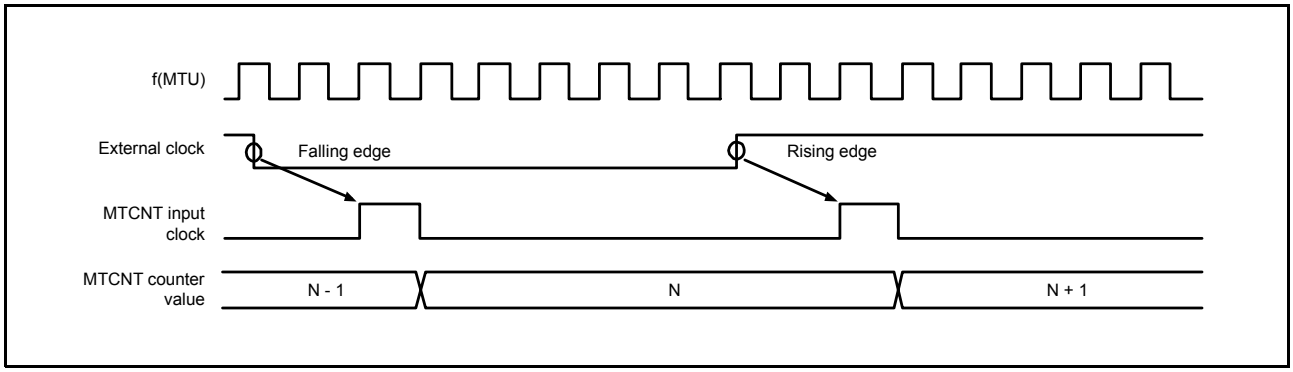


Figure 16.105 Count Timing in External Clock Operation (Channel 0 to 4)

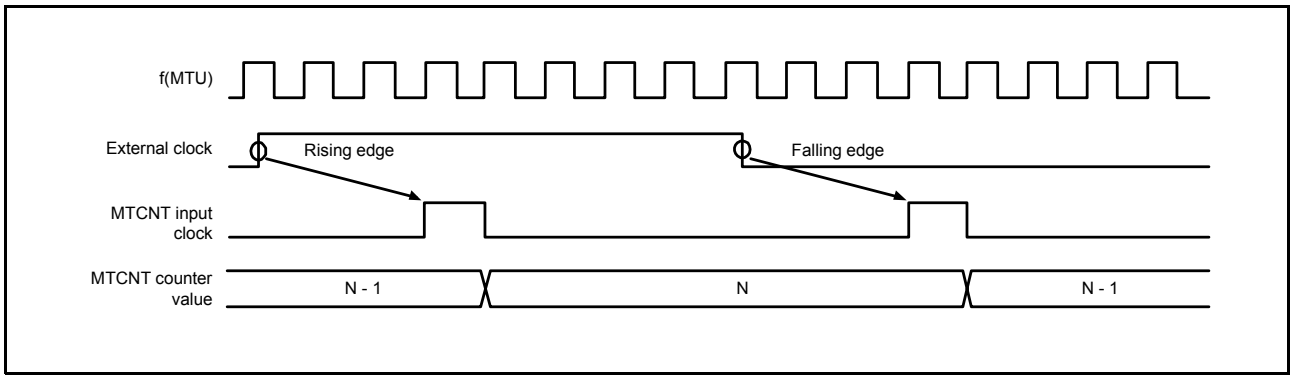


Figure 16.106 Count Timing in External Clock Operation (Phase Counting Mode)

(2) Output Compare Output Timing

A compare match signal is generated in the final state in which the MTCNT counter and MTGR match (the point at which the count value matched by the MTCNT counter is updated). When a compare match signal is generated, the output value set in MTIOCR is output at the output compare output pin (TIOC pin). After a match between the MTCNT counter and MTGR, the compare match signal is not generated until the MTCNT counter input clock is generated.

Figure 16.107 shows output compare output timing (normal mode and PWM mode) and figure 16.108 shows output compare output timing (complementary PWM mode and reset-synchronized PWM mode).

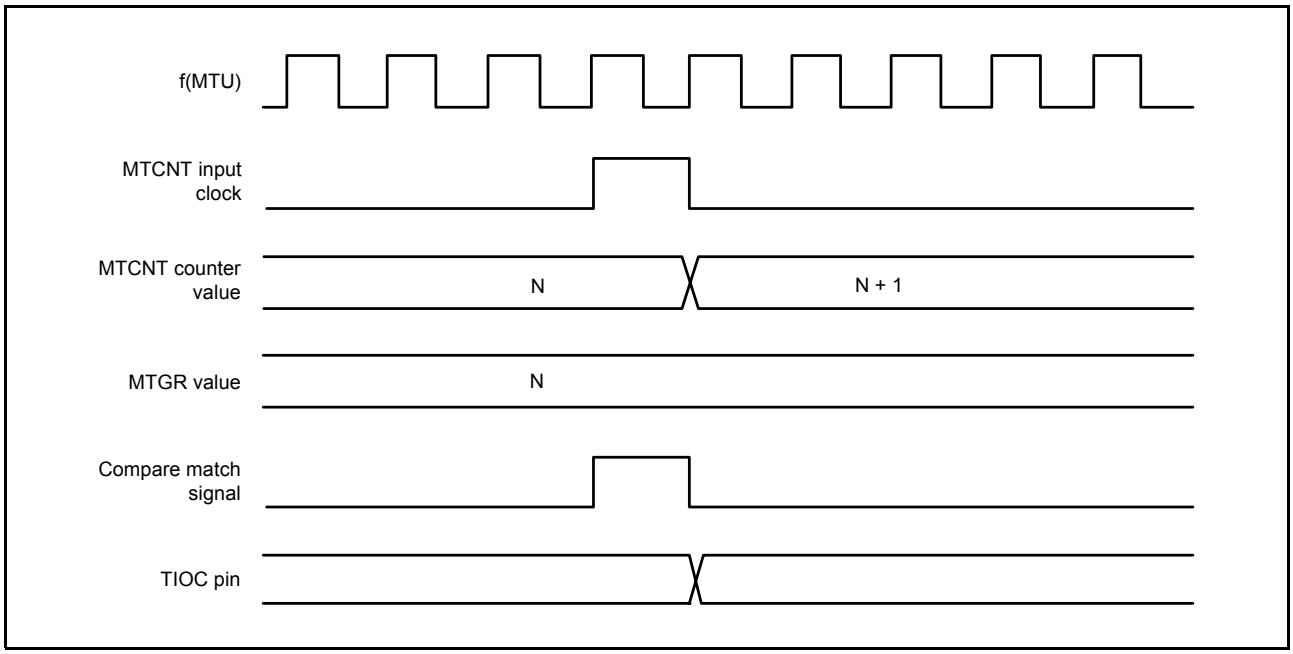


Figure 16.107 Output Compare Output Timing (Normal Mode/PWM Mode)

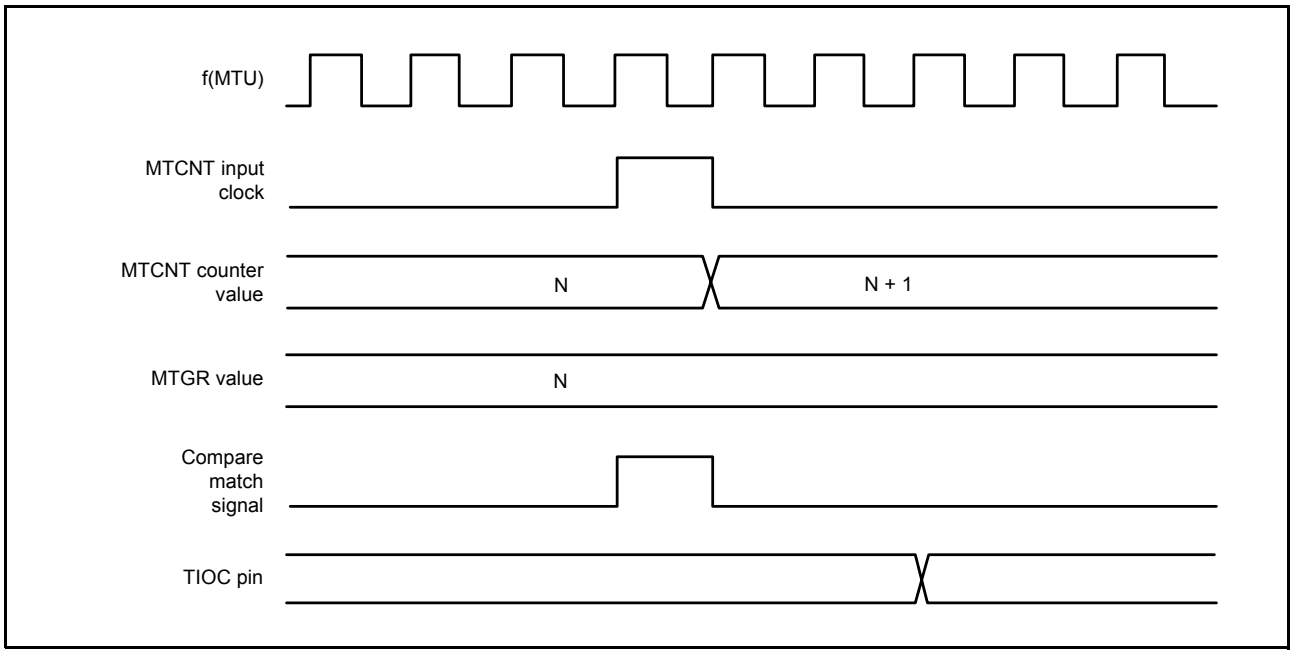


Figure 16.108 Output Compare Output Timing (Complementary PWM Mode/Reset-Synchronized PWM Mode)

(3) Input Capture Signal Timing

Figure 16.109 shows input capture signal timing.

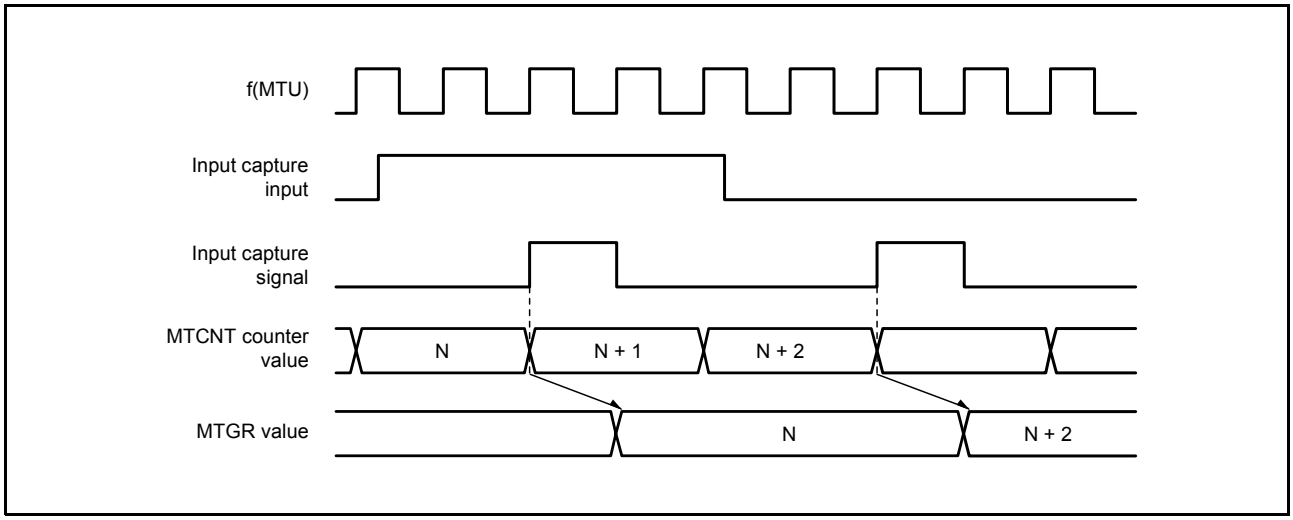


Figure 16.109 Input Capture Input Signal Timing

(4) Timing for Counter Clearing by Compare Match/Input Capture

Figure 16.110 and figure 16.111 show the timing when counter clearing on compare match is specified, and figure 16.112 shows the timing when counter clearing on input capture is specified.

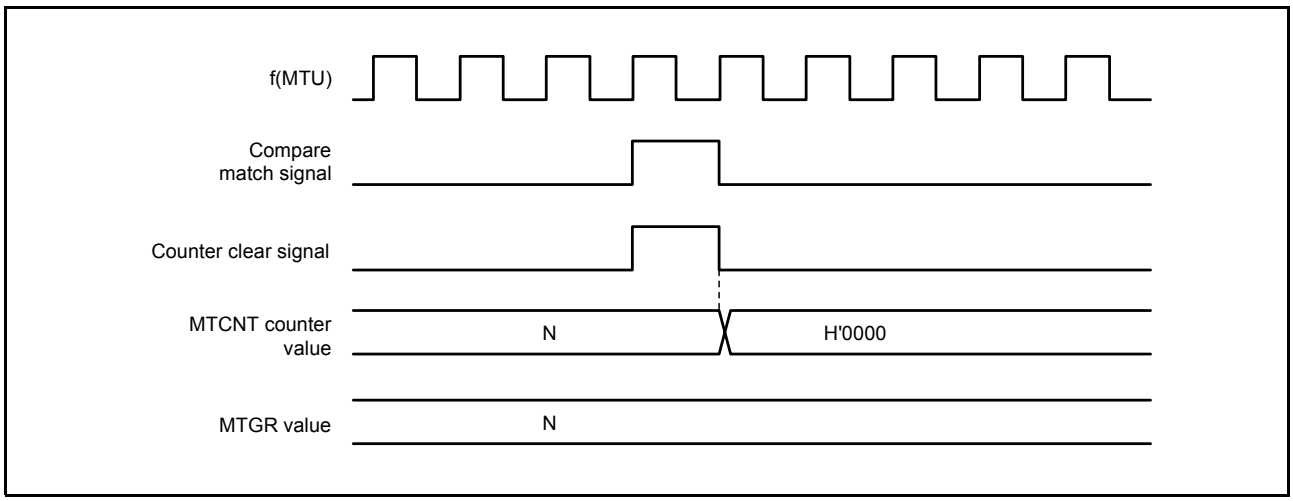


Figure 16.110 Counter Clear Timing (Compare Match) (Channels 0 to 4, 6, and 7)

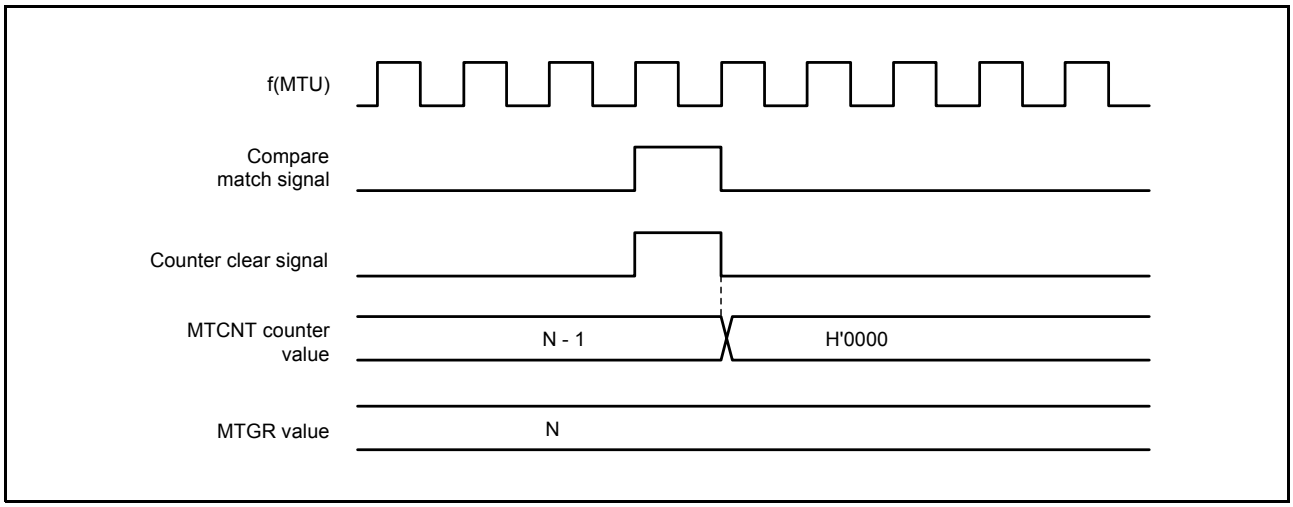


Figure 16.111 Counter Clear Timing (Compare Match) (Channel 5)

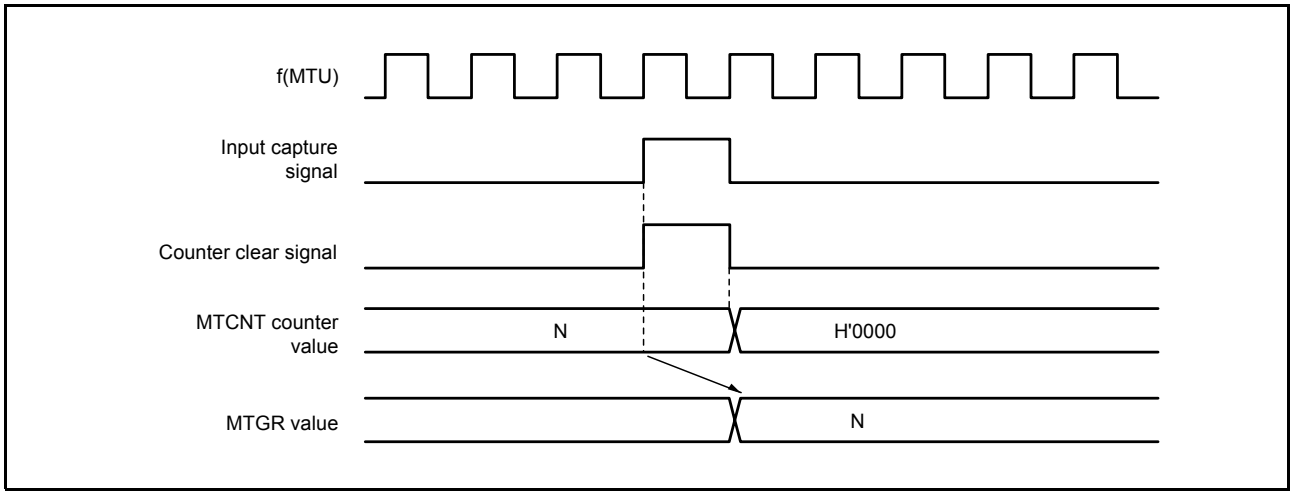


Figure 16.112 Counter Clear Timing (Input Capture) (Channels 0 to 7)

(5) Buffer Operation Timing

Figure 16.113 to figure 16.115 show the timing in buffer operation.

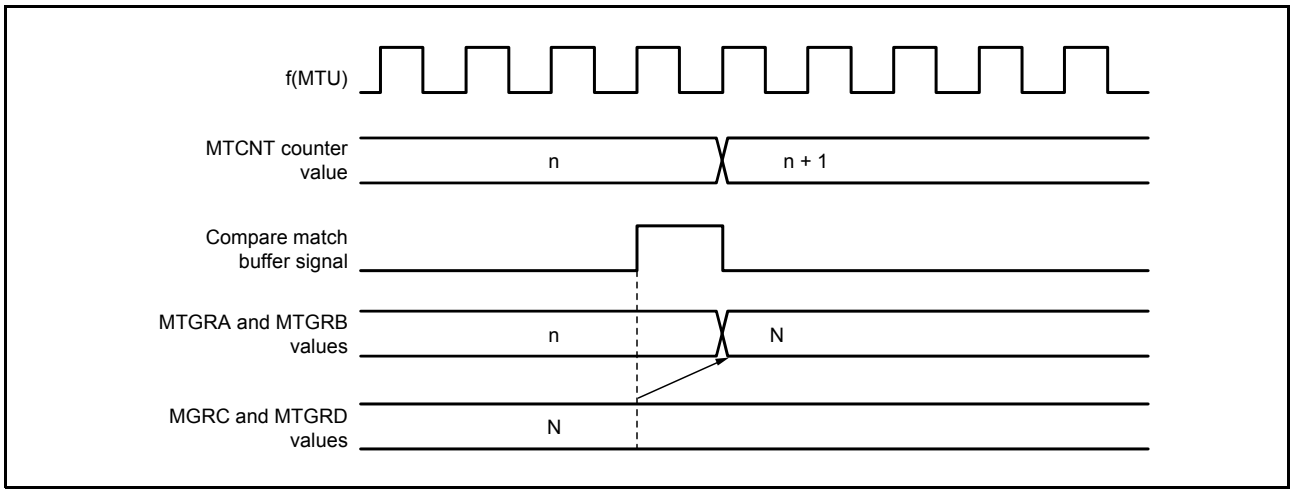


Figure 16.113 Buffer Operation Timing (Compare Match)

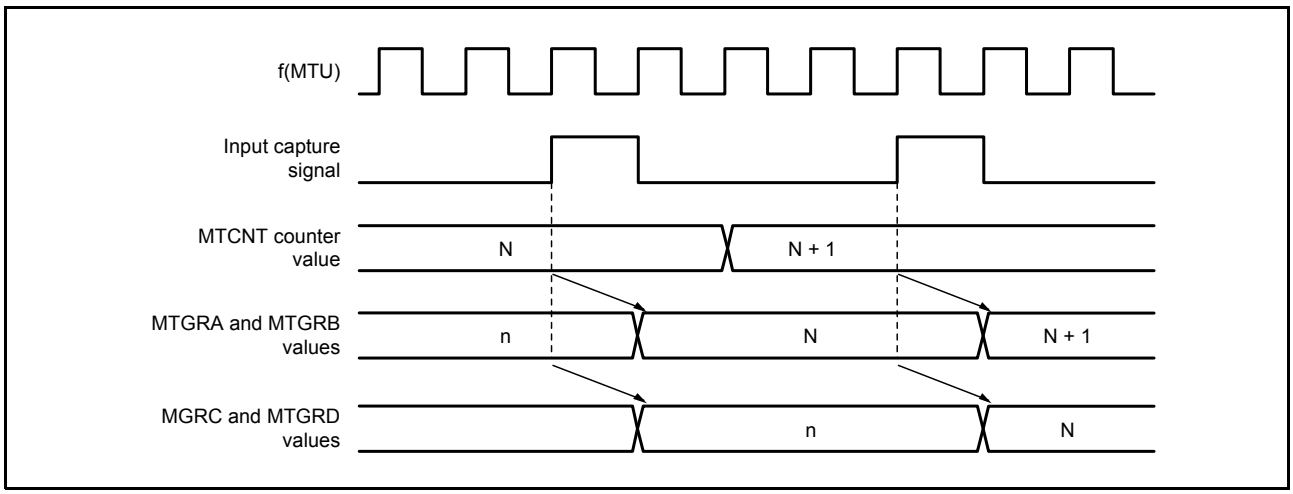


Figure 16.114 Buffer Operation Timing (Input Capture)

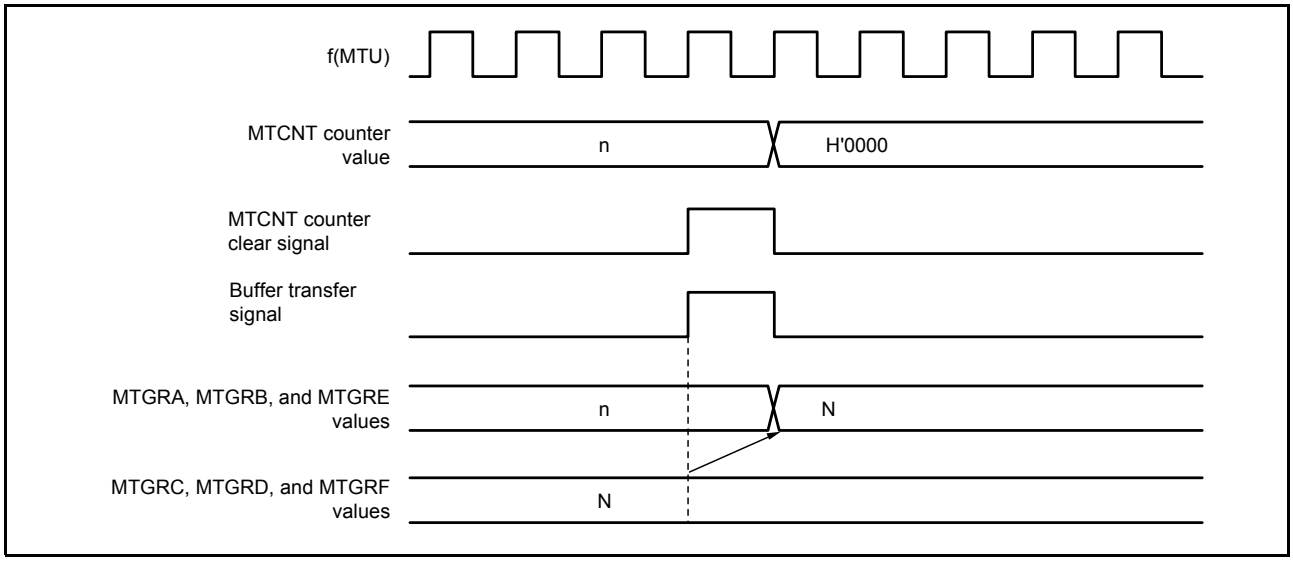


Figure 16.115 Buffer Transfer Timing (when MTCNT Cleared)

(6) Buffer Transfer Timing (Complementary PWM Mode)

Figure 16.116 to figure 16.118 show the buffer transfer timing in complementary PWM mode.

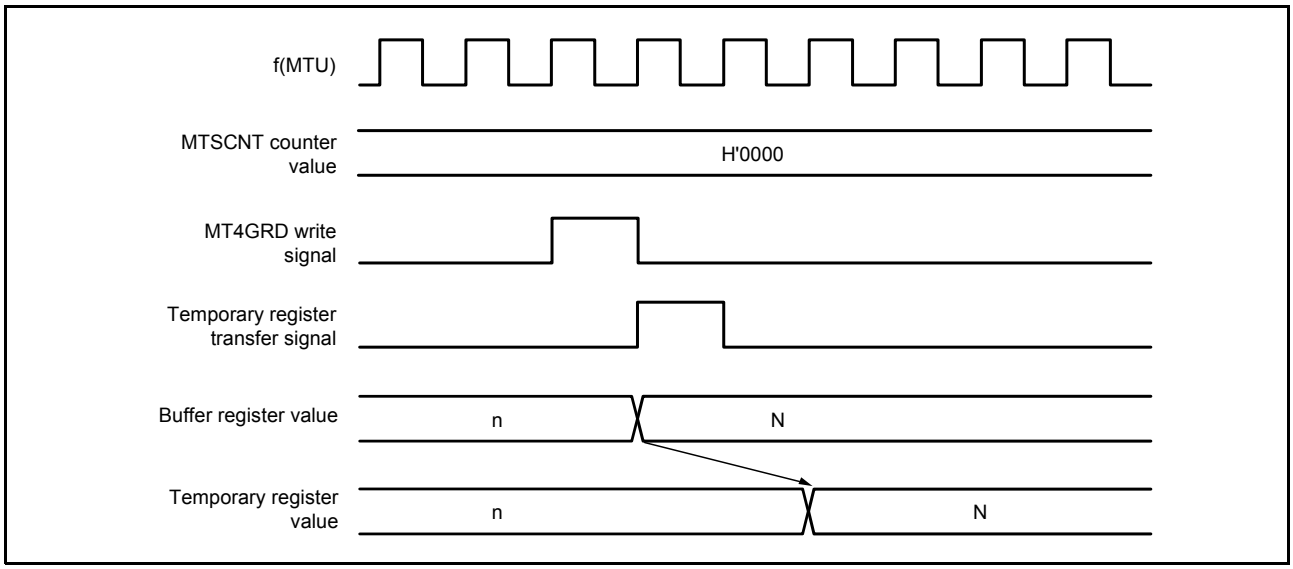


Figure 16.116 Transfer Timing from Buffer Register to Temporary Register (MT34SCNT Stop)

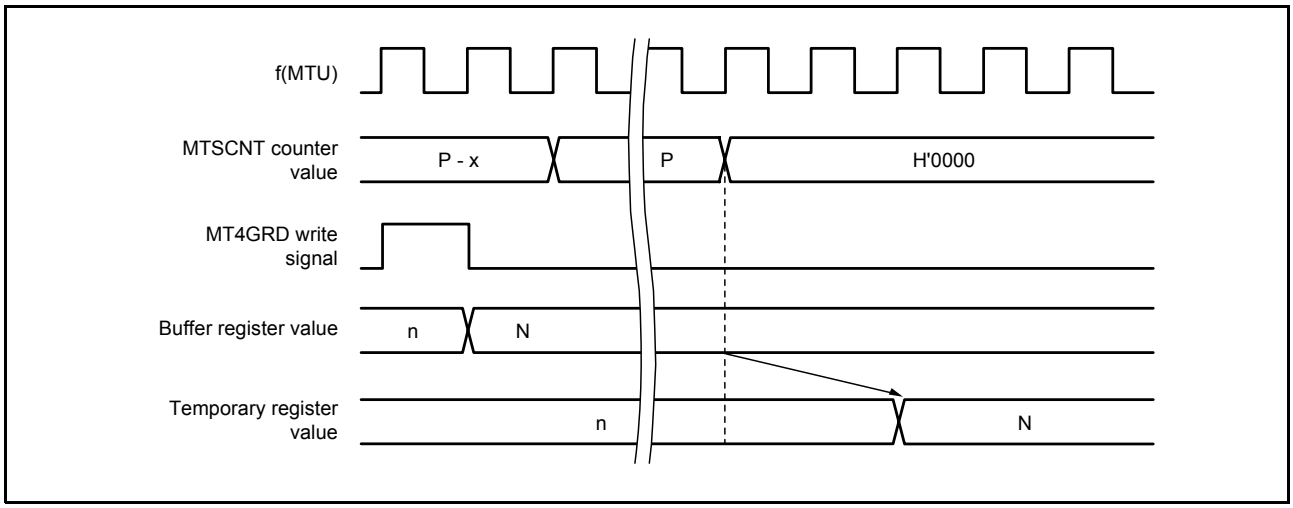


Figure 16.117 Transfer Timing from Buffer Register to Temporary Register (MT34SCNT Operating)

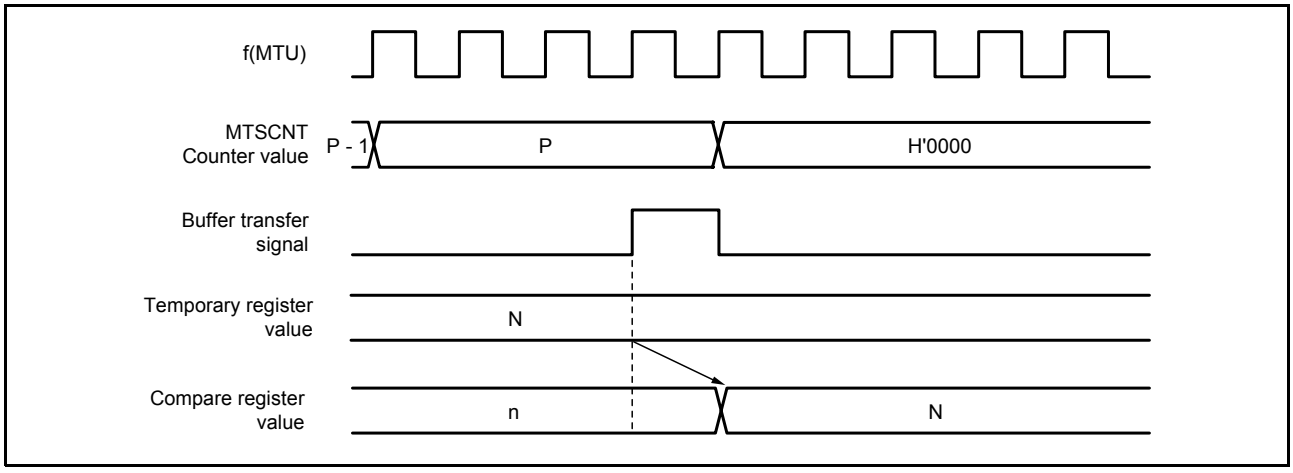


Figure 16.118 Transfer Timing from Temporary Register to Compare Register

16.5.2 Interrupt Signal Timing

(1) TGF Flag Setting Timing in Case of Compare Match

Figure 16.119 and figure 16.120 show the timing for setting of the TGF flag in MTSR on compare match, and TGI interrupt request signal timing.

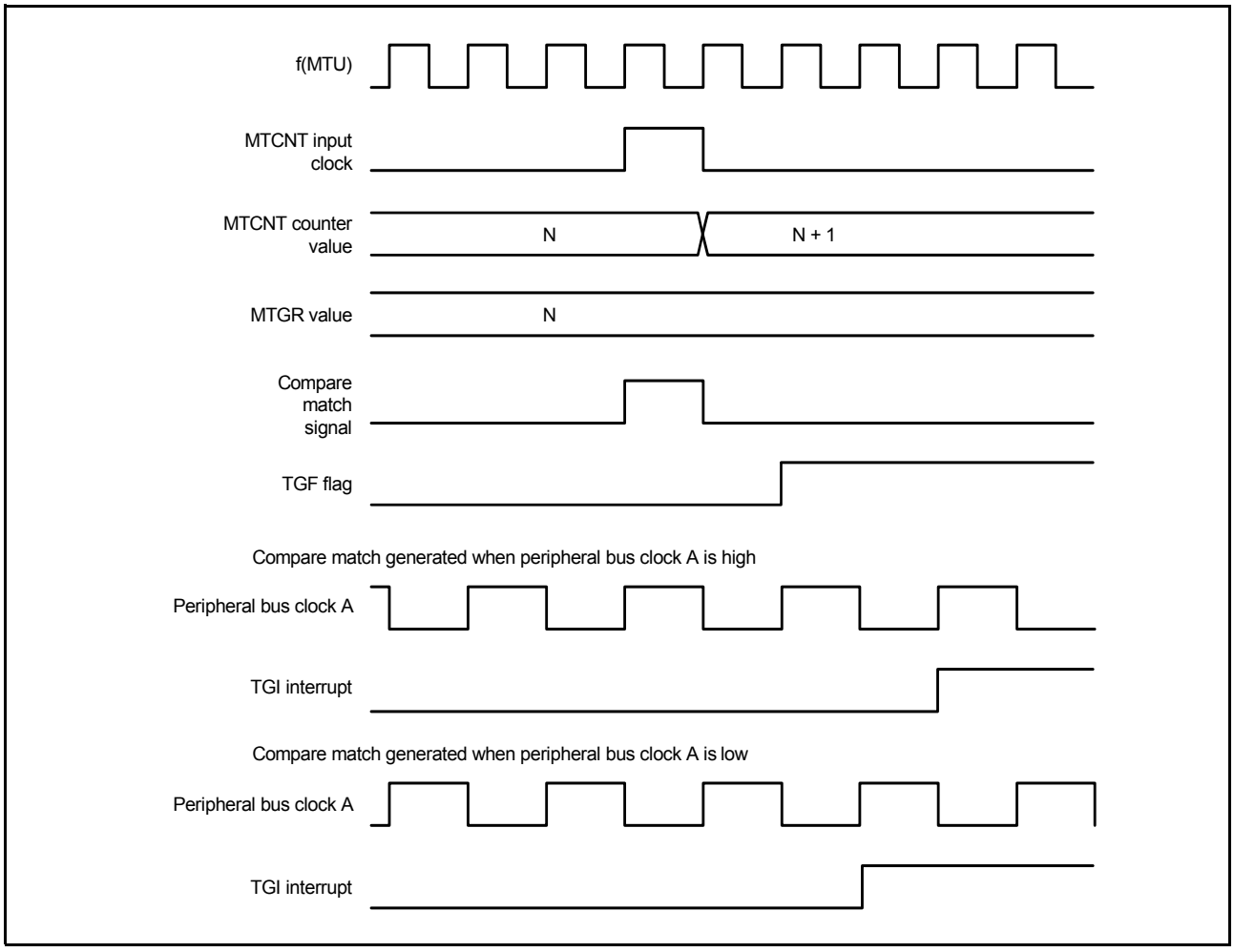


Figure 16.119 TGI Interrupt Timing (Compare Match) (Channels 0 to 4, 6, and 7)

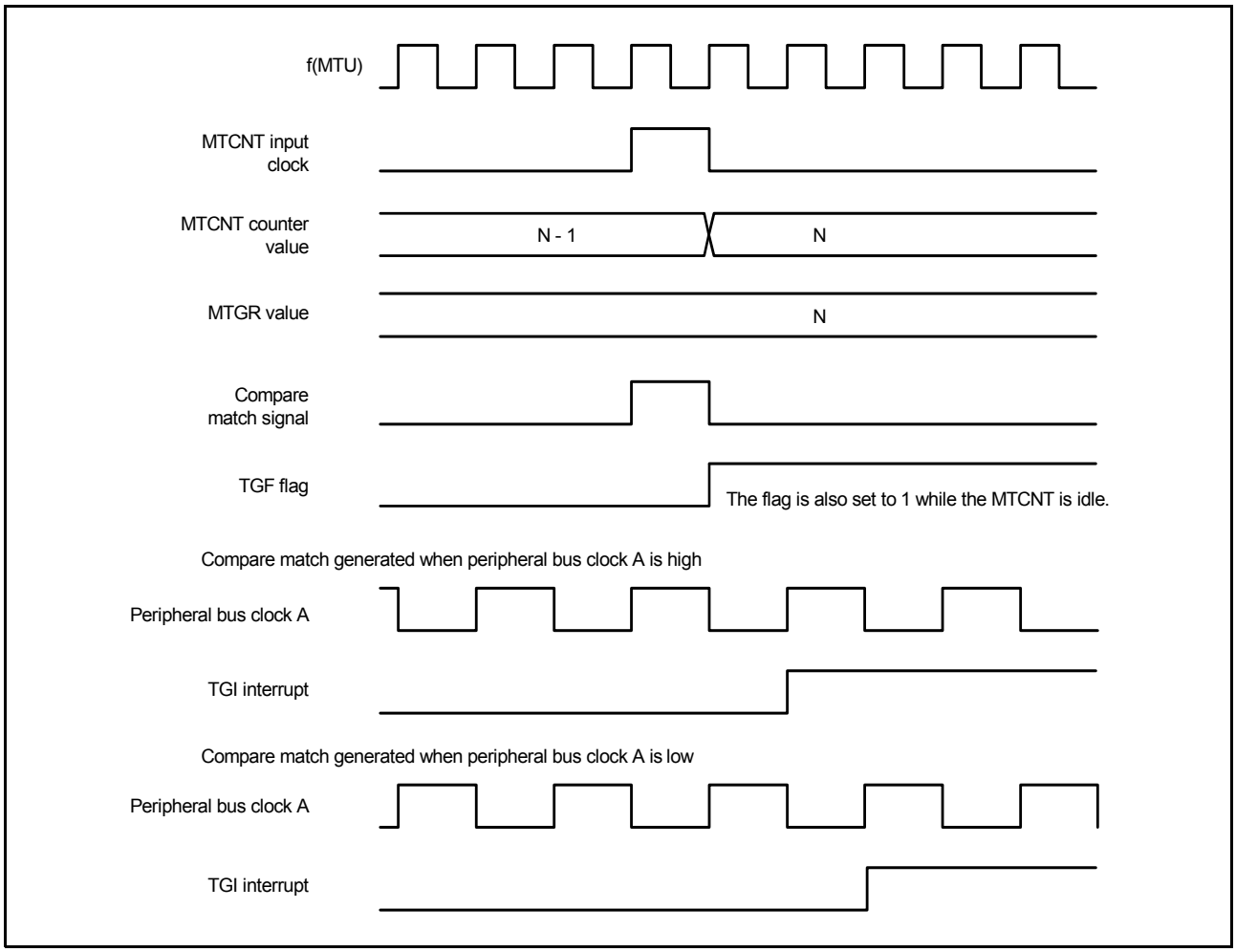


Figure 16.120 TGI Interrupt Timing (Compare Match) (Channel 5)

(2) TGF Flag Setting Timing in Case of Input Capture

Figure 16.121 and figure 16.122 show the timing for setting of the TGF flag in MTSR on input capture, and TGI interrupt request signal timing.

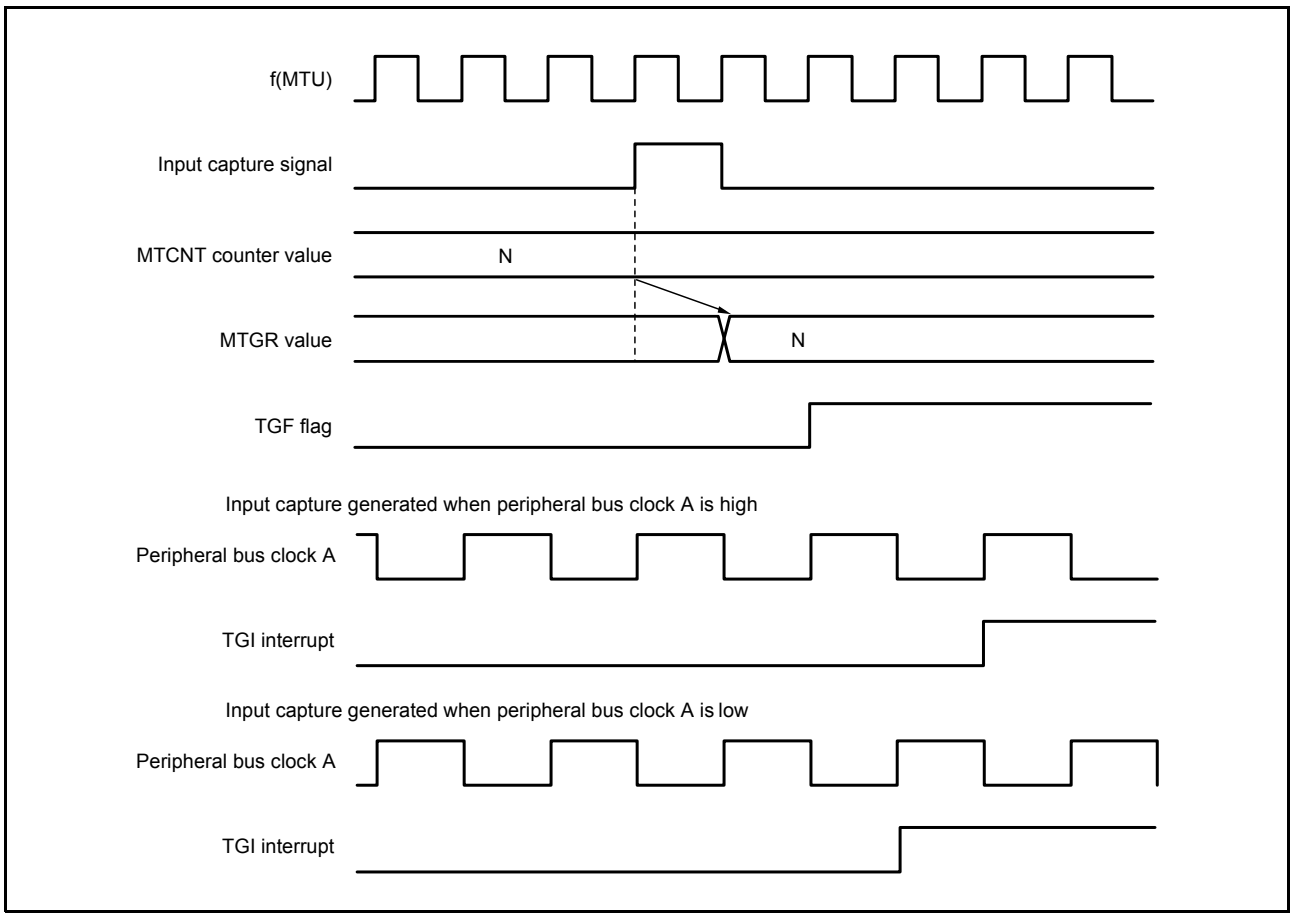


Figure 16.121 TGI Interrupt Timing (Input Capture) (Channels 0 to 4, 6, and 7)

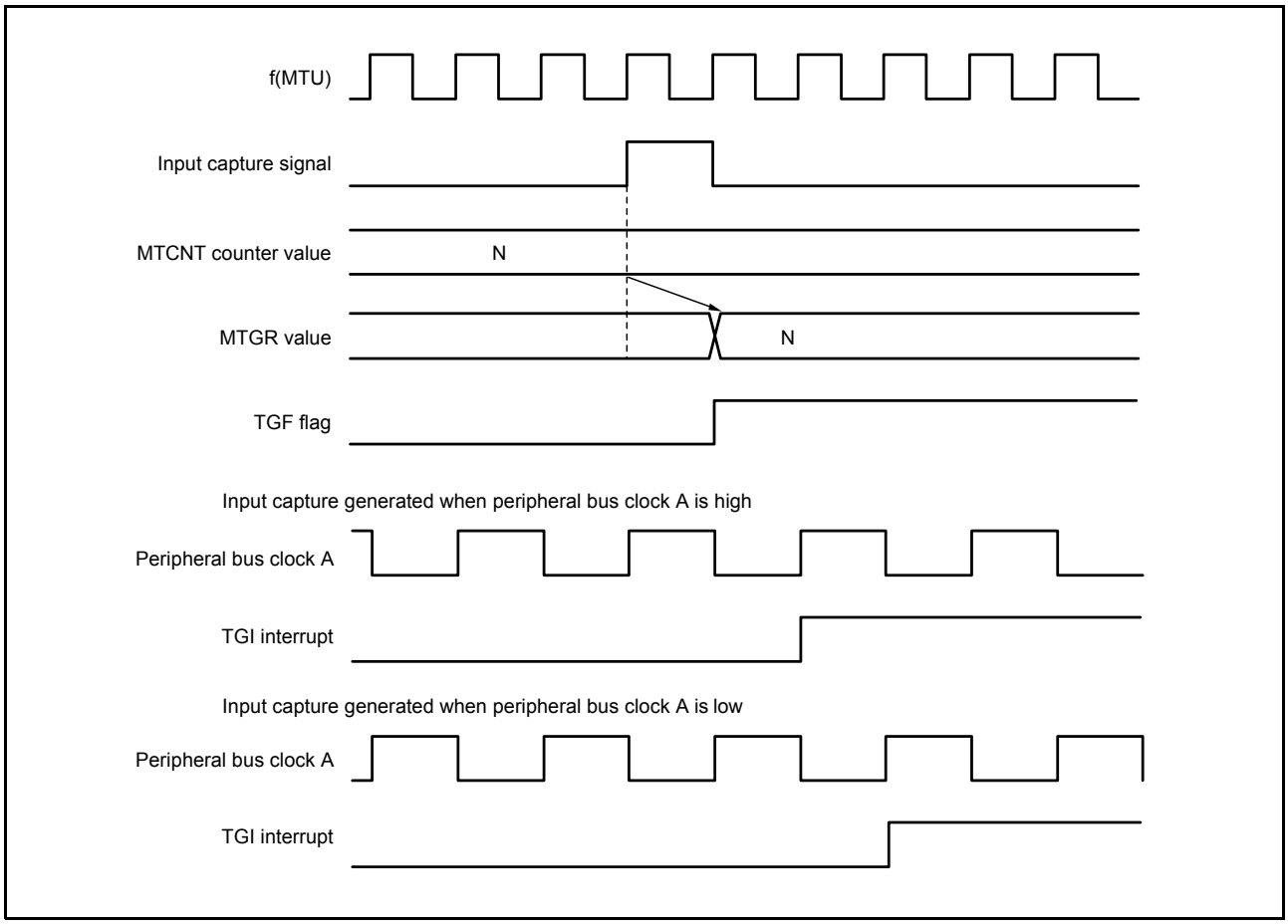


Figure 16.122 TGI Interrupt Timing (Input Capture) (Channel 5)

(3) TCFV Flag/TCFU Flag Setting Timing

Figure 16.123 shows the timing for setting of the TCFV flag in MTSR on overflow, and TCIV interrupt request signal timing.

Figure 16.124 shows the timing for setting of the TCFU flag in MTSR on underflow, and TCIU interrupt request signal timing.

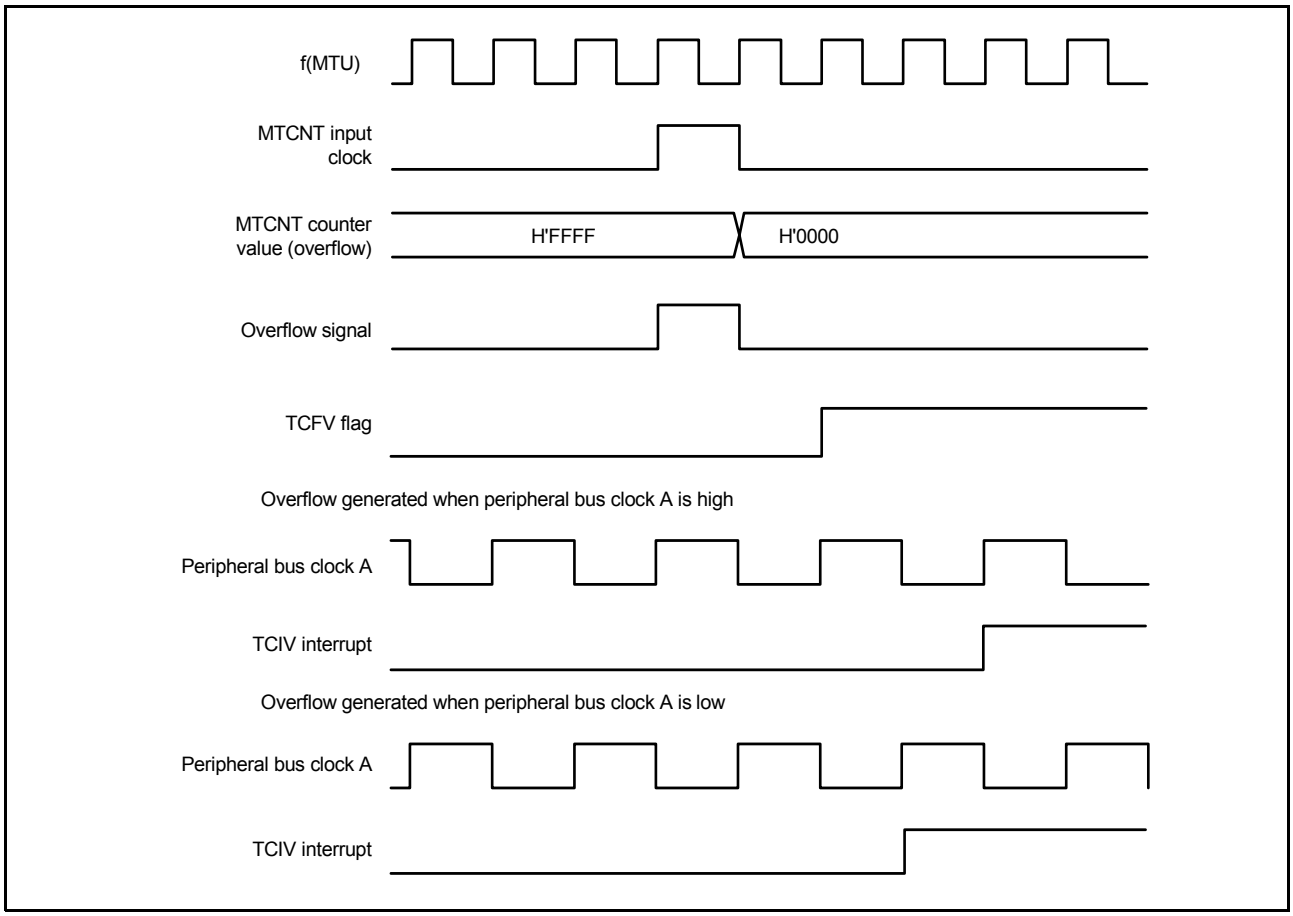


Figure 16.123 TCIV Interrupt Setting Timing

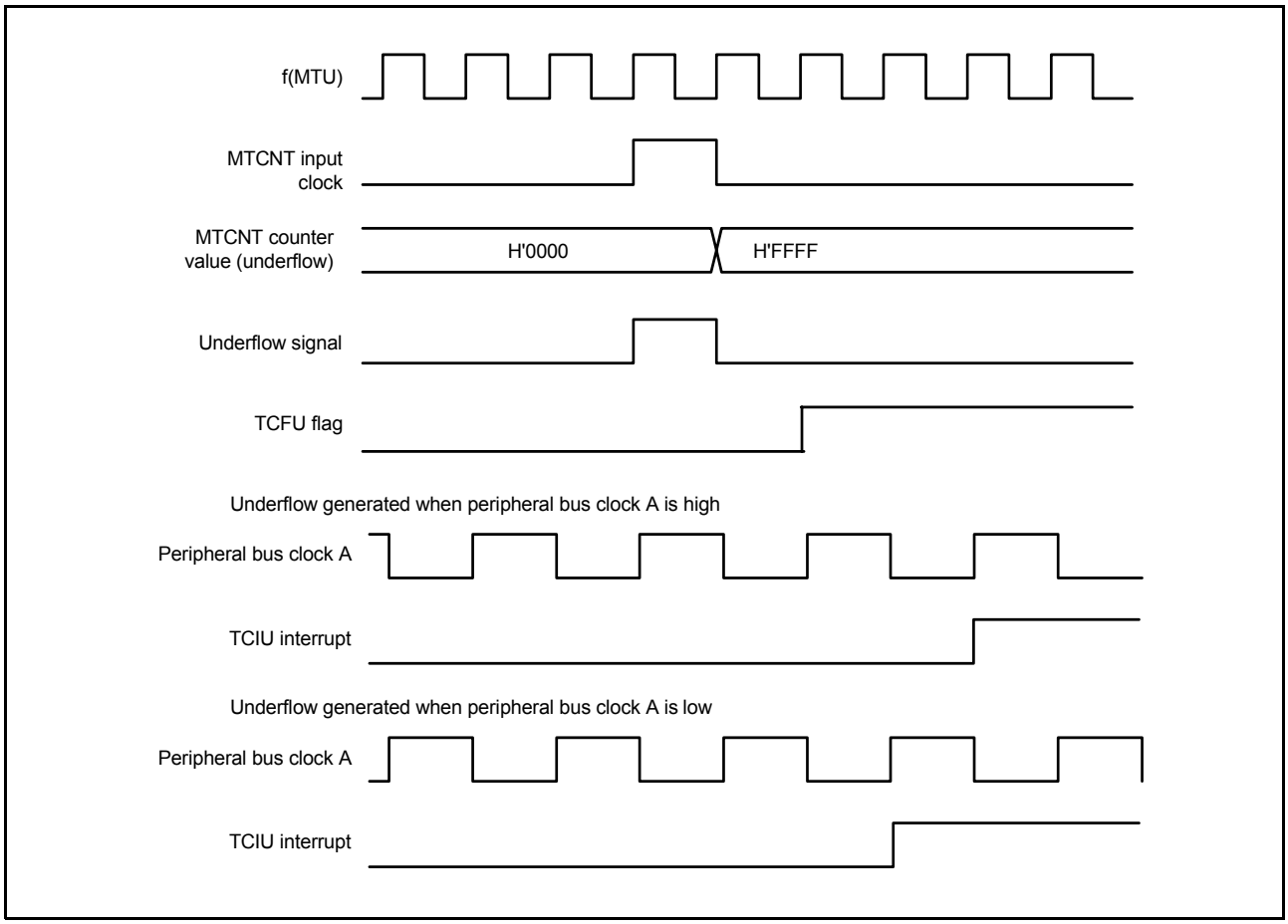


Figure 16.124 TCIU Interrupt Setting Timing

(4) Status Flag Clearing Timing

After the CPU reads a status flag as 1, it is cleared by writing 0 to it. When the DMAC is activated, the flag is cleared automatically. Figure 16.125 shows the timing for status flag clearing by the CPU, and figure 16.126 shows the timing for status flag clearing by the transfer acknowledge signal from the DMAC.

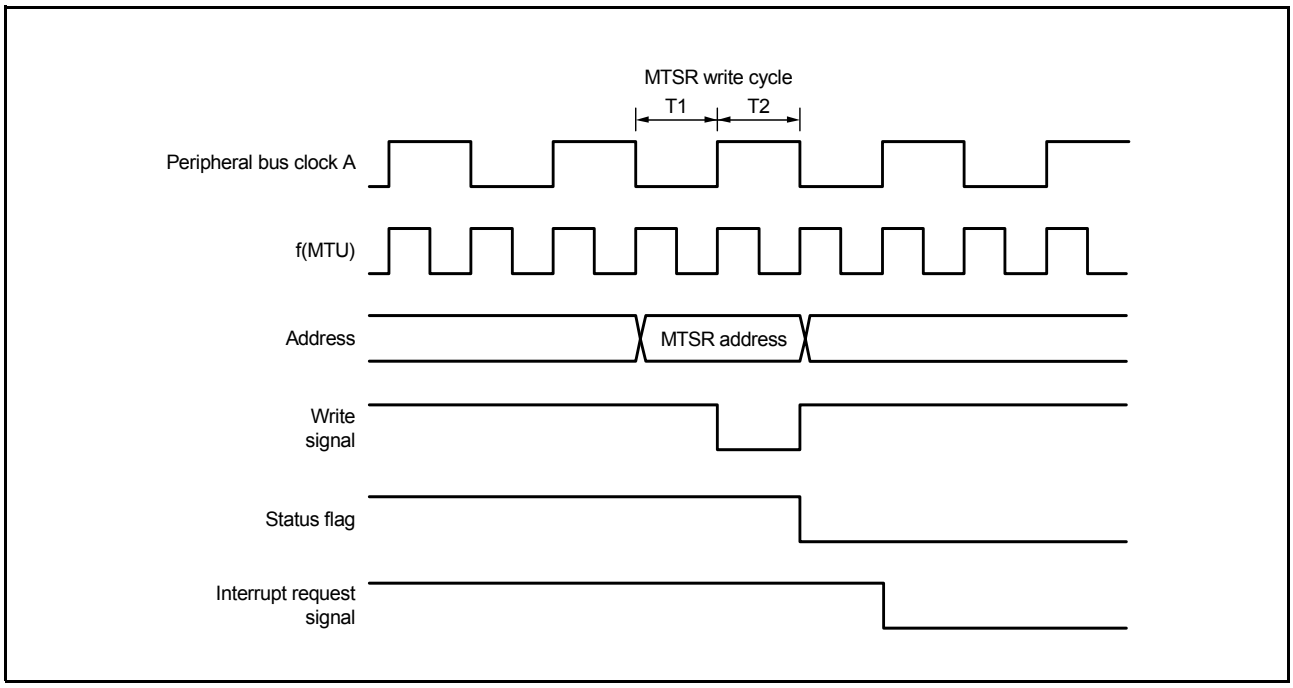


Figure 16.125 Timing for Status Flag Clearing by CPU (Channels 0 to 7)

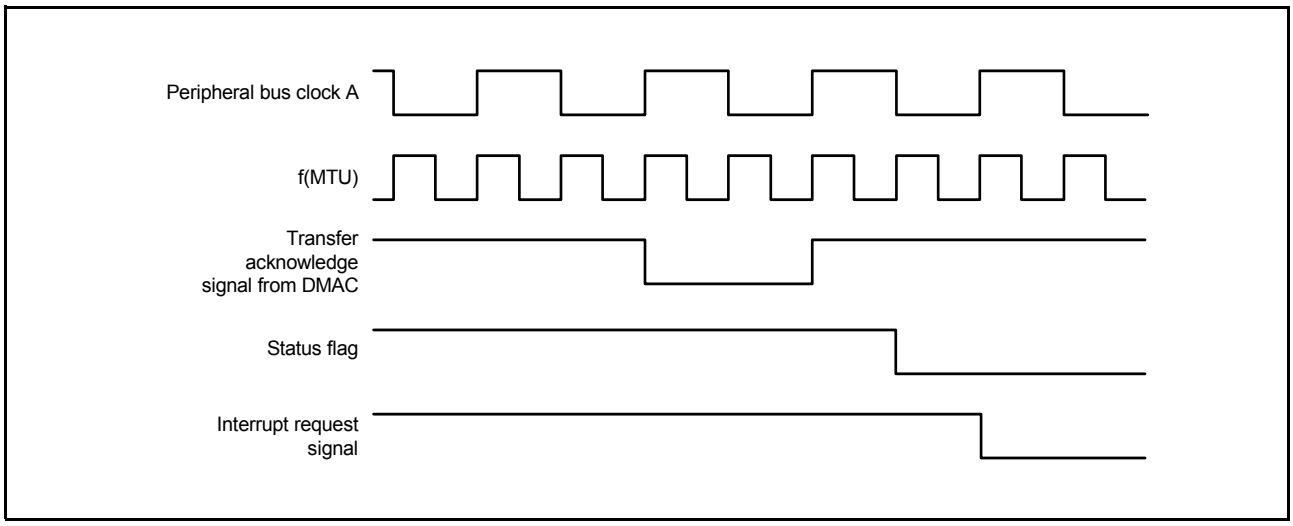


Figure 16.126 Timing for Status Flag Clearing by Transfer Acknowledge Signal from DMAC (Channels 0 to 7)

16.6 MTU-III Usage Notes

16.6.1 Input Clock Restrictions

The input clock pulse width must be at least 1.5 states in the case of single-edge detection, and at least 2.5 states in the case of both-edge detection. The MTU-III will not operate properly at narrower pulse widths.

In phase counting mode, the phase difference and overlap between the two input clocks must be at least 1.5 states, and the pulse width must be at least 2.5 states. Figure 16.127 shows the input clock conditions in phase counting mode.

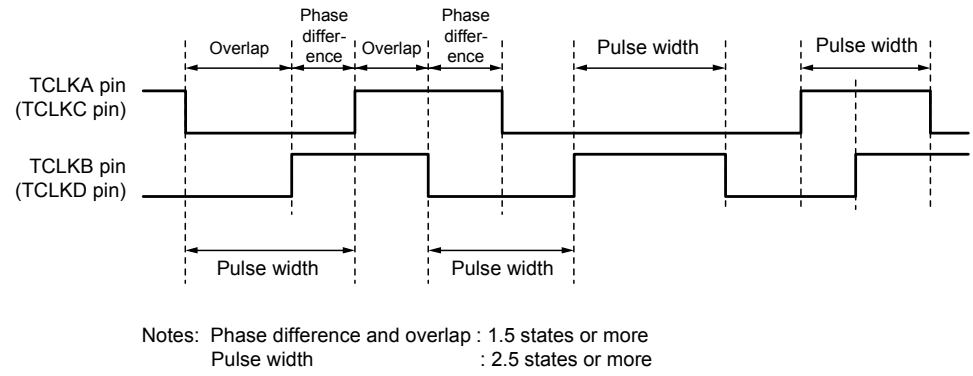


Figure 16.127 Phase Difference, Overlap, and Pulse Width in Phase Counting Mode

16.6.2 Caution on Period Setting

When counter clearing on compare match is set, the MTCNT counter is cleared in the final state in which it matches the MTGR value (the point at which the count value matched by the MTCNT counter is updated). Consequently, the actual counter frequency is given by the following formula:

- Channels 0 to 4, 6, and 7

$$f = \frac{f(MTU)}{(N + 1)}$$

- Channel 5

$$f = \frac{f(MTU)}{N}$$

f: Counter frequency
f(MTU): MTU operating clock
N: MTGR set value

16.6.3 Contention between MTCNT Counter Write and Clear Operations

If the counter clear signal is generated in the T2 state of a MTCNT counter write cycle, MTCNT counter clearing takes precedence and the MTCNT counter write is not performed.
Figure 16.128 shows the timing in this case.

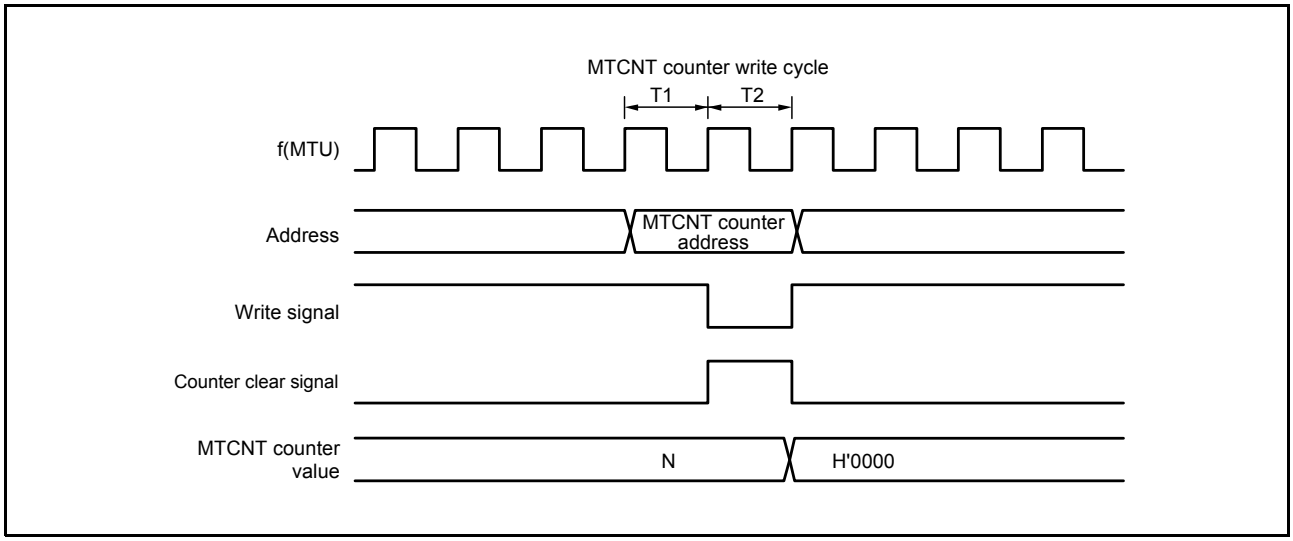


Figure 16.128 Contention between MTCNT Counter Write and Clear Operations

16.6.4 Contention between MTCNT Counter Write and Increment Operations

If incrementing occurs in the T2 state of a MTCNT counter write cycle, the MTCNT counter write takes precedence and the MTCNT counter is not incremented.

Figure 16.129 shows the timing in this case.

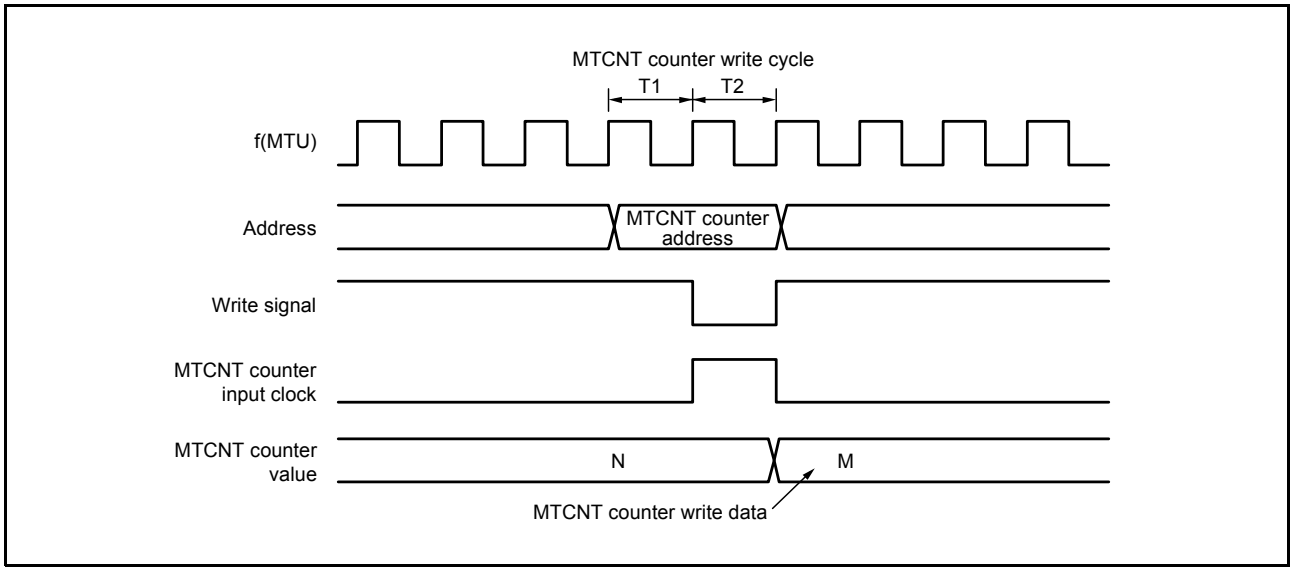


Figure 16.129 Contention between MTCNT Counter Write and Increment Operations

16.6.5 Contention between MTGR Write and Compare Match

If a compare match occurs in the T2 state of a MTGR write cycle, the MTGR write is executed and the compare match signal is also generated.

Figure 16.130 shows the timing in this case.

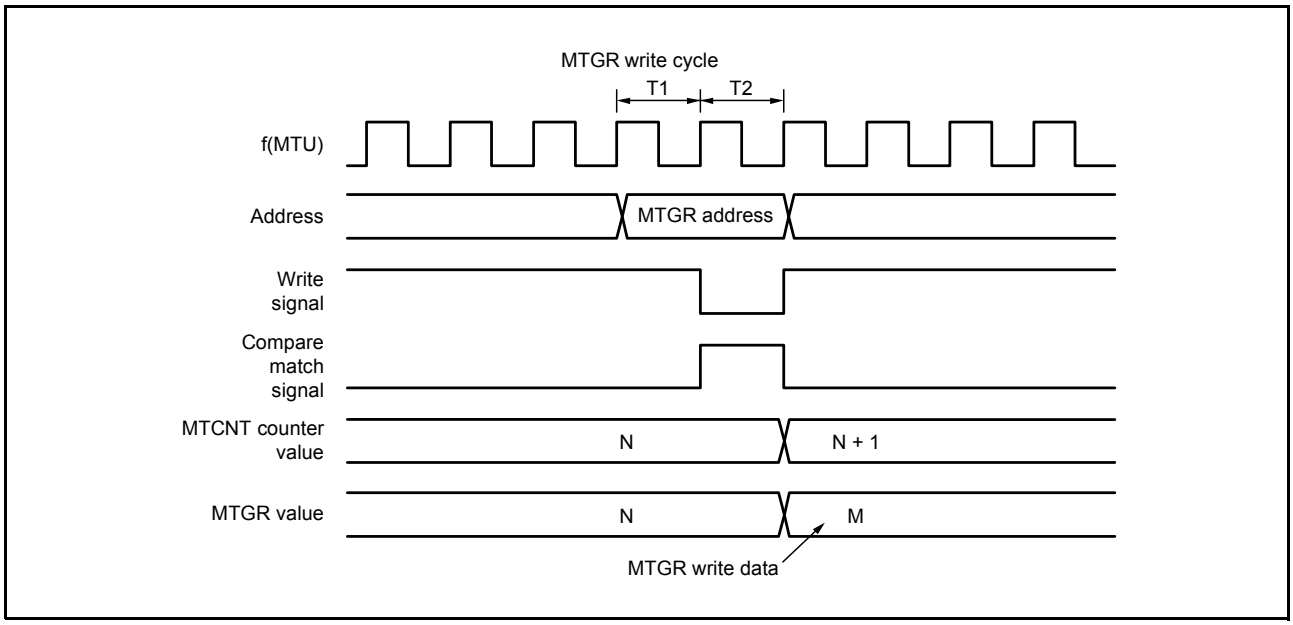


Figure 16.130 Contention between MTGR Write and Compare Match

16.6.6 Contention between Buffer Register Write and Compare Match

If a compare match occurs in the T2 state of a MTGR write cycle, the data that is transferred to MTGR by the buffer operation is the data before write.

Figure 16.131 shows the timing in this case.

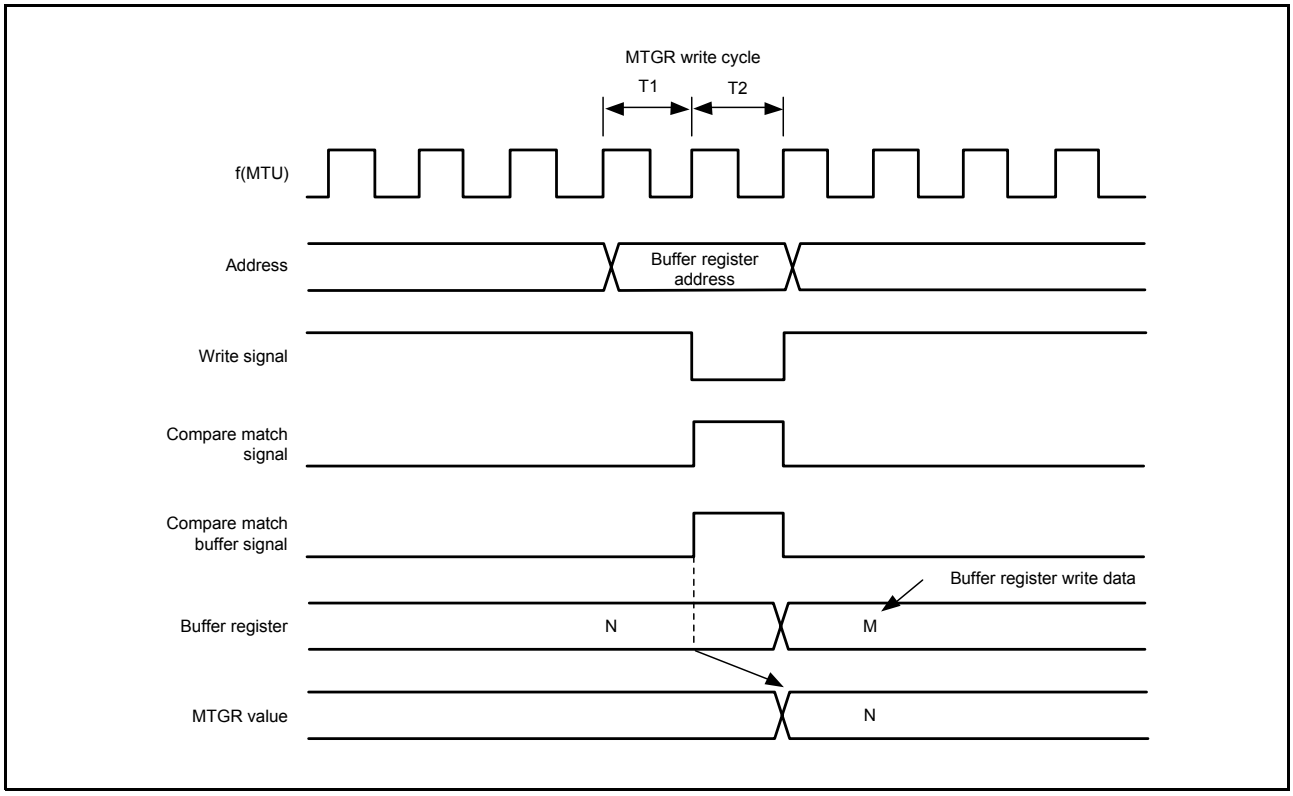


Figure 16.131 Contention between Buffer Register Write and Compare Match

16.6.7 Contention between Buffer Register Write and MTCNT Counter Clear

When the buffer transfer timing is set at the MTCNT counter clear by MTBTM, if the MTCNT counter clear occurs in the T2 state of a MTGR write cycle, the data that is transferred to MTGR by the buffer operation is the data before write.

Figure 16.132 shows the timing in this case.

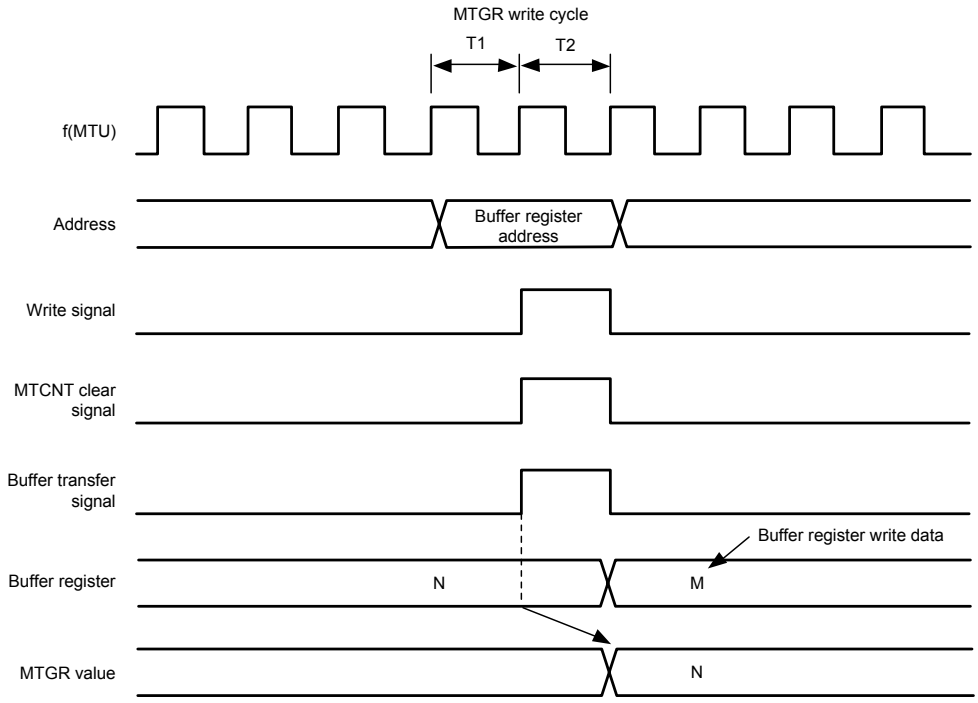


Figure 16.132 Contention between Buffer Register Write and MTCNT Counter Clear

16.6.8 Contention between MTGR Read and Input Capture

If an input capture signal is generated in the T1 state of a MTGR read cycle, the data that is read will be the data in the buffer before input capture transfer for channels 0 to 4, 6, and 7, and the data after input capture transfer for channel 5. Figure 16.133 and figure 16.134 show the timing in this case.

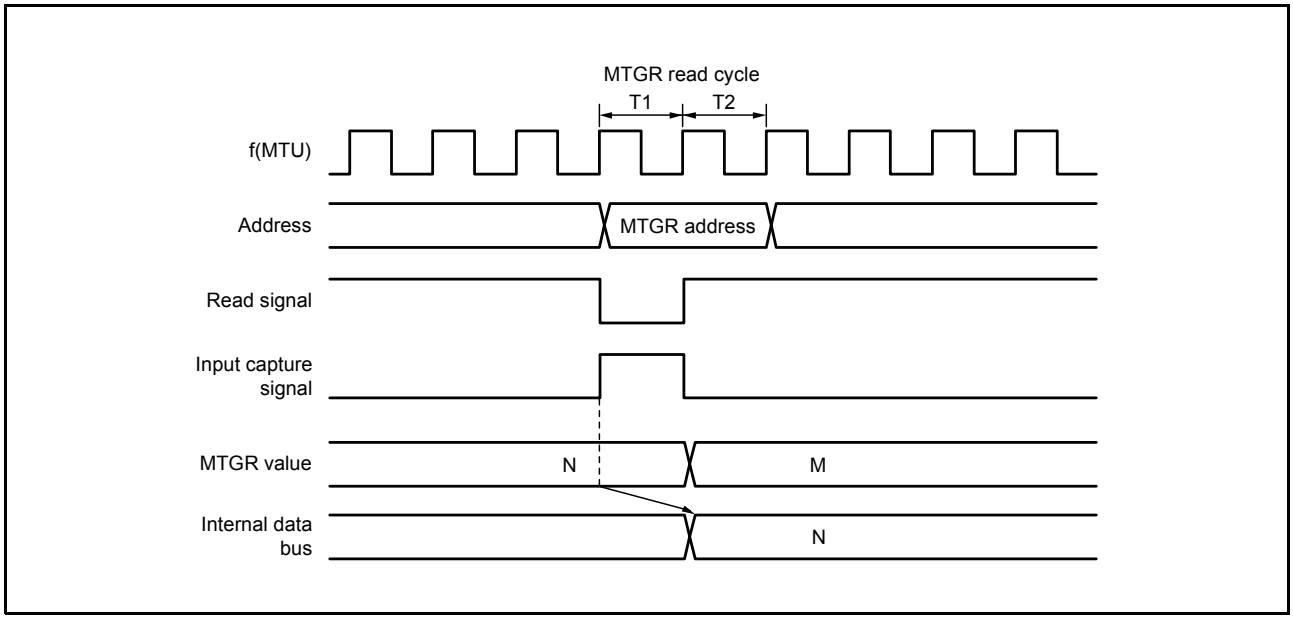


Figure 16.133 Contention between MTGR Read and Input Capture (Channels 0 to 4, 6, and 7)

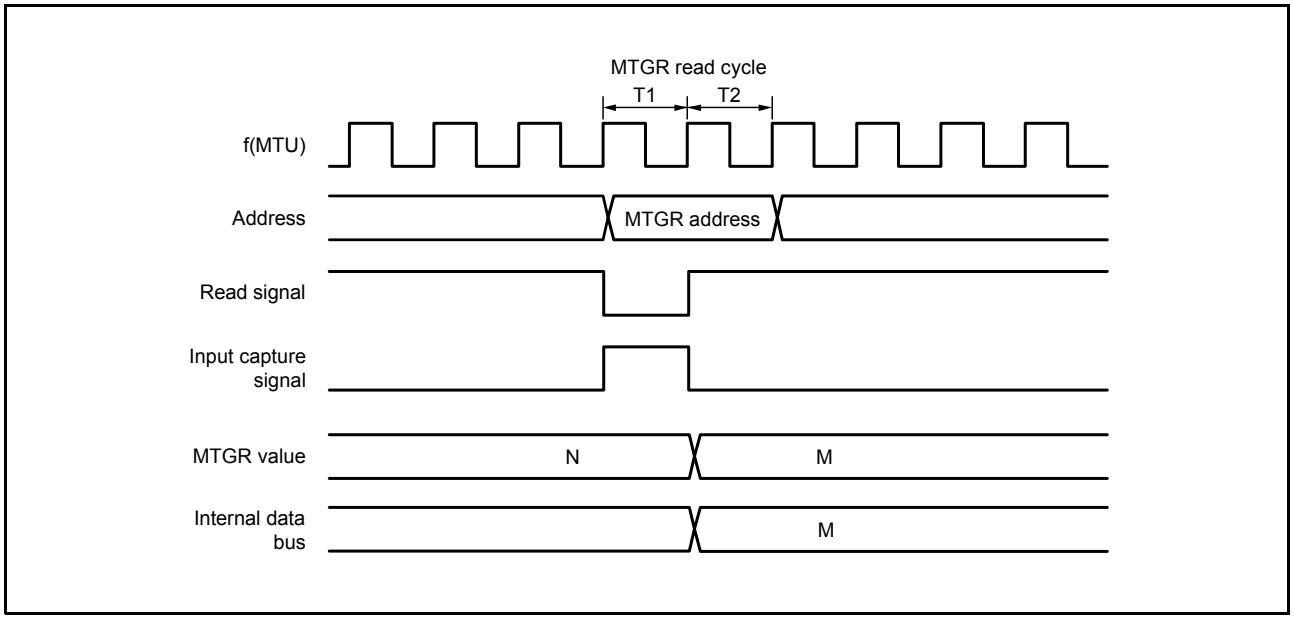


Figure 16.134 Contention between MTGR Read and Input Capture (Channel 5)

16.6.9 Contention between MTGR Write and Input Capture

If an input capture signal is generated in the T2 state of a MTGR write cycle, the input capture operation takes precedence and the write to MTGR is not performed for channels 0 to 4, 6, and 7. For channel 5, write to MTGR is performed and the input capture signal is generated.

Figure 16.135 and figure 16.136 show the timing in this case.

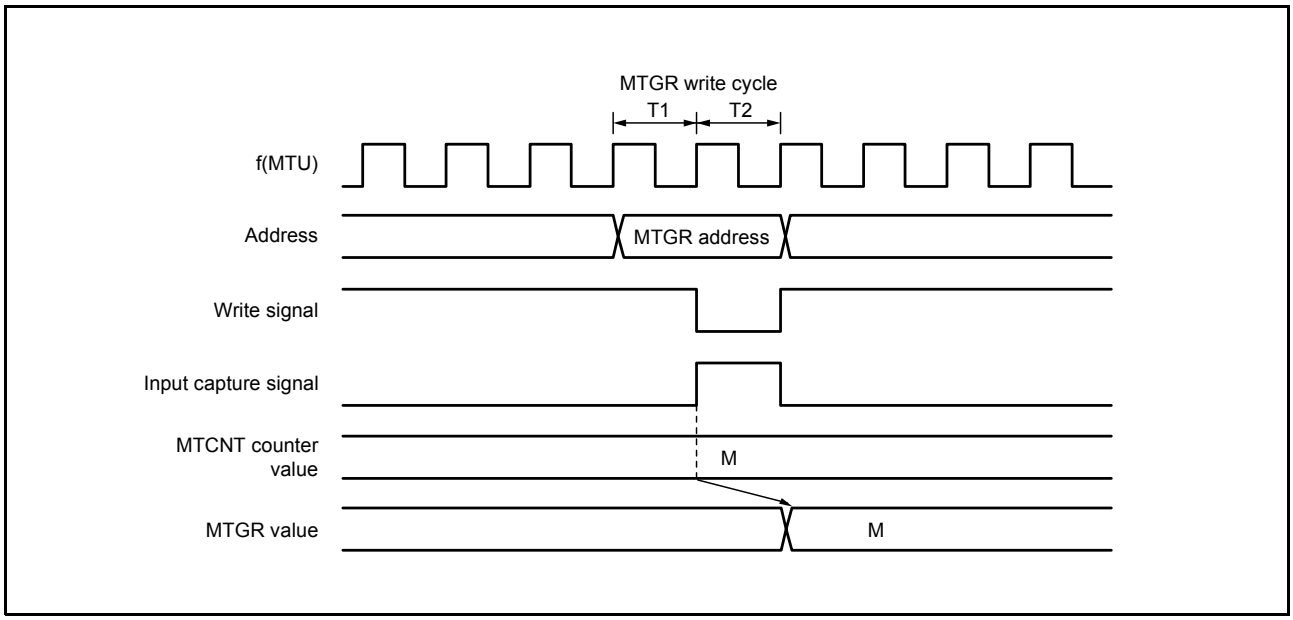


Figure 16.135 Contention between MTGR Write and Input Capture (Channels 0 to 4, 6, and 7)

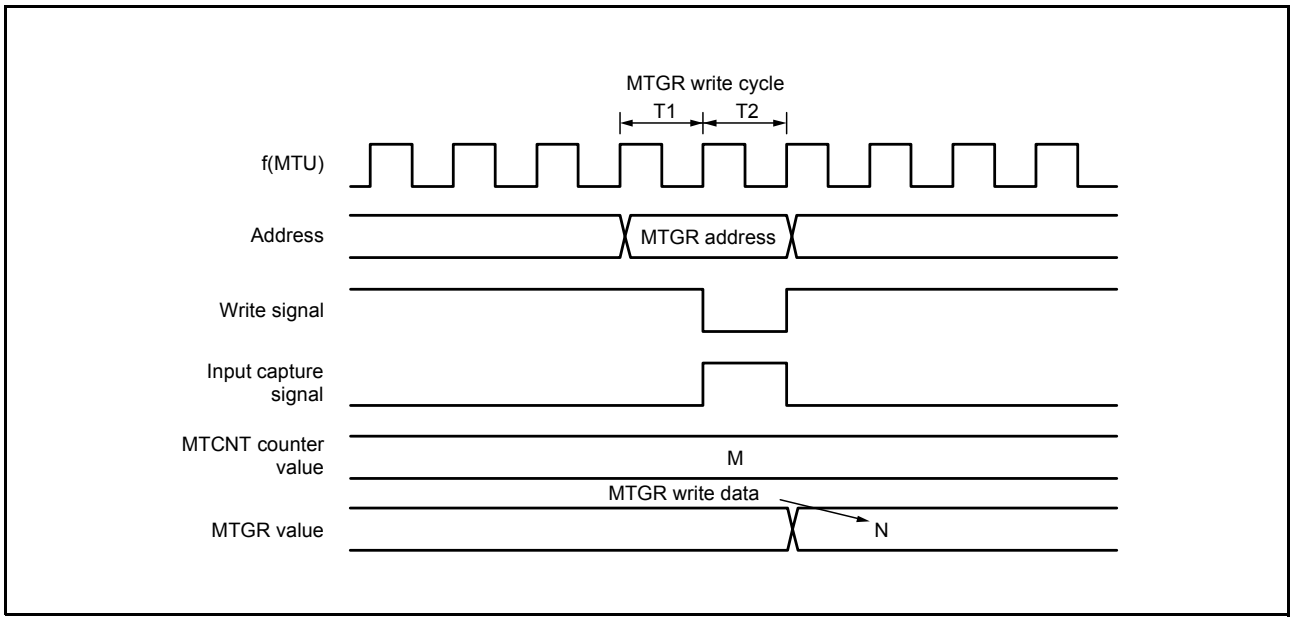


Figure 16.136 Contention between MTGR Write and Input Capture (Channel 5)

16.6.10 Contention between Buffer Register Write and Input Capture

If an input capture signal is generated in the T2 state of a buffer register write cycle, the buffer operation takes precedence and the write to the buffer register is not performed.

Figure 16.137 shows the timing in this case.

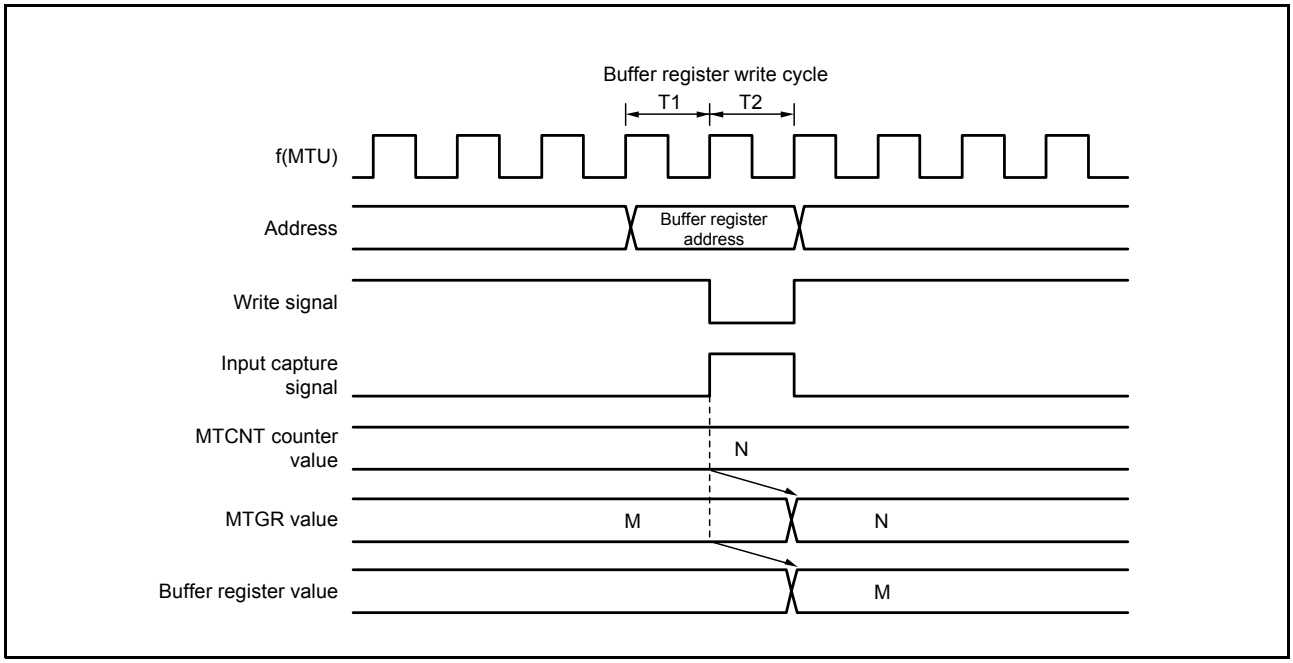


Figure 16.137 Contention between Buffer Register Write and Input Capture

16.6.11 MT2CNT Counter Write and Overflow/Underflow Contention in Cascade Connection

With timer counters MT1CNT and MT2CNT in a cascade connection, when a contention occurs during MT1CNT count (during a MT2CNT counter overflow/underflow) in the T2 state of the MT2CNT counter write cycle, the write to MT2CNT counter is conducted, and the MT1CNT count signal is disabled. At this point, if there is match with MT1GRA and the MT1CNT counter value, a compare signal is issued.

Furthermore, when the MT1CNT count clock is selected as the input capture source of channel 0, MT0GRA to MT0GRD carry out the input capture operation. In addition, when the compare match/input capture of MT0GRC is selected as the input capture source of MT1GRB, MT1GRB carries out input capture operation.

The timing is shown in figure 16.138.

For cascade connections, be sure to synchronize settings for channels 1 and 2 when setting MTCNT counter clearing.

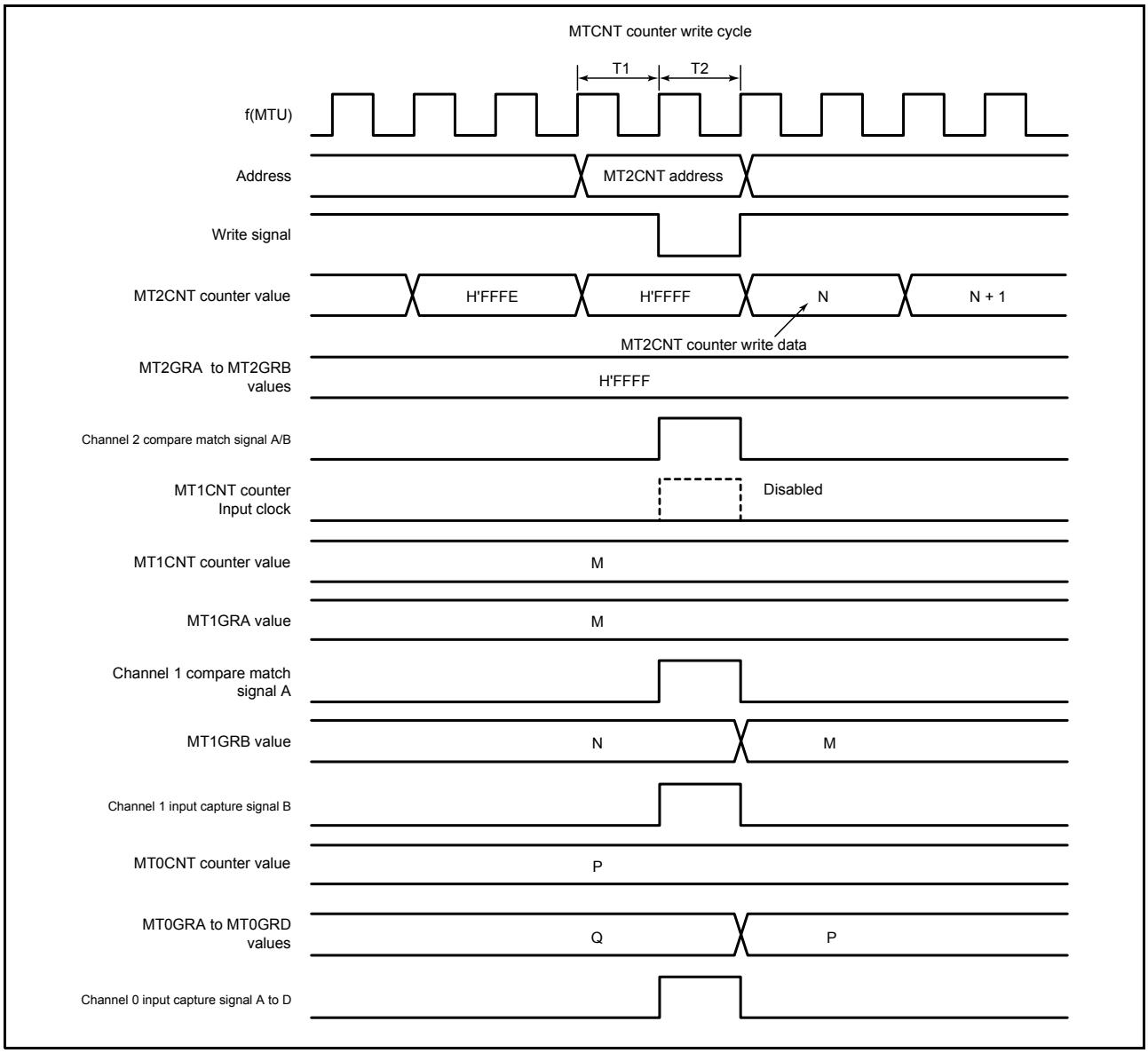


Figure 16.138 MT2CNT Counter Write and Overflow/Underflow Contention in Cascade Connection

16.6.12 Counter Value during Complementary PWM Mode Stop

When counting operation is suspended with counters MT3CNT and MT4CNT (MT6CNT and MT7CNT) in complementary PWM mode, the MT3CNT (MT6CNT) counter has the MTDTD value, and the MT4CNT (MT7CNT) counter is held at H'0000.

When restarting complementary PWM mode, counting begins automatically from the initialized state.

This explanatory diagram is shown in figure 16.139.

When counting begins in another operating mode, be sure that counters MT3CNT and MT4CNT (MT6CNT and MT7CNT) are set to the initial values.

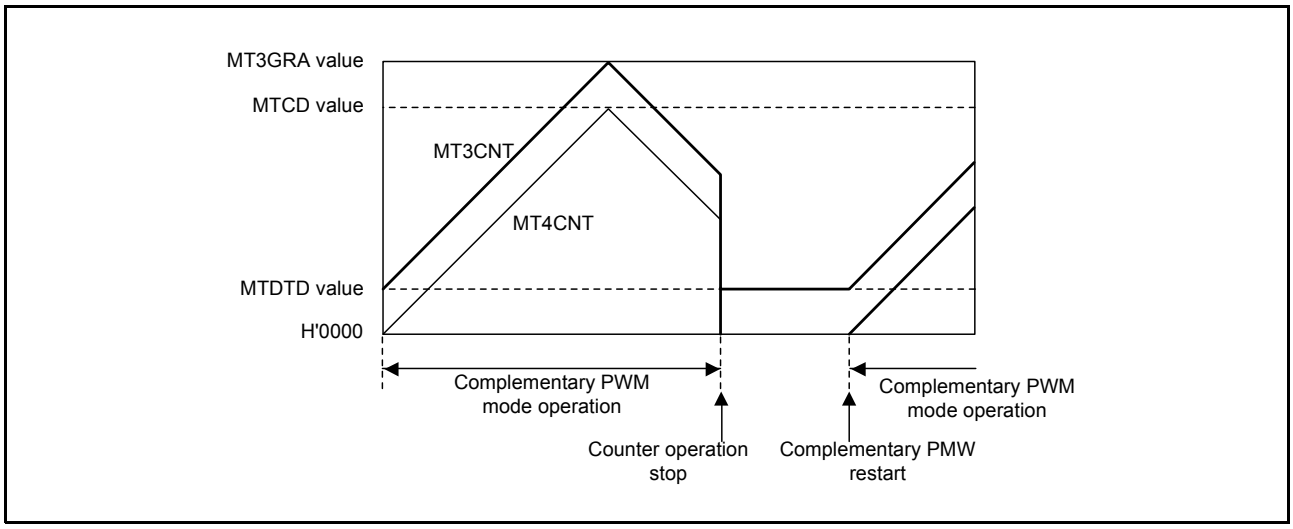


Figure 16.139 Counter Value during Complementary PWM Mode Stop

16.6.13 Buffer Operation Setting in Complementary PWM Mode

In complementary PWM mode, conduct rewrites by buffer operation for the PWM cycle setting register (MT3GRA or MT6GRA), MTCD, and duty setting registers (MT3GRB, MT4GRA, MT4GRB, MT6GRB, MT7GRA, and MT7GRB).

In complementary PWM mode, channel 3 and channel 4 (channel 6 and channel 7) buffers operate in accordance with bit settings BFA and BFB of MT3MD0 (MT6MD0). When the BFA bit in MT3MD0 (MT6MD0) is set to 1, MT3GRC (MT6GRC) functions as a buffer register for MT3GRA (MT6GRA). At the same time, MT4GRC (MT7GRC) functions as the buffer register for MT4GRA (MT7GRA), and MT34CB (MT67CB) functions as the MT34CD (MT67CD) buffer register.

16.6.14 Reset-Synchronized PWM Mode Buffer Operation and Compare Match Flag

When setting buffer operation for reset synchronized PWM mode, set the BFA and BFB bits in MT4MD0 (MT7MD0) to 0. The TIOC4C (TIOC7C) pin will be unable to produce its waveform output if the BFA bit in MT4MD0 (MT7MD0) is set to 1.

In reset synchronized PWM mode, the channel 3 and channel 4 (channel 6 and channel 7) buffers operate in accordance with the BFA and BFB bit settings of MT3MD0 (MT6MD0). For example, when the BFA bit in MT3MD0 (MT6MD0) is set to 1, MT3GRC (MT6GRC) functions as a buffer register for MT3GRA (MT6GRA). At the same time, MT4GRC (MT7GRC) functions as the buffer register for MT4GRA (MT7GRA).

The TGFC flag and TGFD flag in MT3SR0 and MT4SR0 (MT6SR0 and MT7SR0) are not set when MT3GRC (MT6GRC) and MT3GRD (MT6GRD) are operating as buffer registers.

Figure 16.140 shows an operation example in the case when bits BFA and BFB in MT3MD0 (MT6MD0) are set to 1 and bits BFA and BFB in MT4MD0 (MT7MD0) are set to 0.

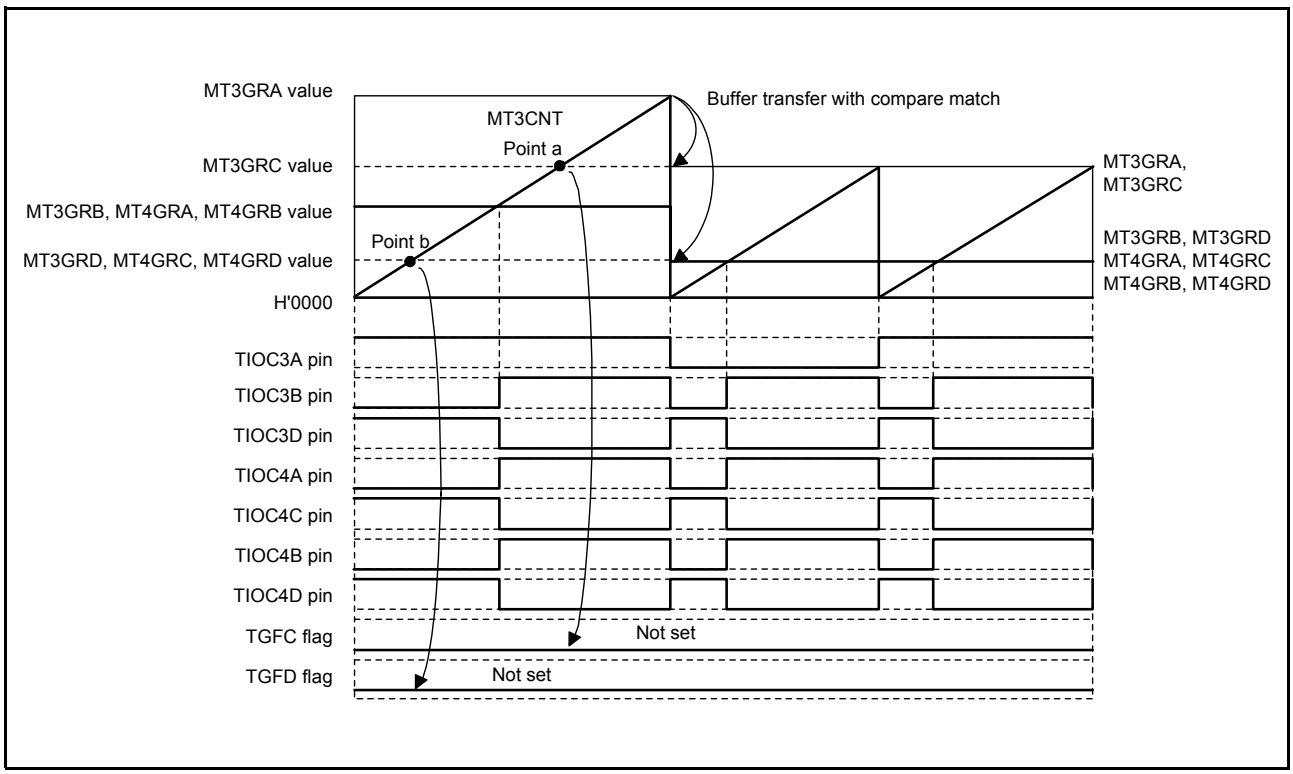


Figure 16.140 Buffer Operation and Compare-Match Flags in Reset-Synchronized PWM Mode

16.6.15 Overflow Flags in Reset-synchronized PWM Mode

When set to reset-synchronized PWM mode, counters MT3CNT and MT4CNT (MT6CNT and MT7CNT) start counting when the CST3 (CST6) bit of MT01234STR (MT67STR) is set to 1. At this point, the MT4CNT (MT7CNT) count clock source and count edge obey the MT3CR (MT6CR) setting.

In reset-synchronized PWM mode, with cycle register MT3GRA (MT6GRA) set value at H'FFFF, when specifying MT3GRA (MT6GRA) compare-match for the counter clear source, MT3CNT and MT4CNT (MT6CNT and MT7CNT) count up to H'FFFF, then a compare-match occurs with MT3GRA (MT6GRA), and counters MT3CNT and MT4CNT (MT4CNT and MT7CNT) are both cleared. At this point, MTSR's overflow flag TCFV bit is not set.

Figure 16.141 shows a TCFV bit operation example in reset-synchronized PWM mode with a set value for cycle register MT3GRA (MT6GRA) of H'FFFF, when a MT3GRA (MT6GRA) compare-match has been specified without synchronous setting for the counter clear source.

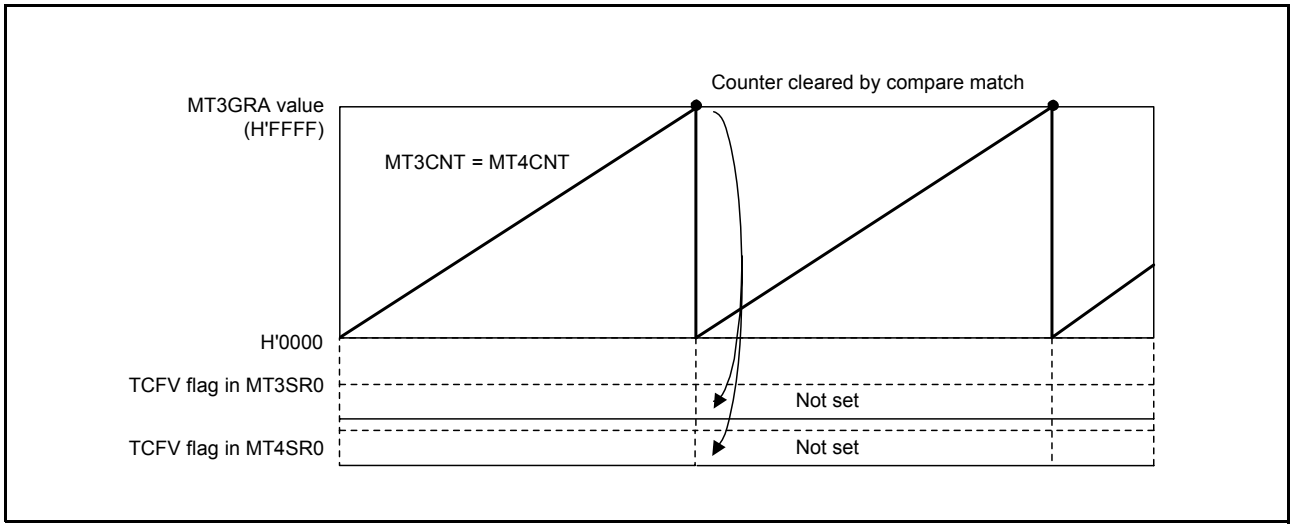


Figure 16.141 Reset-Synchronized PWM Mode Overflow Flag

16.6.16 Contention between Overflow/Underflow and Counter Clearing

If overflow/underflow and counter clearing occur simultaneously, the TCFV/TCFU flag in MTSR is not set and MTCNT counter clearing takes precedence.

Figure 16.142 shows the operation timing when a MTGR compare match is specified as the clearing source, and when H'FFFF is set in MTGR.

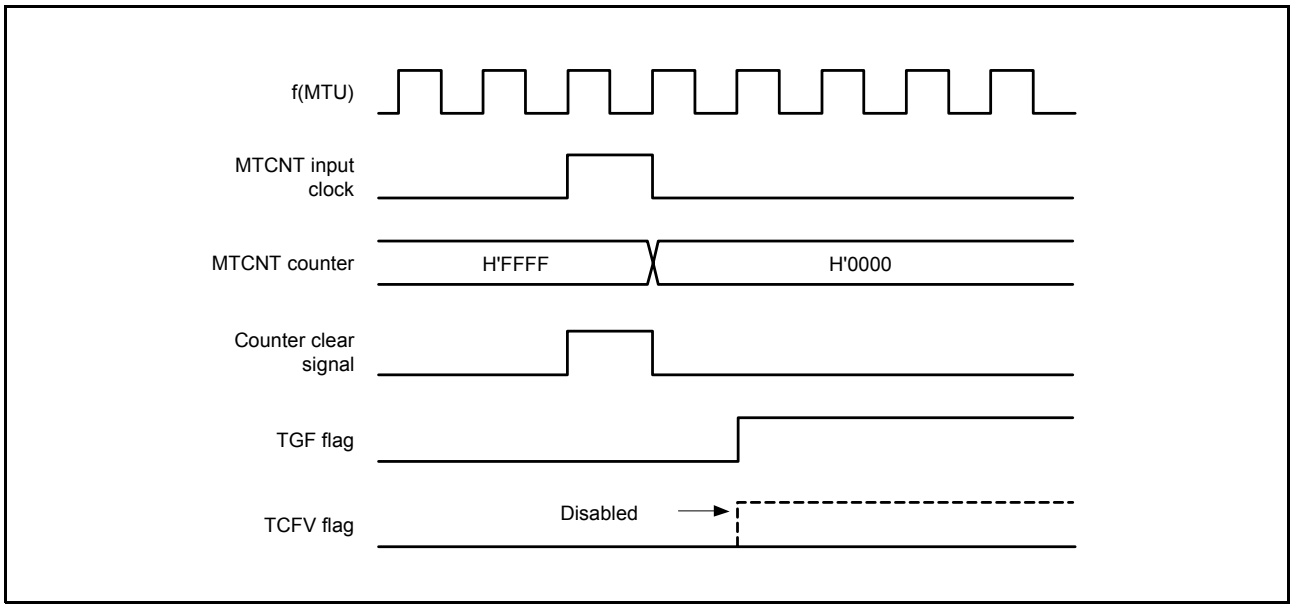


Figure 16.142 Contention between Overflow and Counter Clearing

16.6.17 Contention between MTCNT Counter Write and Overflow/Underflow

If there is an up-count or down-count in the T2 state of an MTCNT counter write cycle, and overflow/underflow occurs, the MTCNT counter write takes precedence and the TCFV/TCFU flag in MTSR is not set.

Figure 16.143 shows the operation timing when there is contention between MTCNT counter write and overflow.

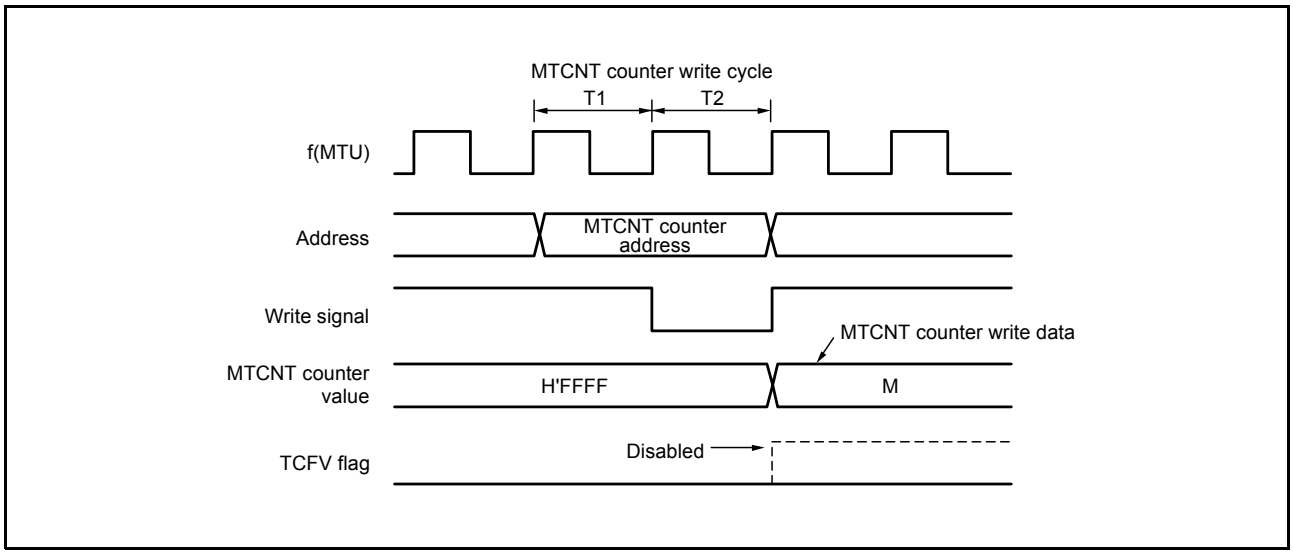


Figure 16.143 Contention between MTCNT Counter Write and Overflow Operations

16.6.18 Cautions on Transition from Normal Operation or PWM Mode 1 to Reset-Synchronized PWM Mode

When making a transition from channel 3 or 4 (channel 6 or 7) normal operation or PWM mode 1 to reset-synchronized PWM mode, if the counter is halted with the output pins (TIOC3B, TIOC3D, TIOC4A, TIOC4C, TIOC4B, TIOC4D, TIOC6B, TIOC6D, TIOC7A, TIOC7C, TIOC7B, TIOC7D) in the high-level state, followed by the transition to reset-synchronized PWM mode and operation in that mode, the initial pin output will not be correct.

When making a transition from normal operation to reset-synchronized PWM mode, write H'11 to registers MT3IOCR0, MT3IOCR1, MT4IOCR0, and MT4IOCR1 (MT6IOCR0, MT6IOCR1, MT7IOCR0, and MT7IOCR1) to initialize the output pins to low level output, then set an initial register value of H'00 before making the mode transition.

When making a transition from PWM mode 1 to reset-synchronized PWM mode, first switch to normal operation, then initialize the output pins to low level output and set an initial register value of H'00 before making the transition to reset-synchronized PWM mode.

16.6.19 Output Level in Complementary PWM Mode and Reset-Synchronized PWM Mode

When channels 3 and 4 (channels 6 and 7) are in complementary PWM mode or reset-synchronized PWM mode, the PWM waveform output level is set with the OLSP and OLSN bits in MTOCR0. In the case of complementary PWM mode or reset-synchronized PWM mode, MTIOCR should be set to H'00.

16.6.20 Simultaneous Capture of Counters MT1CNT and MT2CNT in Cascade Connection

When counters MT1CNT and MT2CNT are operated as a 32-bit counter in cascade connection, the cascade counter value cannot be captured successfully even if input-capture input is simultaneously done to pins TIOC1A and TIOC2A or to TIOC1B and TIOC2B. This is because the input timing of TIOC1A and TIOC2A or of TIOC1B and TIOC2B may not be the same when external input-capture signals to be input into counters MT1CNT and MT2CNT are taken in synchronization with the internal clock.

For example, the MT1CNT counter (the counter for upper 16 bits) does not capture the count-up value by overflow from the MT2CNT counter (the counter for lower 16 bits) but captures the count value before the count-up. In this case, the values of MT1CNT = H'FFF1 and MT2CNT = H'0000 should be transferred to MT1GRA and MT2GRA or to MT1GRB and MT2GRB, but the values of MT1CNT = H'FFF0 and MT2CNT = H'0000 are erroneously transferred.

16.6.21 Control of Output Waveform at Synchronous Counter Clearing in Complementary PWM Mode

When either condition (1) or (2) below is satisfied while control of output waveforms at synchronous counter clearing is enabled (the WRE bit in MTWCR is 1) in complementary PWM mode, the MTU-III will operate as follows.

- The dead time for the PWM output pins will be shorter or no dead time will be generated.
- The negative-phase PWM output pins will output the active level outside the active-level output period.

Condition (1): Synchronous clearing occurs during the PWM output dead time in initial output disabled period (10) (figure 16.144).

Condition (2): Synchronous clearing occurs when MT3GRB ≤ MTDTD, MT4GRA ≤ MTDTD, or MT4GRB ≤ MTDTD (MT6GRB ≤ MTDTD, MT7GRA ≤ MTDTD, or MT7GRB ≤ MTDTD) is satisfied in initial output disabled period (10) or (11) (figure 16.145).

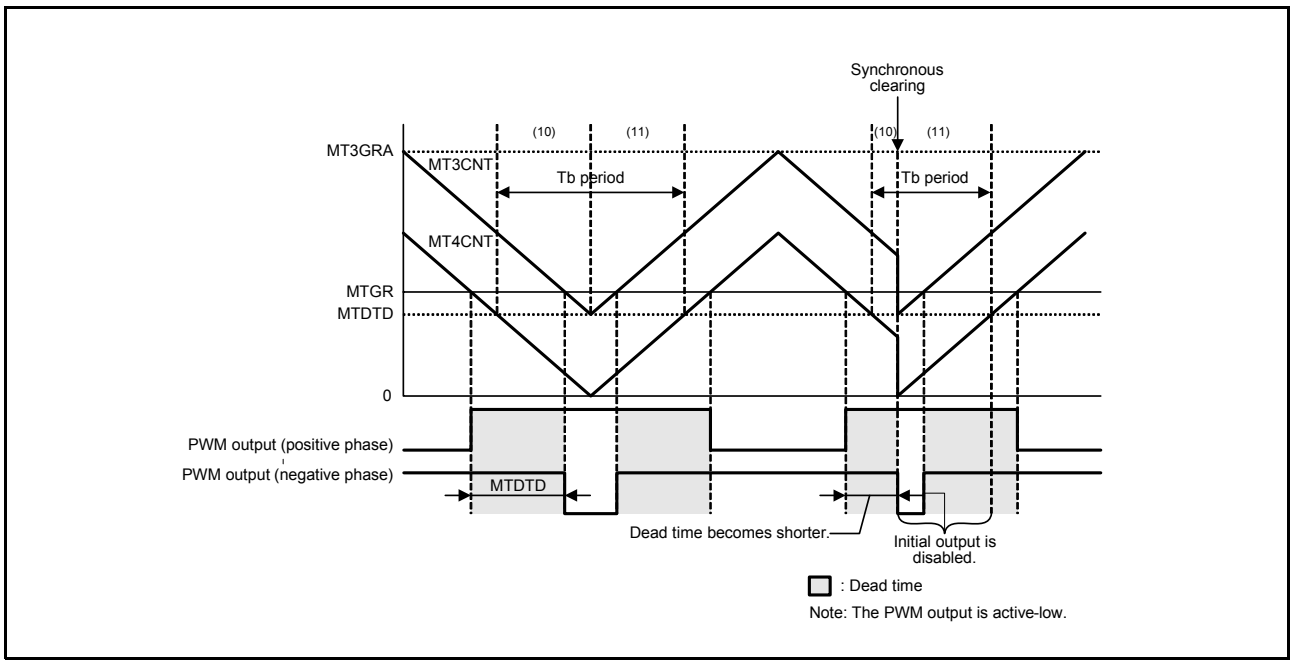


Figure 16.144 Example of Synchronous Clearing under Condition (1)

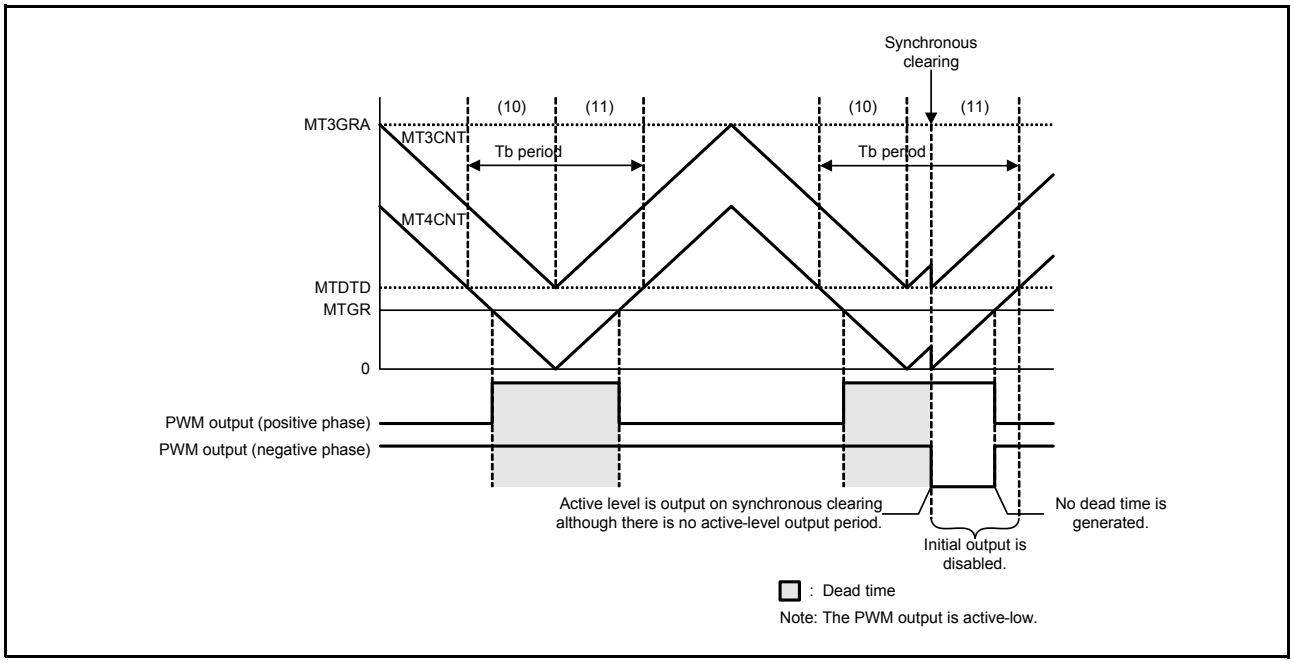


Figure 16.145 Example of Synchronous Clearing under Condition (2)

To avoid this, generate counter clearing only while MT3GRB ≥ MTDTD × 2, MT4GRA ≥ MTDTD × 2, and MT4GRB ≥ MTDTD × 2 (MT6GRB ≥ MTDTD × 2, MT7GRA ≥ MTDTD × 2, and MT7GRB ≥ MTDTD × 2) are all satisfied.

16.6.22 Square Wave Output Switching Function

When using the square wave output switching function, the timer prescaler bits (TPSC[2:0] bits in the MTCR register) in channels 3 and 4 should be set to 000 (internal clock: counts on the MTU operating clock without frequency division). Otherwise, a dead time during square wave switching is not output correctly.

16.6.23 Interrupt Skipping Function 2

When interrupt skipping function 2 is in use and the interval between updating of the values in TADCORA_4 and TADCORB_4 is short, correct counting of the interval to be skipped becomes impossible, and requests for A/D conversion may not be generated with the expected timing. Use this function under the following conditions.
In channels 6 and 7, TADCORA_7 and TADCORB_7 should also be set in the same manner.

(1) Skipping function 2 is used and the skipping count is zero.

- The interval between updating of the TADCORA_4 and TADCORB_4 registers is at least four.
- The interval for comparison with TADCORA_4 is at least 4PMϕ. The updated value of TADCORA_4 is the previous value plus at least four or the previous value minus at least four.
- The interval for comparison with TADCORB_4 is at least 4PMϕ. The updated value of TADCORB_4 is the previous value plus at least four or the previous value minus at least four.

(2) Skipping function 2 is used and the skipping count is one or more.

- The interval between updating of the TADCORA_4 and TADCORB_4 registers is at least two.
- The interval for comparison with TADCORB_4 is at least 2PMϕ. The updated value of TADCORB_4 is the previous value plus at least two or the previous value minus at least two.

16.7 MTU-III Output Pin Initialization

16.7.1 Operating Modes

The MTU-III has the following six operating modes. Waveform output is possible in all of these modes.

- Normal mode (channels 0 to 4, 6, and 7)
- PWM mode 1 (channels 0 to 4, 6, and 7)
- PWM mode 2 (channels 0 to 2)
- Phase counting modes 1 to 4 (channels 1 and 2)
- Complementary PWM mode (channels 3, 4, 6, and 7)
- Reset-synchronized PWM mode (channels 3, 4, 6, and 7)

The MTU-III output pin initialization method for each of these modes is described in this section.

16.7.2 Reset Start Operation

The MTU-III output pins (TIOC*) are initialized low by a reset. For functional settings for MTU-III pins, see section 13, I/O Ports. When the setting is completed, the MTU-III pin states at that point are output to the ports. When the MTU-III output is selected immediately after a reset, the MTU-III output initial level, low, is output directly at the port. When the active level is low, the system will operate at this point, and therefore the pin function setting should be made after initialization of the MTU-III output pins is completed.

Note: * Channel number and port notation

16.7.3 Operation in Case of Re-Setting Due to Error During Operation

If an error occurs during MTU-III operation, MTU-III output should be cut by the system. Cutoff is performed by switching the pin output to port output with the pin function setting and outputting the inverse of the active level. For PWM output pins, output can also be cut by hardware, using the emergency output shutdown function (SD).

The pin initialization procedures for re-setting due to an error during operation, etc., and the procedures for restarting in a different mode after re-setting, are shown below. The MTU3 has six operating modes, as stated above. There are thus 36 mode transition combinations, but some transitions are not available with certain channel and mode combinations. Possible mode transition combinations are shown in Table 16.78.

Table 16.78 Mode Transition Combinations

	Normal	PWM1	PWM2	PCM	CPWM	RPWM
Normal	(1)	(2)	(3)	(4)	(5)	(6)
PWM1	(7)	(8)	(9)	(10)	(11)	(12)
PWM2	(13)	(14)	(15)	(16)	None	None
PCM	(17)	(18)	(19)	(20)	None	None
CPWM	(21)	(22)	None	None	(23) (24)	(25)
RPWM	(26)	(27)	None	None	(28)	(29)

[Legend]
Normal: Normal mode
PWM1: PWM mode 1
PWM2: PWM mode 2
PCM: Phase counting modes 1 to 4
CPWM: Complementary PWM mode
RPWM: Reset-synchronized PWM mode

16.7.4 Overview of Initialization Procedures and Mode Transitions in Case of Error during Operation

- When making a transition to a mode (Normal, PWM1, PWM2, PCM) in which the pin output level is selected by MTIOCR setting, initialize the pins by means of a MTIOCR setting.
- In PWM mode 1, since a waveform is not output to the TIOC*B (TIOC*D) pin*, setting MTIOCR will not initialize it. If initialization is required, carry it out in normal mode, then switch to PWM mode 1.
- In PWM mode 2, since a waveform is not output to the cycle register pin, setting MTIOCR will not initialize it. If initialization is required, carry it out in normal mode, then switch to PWM mode 2.
- In normal mode or PWM mode 2, if MTGRC and MTGRD operate as buffer registers, setting MTIOCR will not initialize the buffer register pins. If initialization is required, clear buffer mode, carry out initialization, then set buffer mode again.
- In PWM mode 1, if one of MTGRC and MTGRD operates as buffer registers, setting MTIOCR will not initialize the MTGRC pin. To initialize the MTGRC pin, clear buffer mode, carry out initialization, then set buffer mode again.
- When making a transition to a mode (CPWM, RPWM) in which the pin output level is selected by MTOCR0 and MTOCR1 setting, switch to normal mode and perform initialization with MTIOCR, then restore MTIOCR to its initial value, and temporarily disable channel 3 and 4 (channel 6 and 7) output with MTOEN. Then operate the unit in accordance with the mode setting procedure (MT34OCR0 setting, MT34OCR1 setting, MTMD0 setting, MT34OEN setting (MT67OCR0 setting, MT67OCR1 setting, MTMD0 setting, MT67OEN setting)).

Note: * Channel number is substituted for * indicated in this article.

Pin initialization procedures are described below for the numbered combinations in table 16.78. The active level is assumed to be low.

(1) Operation when Error Occurs during Normal Mode Operation, and Operation is Restarted in Normal Mode

Figure 16.146 shows an explanatory diagram of the case where an error occurs in normal mode and operation is restarted in normal mode after re-setting.

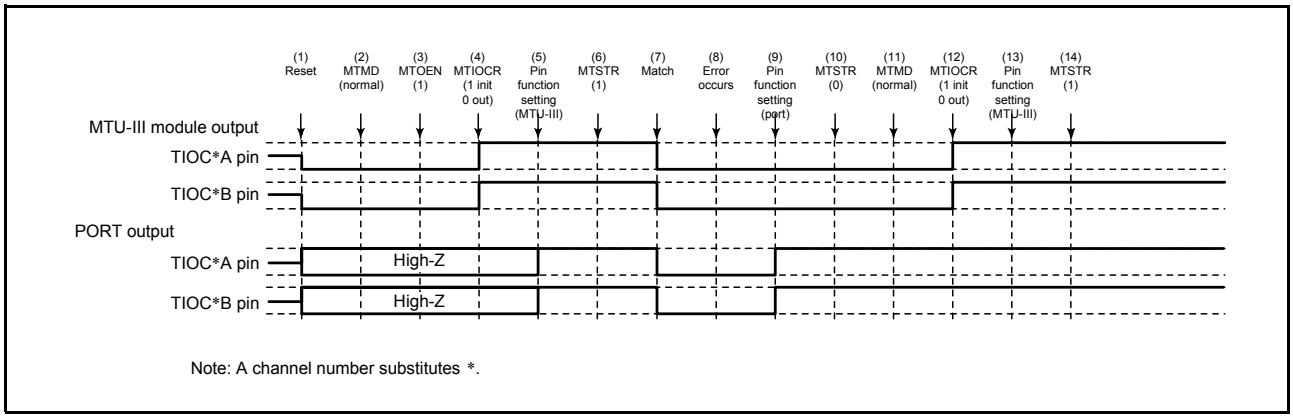


Figure 16.146 Error Occurrence in Normal Mode and Recovery in Normal Mode

(1) After a reset, MTU-III output is low and ports are in the high-impedance state.
(2) After a reset, the MTMD0 setting is for normal mode.
(3) For channels 3 and 4, enable output with MT34OEN before initializing the pins with MTIOCR.
(4) Initialize the pins with MTIOCR. (The example shows initial high output, with low output on compare-match occurrence.)
(5) Set MTU-III output with the pin function setting*.
(6) The count operation is started by MT01234STR.
(7) Output goes low on compare-match occurrence.
(8) An error occurs.
(9) Set port output with the pin function setting* and output the inverse of the active level.
(10) The count operation is stopped by MT01234STR.
(11) Not necessary when restarting in normal mode.
(12) Initialize the pins with MTIOCR.
(13) Set MTU-III output with the pin function setting*.
(14) Operation is restarted by MT01234STR.

Note: * For pin functional settings, see section 13, I/O Ports.

(2) Operation when Error Occurs during Normal Mode Operation, and Operation is Restarted in PWM Mode 1

Figure 16.147 shows an explanatory diagram of the case where an error occurs in normal mode and operation is restarted in PWM mode 1 after re-setting.

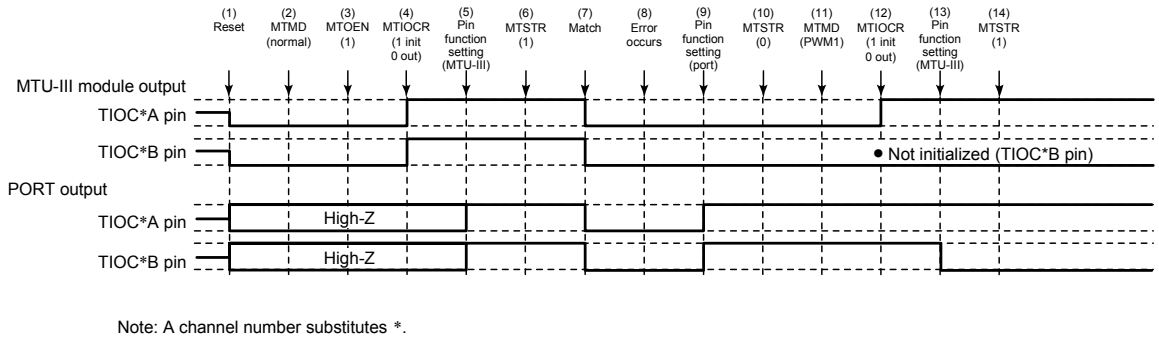


Figure 16.147 Error Occurrence in Normal Mode and Recovery in PWM Mode 1

(1) to (10) are the same as in figure 16.146.
(11) Set PWM mode 1.
(12) Initialize the pins with MTIOCR. (In PWM mode 1, the TIOC*B side is not initialized. If initialization is required, initialize in normal mode, then switch to PWM mode 1.)
(13) Set MTU-III output with the pin function setting*.
(14) Operation is restarted by MT01234STR.

Note: * For pin functional settings, see section 13, I/O Ports.

(3) Operation when Error Occurs during Normal Mode Operation, and Operation is Restarted in PWM Mode 2

Figure 16.148 shows an explanatory diagram of the case where an error occurs in normal mode and operation is restarted in PWM mode 2 after re-setting.

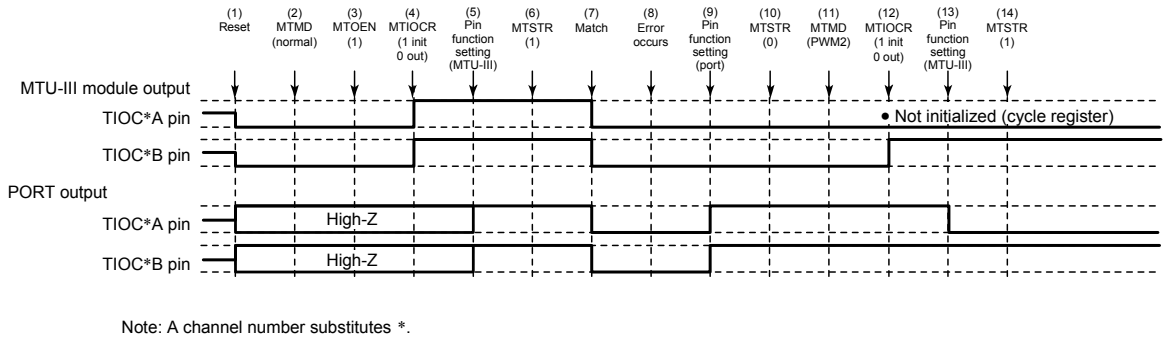


Figure 16.148 Error Occurrence in Normal Mode and Recovery in PWM Mode 2

(1) to (10) are the same as in figure 16.146.
(11) Set PWM mode 2.
(12) Initialize the pins with MTIOCR. (In PWM mode 2, the cycle register pin is not initialized. If initialization is required, carry it out in normal mode, then switch to PWM mode 2.)
(13) Set MTU-III output with the pin function setting*.
(14) Operation is restarted by MT01234STR.

Notes: PWM mode 2 can only be set for channels 0 to 2, and therefore MT34OEN setting is not necessary.
* For pin functional settings, see section 13, I/O Ports.

(4) Operation when Error Occurs during Normal Mode Operation, and Operation is Restarted in Phase Counting Mode

Figure 16.149 shows an explanatory diagram of the case where an error occurs in normal mode and operation is restarted in phase counting mode after re-setting.

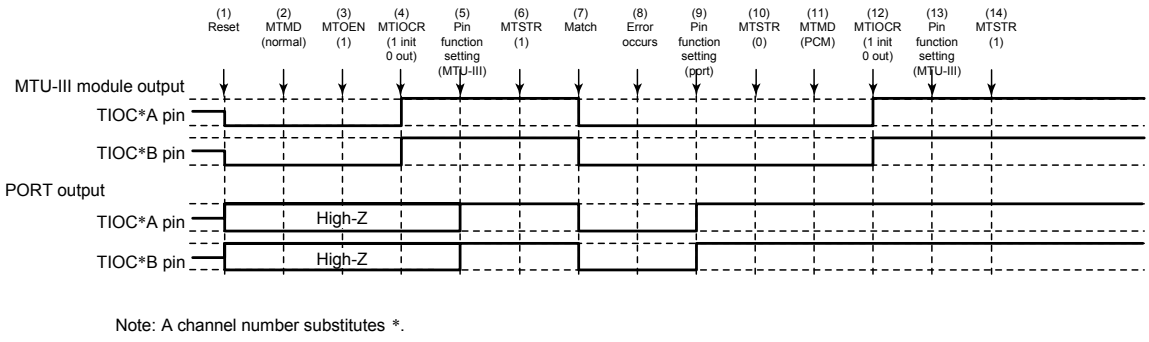


Figure 16.149 Error Occurrence in Normal Mode and Recovery in Phase Counting Mode

(1) to (10) are the same as in figure 16.146.
(11) Set phase counting mode.
(12) Initialize the pins with MTIOCR.
(13) Set MTU-III output with the pin function setting*.
(14) Operation is restarted by MT01234STR.

Notes: Phase counting mode can only be set for channels 1 and 2, and therefore MT34OEN setting is not necessary.
* For pin functional settings, see section 13, I/O Ports.

(5) Operation when Error Occurs during Normal Mode Operation, and Operation is Restarted in Complementary PWM Mode

Figure 16.150 shows an explanatory diagram of the case where an error occurs in normal mode and operation is restarted in complementary PWM mode after re-setting.

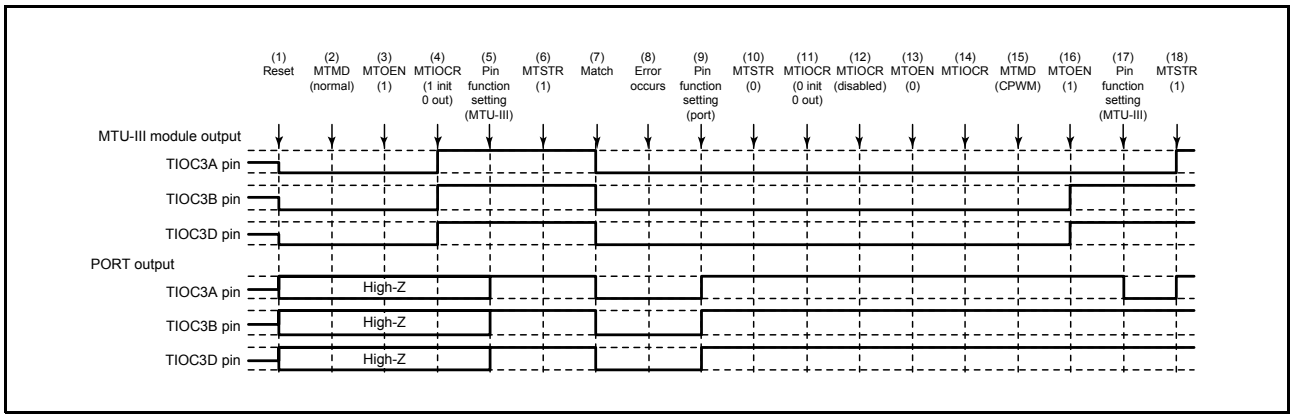


Figure 16.150 Error Occurrence in Normal Mode and Recovery in Complementary PWM Mode

(1) to (10) are the same as in figure 16.146.
(11) Initialize the normal mode waveform generation section with MTIOCR.
(12) Disable operation of the normal mode waveform generation section with MTIOCR.
(13) Disable channel 3 and 4 output with MT34OEN.
(14) Select the complementary PWM output level and cyclic output enabling/disabling with MT34OCR0 and MT34OCR1.
(15) Set complementary PWM.
(16) Enable channel 3 and 4 output with MT34OEN.
(17) Set MTU-III output with the pin function setting*.
(18) Operation is restarted by MT01234STR.

Note: * For pin functional settings, see section 13, I/O Ports.

(6) Operation when Error Occurs during Normal Mode Operation, and Operation is Restarted in Reset-Synchronized PWM Mode

Figure 16.151 shows an explanatory diagram of the case where an error occurs in normal mode and operation is restarted in reset-synchronized PWM mode after re-setting.

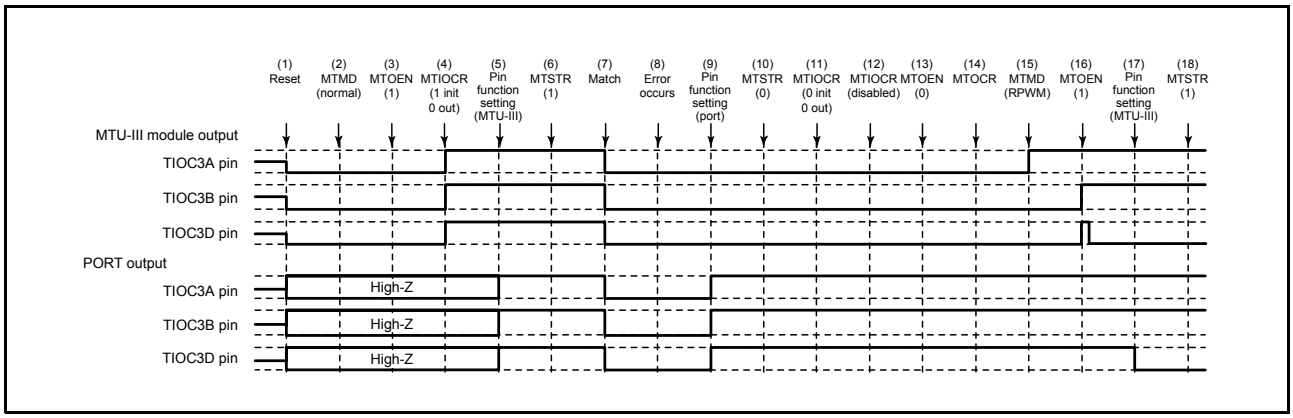


Figure 16.151 Error Occurrence in Normal Mode and Recovery in Reset-Synchronized PWM Mode

(1) to (13) are the same as in figure 16.146.

(14) Select the reset-synchronized PWM output level and cyclic output enabling/disabling with MT34OCR0 and MT34OCR1.

(15) Set reset-synchronized PWM.

(16) Enable channel 3 and 4 output with MT34OEN.

(17) Set MTU-III output with the pin function setting*.

(18) Operation is restarted by MT01234STR.

Note: * For pin functional settings, see section 13, I/O Ports.

(7) Operation when Error Occurs during PWM Mode 1 Operation, and Operation is Restarted in Normal Mode

Figure 16.152 shows an explanatory diagram of the case where an error occurs in PWM mode 1 and operation is restarted in normal mode after re-setting.

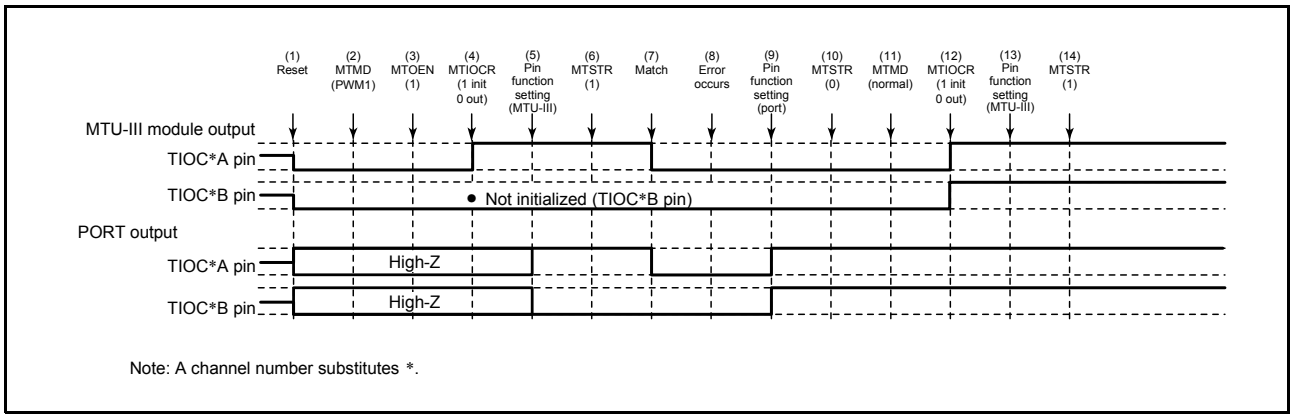


Figure 16.152 Error Occurrence in PWM Mode 1and Recovery in Normal Mode

(1) After a reset, MTU-III output is low and ports are in the high-impedance state.
(2) Set PWM mode 1.
(3) For channels 3 and 4, enable output with MT34OEN before initializing the pins with MTIOCR.
(4) Initialize the pins with MTIOCR. (The example shows initial high output, with low output on compare-match occurrence. In PWM mode 1, the TIOC*B side is not initialized.)
(5) Set MTU-III output with the pin function setting*.
(6) The count operation is started by MT01234STR.
(7) Output goes low on compare-match occurrence.
(8) An error occurs.
(9) Set port output with the pin function setting* and output the inverse of the active level.
(10) The count operation is stopped by MT01234STR.
(11) Set normal mode.
(12) Initialize the pins with MTIOCR.
(13) Set MTU-III output with the pin function setting*.
(14) Operation is restarted by MT01234STR.

Note: * For pin functional settings, see section 13, I/O Ports.

(8) Operation when Error Occurs during PWM Mode 1 Operation, and Operation is Restarted in PWM Mode 1

Figure 16.153 shows an explanatory diagram of the case where an error occurs in PWM mode 1 and operation is restarted in PWM mode 1 after re-setting.

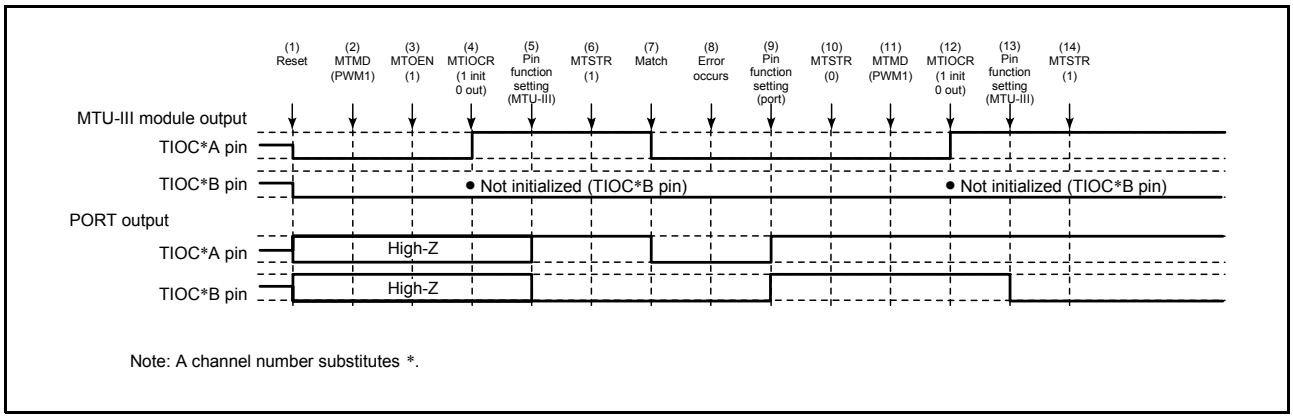


Figure 16.153 Error Occurrence in PWM Mode 1, Recovery in PWM Mode 1

(1) to (10) are the same as in figure 16.152.
(11) Not necessary when restarting in PWM mode 1.
(12) Initialize the pins with MTIOCR. (In PWM mode 1, the TIOC*B side is not initialized.)
(13) Set MTU-III output with the pin function setting*.
(14) Operation is restarted by MT01234STR.

Note: * For pin functional settings, see section 13, I/O Ports.

(9) Operation when Error Occurs during PWM Mode 1 Operation, and Operation is Restarted in PWM Mode 2

Figure 16.154 shows an explanatory diagram of the case where an error occurs in PWM mode 1 and operation is restarted in PWM mode 2 after re-setting.

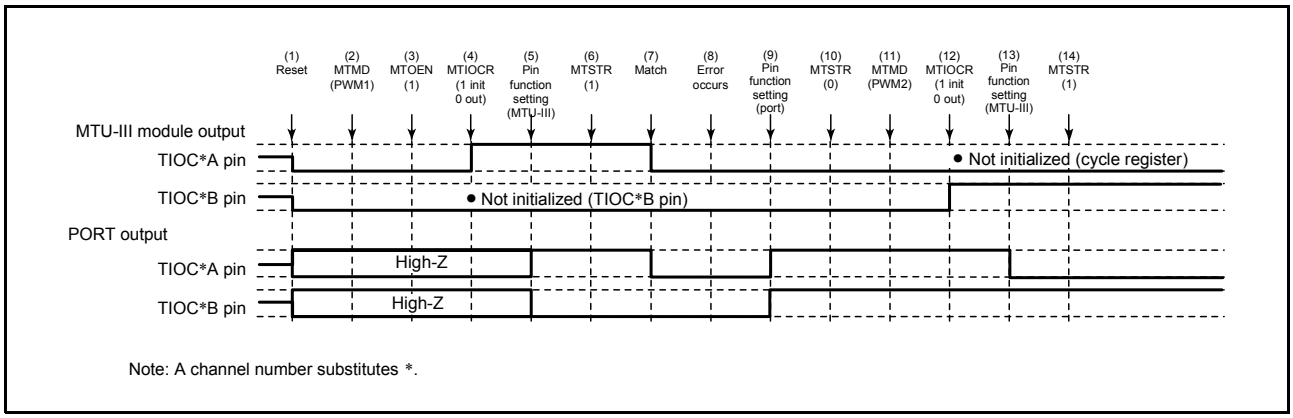


Figure 16.154 Error Occurrence in PWM Mode 1, Recovery in PWM Mode 2

(1) to (10) are the same as in figure 16.152.
(11) Set PWM mode 2.
(12) Initialize the pins with MTIOCR. (In PWM mode 2, the cycle register pin is not initialized.)
(13) Set MTU-III output with the pin function setting*.
(14) Operation is restarted by MT01234STR.

Notes: PWM mode 2 can only be set for channels 0 to 2, and therefore MT34OEN setting is not necessary.
* For pin functional settings, see section 13, I/O Ports.

(10) Operation when Error Occurs during PWM Mode 1 Operation and Operation is Restarted in Phase Counting Mode

Figure 16.155 shows an explanatory diagram of the case where an error occurs in PWM mode 1 and operation is restarted in phase counting mode after re-setting.

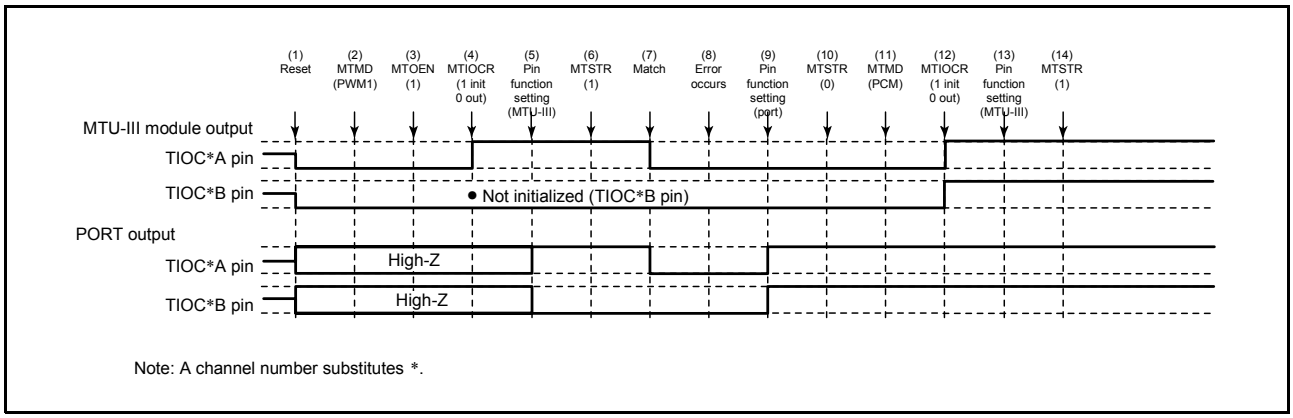


Figure 16.155 Error Occurrence in PWM Mode 1, Recovery in Phase Counting Mode

(1) to (10) are the same as in figure 16.152.
(11) Set phase counting mode.
(12) Initialize the pins with MTIOCR.
(13) Set MTU-III output with the pin function setting*.
(14) Operation is restarted by MT01234STR.

Notes: Phase counting mode can only be set for channels 1 and 2, and therefore MT34OEN setting is not necessary.
* For pin functional settings, see section 13, I/O Ports.

(11) Operation when Error Occurs during PWM Mode 1 Operation, and Operation is Restarted in Complementary PWM Mode

Figure 16.156 shows an explanatory diagram of the case where an error occurs in PWM mode 1 and operation is restarted in complementary PWM mode after re-setting.

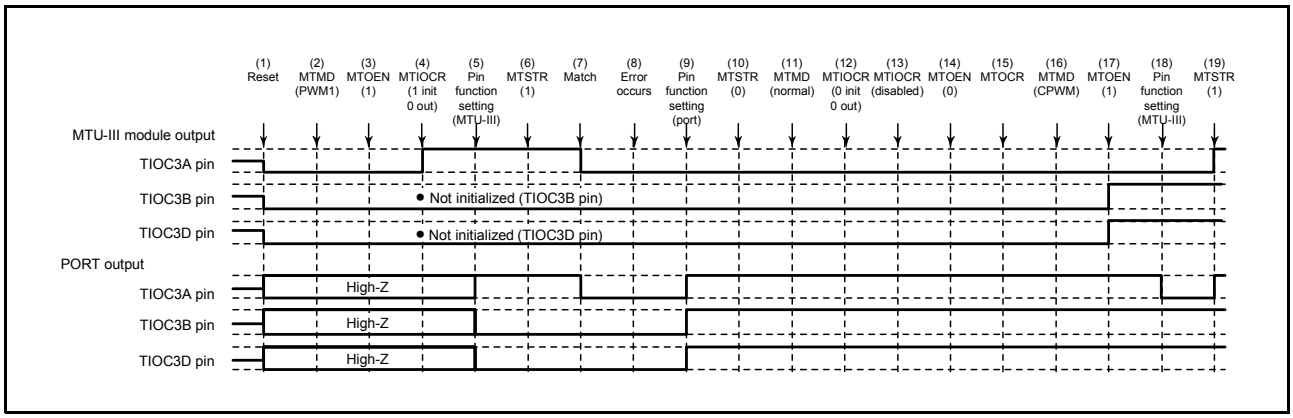


Figure 16.156 Error Occurrence in PWM Mode 1, Recovery in Complementary PWM Mode

(1) to (10) are the same as in figure 16.152.
(11) Set normal mode for initialization of the waveform generation section.
(12) Initialize the PWM mode 1 waveform generation section with MTIOCR.
(13) Disable operation of the PWM mode 1 waveform generation section with MTIOCR.
(14) Disable channel 3 and 4 output with MT34OEN.
(15) Select the complementary PWM output level and cyclic output enabling/disabling with MT34OCR0 and MT34OCR1.
(16) Set complementary PWM.
(17) Enable channel 3 and 4 output with MT34OEN.
(18) Set MTU-III output with the pin function setting*.
(19) Operation is restarted by MT01234STR.

Note: * For pin functional settings, see section 13, I/O Ports.

(12) Operation when Error Occurs during PWM Mode 1 Operation, and Operation is Restarted in Reset-Synchronized PWM Mode

Figure 16.157 shows an explanatory diagram of the case where an error occurs in PWM mode 1 and operation is restarted in reset-synchronized PWM mode after re-setting.

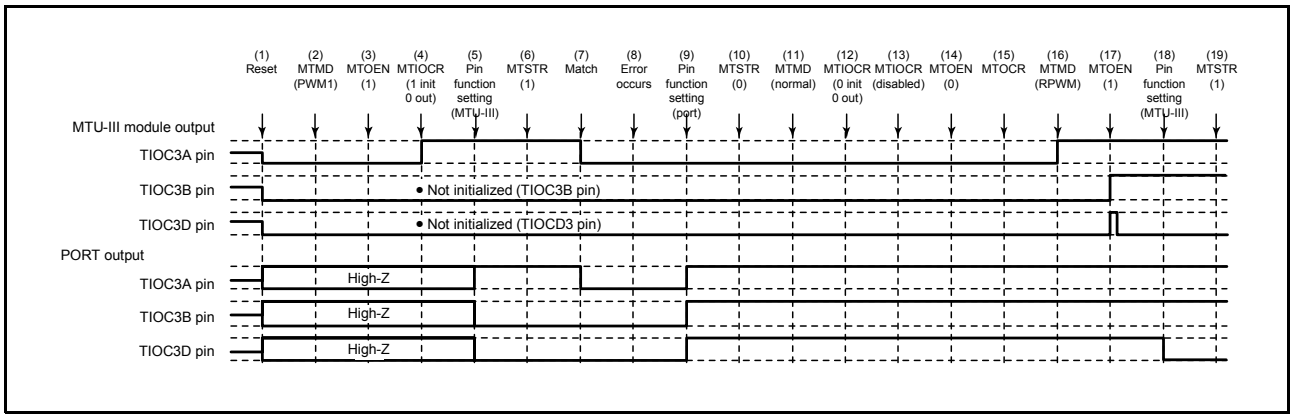


Figure 16.157 Error Occurrence in Complementary PWM Mode 1, Recovery in Reset-Synchronized PWM Mode

(1) to (14) are the same as in figure 16.156.

(15) Select the reset-synchronized PWM output level and cyclic output enabling/disabling with MT34OCR0 and MT34OCR1.

(16) Set reset-synchronized PWM.

(17) Enable channel 3 and 4 output with MT34OEN.

(18) Set MTU-III output with the pin function setting*.

(19) Operation is restarted by MT01234STR.

Note: * For pin functional settings, see section 13, I/O Ports.

(13) Operation when Error Occurs during PWM Mode 2 Operation, and Operation is Restarted in Normal Mode

Figure 16.158 shows an explanatory diagram of the case where an error occurs in PWM mode 2 and operation is restarted in normal mode after re-setting.

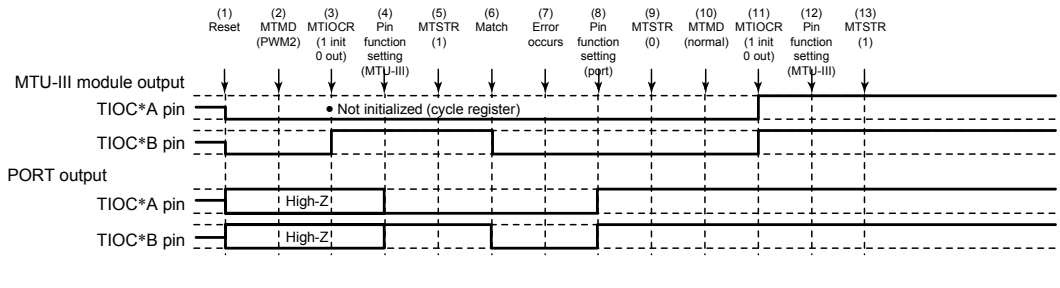


Figure 16.158 Error Occurrence in PWM Mode 2, Recovery in Normal Mode

(1) After a reset, MTU-III output is low and ports are in the high-impedance state.
(2) Set PWM mode 2.
(3) Initialize the pins with MTIOCR. (The example shows initial high output, with low output on compare-match occurrence.
In PWM mode 2, the cycle register pin is not initialized. In the example, TIOC*A is the cycle register.)
(4) Set MTU-III output with the pin function setting*.
(5) The count operation is started by MT01234STR.
(6) Output goes low on compare-match occurrence.
(7) An error occurs.
(8) Set port output with the pin function setting* and output the inverse of the active level.
(9) The count operation is stopped by MT01234STR.
(10) Set normal mode.
(11) Initialize the pins with MTIOCR.
(12) Set MTU-III output with the pin function setting*.
(13) Operation is restarted by MT01234STR.

Note: * For pin functional settings, see section 13, I/O Ports.

(14) Operation when Error Occurs during PWM Mode 2 Operation, and Operation is Restarted in PWM Mode 1

Figure 16.159 shows an explanatory diagram of the case where an error occurs in PWM mode 2 and operation is restarted in PWM mode 1 after re-setting.

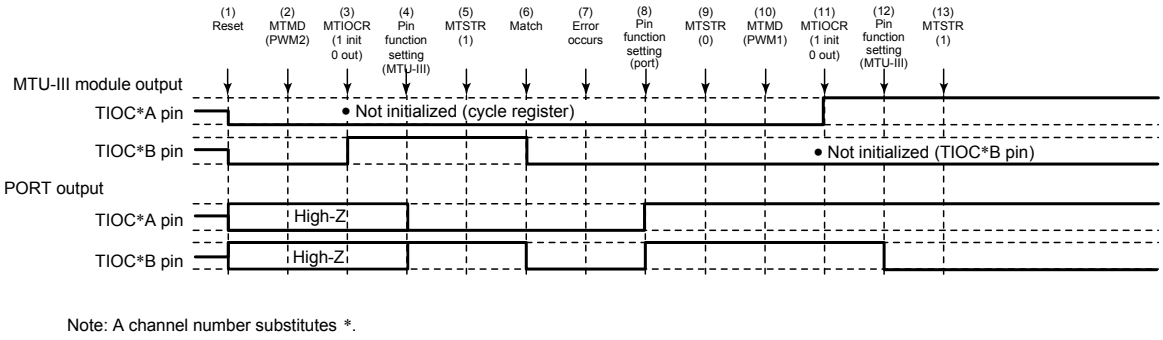


Figure 16.159 Error Occurrence in PWM Mode 2, Recovery in PWM Mode 1

(1) to (9) are the same as in figure 16.158.
(10) Set PWM mode 1.
(11) Initialize the pins with MTIOCR. (In PWM mode 1, the TIOC*B side is not initialized.)
(12) Set MTU-III output with the pin function setting*.
(13) Operation is restarted by MT01234STR.

Note: * For pin functional settings, see section 13, I/O Ports.

(15) Operation when Error Occurs during PWM Mode 2 Operation, and Operation is Restarted in PWM Mode 2

Figure 16.160 shows an explanatory diagram of the case where an error occurs in PWM mode 2 and operation is restarted in PWM mode 2 after re-setting.

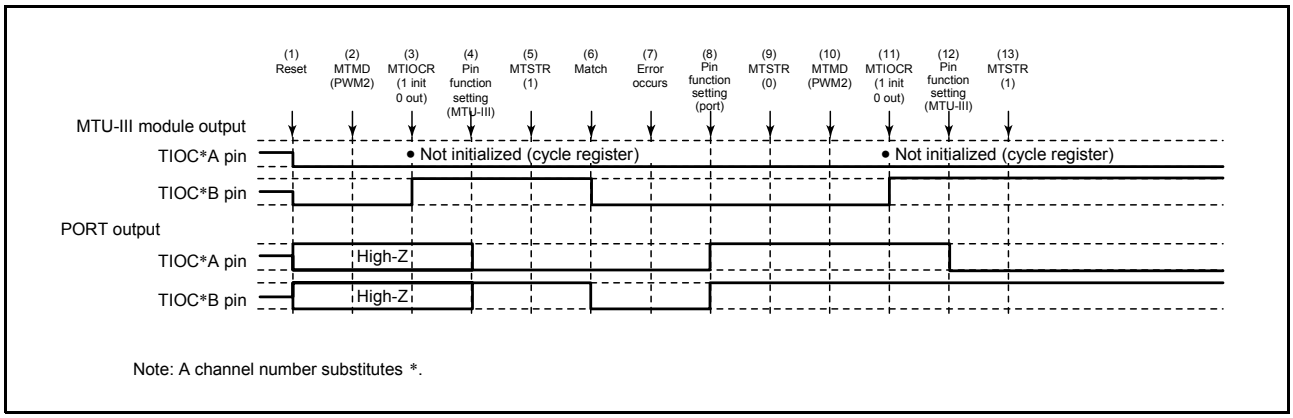


Figure 16.160 Error Occurrence in PWM Mode 2, Recovery in PWM Mode 2

(1) to (9) are the same as in figure 16.158.
(10) Not necessary when restarting in PWM mode 2.
(11) Initialize the pins with MTIOCR. (In PWM mode 2, the cycle register pin is not initialized.)
(12) Set MTU-III output with the pin function setting*.
(13) Operation is restarted by MT01234STR.

Note: * For pin functional settings, see section 13, I/O Ports.

(16) Operation when Error Occurs during PWM Mode 2 Operation and Operation is Restarted in Phase Counting Mode

Figure 16.161 shows an explanatory diagram of the case where an error occurs in PWM mode 2 and operation is restarted in phase counting mode after re-setting.

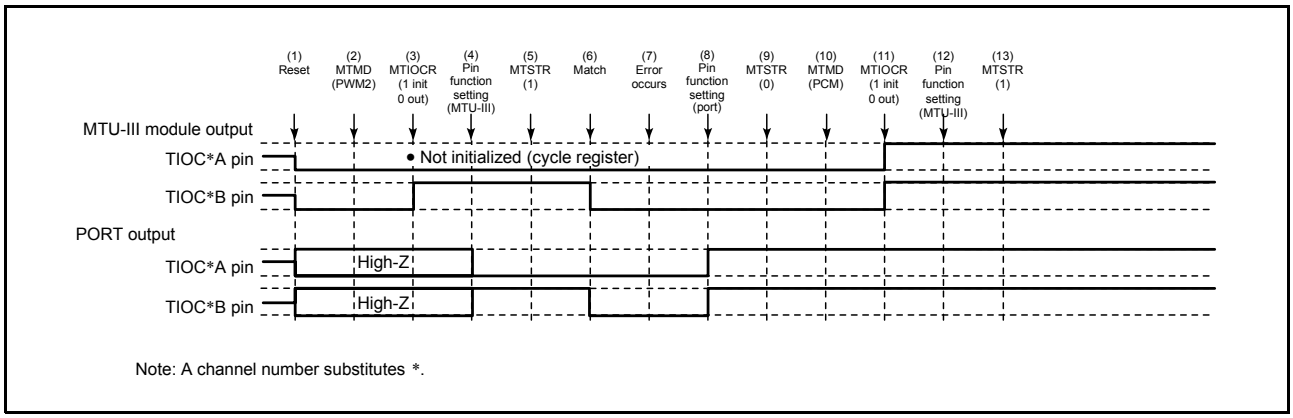


Figure 16.161 Error Occurrence in PWM Mode 2, Recovery in Phase Counting Mode

(1) to (9) are the same as in figure 16.158.
(10) Set phase counting mode.
(11) Initialize the pins with MTIOCR.
(12) Set MTU-III output with the pin function setting*.
(13) Operation is restarted by MT01234STR.

Note: * For pin functional settings, see section 13, I/O Ports.

(17) Operation when Error Occurs during Phase Counting Mode Operation, and Operation is Restarted in Normal Mode

Figure 16.162 shows an explanatory diagram of the case where an error occurs in phase counting mode and operation is restarted in normal mode after re-setting.

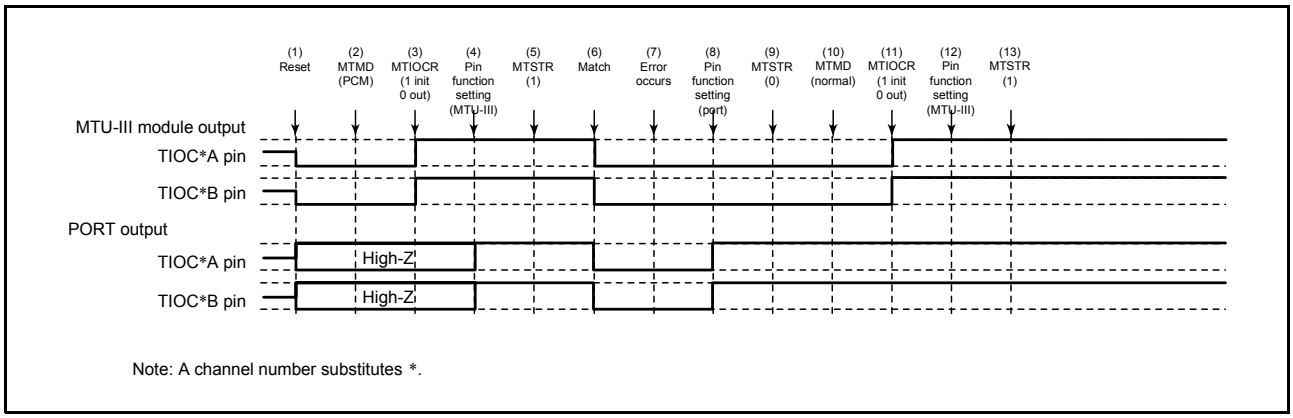


Figure 16.162 Error Occurrence in Phase Counting Mode, Recovery in Normal Mode

(1) After a reset, MTU-III output is low and ports are in the high-impedance state.
(2) Set phase counting mode.
(3) Initialize the pins with MTIOCR. (The example shows initial high output, with low output on compare-match occurrence.)
(4) Set MTU-III output with the pin function setting*.
(5) The count operation is started by MT01234STR.
(6) Output goes low on compare-match occurrence.
(7) An error occurs.
(8) Set port output with the pin function setting* and output the inverse of the active level.
(9) The count operation is stopped by MT01234STR.
(10) Set in normal mode.
(11) Initialize the pins with MTIOCR.
(12) Set MTU-III output with the pin function setting*.
(13) Operation is restarted by MT01234STR.

Note: * For pin functional settings, see section 13, I/O Ports.

(18) Operation when Error Occurs during Phase Counting Mode Operation, and Operation is Restarted in PWM Mode 1

Figure 16.163 shows an explanatory diagram of the case where an error occurs in phase counting mode and operation is restarted in PWM mode 1 after re-setting.

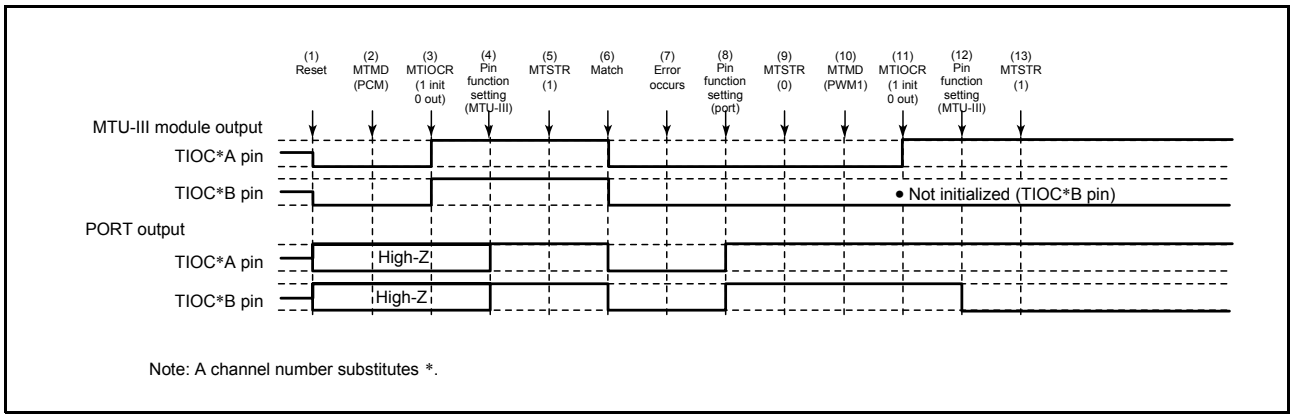


Figure 16.163 Error Occurrence in Phase Counting Mode, Recovery in PWM Mode 1

(1) to (9) are the same as in figure 16.162.
(10) Set PWM mode 1.
(11) Initialize the pins with MTIOCR. (In PWM mode 1, the TIOC*B side is not initialized.)
(12) Set MTU-III output with the pin function setting*.
(13) Operation is restarted by MT01234STR.

Note: * For pin functional settings, see section 13, I/O Ports.

(19) Operation when Error Occurs during Phase Counting Mode Operation, and Operation is Restarted in PWM Mode 2

Figure 16.164 shows an explanatory diagram of the case where an error occurs in phase counting mode and operation is restarted in PWM 2 mode after re-setting.

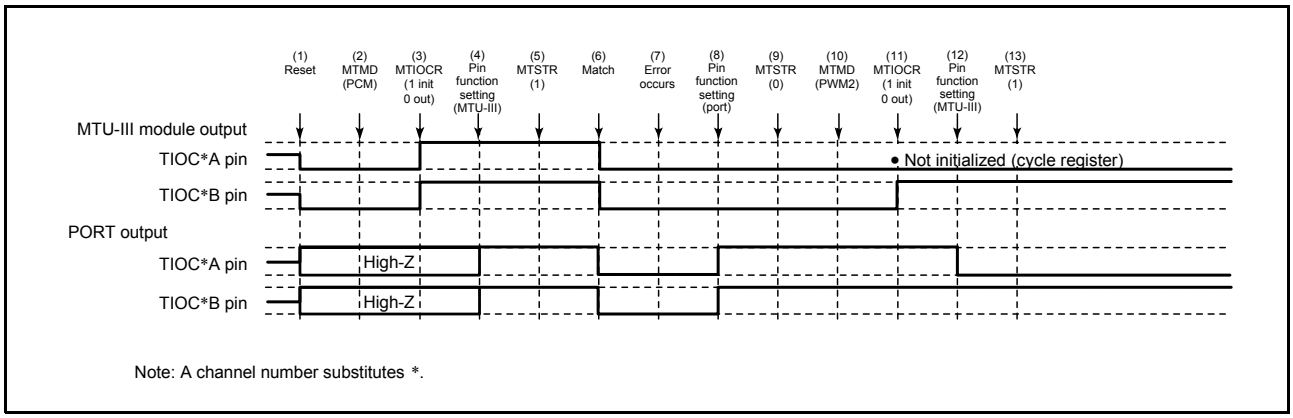


Figure 16.164 Error Occurrence in Phase Counting Mode, Recovery in PWM Mode 2

(1) to (9) are the same as in figure 16.162.
(10) Set PWM mode 2.
(11) Initialize the pins with MTIOCR. (In PWM mode 2, the cycle register pin is not initialized.)
(12) Set MTU-III output with the pin function setting*.
(13) Operation is restarted by MT01234STR.

Note: * For pin functional settings, see section 13, I/O Ports.

(20) Operation when Error Occurs during Phase Counting Mode Operation, and Operation is Restarted in Phase Counting Mode

Figure 16.165 shows an explanatory diagram of the case where an error occurs in phase counting mode and operation is restarted in phase counting mode after re-setting.

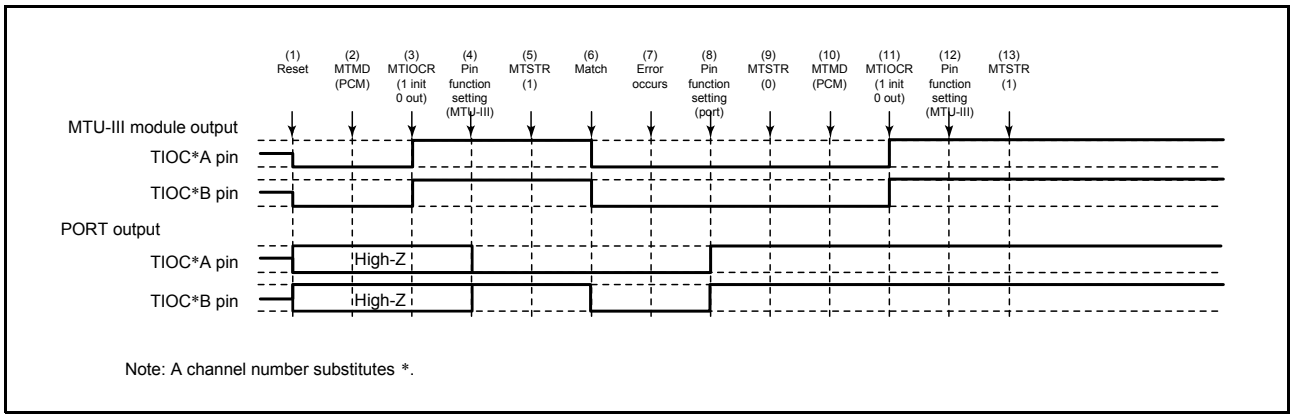


Figure 16.165 Error Occurrence in Phase Counting Mode, Recovery in Phase Counting Mode

(1) to (9) are the same as in figure 16.162.
(10) Not necessary when restarting in phase counting mode.
(11) Initialize the pins with MTIOCR.
(12) Set MTU-III output with the pin function setting*.
(13) Operation is restarted by MT01234STR.

Note: * For pin functional settings, see section 13, I/O Ports.

(21) Operation when Error Occurs during Complementary PWM Mode Operation, and Operation is Restarted in Normal Mode

Figure 16.166 shows an explanatory diagram of the case where an error occurs in complementary PWM mode and operation is restarted in normal mode after re-setting.

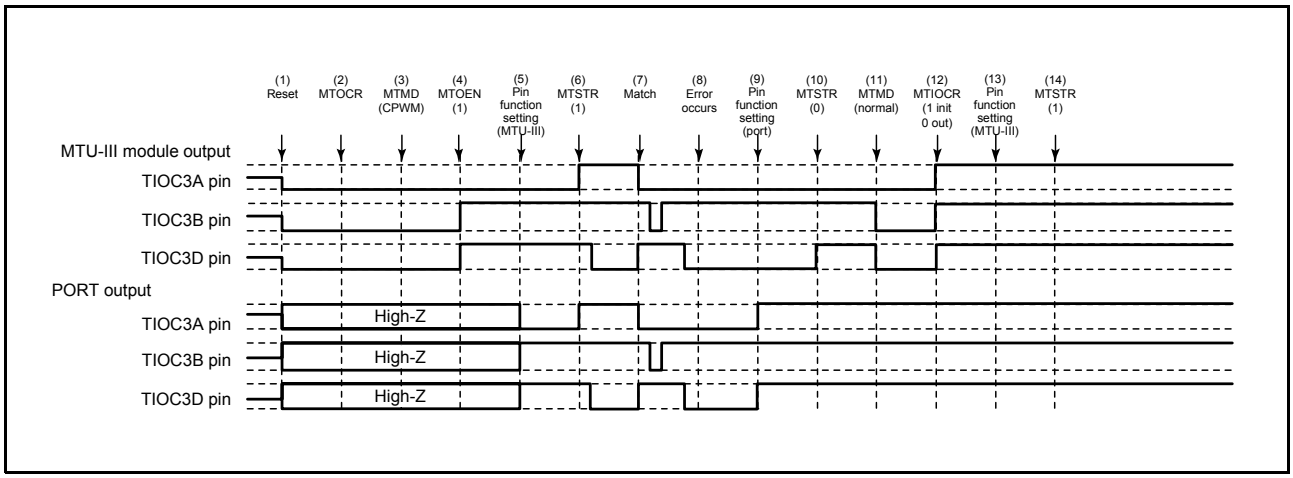


Figure 16.166 Error Occurrence in Complementary PWM Mode, Recovery in Normal Mode

(1) After a reset, MTU-III output is low and ports are in the high-impedance state.
(2) Select the complementary PWM output level and cyclic output enabling/disabling with MT34OCR0 and MT34OCR1.
(3) Set complementary PWM.
(4) Enable channel 3 and 4 output with MT34OEN.
(5) Set MTU-III output with the pin function setting*.
(6) The count operation is started by MT01234STR.
(7) The complementary PWM waveform is output on compare-match occurrence.
(8) An error occurs.
(9) Set port output with the pin function setting* and output the inverse of the active level.
(10) The count operation is stopped by MT01234STR. (MTU-III output becomes the complementary PWM output initial value.)
(11) Set normal mode. (MTU-III output goes low.)
(12) Initialize the pins with MTIOCR.
(13) Set MTU-III output with the pin function setting*.
(14) Operation is restarted by MT01234STR.

Note: * For pin functional settings, see section 13, I/O Ports.

(22) Operation when Error Occurs during Complementary PWM Mode Operation, and Operation is Restarted in PWM Mode 1

Figure 16.167 shows an explanatory diagram of the case where an error occurs in complementary PWM and operation is restarted in PWM mode 1 after re-setting.

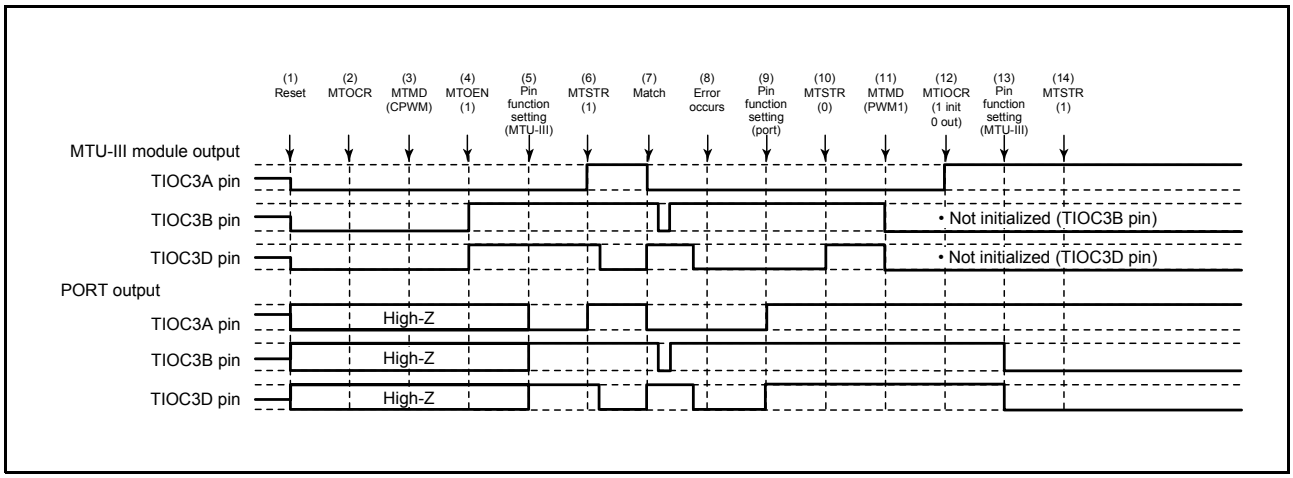


Figure 16.167 Error Occurrence in Complementary PWM Mode, Recovery in PWM Mode 1

(1) to (10) are the same as in figure 16.166.
(11) Set PWM mode 1. (MTU-III output goes low.)
(12) Initialize the pins with MTIOCR. (In PWM mode 1, the TIOC*B side is not initialized.)
(13) Set MTU-III output with the pin function setting*.
(14) Operation is restarted by MT01234STR.

Note: * For pin functional settings, see section 13, I/O Ports.

(23) Operation when Error Occurs during Complementary PWM Mode Operation, and Operation is Restarted in Complementary PWM Mode

Figure 16.168 shows an explanatory diagram of the case where an error occurs in complementary PWM mode and operation is restarted in complementary PWM mode after re-setting (when operation is restarted using the cycle and duty settings at the time the counter was stopped).

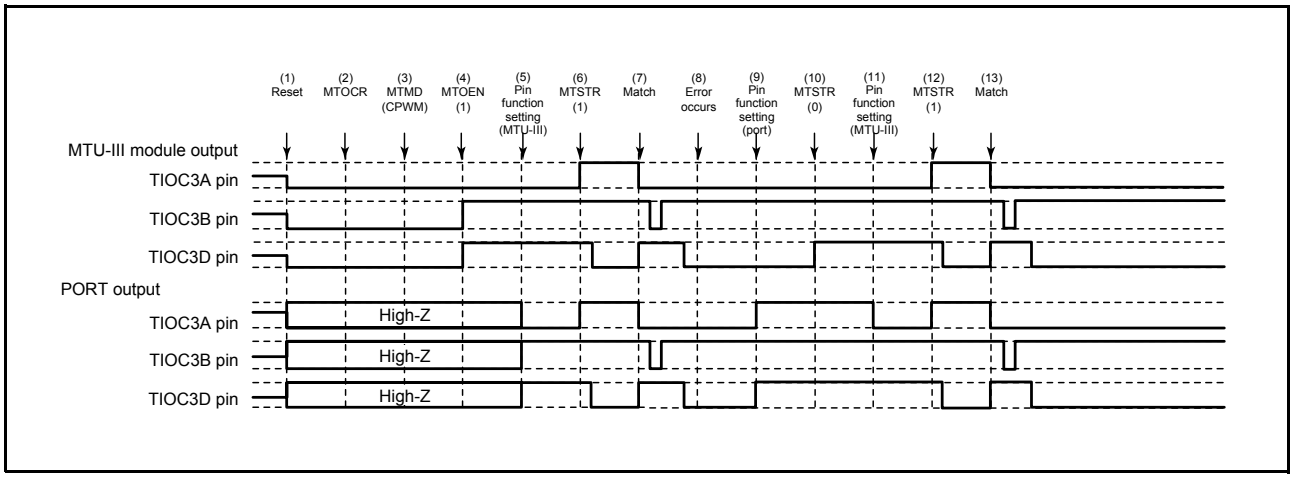


Figure 16.168 Error Occurrence in Complementary PWM Mode, Recovery in Complementary PWM Mode

(1) to (10) are the same as in figure 16.166.

(11) Set MTU-III output with the pin function setting*.

(12) Operation is restarted by MT01234STR.

(13) The complementary PWM waveform is output on compare-match occurrence.

Note: * For pin functional settings, see section 13, I/O Ports.

(24) Operation when Error Occurs during Complementary PWM Mode Operation, and Operation is Newly Restarted in Complementary PWM Mode

Figure 16.169 shows an explanatory diagram of the case where an error occurs in complementary PWM mode and operation is restarted in complementary PWM mode after re-setting (when operation is restarted using completely new cycle and duty settings).

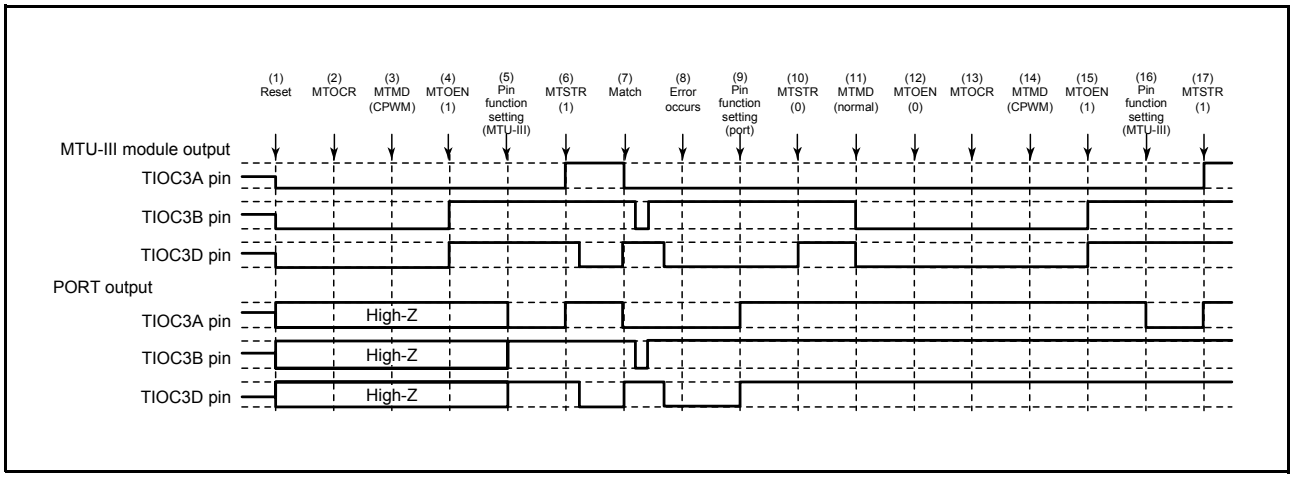


Figure 16.169 Error Occurrence in Complementary PWM Mode, Recovery in Complementary PWM Mode

(1) to (10) are the same as in figure 16.166.
(11) Set normal mode and make new settings. (MTU-III output goes low.)
(12) Disable channel 3 and 4 output with MT34OEN.
(13) Select the complementary PWM mode output level and cyclic output enabling/disabling with MT34OCR0 and MT34OCR1.
(14) Set complementary PWM.
(15) Enable channel 3 and 4 output with MT34OEN.
(16) Set MTU-III output with the pin function setting*.
(17) Operation is restarted by MT01234STR.

Note: * For pin functional settings, see section 13, I/O Ports.

(25) Operation when Error Occurs during Complementary PWM Mode Operation, and Operation is Restarted in Reset-Synchronized PWM Mode

Figure 16.170 shows an explanatory diagram of the case where an error occurs in complementary PWM mode and operation is restarted in reset-synchronized PWM mode after re-setting.

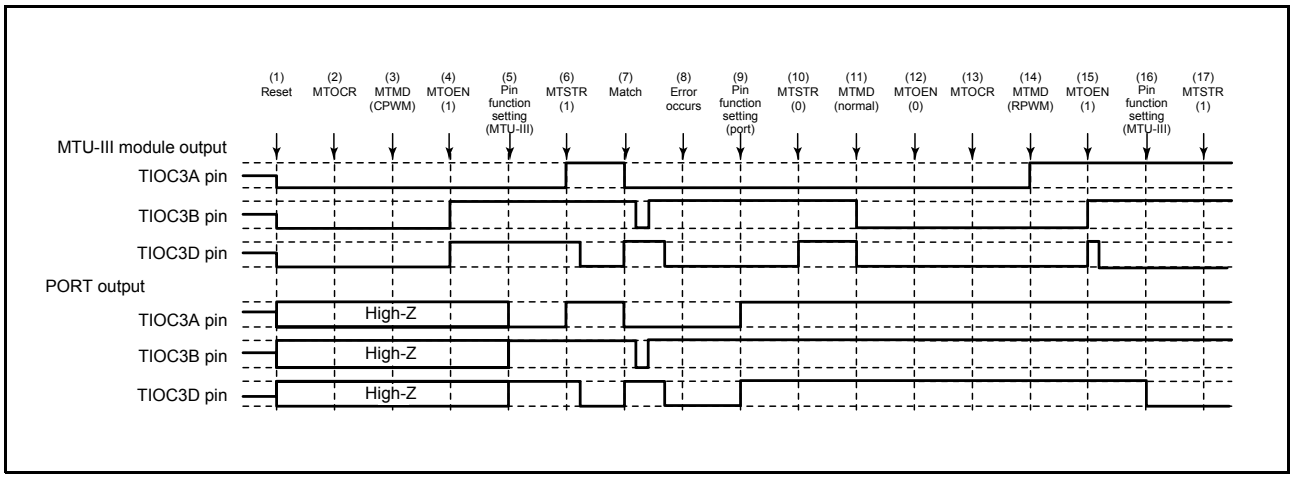


Figure 16.170 Error Occurrence in Complementary PWM Mode, Recovery in Reset-Synchronized PWM Mode

(1) to (10) are the same as in figure 16.166.
(11) Set normal mode. (MTU-III output goes low.)
(12) Disable channel 3 and 4 output with MT34OEN.
(13) Select the reset-synchronized PWM mode output level and cyclic output enabling/disabling with MT34OCR0 and MT34OCR1.
(14) Set reset-synchronized PWM.
(15) Enable channel 3 and 4 output with MT34OEN.
(16) Set MTU-III output with the pin function setting*.
(17) Operation is restarted by MT01234STR.

Note: * For pin functional settings, see section 13, I/O Ports.

(26) Operation when Error Occurs during Reset-Synchronized PWM Mode Operation, and Operation is Restarted in Normal Mode

Figure 16.171 shows an explanatory diagram of the case where an error occurs in reset-synchronized PWM mode and operation is restarted in normal mode after re-setting.

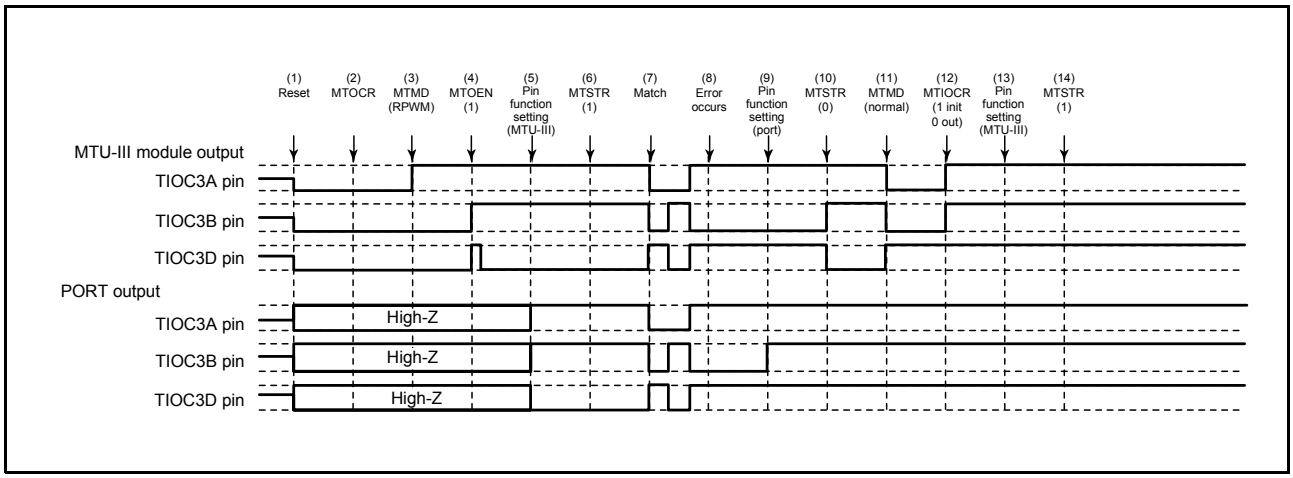


Figure 16.171 Error Occurrence in Reset-Synchronized PWM Mode, Recovery in Normal Mode

(1) After a reset, MTU-III output is low and ports are in the high-impedance state.
(2) Select the reset-synchronized PWM output level and cyclic output enabling/disabling with MT34OCR0 and MT34OCR1.
(3) Set reset-synchronized PWM.
(4) Enable channel 3 and 4 output with MT34OEN.
(5) Set MTU-III output with the pin function setting*.
(6) The count operation is started by MT01234STR.
(7) The reset-synchronized PWM waveform is output on compare-match occurrence.
(8) An error occurs.
(9) Set port output with the pin function setting* and output the inverse of the active level.
(10) The count operation is stopped by MT01234STR. (MTU-III output becomes the reset-synchronized PWM output initial value.)
(11) Set normal mode. (MTU-III positive phase output is low, and negative phase output is high.)
(12) Initialize the pins with MTIOCR.
(13) Set MTU-III output with the pin function setting*.
(14) Operation is restarted by MT01234STR.

Note: * For pin functional settings, see section 13, I/O Ports.

(27) Operation when Error Occurs during Reset-Synchronized PWM Mode Operation, and Operation is Restarted in PWM Mode 1

Figure 16.172 shows an explanatory diagram of the case where an error occurs in reset-synchronized PWM mode and operation is restarted in PWM mode 1 after re-setting.

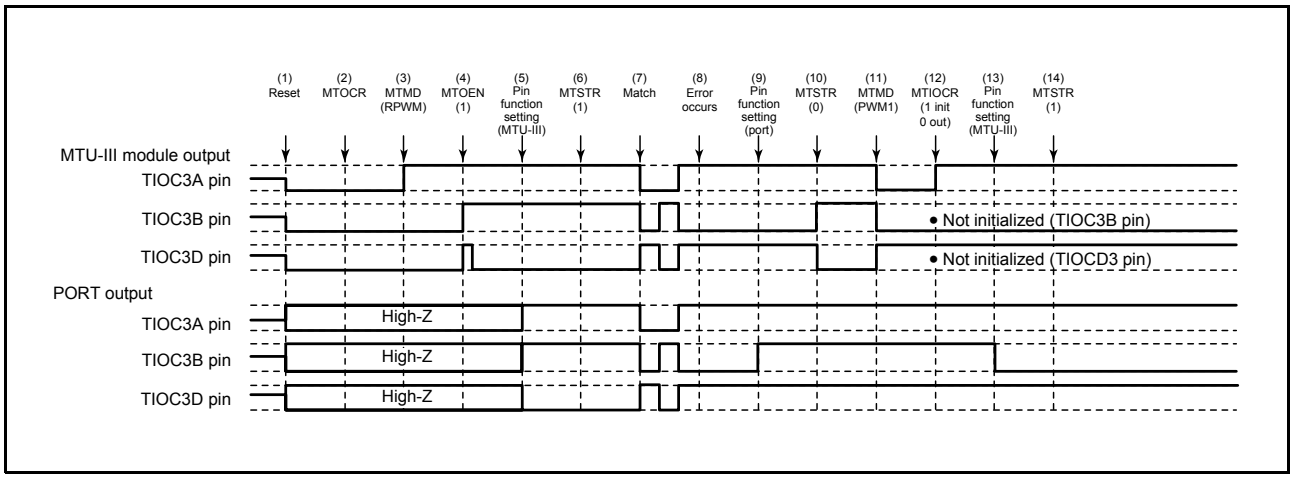


Figure 16.172 Error Occurrence in Reset-Synchronized PWM Mode, Recovery in PWM Mode 1

(1) to (10) are the same as in figure 16.171.
(11) Set PWM mode 1. (MTU-III positive phase output is low, and negative phase output is high.)
(12) Initialize the pins with MTIOCR. (In PWM mode 1, the TIOC*B side is not initialized.)
(13) Set MTU-III output with the pin function setting*.
(14) Operation is restarted by MT01234STR.

Note: * For pin functional settings, see section 13, I/O Ports.

(28) Operation when Error Occurs during Reset-Synchronized PWM Mode Operation, and Operation is Restarted in Complementary PWM Mode

Figure 16.173 shows an explanatory diagram of the case where an error occurs in reset-synchronized PWM mode and operation is restarted in complementary PWM mode after re-setting.

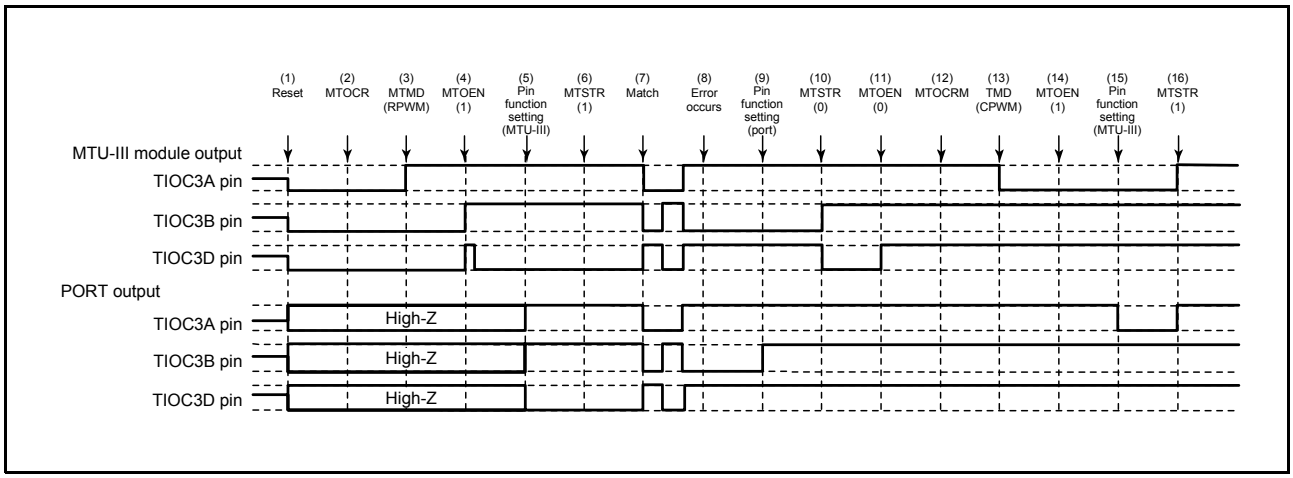


Figure 16.173 Error Occurrence in Reset-Synchronized PWM Mode, Recovery in Complementary PWM Mode

(1) to (10) are the same as in figure 16.171.
(11) Disable channel 3 and 4 output with MT34OEN.
(12) Select the complementary PWM output level and cyclic output enabling/disabling with MT34OCR0 and MT34OCR1.
(13) Set complementary PWM. (The MTU-III cyclic output pin goes low.)
(14) Enable channel 3 and 4 output with MT34OEN.
(15) Set MTU-III output with the pin function setting*.
(16) Operation is restarted by MT01234STR.

Note: * For pin functional settings, see section 13, I/O Ports.

(29) Operation when Error Occurs during Reset-Synchronized PWM Mode Operation, and Operation is Restarted in Reset-Synchronized PWM Mode

Figure 16.174 shows an explanatory diagram of the case where an error occurs in reset-synchronized PWM mode and operation is restarted in reset-synchronized PWM mode after re-setting.

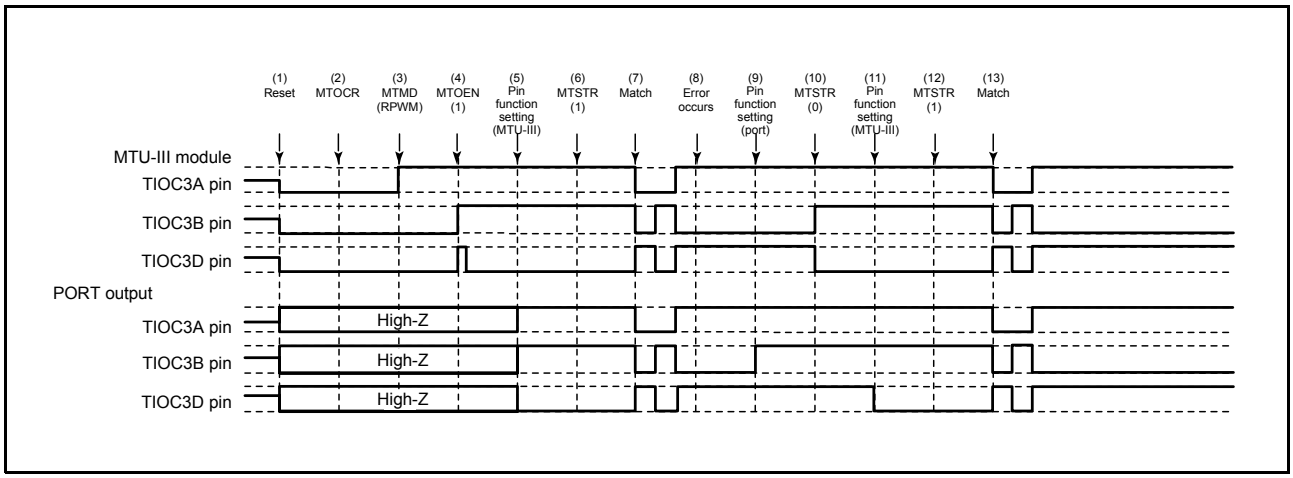


Figure 16.174 Error Occurrence in Reset-Synchronized PWM Mode, Recovery in Reset-Synchronized PWM Mode

(1) to (10) are the same as in figure 16.171.

(11) Set MTU-III output with the pin function setting*.

(12) Operation is restarted by MT01234STR.

(13) The reset-synchronized PWM waveform is output on compare-match occurrence.

Note: * For pin functional settings, see section 13, I/O Ports.

17. Serial Communication Interface (SCI)

The serial communication interface (SCI) can handle both asynchronous and clock synchronous serial communication. Asynchronous serial data communication can be carried out with standard asynchronous communication LSI such as a Universal Asynchronous Receiver/Transmitter (UART) or Asynchronous Communication Interface Adapter (ACIA). In clock synchronous mode, there are normal mode in which serial communication is performed synchronized with a clock, and extended mode in which serial communication is performed varying the polarity and phase of a transmit/receive clock.
This LSI has four channels.

17.1 Introduction

Table 17.1 lists the Serial Communication Interface (SCI) Specifications.

Note: The SCK3 input/output pin is not available in the SH72A0 group.
Accordingly, input and output of the clock signal through the SCK3 pin cannot be performed in the SH72A0 group. Channel 3 is thus dedicated to asynchronous communication.

Table 17.1 Serial Communication Interface (SCI) Specifications

Item	Description
Serial data communication mode	• Asynchronous mode, clock synchronous mode
Communication system	• Full-duplex communication The transmitter and receiver are mutually independent, enabling transmission and reception to be executed simultaneously. Double-buffering is used in both the transmitter and the receiver, enabling continuous transmission and continuous reception.
Bit rate	• On-chip baud rate generator allows selection of the desired bit rate • The external clock can be selected as a transmit/receive clock source
Data format	• Data LSB first/MSB first can be selected (except for 7-bit data in asynchronous mode). [Asynchronous Mode] • Data length: 7 or 8 bits • Stop bit length: 1 or 2 bits • Parity: Even, odd, or none [Clock Synchronous Mode] • Data length: 8 bits • Clock phase and polarity can be selected.
Receive error detection	[Asynchronous Mode] • Parity, overrun, and framing errors [Clock Synchronous Mode] • Overrun error
Interrupt requests	• 4 interrupt requests SCIi transmit end interrupt, SCIi transmit buffer empty interrupt, SCIi receive buffer full interrupt, SCIi receive error interrupt (overrun error, framing error, parity error) • Interrupt sources of transmit data empty and receive data full allow DMAC to be activated.
Others	• Module stop mode can be set [Asynchronous Mode] • Break detection: Break can be detected by reading the RxDi pin level directly in case of a framing error

Figure 17.1 shows the Serial Communication Interface (SCI) Block Diagram. This LSI has four channels, but only one channel of them is shown in the block diagram of Figure 17.1.

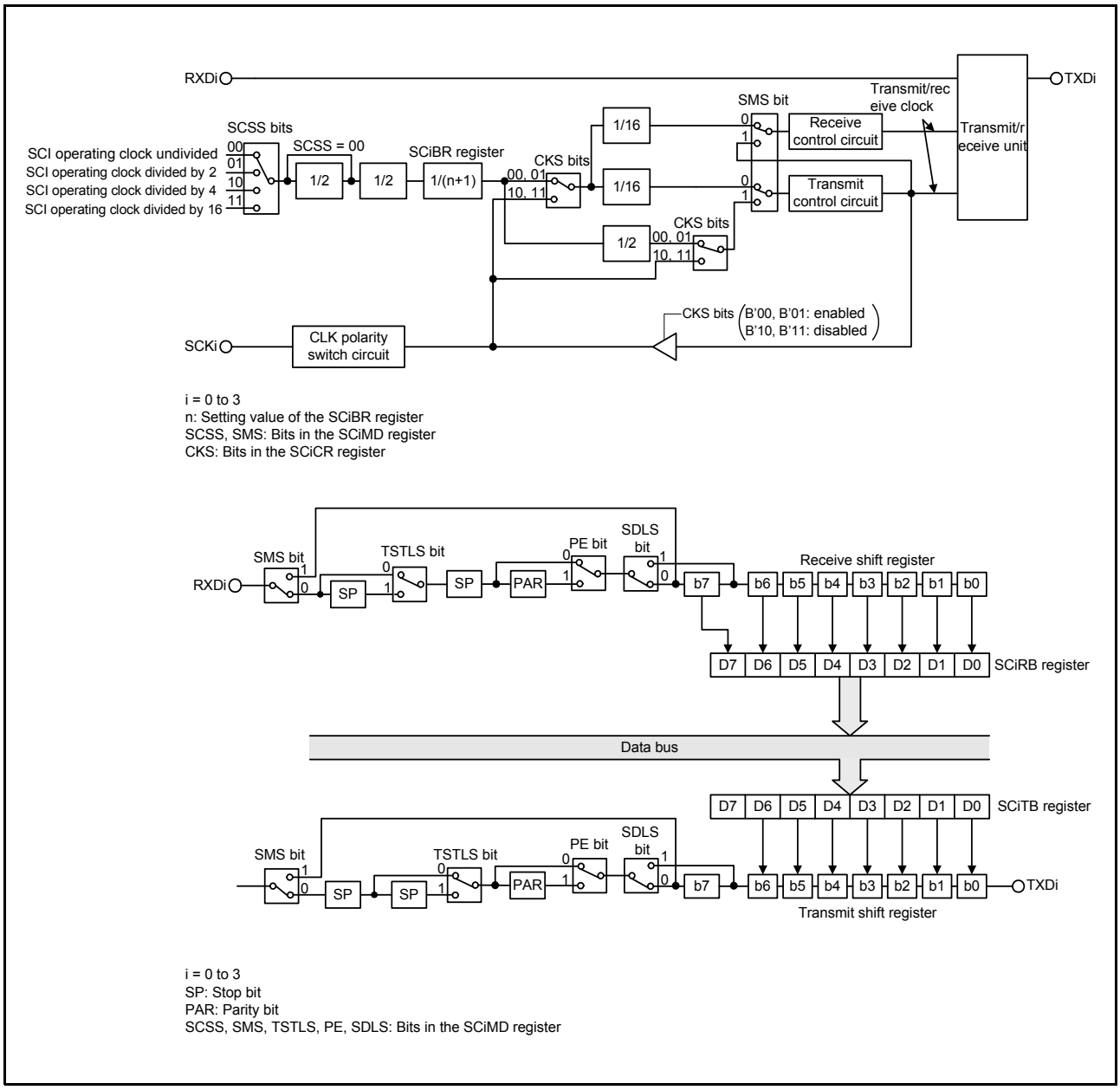


Figure 17.1 Serial Communication Interface (SCI) Block Diagram

Table 17.2 lists the SCI I/O Pins.

Table 17.2 SCI I/O Pins

Pin Name	I/O	Description
SCK0 to SCK3*	I/O	SCI clock input/output
RXD0 to RXD3	Input	SCI receive data input
TXD0 to TXD3	Output	SCI transmit data output

Note: * The SCK3 pin is not available in the SH72A0 group.

17.2 Registers

Table 17.3 lists the SCI Registers.

Table 17.3 SCI Registers

Channel	Register Name	Symbol	After Reset	Address	Access Size
Channel 0	SCI0 transmit buffer register	SC0TB	Undefined	H'FFFF 6003	8
	SCI0 control register	SC0CR	H'00	H'FFFF 6002	8
	SCI0 bit rate register	SC0BR	H'FF	H'FFFF 6001	8
	SCI0 mode register	SC0MD	H'00	H'FFFF 6000	8
	SCI0 receive buffer register	SC0RB	Undefined	H'FFFF 6005	8
	SCI0 status register	SC0SR	H'84	H'FFFF 6004	8
	SCI0 extension mode register	SC0EMD	H'00	H'FFFF 6008	8
Channel 1	SCI1 transmit buffer register	SC1TB	Undefined	H'FFFF 6103	8
	SCI1 control register	SC1CR	H'00	H'FFFF 6102	8
	SCI1 bit rate register	SC1BR	H'FF	H'FFFF 6101	8
	SCI1 mode register	SC1MD	H'00	H'FFFF 6100	8
	SCI1 receive buffer register	SC1RB	Undefined	H'FFFF 6105	8
	SCI1 status register	SC1SR	H'84	H'FFFF 6104	8
	SCI1 extension mode register	SC1EMD	H'00	H'FFFF 6108	8
Channel 2	SCI2 transmit buffer register	SC2TB	Undefined	H'FFFF 6203	8
	SCI2 control register	SC2CR	H'00	H'FFFF 6202	8
	SCI2 bit rate register	SC2BR	H'FF	H'FFFF 6201	8
	SCI2 mode register	SC2MD	H'00	H'FFFF 6200	8
	SCI2 receive buffer register	SC2RB	Undefined	H'FFFF 6205	8
	SCI2 status register	SC2SR	H'84	H'FFFF 6204	8
	SCI2 extension mode register	SC2EMD	H'00	H'FFFF 6208	8
Channel 3	SCI3 transmit buffer register	SC3TB	Undefined	H'FFFF 6303	8
	SCI3 control register	SC3CR	H'00	H'FFFF 6302	8
	SCI3 bit rate register	SC3BR	H'FF	H'FFFF 6301	8
	SCI3 mode register	SC3MD	H'00	H'FFFF 6300	8
	SCI3 receive buffer register	SC3RB	Undefined	H'FFFF 6305	8
	SCI3 status register	SC3SR	H'84	H'FFFF 6304	8
	SCI3 extension mode register	SC3EMD	H'00	H'FFFF 6308	8

17.2.1 SCIi Transmit Buffer Register (SCiTB) (i = 0 to 3)

Address SC0TB: H'FFFF 6003, SC1TB: H'FFFF 6103, SC2TB: H'FFFF 6203, SC3TB: H'FFFF 6303

	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	x	x	x	x	x	x	x	x

Bit	Description	R/W
b7 to b0	Transmit buffer data This register stores the transmit data.	R/W

The transmit data set into the SCiTB register is transferred to the transmit shift register (see Figure 17.1) to start transmission to the TXDi pin, when the transmit shift register is detected to be empty. The SCiTB register and transmit shift register have a double buffer structure to enable continuous transmission. When one frame of the data has been transmitted and the next data is set in the SCiTB register, the transmission is continued by transferring the data to the shift register.

The SCiTB register can be read from or written to by the CPU and the DMAC at all times. Write transmit data to the SCiTB register only after confirming the TBEF flag in the SCiSR register is set to 1 (no data in the transmit buffer register).

The transmit shift register cannot be directly accessed.

17.2.2 SCIi Control Register (SCiCR) (i = 0 to 3)

Address SC0CR: H'FFFF 6002, SC1CR: H'FFFF 6102, SC2CR: H'FFFF 6202, SC3CR: H'FFFF 6302

	b7	b6	b5	b4	b3	b2	b1	b0
	TIE	RIE	TE	RE	—	TEIE	CKS[1:0]	
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	TIE	SCIi Transmit Buffer Empty Interrupt Enable Bit	0: SCIi transmit buffer empty interrupt disabled 1: SCIi transmit buffer empty interrupt enabled	R/W
b6	RIE	SCIi Receive Buffer Full Interrupt Enable Bit	0: SCIi receive buffer full interrupt disabled 1: SCIi receive buffer full interrupt enabled	R/W
b5	TE	Transmit Enable Bit	0: Transmission disabled 1: Transmission enabled	R/W
b4	RE	Receive Enable Bit	0: Reception disabled 1: Reception enabled	R/W
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R
b2	TEIE	SCIi Transmit End Interrupt Enable Bit	0: SCIi transmit end interrupt disabled 1: SCIi transmit end interrupt enabled	R/W
b1, b0	CKS[1:0]	Clock Select Bits	In asynchronous mode b1 b0 0 0 : On-chip baud rate generator (SCK pin can be used as the I/O port) 0 1 : On-chip baud rate generator (Clock signal with the same frequency as the bit rate is output at the SCK pin) 1 X : External clock (Input the clock signal operating at sixteen times the frequency of the bit rate at the SCK pin) In clock synchronous mode b1 b0 0 X : Internal clock (SCK pin turns out to be a clock output pin) 1 X : External clock (SCK pin turns out to be a clock input pin)	R/W

For details on interrupt requests, refer to section 17.4.1, Interrupt Sources. Interrupt requests must be set enabled before (or at least simultaneously with) setting the TE or RE bit.

TIE Bit

Whether or not to use a SCIi transmit buffer empty interrupt is selected by using the TIE bit.

The SCIi transmit buffer empty interrupt cannot be used simultaneously with the SCIi transmit end interrupt. Set the TEIE bit to 0 when the TIE bit is set to 1.

RIE Bit

Whether or not to use an SCIi receive buffer full interrupt is selected by using the RIE bit.

TE Bit

Whether or not to transmit the data is selected by using the TE bit. A serial transmission starts by setting the transmit data into the SCiTB register when the TE bit is 1 and the TBEF flag in the SCiSR register is 1 (no data in the transmit buffer register). The SCiMD register must be set to decide a transmission format, before the TE bit is set to 1.
The TBEF flag is fixed to 1 when the TE bit is set to 0.
The TE bit can be set to 1 when bits TE and RE are 0.
To change the bit value when the TE bit, the RE bit, or both are set to 1, write 0 to both bits simultaneously.

RE Bit

Whether or not to receive the data is selected by using the RE bit. A serial reception starts by detecting a start bit in asynchronous mode or a clock signal input in clock synchronous mode. The SCiMD register must be set to decide a reception format, before the RE bit is set to 1.
The RE bit can be set to 1 when bits TE and RE are 0.
To change the bit value when the TE bit, the RE bit, or both are set to 1, write 0 to both bits simultaneously.

TEIE Bit

Whether or not to use an SCIi transmit end interrupt is selected by using the TEIE bit.
The SCIi transmit end interrupt cannot be used simultaneously with the SCIi transmit buffer empty interrupt. Set the TIE bit to 0 when the TEIE bit is set to 1.

CKS Bits

A clock source and an SCKi pin function are selected by using the CKS bits. Set these bits only when bits TE and RE are 0.

17.2.3 SCIi Mode Register (SCiMD) (i = 0 to 3)

Address SC0MD: H'FFFF 6000, SC1MD: H'FFFF 6100, SC2MD: H'FFFF 6200, SC3MD: H'FFFF 6300

	b7	b6	b5	b4	b3	b2	b1	b0
	SMS	SDLS	PE	OES	TSTLS	—	SCSS[1:0]	
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	SMS	SCI Mode Select Bit	0: Asynchronous mode 1: Clock synchronous mode	R/W
b6	SDLS	Data Length Select Bit	0: The data length is 8 bits in transmission/reception 1: The data length is 7 bits in transmission/reception	R/W
b5	PE	Parity Enable Bit	[Only valid in asynchronous mode] 0: No parity is in transmission and parity checks are not performed in reception. 1: Parity bits are added in transmission and parity checks are performed in reception.	R/W
b4	OES	Parity Select Bit	[Only valid in asynchronous mode] 0: Even parity 1: Odd parity	R/W
b3	TSTLS	Stop Bit Length Select Bit	[Only valid in asynchronous mode] 0: 1 stop bit 1: 2 stop bits	R/W
b2	—	Reserved	This bit is read as 0. The write value should be 0.	R
b1, b0	SCSS[1:0]	Count Source Select Bits	b1 b0 0 0 : SCI operating clock undivided 0 1 : SCI operating clock divided by 4 1 0 : SCI operating clock divided by 16 1 1 : SCI operating clock divided by 64	R/W

Set the SCiMD register when bits TE and RE in the SCiCR register are 0.

SMS Bit

Asynchronous mode or clock synchronous mode is selected by using the SMS bit.

SDLS Bit

This bit is valid only in asynchronous mode. When the data length is 7 bits in asynchronous mode, LSB first is fixed and bit 7 of the SCiTB register is not transmitted in transmission. In clock synchronous mode, a fixed data length of 8 bits is used.

TSTLS Bit

In reception, only the first stop bit is checked regardless of this bit setting. If the second stop bit is 0, it is treated as the start bit of the next transmit frame.

SCSS Bits

A count source clock ratio for a baud rate generator is selected by using the SCSS bits.

17.2.4 SCIi Bit Rate Register (SCiBR) (i = 0 to 3)

Address SC0BR: H'FFFF 6001, SC1BR: H'FFFF 6101, SC2BR: H'FFFF 6201, SC3BR: H'FFFF 6301

	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	1	1	1	1	1	1	1	1

Bit	Description	R/W
b7 to b0	Bit rate If a setting value is n, a count source is divided by n+1.	R/W

As each SCI channel has an independent baud rate generator, different bit rates can be set for each channel. Table 17.4 lists the relationship between the n setting value in the SCiBR register and bit rate B for normal asynchronous mode and normal clock synchronous mode. The initial value of the SCiBR register is H'FF. The SCiBR register can always be read from by the CPU but can only be written to when TE = RE = 0.

The setting value of the SCiBR register is obtained by the following equations.

(1) Asynchronous mode

$$\text{Value of the SCiBR register} = \left(\frac{f(SCI)}{64 \times 2^{2m-1} \times B}\right) - 1$$

(2) Clock synchronous mode

$$\text{Value of the SCiBR register} = \left(\frac{f(SCI)}{8 \times 2^{2m-1} \times B}\right) - 1$$

f (SCI): SCI operating clock

Input clock frequency (Hz) for SCI module

B: Bit rate (bps)

m: See Table 17.4.

Table 17.4 The Value m in the Equations for the Value of the SCiBR Register

SCSS Bits in the SCiMD Register	m
00	0
01	1
10	2
11	3

(Example calculation)

Asynchronous mode: SCI operating clock f (SCI) = 10 MHz, Bit rate = 9600 bps, SCSS bits = B'00

$$\text{Value of the SCiBR register} = \left(\frac{10000000}{64 \times 2^{2\times0-1} \times 9600}\right) - 1 = 32.5 - 1 \approx \underline{32}$$

17.2.5 SCIi Receive Buffer Register (SCiRB) (i = 0 to 3)

Address SC0RB: H'FFFF 6005, SC1RB: H'FFFF 6105, SC2RB: H'FFFF 6205, SC3RB: H'FFFF 6305

	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	x	x	x	x	x	x	x	x

Bit	Description	R/W
b7 to b0	Receive buffer data This register stores the receive data.	R

When one frame of the data has been received from the RXDi pin, the receive data is automatically transferred to the SCiRB register from the receive shift register (see Figure 17.1) and the next data can be received. The SCiRB register and the receive shift register have a double buffer function to enable continuous reception.
The RBFF flag in the SCiSR register must be 1 (data in the receive buffer register) when reading the SCiRB register.

The receive shift register cannot be directly accessed.

17.2.6 SCIi Status Register (SCiSR) (i = 0 to 3)

Address SC0SR: H'FFFF 6004, SC1SR: H'FFFF 6104, SC2SR: H'FFFF 6204, SC3SR: H'FFFF 6304

	b7	b6	b5	b4	b3	b2	b1	b0
	TBEF	RBFF	OREF	FREF	PERF	TSEF	—	—
After Reset	1	0	0	0	0	1	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	TBEF	Transmit Buffer Register Empty Flag	0: Data 1: No data	R/(W) *
b6	RBFF	Receive Buffer Register Full Flag	0: No data 1: Data	R/(W) *
b5	OREF	Overrun Error Flag	0: No overrun error has occurred 1: Overrun error has occurred	R/(W) *
b4	FREF	Framing Error Flag	0: No framing error has occurred 1: Framing error has occurred	R/(W) *
b3	PERF	Parity Error Flag	0: No parity error has occurred 1: Parity error has occurred	R/(W) *
b2	TSEF	Transmit Shift Register Empty Flag	0: Data in the transmit shift register (during transmission) 1: No data in the transmit shift register (transmit end)	R
b1, b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

Note: * Only 0 can be written after 1 is read to clear the flag.

Flags TBEF, RBFF, OREF, FREF, and PERF can only be cleared.

TBEF Flag

Whether or not the transmit data is in the SCiTB register is determined by using the TBEF flag. Only 0 can be written to clear the flag.

[Conditions to become 1]

- When the TE bit in the SCiCR register is set to 0 (transmission disabled)
- When the data is transferred to the transmit shift register from the SCiTB register and the data can be written to the SCiTB register

[Conditions to become 0]

- When 0 is written after reading that the TBEF flag is 1
- When the transmit data is written to the SCiTB register

Read the TBEF flag to ensure that the flag is 0 after writing 0, when clearing the flag.

RBFF Flag

Whether or not the receive data is in the SCiRB register is determined by using the RBFF flag. Only 0 can be written to clear the flag.

[Condition to become 1]

When reception ends normally and the receive data is transferred from the receive shift register to the SCiRB register

[Conditions to become 0]

- When 0 is written after reading that the RBFF flag is 1
- When the receive data is read from the SCiRB register

When 0 is written to the RBFF flag for clearing, make sure that 0 can be read after writing 0. Serial communications cannot be continued while the RBFF flag is 1. If the next data reception is completed with the RBFF flag set to 1, an overrun error is generated and the receive data is erased.

OREF Flag

Whether an overrun error has occurred when receiving the data is determined by using the OREF flag. When the overrun error has occurred, the data before the overrun error is held in the SCiRB register and the data received while the OREF flag is 1 is not transferred to the SCiRB register.

Serial communication cannot be continued while OREF = 1. Thus, set the OREF flag to 0 to continue serial communication. Only 0 can be written to clear the flag.

[Condition to become 1]

- When the next data is received while the RBFF flag is 1

[Condition to become 0]

- When 0 is written after reading that the OREF flag is 1

Read the OREF flag to ensure that the flag is 0 after writing 0, when clearing the flag by using an interrupt.

FREF Flag

Whether a framing error has occurred when receiving the data in asynchronous mode is determined by using the FREF flag. When the TSTLS bit is 1 (2 stop bits), only the first bit is checked. The second bit is regarded as the start bit for the next transmit frame and the FREF flag is not set if the second bit is 0. Although the receive data is transferred to the SCiRB register even if the framing error has occurred, the RBFF flag is not set. Serial communication cannot be continued while FREF = 1. Thus, set the FREF flag to 0 to continue serial communication. Only 0 can be written to clear the flag.

[Condition to become 1]

- When the number of stop bits set by the TSTLS bit in the SCiMD register is not detected

[Condition to become 0]

- When 0 is written after reading that the FREF flag is 1

Read the FREF flag to ensure that the flag is 0 after writing 0, when clearing the flag by using an interrupt.

PERF Flag

Whether a parity error has occurred when receiving the data in asynchronous mode is determined by using the PERF flag. The parity error can be detected when the PE bit in the SCiMD register is 1 (parity enabled). Although the receive data is transferred to the SCiRB register even if the parity error has occurred, the RBFF flag is not set. Serial communication cannot be continued while PERF = 1. Thus, set the PERF flag to 0 to continue serial communication. Only 0 can be written to clear the flag.

[Condition to become 1]

- When the parity error has occurred during the reception

[Condition to become 0]

- When 0 is written after reading that the PERF flag is 1

Read the PERF flag to ensure that the flag is 0 after writing 0, when clearing the flag by using an interrupt.

TSEF Flag

Whether the transmit shift register is empty is determined by using the TSEF flag.

[Conditions to become 1]

- When the TE bit in the SCiCR register is set to 0 (transmission disabled)
- When the TBEF flag is 1 during the transmission of the last bit of the transmit data.

[Conditions to become 0]

- When 0 is written to the TBEF flag after reading that the flag is 1
- Data is written to the transmit buffer register when the TE bit is 1, and then the data is transferred to the transmit shift register.

17.2.7 SCIi Extension Mode Register (SCiEMD) (i = 0 to 3)

Address SC0EMD: H'FFFF 6008, SC1EMD: H'FFFF 6108, SC2EMD: H'FFFF 6208, SC3EMD: H'FFFF 6308

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	SDIR	—	CKPOS	CKPHS
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R
b3	SDIR	Data Direction Select Bit	0: LSB first is used in transmission/reception 1: MSB first is used in transmission/reception	R/W
b2	—	Reserved	This bit is read as 0. The write value should be 0.	R
b1	CKPOS	Clock Polarity Select Bit	[Only valid in clock synchronous mode] 0: Polarity is not inverted. Transmit data synchronized with a falling edge is output and receive data synchronized with a rising edge is input for the transmit/receive clock 1: Polarity is inverted. Transmit data synchronized with a rising edge is output and receive data synchronized with a falling edge is input for the transmit/receive clock	R/W
b0	CKPHS	Clock Phase Select Bit	[Only valid in clock synchronous mode] 0: Clock phase is not delayed. 1: Clock phase is delayed by a half phase.	R/W

Set the function when bits TE and RE in the SCiCR register are set to 0.

SDIR Bit

This bit selects whether LSB first or MSB first is used in transmission/reception.
This bit is valid only when the transmit/receive format is set to 8-bit data. When the transmit/receive format is set to 7-bit data, LSB first is fixed.

CKPOS Bit

This bit selects whether or not to invert the clock polarity.

CKPHS Bit

This bit selects whether or not to delay the clock phase by a half phase.

17.3 Operations

In the operations of this module, there are two modes: asynchronous mode and clock synchronous mode. Each of their operations and usage examples is shown below.

17.3.1 Asynchronous Mode

Asynchronous mode enables data transmission/reception synchronized with an internal clock generated by a trigger on the falling edge of the start bit. Table 17.5 lists the Asynchronous Mode Specifications.

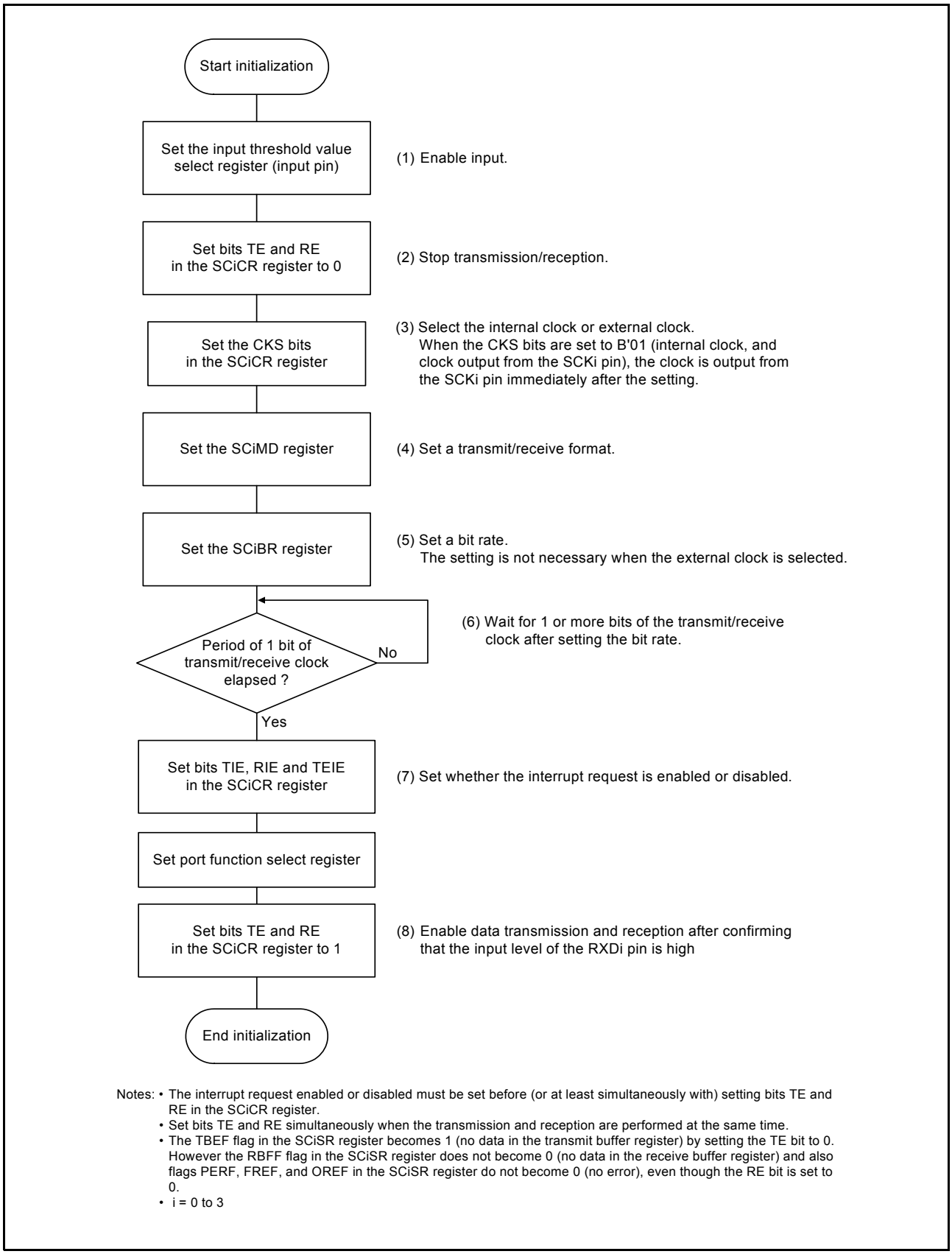
Table 17.5 Asynchronous Mode Specifications

Item	Specification
Data format	• Start bit: 1 bit-length • Data length: 7 or 8 bits can be selected. • Parity bit: Odd, even, or none parity can be selected. • Stop bit length: 1 bit or 2 bits can be selected. • LSB first or MSB first can be selected (only LSB first can be selected in 7-bit data length of asynchronous mode).
I/O pins	• SCKi pin (input/output): Clock input/output • RXDi pin (input): Data input pin • TXDi pin (output): Data output pin
Transmit/receive clock	• When the CKS bits in the SCiCR register are B'00 or B'01 (internal clock) Bit rate = $\frac{\text{f (SCI)}}{64 \times 2^{2m-1} \times (\text{SCiBR register setting value} + 1)}$ f (SCI): SCI operating clock Input clock frequency (Hz) for SCI module m: See Table 17.4 • When the CKS bits in the SCiCR register are B'10 or B'11 (external clock) Clock input to SCKi pin
Transmit start conditions	When the TE bit in the SCiCR register is 1 (transmission enabled), data has been written to the transmit buffer register, and the data has been transferred to the transmit shift register.
Receive start conditions	When the start bit is detected with the RE bit in the SCiCR register being 1 (reception enabled).
Error detection	• Overrun error This error occurs when the next reception is completed while the RBFF flag in the SCiSR register is 1 (data in the receive buffer register). • Framing error Determined by the first bit regardless of the setting of the TSTLS bit in the SCiMD register. • Parity error Occurs when parity check is performed and the number (odd/even) of 1s in the received data character and the parity bit does not match the setting (odd/even).
Interrupt requests	4 interrupt requests SCIi transmit end interrupt, SCIi transmit buffer empty interrupt, SCIi receive buffer full interrupt, SCIi receive error interrupt (overrun error, framing error, parity error)

Note: i = 0 to 3

17.3.1.1 Initialization in Asynchronous Mode

Figure 17.2 shows the Initialization Flowchart in Asynchronous Mode.



Notes:
- The interrupt request enabled or disabled must be set before (or at least simultaneously with) setting bits TE and RE in the SCiCR register.
- Set bits TE and RE simultaneously when the transmission and reception are performed at the same time.
- The TBEF flag in the SCiSR register becomes 1 (no data in the transmit buffer register) by setting the TE bit to 0. However the RBFF flag in the SCiSR register does not become 0 (no data in the receive buffer register) and also flags PERF, FREF, and OREF in the SCiSR register do not become 0 (no error), even though the RE bit is set to 0.
- i = 0 to 3

Figure 17.2 Initialization Flowchart in Asynchronous Mode

17.3.1.2 Data transmission in Asynchronous Mode

The following is an operation example of transmission in asynchronous mode.

The transmit data is transferred to the transmit shift register (see Figure 17.1) from the SCiTB register by writing the transmit data into the SCiTB register and setting the TBEF flag in the SCiSR register to 0 (data in the transmit buffer register) when the TE bit in the SCiCR register is 1 (transmission enabled) and the TBEF flag is 1 (no data in the transmit buffer register). After that, the TBEF flag becomes 1 and the data transmission is started. An SCIi transmit buffer empty interrupt occurs when the TIE bit in the SCiCR register is 1 (SCIi transmit buffer empty interrupt enabled). The start bit, transmit data, parity bit and stop bit are transmitted from the TXDi pin in this order.

While the stop bit is transmitted, the TBEF flag is checked. The next transmit data is transferred to the transmit shift register from the SCiTB register when the TBEF flag is 0, and is transmitted after the stop bit is transmitted.

When the stop bit has been transmitted later, the TSEF flag in the SCiSR register becomes 1 (no data in the transmit shift register), and an SCIi transmit end interrupt occurs when the TEIE bit in the SCiCR register is 1 (SCIi transmit end interrupt enabled). The TXDi pin becomes high level after the transmission is completed.

Figure 17.3 shows the Operation Example of Transmission in Asynchronous Mode.

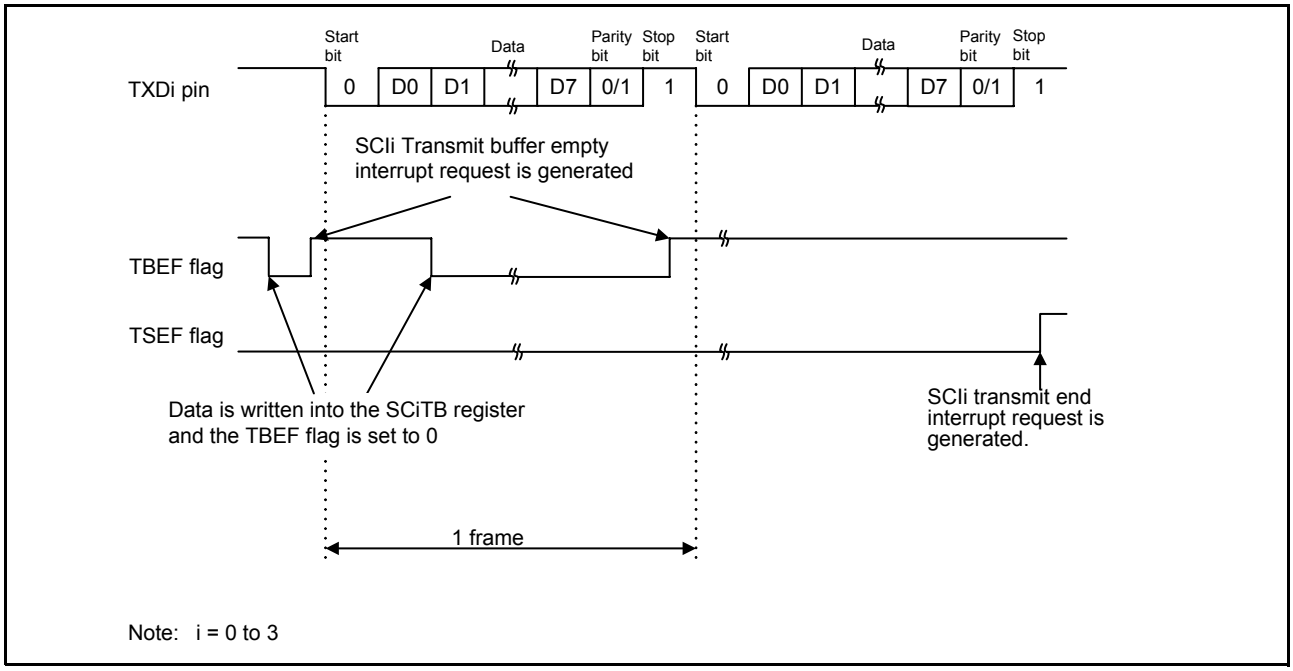


Figure 17.3 Operation Example of Transmission in Asynchronous Mode

Figure 17.4 shows the Data Transmission Procedure in Asynchronous Mode.

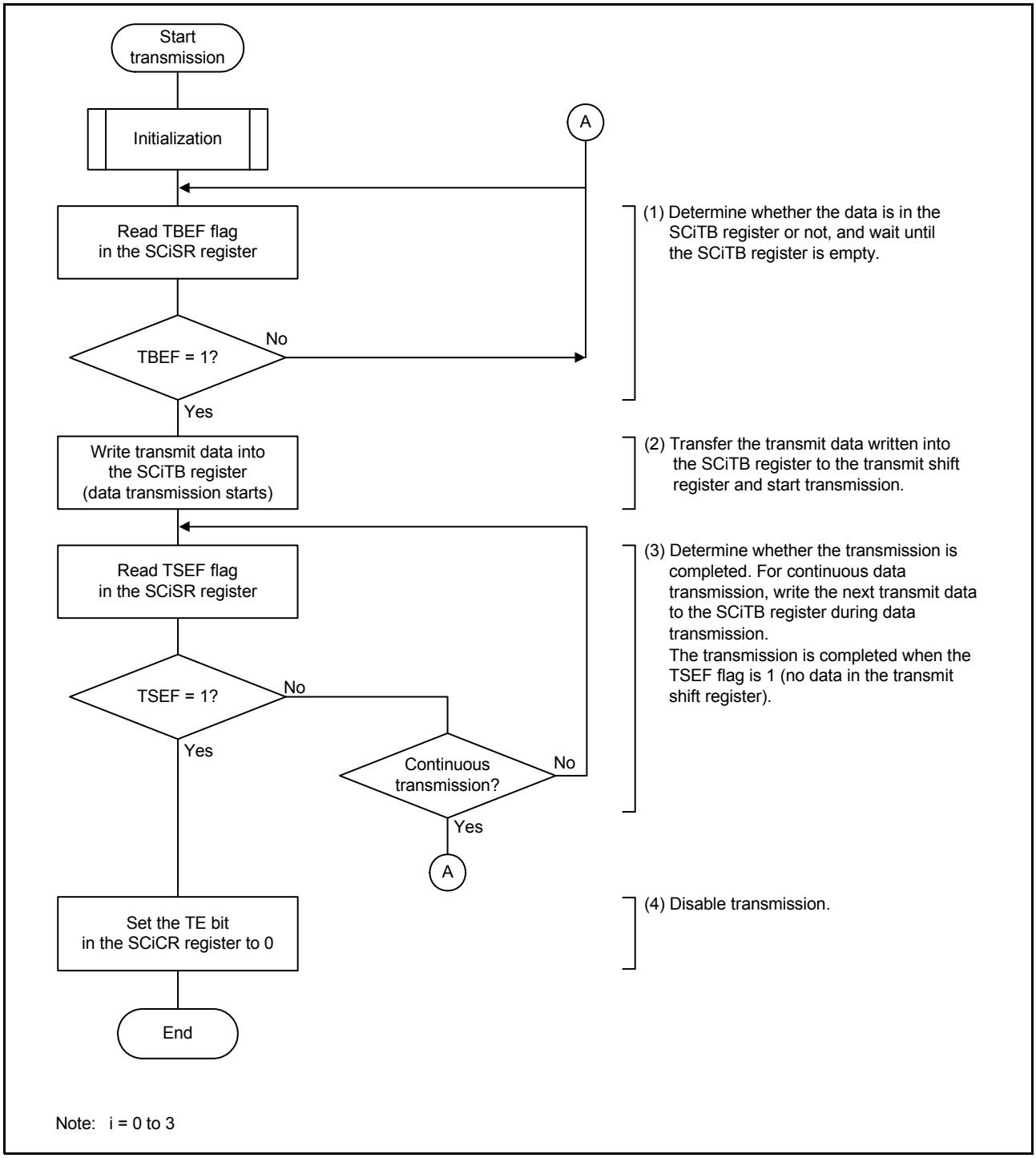


Figure 17.4 Data Transmission Procedure in Asynchronous Mode

17.3.1.3 Data Reception in Asynchronous Mode

The following is an operation example of reception in asynchronous mode.

The data reception is started by detecting the start bit and the receive data is input into the receive shift register (see Figure 17.1). After completing the data reception, the parity bit and stop bit are checked.

The RBFF flag in the SCiSR register becomes 1 (data in the receive buffer register) when the data reception is successfully completed, and the receive data is transferred to the SCiRB register from the receive shift register.

The PERF or FREF flag in the SCiSR register is set to 1 (error has occurred) and the receive data is transferred to the SCiRB register from the receive shift register, when parity error/framing error has occurred.

The OREF flag in the SCiSR register becomes 1 (overrun error has occurred) and the receive data is not transferred to the SCiRB register from the receive shift register, when an overrun error has occurred. The RBFF flag holds 1.

A receive interrupt request is generated regardless of whether or not the receive error has occurred when the reception is completed, if the RIE bit in the SCiCR register is 1 (receive interrupt enabled).

Figure 17.5 shows the Operation Example of Reception in Asynchronous Mode.

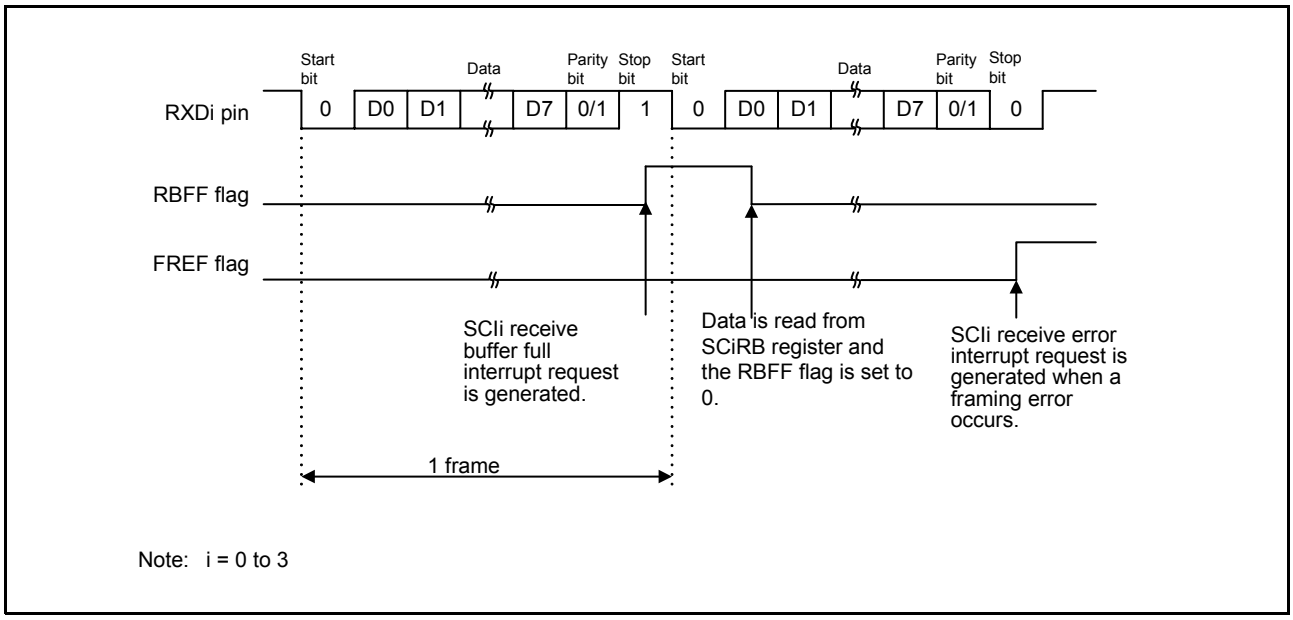


Figure 17.5 Operation Example of Reception in Asynchronous Mode

Figure 17.6 shows the Data Reception Procedure Example in Asynchronous Mode.

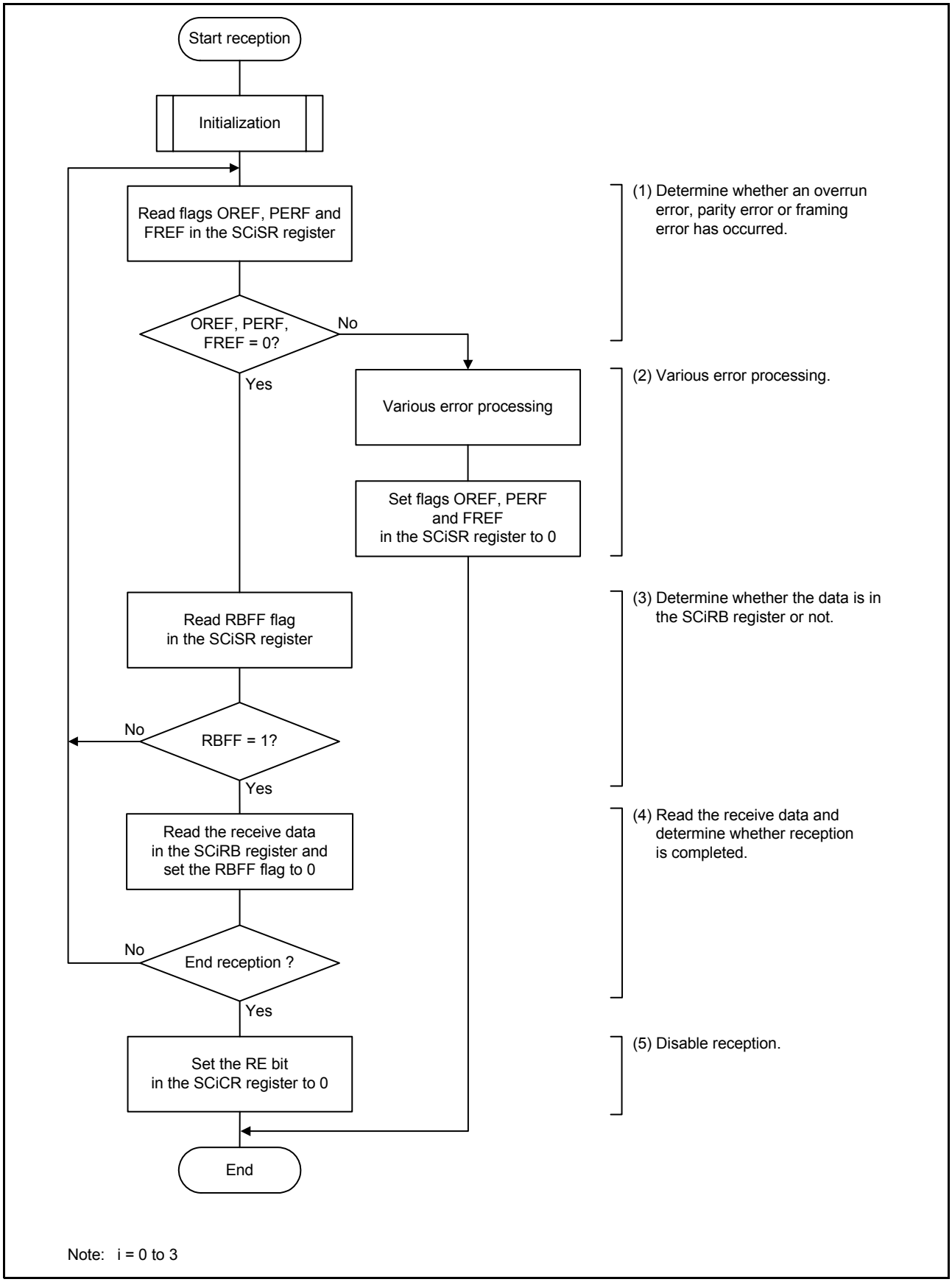


Figure 17.6 Data Reception Procedure Example in Asynchronous Mode

Table 17.6 lists the status of registers and receive data when receive errors are detected.

Table 17.6 Status Flag and Receive Data in Receive Error

Status Flags in the SCiSR Register				Receive Error Status	Receive Data
RBFF	OREF	FREF	PERF		
1	1	0	0	Overrun error	Not transferred to the SCiRB register
0	0	1	0	Framing error	Transferred to the SCiRB register
0	0	0	1	Parity error	Transferred to the SCiRB register
1	1	1	0	Overrun error + framing error	Not transferred to the SCiRB register
1	1	0	1	Overrun error + parity error	Not transferred to the SCiRB register
0	0	1	1	Framing error + parity error	Transferred to the SCiRB register
1	1	1	1	Overrun error + framing error + parity error	Not transferred to the SCiRB register

Note: i = 0 to 3

17.3.1.4 Clock in Asynchronous Mode

When an internal clock is used in asynchronous mode, the internal clock operates at sixteen times the frequency of the bit rate. Thus, when an external clock input is used in asynchronous mode, the clock operating at sixteen times the frequency of the bit rate is required to be input at the SCKi pin. Figure 17.7 shows the Timing of Receive Data Sampling in Asynchronous Mode.

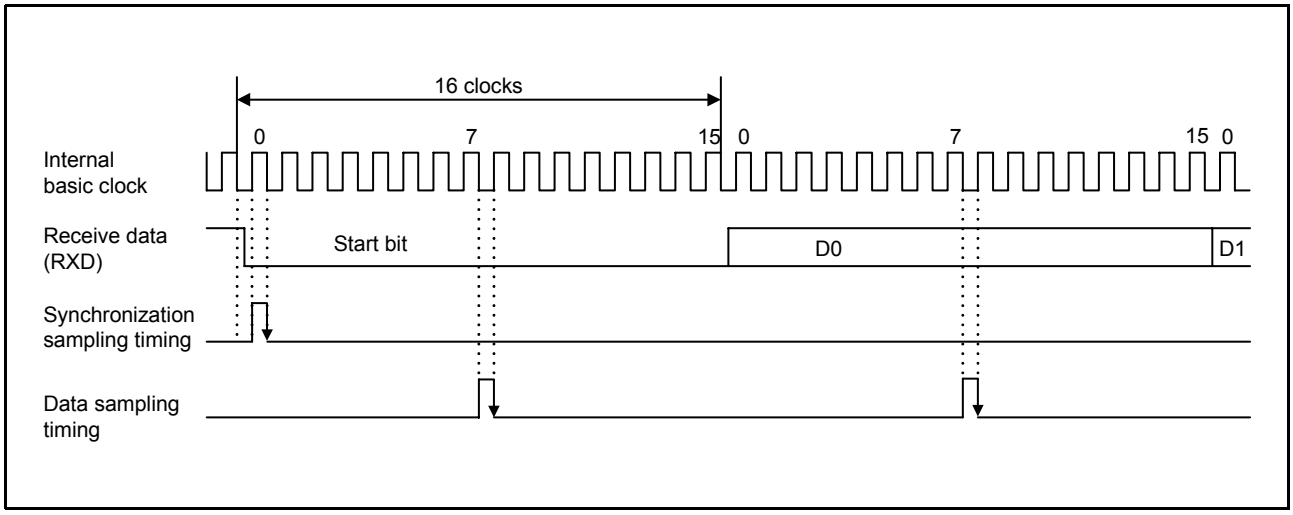


Figure 17.7 Timing of Receive Data Sampling in Asynchronous Mode

17.3.2 Clock Synchronous Mode

Clock synchronous mode allows data transmission/reception synchronized with transmit/receive clock.

Table 17.7 lists the Clock Synchronous Mode Specifications.

Channel 3 is not available in clock synchronous mode in the SH72A0 group.

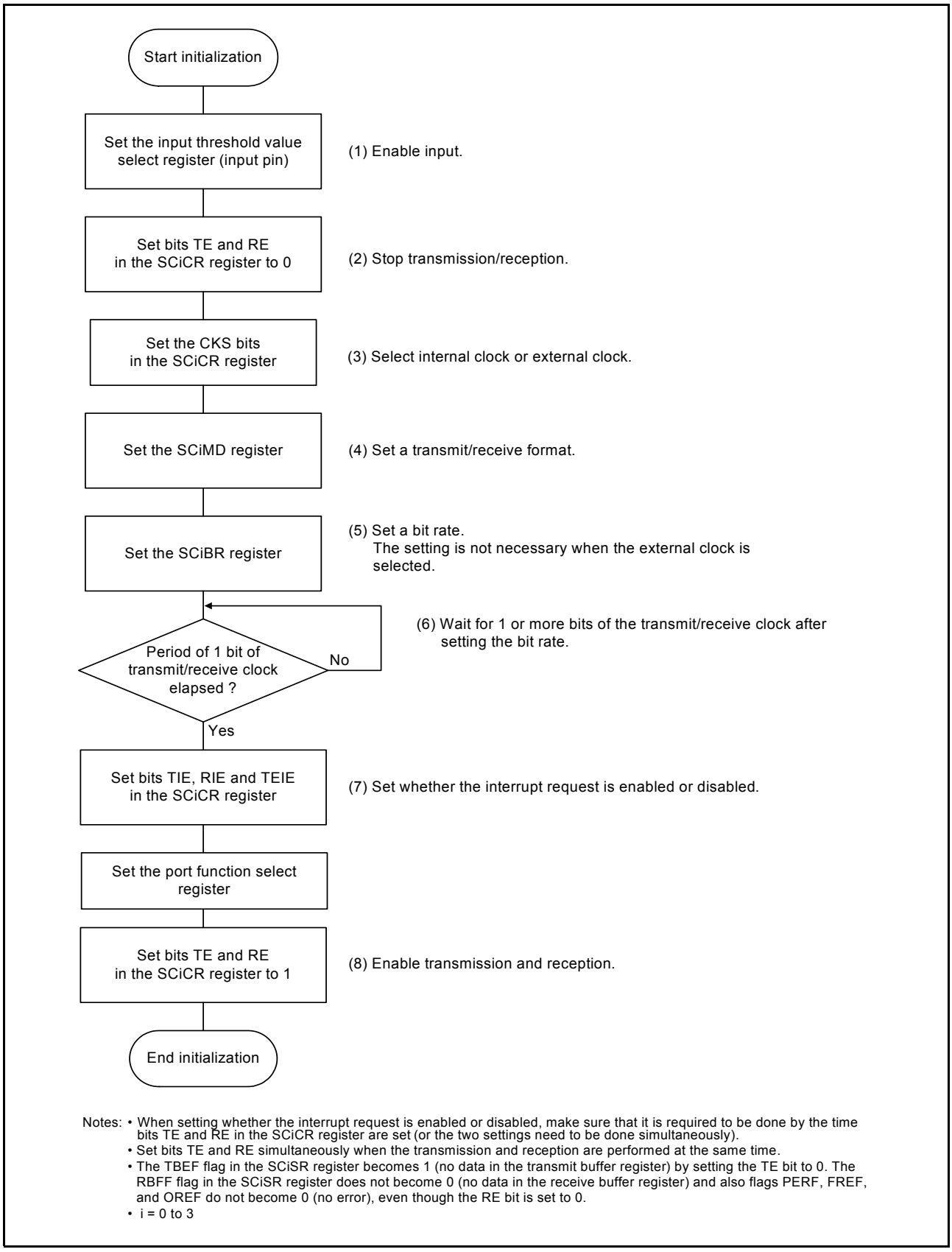
Table 17.7 Clock Synchronous Mode Specifications

Item	Specification
Data format	• Data length: 8 bits • LSB first or MSB first can be selected. • Clock phase and polarity can be selected.
I/O pins	• SCKi pin (I/O): Clock input pin • RXDi pin (input): Data input pin • TXDi pin (output): Data output pin
Transmit/receive clock	• When the CKS bits in the SCiCR register are B'00 or B'01 (internal clock) Bit rate = $\frac{f(SCI)}{8 \times 2^{2m-1} \times (\text{SCiBR register setting value} + 1)}$ f (SCI): SCI operating clock Input clock frequency (Hz) for SCI module m: See Table 17.4 • When the CKS bits in the SCiCR register are B'10 or B'11 (external clock) Clock input to SCKi pin
Transmit start conditions	When the TE bit in the SCiCR register is 1 (transmission enabled), data has been written to the transmit buffer register, and the data has been transferred to the transmit shift register.
Receive start conditions	When the clock input at the SCKi pin is detected while the RE bit in the SCiCR register is 1 (reception enabled)
Error detection	Overrun error This error occurs when the next reception is completed while the RBFF flag in the SCiSR register is 1 (data in the receive buffer register).
Interrupt requests	4 interrupt requests SCIi transmit end interrupt, SCIi transmit buffer empty interrupt, SCIi receive buffer full interrupt, SCIi receive error interrupt (overrun error)

Note: i = 0 to 3

17.3.2.1 Initialization in Clock Synchronous Mode

Figure 17.8 shows the Initialization Flowchart in Clock Synchronous Mode.



Notes: • When setting whether the interrupt request is enabled or disabled, make sure that it is required to be done by the time bits TE and RE in the SCiCR register are set (or the two settings need to be done simultaneously).
• Set bits TE and RE simultaneously when the transmission and reception are performed at the same time.
• The TBEF flag in the SCiSR register becomes 1 (no data in the transmit buffer register) by setting the TE bit to 0. The RBFF flag in the SCiSR register does not become 0 (no data in the receive buffer register) and also flags PERF, FREF, and OREF do not become 0 (no error), even though the RE bit is set to 0.
• i = 0 to 3

Figure 17.8 Initialization Flowchart in Clock Synchronous Mode

17.3.2.2 Data Transmission in Clock Synchronous Mode

In data transmission, clock of a bit rate set by the SCiBR register is used as the transmit/receive clock when the clock is the internal clock, and the clock input to SCKi pin is used as the transmit/receive clock when the clock is the external clock. In both cases the transmit data is synchronized with the transmit/receive clock and output from the TXDi pin.

The following is an operation example of transmission in clock synchronous mode.

The transmit data is transferred to the transmit shift register (see Figure 17.1) from the SCiTB register by writing the transmit data into the SCiTB register and setting the TBEF flag in the SCiSR register to 0 (data in the transmit buffer register) when the TE bit in the SCiCR register is 1 (transmission enabled) and the TBEF flag is 1 (no data in the transmit buffer register). After that, the TBEF flag becomes 1 and the data transmission is started. An SCIi transmit buffer empty interrupt occurs when the TIE bit in the SCiCR register is 1 (SCIi transmit buffer empty interrupt enabled). To perform the continuous transmission, set the TBEF flag to 0 by writing the next data into the SCiTB register during the transmission of the transmit data after confirming that the TBEF flag is 1. After the eighth bit of the data in transmission is transmitted, the next transmit data is transferred to the transmit shift register from the SCiTB register and the transmission is continued. When the continuous transmission is not performed, the TSEF flag in the SCiSR register becomes 1 (no data in the transmit shift register) by transmitting the eighth bit of the transmit data, and the transmission is completed. An SCIi transmit end interrupt occurs when the TEIE bit in the SCiCR register is 1 (SCIi transmit end interrupt enabled). After the transmission, TXDi pin holds the output level of the last bit and the SCKi pin becomes high level.

Figure 17.9 shows the Operation Example of Transmission in Clock Synchronous Mode.

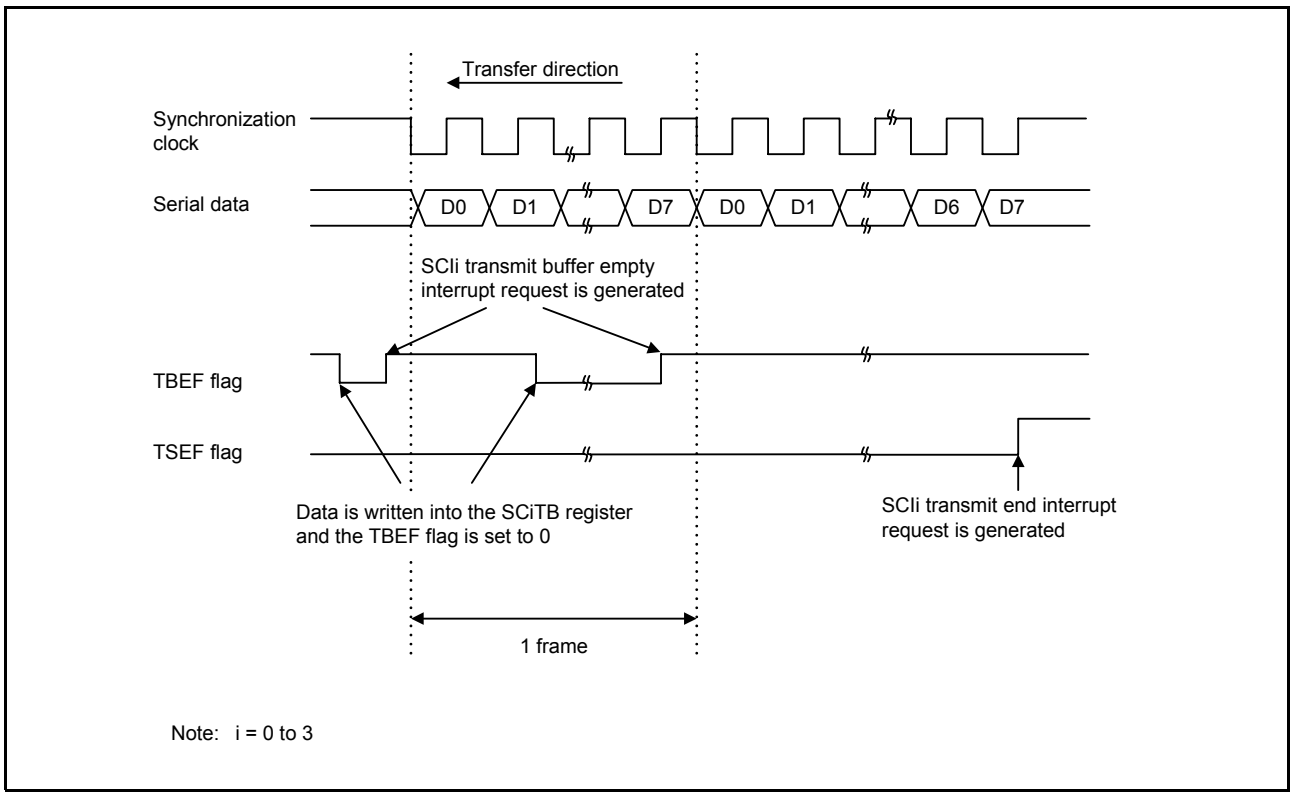


Figure 17.9 Operation Example of Transmission in Clock Synchronous Mode

Figure 17.10 shows the Data Transmission Procedure Example in Clock Synchronous Mode.

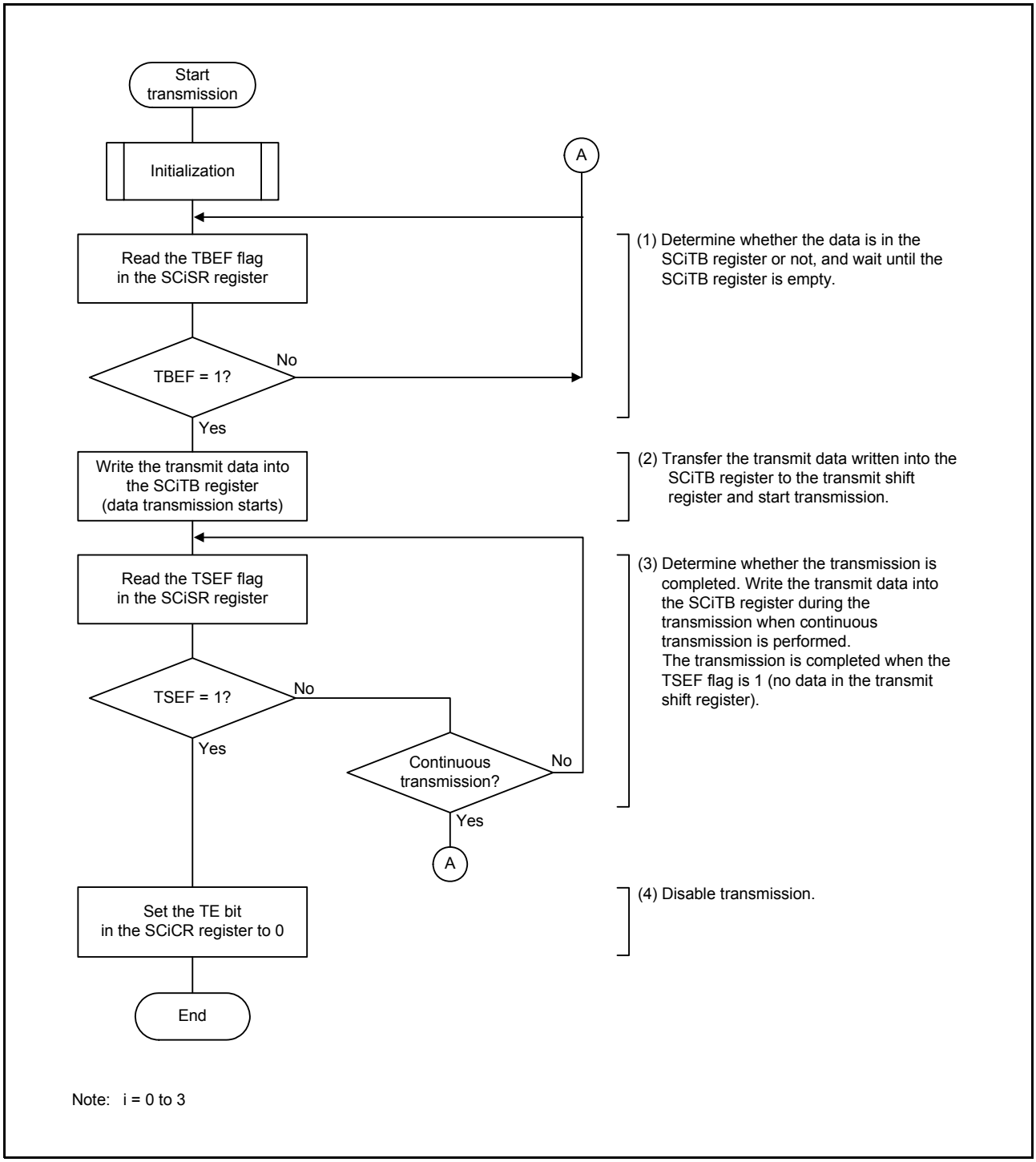


Figure 17.10 Data Transmission Procedure Example in Clock Synchronous Mode

17.3.2.3 Data Reception in Clock Synchronous Mode

In data reception, clock of a bit rate set by the SCiBR register is used as the transmit/receive clock when the clock is the internal clock, and the clock input to the SCKi pin is used as the transmit/receive clock when the clock is the external clock. In both cases, the receive data is synchronized with the transmit/receive clock and input to the RXDi pin.

The following is an operation example of reception in clock synchronous mode.

The RBFF flag in the SCiSR register becomes 1 (data in the receive buffer register) by receiving 8 bits of the data, and the receive data is transferred to the SCiRB register from the receive shift register (see Figure 17.1). A receive interrupt occurs when the RIE bit in the SCiCR register is 1 (receive interrupt enabled).

The OREF flag in the SCiSR register becomes 1 (overrun error has occurred) and an overrun error occurs by completing the reception when the RBFF flag is 1, and an SCIi receive error interrupt request is generated. The receive data is not transferred to the SCiRB register from the receive shift register and the RBFF flag holds 1. The serial transmission/reception cannot be performed when the receive error flag is set.

Set flags OREF, FREF, PERF, and RBFF to 0 to continue the data reception.

Figure 17.11 shows the Operation Example of Reception in Clock Synchronous Mode.

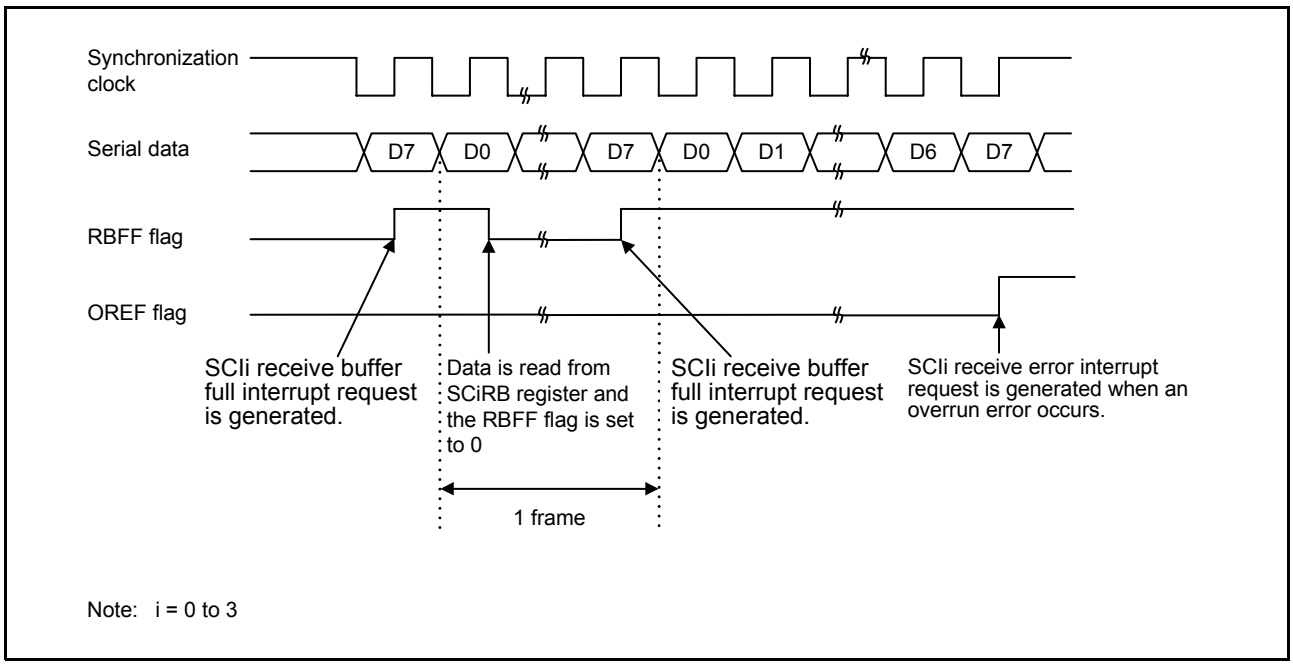


Figure 17.11 Operation Example of Reception in Clock Synchronous Mode

Figure 17.12 shows the Data Reception Procedure Example in Clock Synchronous Mode.

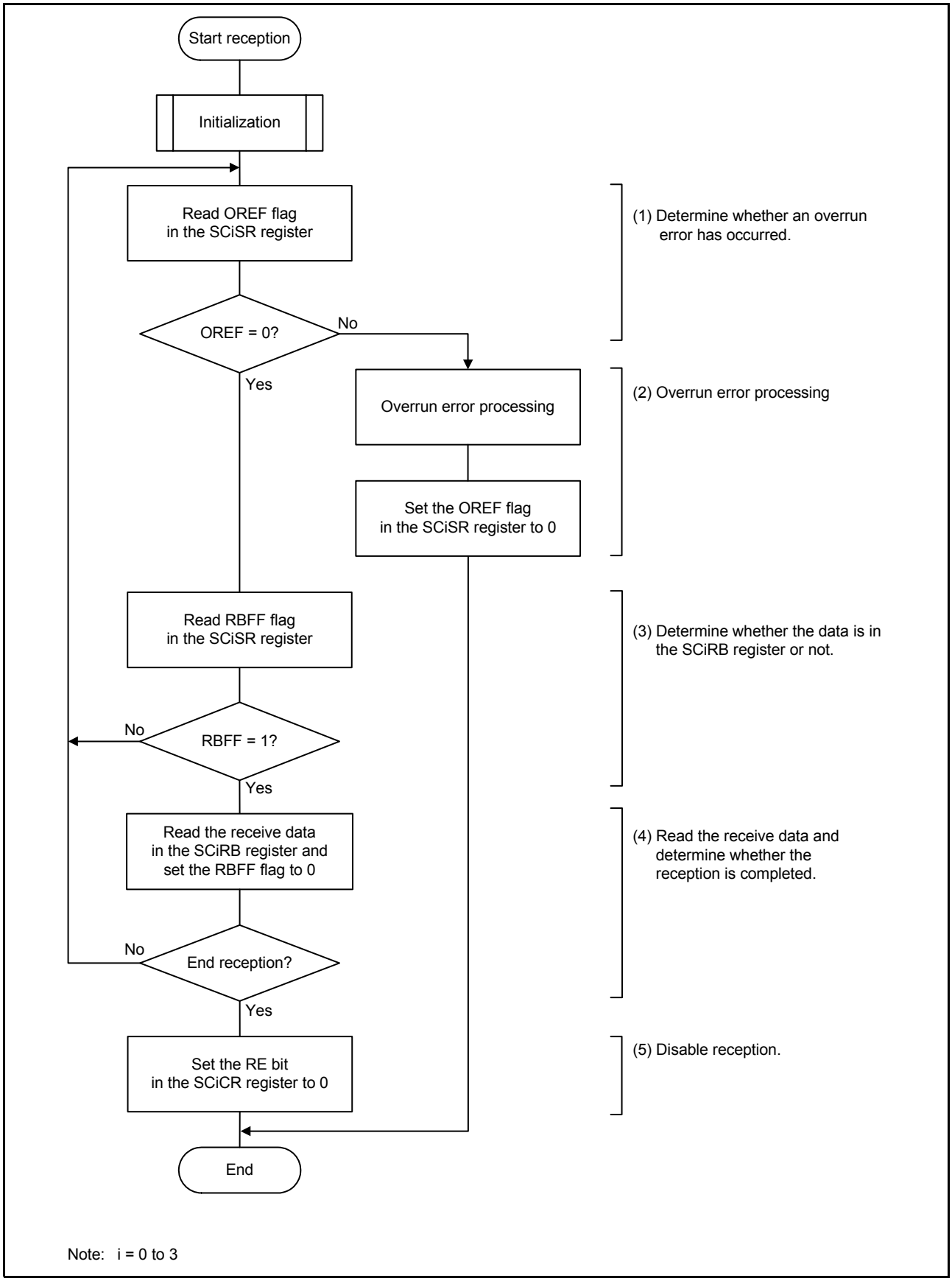


Figure 17.12 Data Reception Procedure Example in Clock Synchronous Mode

17.3.2.4 Clock Phase and Polarity Selection in Clock Synchronous Mode

In clock synchronous mode, communications can be performed varying the polarity and phase of a transmit/receive clock. The transmit/receive clock phase and polarity can be varied by bits CKPHS and CKPOS in the SCiEMD register. Set bits CKPHS and CKPOS after setting the SMS bit in the SCiMD register to 1 (clock synchronous mode). Set bits CKPHS and CKPOS to 0 when the SMS bit is 0 (asynchronous mode).

Figure 17.13 and Figure 17.14 show respectively a transmit/receive timing in master mode (internal clock is selected by the CKS bits in the SCiCR register) and in slave mode (external clock is selected by the CKS bits).

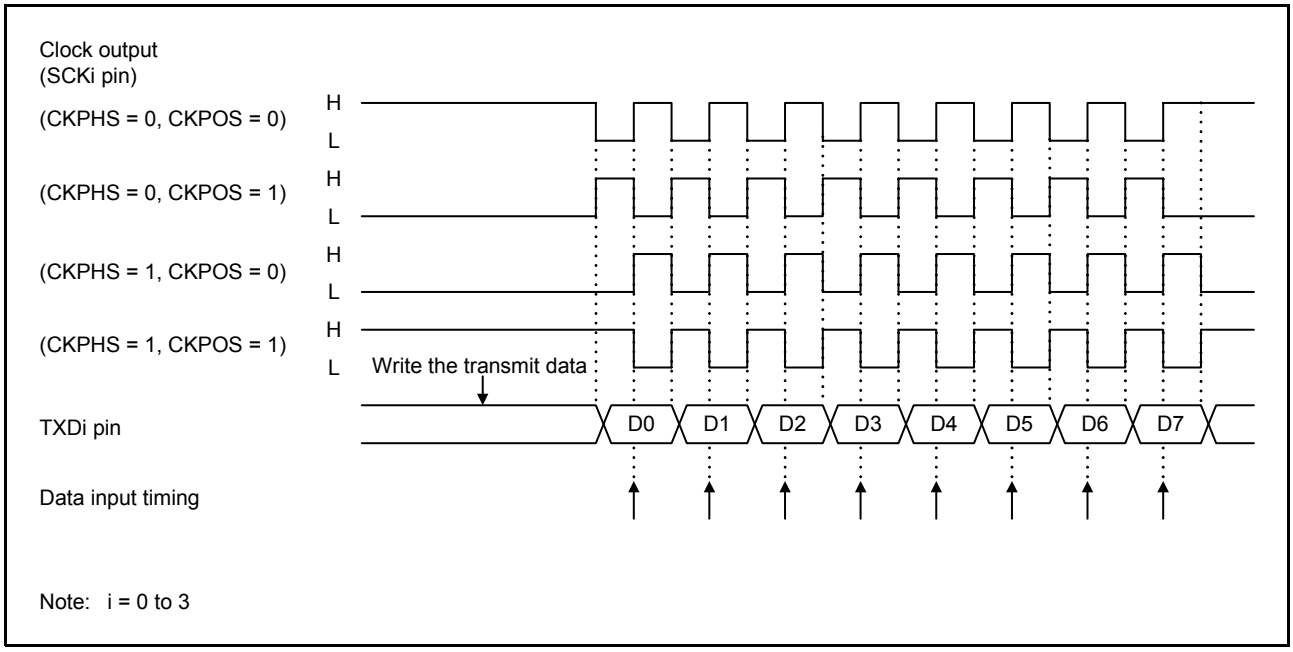


Figure 17.13 Transmit/Receive Timing in Master Mode

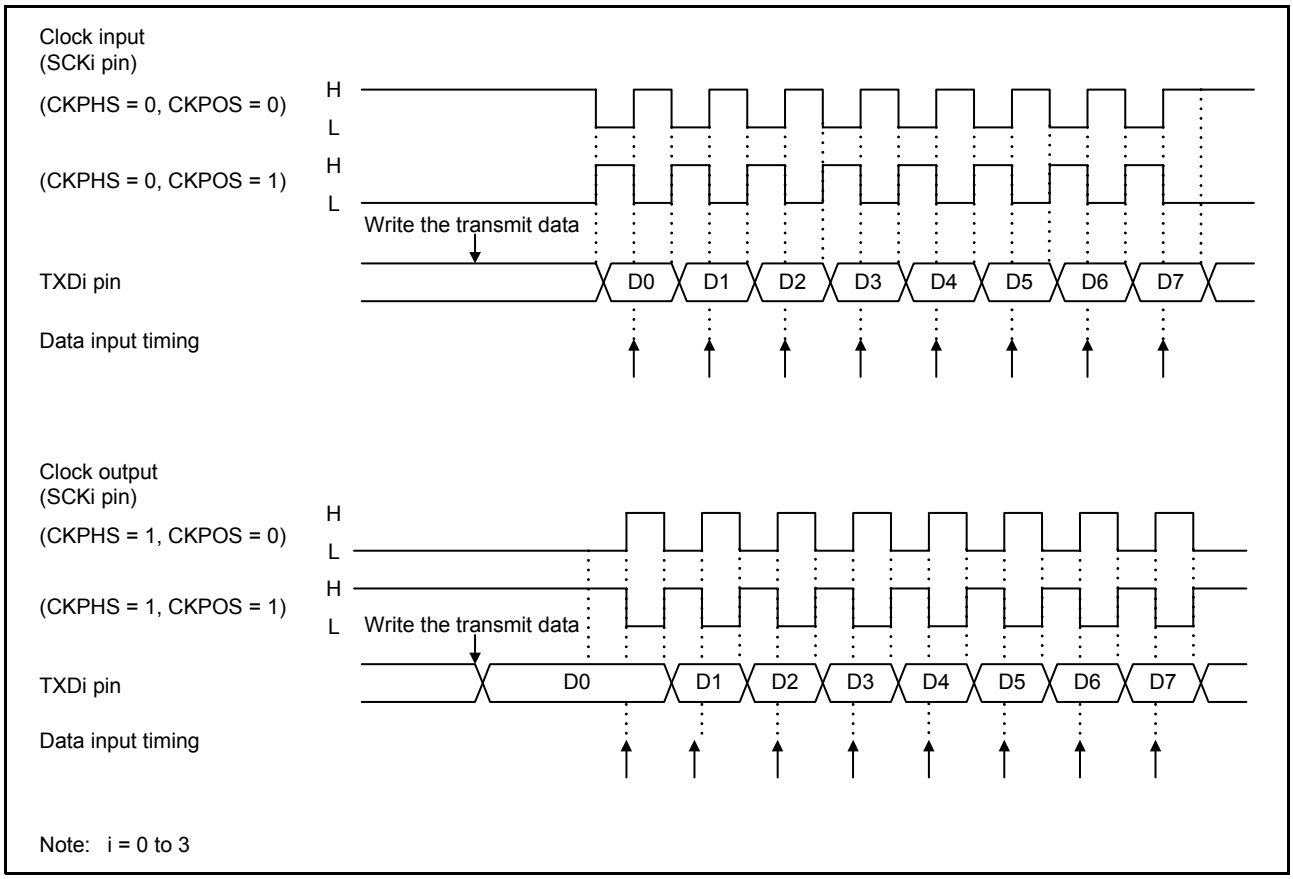


Figure 17.14 Transmit/Receive Timing in Slave Mode

17.4 Interrupts

17.4.1 Interrupt Sources

The SCI has the SCIi transmit end interrupt, the SCIi transmit buffer empty interrupt, the SCIi receive buffer full interrupt, and the SCIi receive error interrupt (overrun errors, framing errors, and parity errors) per channel. Table 17.8 lists a condition to generate an interrupt request.

When the CPU interrupt is operated by interrupt requests, the priority between channels can be switched by setting the interrupt controller (INTC). For details, refer to section 8, Interrupt Controller (INTC).

Table 17.8 SCI Interrupt Sources

Interrupt Request Sources	Condition	DMAC Activation
SCIi transmit end interrupt	When the TSEF flag is 1	Not possible
SCIi transmit buffer empty interrupt	When the TBEF flag changes from 0 to 1	Possible
SCIi receive buffer full interrupt	When the RBFF flag changes from 0 to 1	Possible
SCIi receive error interrupt	When at least one flag among flags OREF, FREF, and PERF is 1	Not possible

Notes: • The SCIi transmit end interrupt and the SCIi transmit buffer empty interrupt cannot be used simultaneously.
• i = 0 to 3

17.5 Notes on SCI

17.5.1 Receive Error Flags and Transmit Operations (Clock Synchronous Mode Only)

When the transmission and reception are operated in clock synchronous mode, the transmission cannot be started with the overrun error flag (OREF) in the SCiSR register set to 1, even if the TBEF flag is cleared to 0. Make sure to clear the overrun error flag (OREF), the framing error flag (FREF), the parity error flag (PERF) to 0 by the time the transmission is started. Also, note that the receive error flags (flags OREF, FREF, and PERF) cannot be cleared to 0 even if the RE bit in the SCiCR register is cleared to 0.

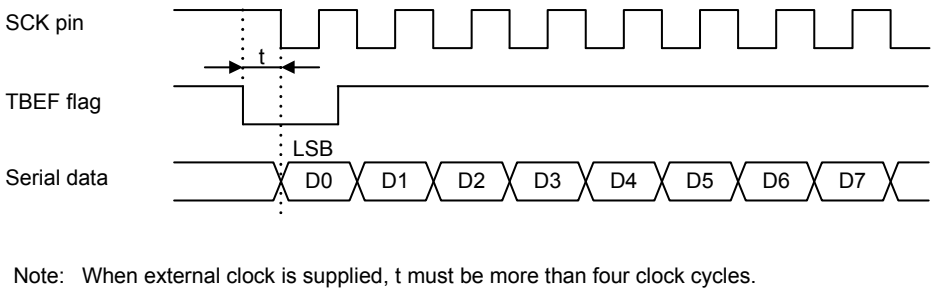
17.5.2 Relation between Writing into SCiTB Register and TBEF Flag

The TBEF flag in the SCiSR register is a status flag which indicates that transmit data have been transferred from the SCiTB register to the transmit shift register. When data is transferred from the SCiTB register to the transmit shift register, the TBEF flag is set to 1.

Data can be written into the SCiTB register, irrespective of the TBEF flag status. However, if a new data is written into the SCiTB register while the TBEF flag is 0, the data stored in the SCiTB register are erased because the stored data has not been transferred to the transmit shift register. Thus, write the transmit data into the SCiTB register after verifying that the TBEF flag is set to 1.

17.5.3 Restrictions in Clock Synchronous Transmission

When the external clock source is used as a synchronization clock, clear the TBEF flag to 0 and then input the transmit clock after waiting for at least five clock cycles of the SCI operating clock. If the transmit clock is input within four clock cycles after the SCiTB register is updated, there is a risk of malfunction (see Figure 17.15).



Note: When external clock is supplied, t must be more than four clock cycles.

Figure 17.15 Restrictions in Clock Synchronous Transmission

17.5.4 External Clock Input in Clock Synchronous Mode

Regarding the SCKi input of the external clock in clock synchronous mode, each of the high pulse period and the low pulse period needs to be specified as two or more clocks, and the cycle needs be specified as six or more clocks.

18. Serial Bus Interface (SBI)

This LSI includes four-channel serial bus interface (SBI) in the SH72A2 group and three-channel SBI (SBI0 to SBI2) in the SH72A0 group.

The SBI has channels (transmission only) which are independent of each other and is capable of full-duplex high-speed serial communication with multiple processors and peripheral devices. In this section, note the i in the SBIi and the i used in pin and signal names represent values from 0 to 3.

18.1 Introduction

Table 18.1 lists the Specifications of Serial Bus Interface (SBI).

Note: The SH72A0 group does not have the SSL10 input/output pin, the SSL03 output pin, and the SSL11 to SSL13 output pins. The number of available CS pins is limited according to the channel in products of the SH72A0 group.

Table 18.1 Specifications of Serial Bus Interface (SBI)

Item	Description
SBI transfer functions	• Use of MOSI (master out/slave in), MISO (master in/slave out), SSL (slave select), and RSPCK (SBI clock) signals allows serial communications through SBI operation (four-wire method) or clock synchronous operation (three-wire method). • Capable of transmit-only operations • Capable of serial communications in master/slave mode • Supports mode fault error detection • Supports overrun error detection • Switching of the polarity of the serial transfer clock • Switching of the clock phase of serial transfer
Data format	• MSB-first/LSB-first selectable • Transfer bit length is selectable as 8, 9, 10, 11, 12, 13, 14, 15, 16, 20, 24, or 32 bits. • 128-bit transmit/receive buffers • Up to four frames can be transferred in one round of transmission/reception (each frame consisting of up to 32 bits).
Buffer configuration	• Double buffer configuration for the transmit/receive buffers
SSL control function	• Four SSL signals (SSLi0 to SSLi3) for each channel of SBI • In single-master mode, outputs SSLi0 to SSLi3 signals • In multi-master mode, SSLi0 signal for input, and SSLi1 to SSLi3 signals for either output or Hi-Z. • In slave mode, SSLi0 signal for input, and SSLi1 to SSLi3 signals for Hi-Z. • Controllable delay from SSL output assertion to RSPCK operation (RSPCK delay) Range: 1 to 8 RSPCK cycles (set in RSPCK-cycle units) • Controllable delay from RSPCK stoppage to SSL output negation (SSL negation delay) Range: 1 to 8 RSPCK cycles (set in RSPCK-cycle units) • Controllable wait for next-access SSL output assertion (next-access delay) Range: 1 to 8 RSPCK cycles (set in RSPCK-cycle units) • Function for changing SSL polarity
Control in master transfer	• A transfer of up to 8 commands can be executed sequentially in looped execution. • For each command, the following can be set: • SSL signal value, bit rate, RSPCK polarity/phase, transfer data length, LSB/MSB first, burst, RSPCK delay, SSL negation delay, and next-access delay. • A transfer can be initiated by writing to the transmit buffer. • A transfer can be initiated by the CPU clearing the SPTEF flag. • MOSI signal value specifiable in SSL negation
Interrupt sources	• Maskable interrupt sources: SBI receive interrupt (receive buffer full) SBI transmit interrupt (transmit buffer empty) SBI error interrupt (mode fault, overrun, parity error) SBI idle interrupt (SBI idle)
Others	• Provides loop back mode. • Provides a function for switching between CMOS output and open-drain output. • Provides a function for disabling (initializing) the SBI.

Note: i = 0 to 3

18.1.1 Internal Block Diagram

Figure 18.1 shows the SBI block diagram.

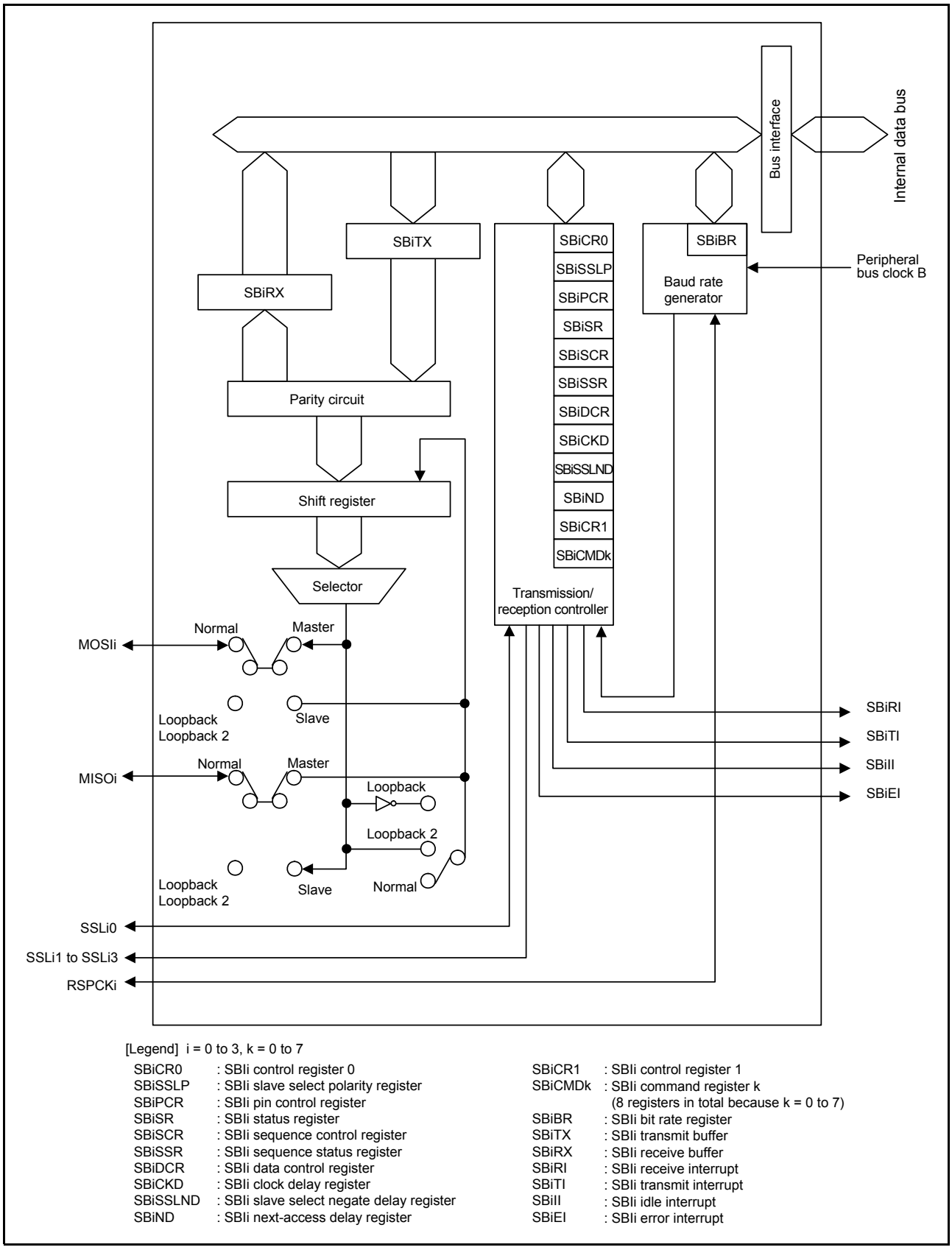


Figure 18.1 SBI Block Diagram for one channel

18.1.2 Pin Configurations

The serial pins listed in Table 18.2 are provided for each channel of the SBI. The SBI automatically switches the input/output direction of the SSLi0 pin (i = 0 to 3). SSLi0 is set as an output when the SBI is a single master and as an input when the SBI is a multi-master or a slave. Pins RSPCK, MOSIi, and MISOi are automatically set as inputs or outputs according to the setting of master or slave and the level input on SSLi0 (refer to section 18.3.2, Controlling SBI Pins).

Table 18.2 SBI I/O Pins

Channel	Pin Name	I/O	Description
Channel 0	RSPCK0	I/O	SBI0 clock input/output
	MOSI0	I/O	SBI0 master transmit data
	MISO0	I/O	SBI0 slave transmit data
	SSL00	I/O	SBI0 slave selection 0
	SSL01	Output	SBI0 slave selection 1
	SSL02	Output	SBI0 slave selection 2
	SSL03*	Output	SBI0 slave selection 3
Channel 1	RSPCK1	I/O	SBI1 clock input/output
	MOSI1	I/O	SBI1 master transmit data
	MISO1	I/O	SBI1 slave transmit data
	SSL10*	I/O	SBI1 slave selection 0
	SSL11*	Output	SBI1 slave selection 1
	SSL12*	Output	SBI1 slave selection 2
	SSL13*	Output	SBI1 slave selection 3
Channel 2	RSPCK2	I/O	SBI2 clock input/output
	MOSI2	I/O	SBI2 master transmit data
	MISO2	I/O	SBI2 slave transmit data
	SSL20	I/O	SBI2 slave selection 0
	SSL21	Output	SBI2 slave selection 1
	SSL22	Output	SBI2 slave selection 2
	SSL23	Output	SBI2 slave selection 3
Channel 3*	RSPCK3	I/O	SBI3 clock input/output
	MOSI3	I/O	SBI3 master transmit data
	MISO3	I/O	SBI3 slave transmit data
	SSL30	I/O	SBI3 slave selection 0
	SSL31	Output	SBI3 slave selection 1
	SSL32	Output	SBI3 slave selection 2
	SSL33	Output	SBI3 slave selection 3

Note: * Not available in the SH72A0 group.

18.2 Registers

Table 18.3 and Table 18.4 list the SBI registers. These registers enable the SBI to perform the following controls: specifying master/slave modes, specifying a transfer format, and controlling the transmitter and receiver.

Table 18.3 SBI Registers (1)

Channel	Register Name	Symbol	After Reset	Address	Access Size
Channel 0	SBI0 control register 0	SB0CR0	H'00	H'FF62 0000	8, 16
	SBI0 slave select polarity register	SB0SSLP	H'00	H'FF62 0001	8, 16
	SBI0 pin control register	SB0PCR	H'00	H'FF62 0002	8, 16
	SBI0 status register	SB0SR	H'20	H'FF62 0003	8, 16
	SBI0 data register	SB0DR	H'0000 0000	H'FF62 0004	16, 32
	SBI0 sequence control register	SB0SCR	H'00	H'FF62 0008	8, 16
	SBI0 sequence status register	SB0SSR	H'00	H'FF62 0009	8, 16
	SBI0 bit rate register	SB0BR	H'FF	H'FF62 000A	8, 16
	SBI0 data control register	SB0DCR	H'00	H'FF62 000B	8, 16
	SBI0 clock delay register	SB0CKD	H'00	H'FF62 000C	8, 16
	SBI0 slave select negate delay register	SB0SSLND	H'00	H'FF62 000D	8, 16
	SBI0 next-access delay register	SB0ND	H'00	H'FF62 000E	8, 16
	SBI0 control register 1	SB0CR1	H'00	H'FF62 000F	8, 16
	SBI0 command register 0	SB0CMD0	H'070D	H'FF62 0010	16
	SBI0 command register 1	SB0CMD1	H'070D	H'FF62 0012	16
	SBI0 command register 2	SB0CMD2	H'070D	H'FF62 0014	16
	SBI0 command register 3	SB0CMD3	H'070D	H'FF62 0016	16
	SBI0 command register 4	SB0CMD4	H'070D	H'FF62 0018	16
	SBI0 command register 5	SB0CMD5	H'070D	H'FF62 001A	16
	SBI0 command register 6	SB0CMD6	H'070D	H'FF62 001C	16
	SBI0 command register 7	SB0CMD7	H'070D	H'FF62 001E	16
Channel 1	SBI1 control register 0	SB1CR0	H'00	H'FF62 0100	8, 16
	SBI1 slave select polarity register	SB1SSLP	H'00	H'FF62 0101	8, 16
	SBI1 pin control register	SB1PCR	H'00	H'FF62 0102	8, 16
	SBI1 status register	SB1SR	H'20	H'FF62 0103	8, 16
	SBI1 data register	SB1DR	H'0000 0000	H'FF62 0104	16, 32
	SBI1 sequence control register	SB1SCR	H'00	H'FF62 0108	8, 16
	SBI1 sequence status register	SB1SSR	H'00	H'FF62 0109	8, 16
	SBI1 bit rate register	SB1BR	H'FF	H'FF62 010A	8, 16
	SBI1 data control register	SB1DCR	H'00	H'FF62 010B	8, 16
	SBI1 clock delay register	SB1CKD	H'00	H'FF62 010C	8, 16
	SBI1 slave select negate delay register	SB1SSLND	H'00	H'FF62 010D	8, 16
	SBI1 next-access delay register	SB1ND	H'00	H'FF62 010E	8, 16
	SBI1 control register 1	SB1CR1	H'00	H'FF62 010F	8, 16
	SBI1 command register 0	SB1CMD0	H'070D	H'FF62 0110	16
	SBI1 command register 1	SB1CMD1	H'070D	H'FF62 0112	16
	SBI1 command register 2	SB1CMD2	H'070D	H'FF62 0114	16
	SBI1 command register 3	SB1CMD3	H'070D	H'FF62 0116	16
	SBI1 command register 4	SB1CMD4	H'070D	H'FF62 0118	16
	SBI1 command register 5	SB1CMD5	H'070D	H'FF62 011A	16

Table 18.4 SBI Registers (2)

Channel	Register Name	Symbol	After Reset	Address	Access Size
Channel 1	SBI1 command register 6	SB1CMD6	H'070D	H'FF62 011C	16
	SBI1 command register 7	SB1CMD7	H'070D	H'FF62 011E	16
Channel 2	SBI2 control register 0	SB2CR0	H'00	H'FF62 0200	8, 16
	SBI2 slave select polarity register	SB2SSLP	H'00	H'FF62 0201	8, 16
	SBI2 pin control register	SB2PCR	H'00	H'FF62 0202	8, 16
	SBI2 status register	SB2SR	H'20	H'FF62 0203	8, 16
	SBI2 data register	SB2DR	H'0000 0000	H'FF62 0204	16, 32
	SBI2 sequence control register	SB2SCR	H'00	H'FF62 0208	8, 16
	SBI2 sequence status register	SB2SSR	H'00	H'FF62 0209	8, 16
	SBI2 bit rate register	SB2BR	H'FF	H'FF62 020A	8, 16
	SBI2 data control register	SB2DCR	H'00	H'FF62 020B	8, 16
	SBI2 clock delay register	SB2CKD	H'00	H'FF62 020C	8, 16
	SBI2 slave select negate delay register	SB2SSLND	H'00	H'FF62 020D	8, 16
	SBI2 next-access delay register	SB2ND	H'00	H'FF62 020E	8, 16
	SBI2 control register 1	SB2CR1	H'00	H'FF62 020F	8, 16
	SBI2 command register 0	SB2CMD0	H'070D	H'FF62 0210	16
	SBI2 command register 1	SB2CMD1	H'070D	H'FF62 0212	16
	SBI2 command register 2	SB2CMD2	H'070D	H'FF62 0214	16
	SBI2 command register 3	SB2CMD3	H'070D	H'FF62 0216	16
	SBI2 command register 4	SB2CMD4	H'070D	H'FF62 0218	16
	SBI2 command register 5	SB2CMD5	H'070D	H'FF62 021A	16
	SBI2 command register 6	SB2CMD6	H'070D	H'FF62 021C	16
	SBI2 command register 7	SB2CMD7	H'070D	H'FF62 021E	16
Channel 3	SBI3 control register 0	SB3CR0	H'00	H'FF62 0300	8, 16
	SBI3 slave select polarity register	SB3SSLP	H'00	H'FF62 0301	8, 16
	SBI3 pin control register	SB3PCR	H'00	H'FF62 0302	8, 16
	SBI3 status register	SB3SR	H'20	H'FF62 0303	8, 16
	SBI3 data register	SB3DR	H'0000 0000	H'FF62 0304	16, 32
	SBI3 sequence control register	SB3SCR	H'00	H'FF62 0308	8, 16
	SBI3 sequence status register	SB3SSR	H'00	H'FF62 0309	8, 16
	SBI3 bit rate register	SB3BR	H'FF	H'FF62 030A	8, 16
	SBI3 data control register	SB3DCR	H'00	H'FF62 030B	8, 16
	SBI3 clock delay register	SB3CKD	H'00	H'FF62 030C	8, 16
	SBI3 slave select negate delay register	SB3SSLND	H'00	H'FF62 030D	8, 16
	SBI3 next-access delay register	SB3ND	H'00	H'FF62 030E	8, 16
	SBI3 control register 1	SB3CR1	H'00	H'FF62 030F	8, 16
	SBI3 command register 0	SB3CMD0	H'070D	H'FF62 0310	16
	SBI3 command register 1	SB3CMD1	H'070D	H'FF62 0312	16
	SBI3 command register 2	SB3CMD2	H'070D	H'FF62 0314	16
	SBI3 command register 3	SB3CMD3	H'070D	H'FF62 0316	16
	SBI3 command register 4	SB3CMD4	H'070D	H'FF62 0318	16
	SBI3 command register 5	SB3CMD5	H'070D	H'FF62 031A	16
	SBI3 command register 6	SB3CMD6	H'070D	H'FF62 031C	16
	SBI3 command register 7	SB3CMD7	H'070D	H'FF62 031E	16

18.2.1 SBIi Control Register 0 (SBiCR0) (i = 0 to 3)

Address SB0CR: H'FF62 0000, SB1CR: H'FF62 0100, SB2CR: H'FF62 0200, SB3CR: H'FF62 0300

	b7	b6	b5	b4	b3	b2	b1	b0
	SPRIE	SPE	SPTIE	SPEIE	MSTR	MODFEN	TXMD	SPMS
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	SPRIE	SBI Receive Interrupt Enable Bit	0: Disables the generation of SBI receive interrupt requests. 1: Enables the generation of SBI receive interrupt requests.	R/W
b6	SPE	SBI Function Enable Bit	0: Disables the SBI function 1: Enables the SBI function	R/W
b5	SPTIE	SBI Transmit Interrupt Enable Bit	0: Disables the generation of SBI transmit interrupt requests. 1: Enables the generation of SBI transmit interrupt requests.	R/W
b4	SPEIE	SBI Error Interrupt Enable Bit	0: Disables the generation of SBI error interrupt requests. 1: Enables the generation of SBI error interrupt requests.	R/W
b3	MSTR	SBI Master/Slave Mode Select Bit	0: Slave mode 1: Master mode	R/W
b2	MODFEN	Mode Fault Error Detection Enable Bit	0: Disables the detection of mode fault error 1: Enables the detection of mode fault error	R/W
b1	TXMD	Communication Operating Mode Select Bit	0: Full-duplex synchronous serial communication 1: Transmit-only operation	R/W
b0	SPMS	SBI Mode Select Bit	0: SBI operation (four-wire method) 1: Clock synchronous operation (three-wire method)	R/W

If the MSTR, MODFEN and TXMD bits are changed while the SBI function is enabled by setting the SPE bit to 1, subsequent operations cannot be guaranteed.

SPRIE Bit

If the SBI has detected a receive buffer write after completion of a serial transfer and the SPRF flag in the SBIi status register (SBiSR) is set to 1, this bit enables or disables the generation of an SBI receive interrupt request.

SPE Bit

Setting this bit to 1 enables the SBI function. When the MODF flag in the SBIi status register (SBiSR) is 1, the SPE bit cannot be set to 1 (refer to section 18.3.8, Error Detection). Setting the SPE bit to 0 disables the SBI function, and initializes a part of the module function (refer to section 18.3.9, Initializing SBI).

SPTIE Bit

Enables or disables the generation of SBI transmit interrupt requests when the SBI detects transmit buffer empty and sets the SPTEF flag in the SBIi status register (SBiSR) to 1.
In the SBI disabled (with the SPE bit 0) status, the SPTEF flag is 1. Therefore, note that setting the SPTIE bit to 1 when the SBI is in the disabled status generates an SBI transmit interrupt request.

SPEIE Bit

Enables or disables the generation of SBI error interrupt requests when the SBI detects a mode fault error and sets the MODF flag in the SBIi status register (SBiSR) to 1, or when the SBI detects an overrun error and sets the OVRF flag in the SBiSR register to 1 (refer to section 18.3.8, Error Detection).

MSTR Bit

Selects master/slave mode of SBI. According to MSTR bit settings, the SBI determines the direction of pins RSPCK, MOSIi, MISOi, and SSLi1 to SSLi3.

MODFEN Bit

Enables or disables the detection of mode fault error (refer to section 18.3.8, Error Detection). In addition, the SBI determines the input/output directions of the SSLi0 pin based on combinations of the MODFEN and MSTR bits (refer to section 18.3.2, Controlling SBI Pins).

TXMD Bit

Selects the full-duplex synchronous serial communication or the transmit-only operation.
When communications are performed with the TXMD bit being 1, only transmissions and no receptions are done (refer to section 18.3.6, Communication Operating Mode).
When the TXMD bit is set to 1, the receive buffer full interrupt request cannot be used.

SPMS Bit

Selects SBI operation (four-wire method) or clock synchronous operation (three-wire method).
The SSLi pins are not used in clock synchronous operation. The three pins RSPCK, MOSIi, and MISOi handle communications. For clock synchronous operation, set the CPHA bit in the SBIi command register k (SBiCMDk) to 1. When the CPHA bit is set to 0, operation is not guaranteed.

18.2.2 SBIi Slave Select Polarity Register (SBiSSLP) (i = 0 to 3)

Address SB0SSLP: H'FF62 0001, SB1SSLP: H'FF62 0101, SB2SSLP: H'FF62 0201, SB3SSLP: H'FF62 0301

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	SSL3P	SSL2P	SSL1P	SSL0P
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R
b3	SSL3P	SSLi3 Signal Polarity Setting Bit	0: SSLi3 signal low-active 1: SSLi3 signal high-active	R/W
b2	SSL2P	SSLi2 Signal Polarity Setting Bit	0: SSLi2 signal low-active 1: SSLi2 signal high-active	R/W
b1	SSL1P	SSLi1 Signal Polarity Setting Bit	0: SSLi1 signal low-active 1: SSLi1 signal high-active	R/W
b0	SSL0P	SSLi0 Signal Polarity Setting Bit	0: SSLi0 signal low-active 1: SSLi0 signal high-active	R/W

If the contents of the SBiSSLP register are changed by the CPU while the SBI function is enabled by setting the SPE bit in the SBIi control register 0 (SBiCR0) to 1, subsequent operations are not guaranteed.

SSL3P Bit

This bit set the polarity of the SSLi3 signal.

SSL2P Bit

This bit set the polarity of the SSLi2 signal.

SSL1P Bit

This bit set the polarity of the SSLi1 signal.

SSL0P Bit

This bit set the polarity of the SSLi0 signal.

18.2.3 SBIi Pin Control Register (SBiPCR) (i = 0 to 3)

Address SB0PCR: H'FF62 0002, SB1PCR: H'FF62 0102, SB2PCR: H'FF62 0202, SB3PCR: H'FF62 0302

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	MOIFE	MOIFV	—	SPOM	SPLP2	SPLP
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R
b5	MOIFE	MOSI Idle Value Fixing Enable Bit	0: MOSIi output value equals final data from previous transfer 1: MOSIi output value equals the value set in the MOIFV bit	R/W
b4	MOIFV	MOSI Idle Fixed Value	0: MOSIi Idle fixed value equals 0 1: MOSIi Idle fixed value equals 1	R/W
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R
b2	SPOM	SBI Output Pin Mode Bit	0: CMOS output 1: Open-drain output	R/W
b1	SPLP2	SBI Loopback 2 Bit	0: Normal mode 1: Loopback mode (transmit data = receive data)	R/W
b0	SPLP	SBI Loopback Bit	0: Normal mode 1: Loopback mode (transmit data inverted = receive data)	R/W

If the contents of this register is changed by the CPU while the SBI function is enabled by setting the SPE bit in the SBIi control register 0 (SBiCR0) to 1, subsequent operations cannot be guaranteed.

MOIFE Bit

Fixes the MOSIi output value when the SBI in master mode is in an SSL negation period *Note:. When MOIFE is 0, the SBI outputs the last data from the previous serial transfer during the SSL negation period. When the MOIFE bit is 1, the SBI outputs the fixed value set in the MOIFV bit to the MOSIi pin.

MOIFV Bit

If the MOIFE bit is 1 in master mode, the SBI, according to MOIFV bit settings, determines the MOSIi signal value during the SSL negation period *.

SPOM Bit

Sets the SBI output pins to CMOS output/open drain output.

SPLP2 Bit

When the SPLP2 bit is set to 1, the SBI shuts off the path between the MISOi pin and the shift register, and between the MOSIi pin and the shift register, and connects (reverses) the input path and the output path for the shift register (loopback mode).

SPLP Bit

When the SPLP bit is set to 1, the SBI shuts off the path between the MISOi pin and the shift register, and between the MOSIi pin and the shift register, and connects (reverses) the input path and the output path for the shift register (loopback mode).

Note: * Including the SSL retention period during a burst transfer.

18.2.4 SBIi Status Register (SBiSR) (i = 0 to 3)

Address SB0SR: H'FF62 0003, SB1SR: H'FF62 0103, SB2SR: H'FF62 0203, SB3SR: H'FF62 0303

	b7	b6	b5	b4	b3	b2	b1	b0
	SPRF	—	SPTEF	—	PERF	MODF	IDLNF	OVRF
After Reset	0	0	1	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	SPRF	SBI Receive Buffer Full Flag	0: No receive valid data in the SBiDR register 1: Receive valid data found in the SBiDR register	R/W *
b6	—	Reserved	This bit is read as 0. The write value should be 0.	R
b5	SPTEF	SBI Transmit Buffer Empty Flag	0: Data found in the transmit buffer 1: No data in the transmit buffer	R/W *
b4	—	Reserved	This bit is read as 0. The write value should be 0.	R
b3	PERF	Parity Error Flag	0: No parity error 1: A parity error	R/W *
b2	MODF	Mode Fault Error Flag	0: No mode fault error occurs 1: A mode fault error occurs	R/W *
b1	IDLNF	SBI Idle Flag	0: SBI idle state 1: SBI transfer state	R
b0	OVRF	Overrun Error Flag	0: No overrun error occurs 1: An overrun error occurs	R/W *

Note: * Only 0 can be written to clear the flag after reading 1.

Writing to the SBiSR register can only be performed by the CPU under certain conditions.

SPRF Flag

Indicates the status of the receive buffer for the SBIi data register (SBiDR). Upon completion of a serial transfer with the communication operating mode select bit (TXMD) in the SBIi control register 0 (SBiCR0) and the SPRF flag set to 0, the SBI transfers the receive data from the shift register to the SBiDR register, and sets this bit to 1. As the SBI handles full-duplex synchronous serial communications while the TXDM bit is 0, this means that the last bit of transmit data has been transmitted. The SPRF flag is cleared to 0 under the following conditions:

- The CPU reads the SBiSR register when the SPRF flag is 1, and then the CPU writes a 0 to the SPRF flag.
- Received data is read from the SBiDR register.
- The reset is performed.

If a serial transfer ends while the SPRF flag is 1, the SBI does not transfer the received data from the shift register to the SBiDR register. When the OVRF flag in the SBiSR register is 1, the SPRF flag cannot be changed from 0 to 1 (refer to section 18.3.8, Error Detection).

SPTEF Flag

Indicates the status of the transmit buffer for the SBIi data register (SBiDR). After the initialization of SBI or after transmit data is transferred from the transmit buffer to the shift register, the SBI sets the SPTEF flag to 1. The SPTEF flag is cleared to 0 under the following conditions. If the SPTEF flag is cleared and the shift register is empty, the data is copied from the transmit buffer to the shift register.

- The CPU reads the SBiSR register when the SPTEF flag is 1, and then the CPU writes 0 to the SPTEF flag.
- The transmit data is written to the SBiDR register.

Data can be written to the SBiDR register only when the SPTEF flag is 1. If data is written to the transmit buffer of the SBiDR register when the SPTEF flag is 0, the data in the transmit buffer is not updated.

PERF Flag

Indicates the occurrence of a parity error. If a serial transfer ends while the TXMD bit in the SBiCR0 register is 0 and the SPPE bit in the SBiCR1 register is 1, the SBI detects a parity error. When a parity error occurs, the PERF flag is set to 1. The PERF flag is cleared to 0 under the following conditions.

- The CPU reads the SBiSR register while the PERF flag is 1, and then writes a 0 to the PERF flag.
- The reset is performed.

MODF Flag

Indicates the occurrence of a mode fault error. When the input level of the SSLi0 pin changes to the active level while the MSTR bit in the SBIi control register 0 (SBiCR0) is 1 and the MODFEN bit is 1 with the SBI being in multi-master mode, the SBI detects a mode fault error and sets the MODF flag to 1. Similarly, if the MODFEN bit is set to 1 when the MSTR bit is 0 and the SBI is in slave mode, and the SSLi0 pin is negated before the RSPCK cycle necessary for data transfer ends, the SBI detects a mode fault error. The active level of the SSLi0 signal is determined by the SSL0P bit in the SBIi slave select polarity register (SBiSSLP). The MODF flag is cleared to 0 under the following conditions.

- The CPU reads the SBiSR register when the MODF flag is 1, and then writes 0 to the MODF flag.
- The reset is performed.

IDLNF Flag

Indicates the SBI transfer status. This flag is cleared to 0 with the SBI not transferred in master mode, because the SPCP bits in the SBIi sequence status register (SBiSSR) are set to 000 and at the head of the sequence, and the data to be transferred next is not set with the SPTEF flag in the SBiSR register being 1.

Both in master mode and slave mode, this flag is cleared to 0 when the SPE bit in the SBiCR0 register is 0 and the SBI function is disabled.

In master mode (single-master mode/multi-master mode), the IDLNF flag is cleared to 0 under the following conditions:

- The SPE bit in the SBiCR0 register is 0 (the SBI initialization) or the SPTEF flag in the SBiSR register is 1 (the data to be transferred next is not set).
- The SPCP bits in the SBiSSR register is 000 (the sequence control is at the first command pointer of the loop).
- The SBI internal sequencer is changed to be in the idle state (the state in which the operations up to the next access delay are completed).

The IDLNF flag is cleared to 0 when all the above conditions are met.

The IDLNF flag is set when the above conditions are not met.

In slave mode, the IDLNF flag is cleared to 0 under the following condition:

- The SPE bit in the SBiCR0 register is 0 (the SBI initialization)

The IDLNF flag is set to 1 when the SPE bit is set to 1.

OVRF Flag

Indicates the occurrence of an overrun error. If a serial transfer ends while the communication operating mode select bit (TXMD) in the SBIi control register 0 (SBiCR0) is 0 and the SPRF flag is 1, the SBI detects an overrun error, and sets the OVRF flag to 1. The OVRF flag is cleared to 0 under the following conditions.

- The CPU reads the SBiSR register when the OVRF flag is 1, and then writes 0 to the OVRF flag.
- The reset is performed.

18.2.5 SBIi Data Register (SBiDR) (i = 0 to 3)

Address SB0DR: H'FF62 0004, SB1DR: H'FF62 0104, SB2DR: H'FF62 0204, SB3DR: H'FF62 0304

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
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
After Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Description	R/W
b31 to b0	This register stores the SBI transmit/receive data.	R/W

The SBIi data register (SBiDR) is a buffer that holds data for transmission and reception by the SBI.

The transmit buffer (SBiTX) and receive buffer (SBiRX) are independent and are mapped to the SBiDR register.

When reading from or writing to the SBiDR register, set the SBI longword access/word access specification bit (SPLW) in the SBIi data control register (SBiDCR) in longword or word units. When the SPLW bit is 0, the SBiDR register is a 64-bit buffer that consists of up to four 16-bit frames. When the SPLW bit is 1, the SBiDR register is a 128-bit buffer that consists of up to four 32-bit frames.

The frame length used by the SBiDR register is determined by the number of frames specification bits (SPFC) in the SBIi data control register (SBiDCR), and the bit length to be used is determined by the SBI data length specification bits (SPB) in the SBIi command register k (SBiCMDk).

If the SPTEF flag in the SBIi status register (SBiSR) is 1 when data is written to the SBiDR register, the SBI allows writing of the data to the SBiDR register transmit buffer. If the SPTEF flag is 0, the SBI does not allow updating of the SBiDR register transmit buffer.

If the SBI receive/transmit data selection bit (SPRDTD) in the SBIi data control register (SBiDCR) is 0 when data is read from the SBiDR register, the SBI allows reading of the receive buffer. If the SPRDTD bit is 1, the SBI allows reading of the transmit buffer.

When reading from the transmit buffer, the value written to the buffer immediately before the read operation is read. All the read value are 0 while the SPTEF flag in the SBIi status register (SBiSR) is 0.

In normal operation method, data is read from the receive buffer when the SPRDTD bit is 0, and the SPRF flag in the SBiSR register is 1 (a condition in which the receive buffer holds data that has not yet been read out). When the SPRF flag or the OVRF flag in the SBiSR register is 1, the SBI does not update the receive buffer of the SBiDR register at the end of a serial transfer.

18.2.6 SBIi Sequence Control Register (SBiSCR) (i = 0 to 3)

Address SB0SCR: H'FF62 0008, SB1SCR: H'FF62 0108, SB2SCR: H'FF62 0208, SB3SCR: H'FF62 0308

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	SPSLN[2:0]		
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R
b2 to b0	SPSLN[2:0]	SBI Sequence Length Specification Bits	Refer to the table below.	R/W

The SBIi sequence control register (SBiSCR) sets the sequence control method when the SBI operates in master mode. When the CPU rewrites the SPSLN bits in the SBiSCR register with the MSTR and SPE bits in the SBIi control register 0 (SBiCR0) being 1 and the SBI function in master mode being enabled, rewrite the SPSLN bits with the IDLNF flag in the SBIi status register (SBiSR) being 0.

SPSLN Bit

These bits specify a sequence length when the SBI in master mode performs sequential operations. The SBI in master mode changes SBIi command registers 0 to 7 (SBiCMD0 to SBiCMD7) to be referenced and the order in which they are referenced according to the sequence length that is set in the SPSLN bits. The relationship among the setting of the SPSLN bits, sequence length, and the SBiCMD0 to SBiCMD7 registers referenced by the SBI is shown below.

SPSLN	Sequence Length	Referenced SBiCMDk Register NO.
000	1	0 → 0 → ...
001	2	0 → 1 → 0 → ...
010	3	0 → 1 → 2 → 0 → ...
011	4	0 → 1 → 2 → 3 → 0 → ...
100	5	0 → 1 → 2 → 3 → 4 → 0 ...
101	6	0 → 1 → 2 → 3 → 4 → 5 → 0 ...
110	7	0 → 1 → 2 → 3 → 4 → 5 → 6 → 0 ...
111	8	0 → 1 → 2 → 3 → 4 → 5 → 6 → 7 → 0 ...

Note: i = 0 to 3, k = 0 to 7

18.2.7 SBIi Sequence Status Register (SBiSSR) (i = 0 to 3)

Address SB0SSR: H'FF62 0009, SB1SSR: H'FF62 0109, SB2SSR: H'FF62 0209, SB3SSR: H'FF62 0309

	b7	b6	b5	b4	b3	b2	b1	b0
	—	SPECM [2:0]			—	SPCP [2:0]		
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R
b6 to b4	SPECM [2:0]	SBI Error Command Bits	b6 b5 b4 0 0 0 : SBiCMD0 0 0 1 : SBiCMD1 0 1 0 : SBiCMD2 0 1 1 : SBiCMD3 1 0 0 : SBiCMD4 1 0 1 : SBiCMD5 1 1 0 : SBiCMD6 1 1 1 : SBiCMD7	R
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R
b2 to b0	SPCP [2:0]	SBI Command Pointer Bits	b2 b1 b0 0 0 0 : SBiCMD0 0 0 1 : SBiCMD1 0 1 0 : SBiCMD2 0 1 1 : SBiCMD3 1 0 0 : SBiCMD4 1 0 1 : SBiCMD5 1 1 0 : SBiCMD6 1 1 1 : SBiCMD7	R

The SBIi sequence status register (SBiSSR) indicates the sequence control status when the SBI is in the master operation. The write into the SBiSSR register from the CPU is ignored.

SPECM Bit

These bits indicate SBIi command register k (SBiCMDk) that are pointed to by command pointers (the SPCP bits) when an error is detected during sequence control by the SBI. The SBI updates the SPECM bits only when an error is detected. If the OVRF, MODF, and PERF flags in the SBIi status register (SBiSR) are 0 and there is no error, the values of the SPECM bits have no meaning.

For the SBI's error detection function, refer to section 18.3.8, Error Detection. For the SBI's sequence control, refer to section 18.3.10, SBI Operations (1), Master Mode Operation.

SPCP Bit

During SBI sequence control, these bits indicate SBIi command register k (SBiCMDk), which are currently pointed to by the pointers.

For the SBI's sequence control, refer to section 18.3.10, SBI Operations (1), Master Mode Operation.

18.2.8 SBIi Bit Rate Register (SBiBR) (i = 0 to 3)

Address SB0BR: H'FF62 000A, SB1BR: H'FF62 010A, SB2BR: H'FF62 020A, SB3BR: H'FF62 030A

	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	1	1	1	1	1	1	1	1

Bit	Description	R/W
b7 to b0	This register sets the bit rate in master mode.	R/W

The SBIi bit rate register (SBiBR) sets the bit rate in master mode. If the contents of the SBiBR register is changed by the CPU while the MSTR and SPE bits in the SBIi control register 0 (SBiCR0) are 1 with the SBI function in master mode enabled, operation cannot be guaranteed.

When the SBI is used in slave mode, the bit rate depends on the bit rate of the input clock regardless of the settings of the SBiBR register and the BRDV bit in the SBIi command register k (SBiCMDk). (set the bit rate which meets electrical characteristics.)

The bit rate is determined by combinations of the SBiBR register settings and the BRDV bit settings. The equation for calculating the bit rate is given below. In the equation, n denotes an the SBiBR register setting (0, 1, 2, ..., 255), and N denotes bit settings in the BRDV bits.

$$\text{Bit rate} = \frac{f(PBB)}{2 \times (n + 1) \times 2^{N}}$$

Note: * f (PBB): Peripheral bus clock B

Table 18.5 lists Relationship between SBiBR and BRDV Settings.

Table 18.5 Relationship between SBiBR and BRDV Settings

SBiBR(n)	BRDV(N)	Division Ratio	Bit Rate	
			f(PBB) = 32 MHZ	f(PBB) = 40 MHZ
0	0	2	16.0 Mbps *	20.0 Mbps *
1	0	4	8.00 Mbps	10.0 Mbps
2	0	6	5.33 Mbps	6.67 Mbps
3	0	8	4.00 Mbps	5.00 Mbps
4	0	10	3.20 Mbps	4.00 Mbps
5	0	12	2.67 Mbps	3.33 Mbps
5	1	24	1.33 kbps	1.67 Mbps
5	2	48	677 kbps	833 kbps
5	3	96	333 kbps	417 kbps
255	3	4096	7.81 kbps	9.78 kbps

Note: * Cannot be set in this LSI.

18.2.9 SBIi Data Control Register (SBiDCR) (i = 0 to 3)

Address SB0DCR: H'FF62 000B, SB1DCR: H'FF62 010B, SB2DCR: H'FF62 020B, SB3DCR: H'FF62 030B

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	SPLW	SPRDTD	SLSEL[1:0]		SPFC[1:0]	
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R
b5	SPLW	SBI Longword Access/ Word Access Specification Bit	0: The SBiDR register is accessed in words. 1: The SBiDR register is accessed in longwords.	R/W
b4	SPRDTD	SBI Receive/ Transmit Data Selection Bit	0: The SBiDR register reads the receive buffer. 1: The SBiDR register reads the transmit buffer (only if the SPTEF flag is 1).	R/W
b3, b2	SLSEL[1:0]	SSL Pin Output Selection Bits	b3 b2 0 0 : SSL output at pins SSLi3 to SSLi0 0 1 : SSL output at the SSLi0 pin Pins SSLi3 to SSLi1 can be used as I/O ports 1 0 : SSL output at pins SSLi1 and SSLi0 Pins SSLi3 and SSLi2 can be used as I/O ports 1 1 : Setting prohibited	R/W
b1, b0	SPFC[1:0]	Number of Frames Specification Bits	Refer to Table 18.6.	R/W

The SBIi data control register (SBiDCR) is a register used to set the number of frames that can be stored in the SBiDR register and reading from the SBiDR register, and select the width (longword or word) for access to the SBiDR register.

By the combination of the SBI data length specification bits (SPB) in the SBIi command register k (SBiCMDk), the SBI sequence length specification bits (SPSLN) in the SBIi sequence control register (SBiSCR), and the number of frames specification bits (SPFC) in the SBIi data control register (SBiDCR), up to four frames can be transmitted or received in one round of transmission or reception activation.

When the CPU rewrites the SPFC bits in the SBiDCR register with the SPE bit in the SBIi control register 0 (SBiCR0) being 1 and the SBI function being enabled, rewrite the SPFC bits with the IDLNF flag in the SBIi status register (SBiSR) being 0.

SPLW Bit

The SBI longword access/word access specification bit (SPLW) sets the width for the access to the SBIi data register (SBiDR). Access the SBiDR register in words when the SPLW bit is 0 and access the SBiDR register in longwords when the SPLW bit is 1.

When the SPLW bit is 0, set the SBI data length specification bits (SPB) in the SBIi command register k (SBiCMDk) to between 8 and 16 bits. Operations cannot be guaranteed when the bits are set to 20, 24, or 32 bits.

SPRDTD Bit

The SBI receive/transmit data selection bit (SPRDTD) selects whether the value which the SBIi data register (SBiDR) reads is the receive buffer or the transmit buffer. When the transmit buffer is read, the very last value written into the SBiDR register is read. Read the transmit buffer when the SPTEF flag in the SBIi status register (SBiSR) is 1.

SLSEL Bit

These bits control the SSLi pin output in master mode. The pins unused can be used as I/O ports.

SPFC Bit

These bits specify the number of frames that can be stored in the SBiDR register. By the combination of the SBI data length specification bits (SPB) in the SBIi command register k (SBiCMDk), the SBI sequence length specification bits (SPSLN) in the SBIi sequence control register (SBiSCR), and the number of frames specification bits (SPFC) in the SBIi data control register (SBiDCR), up to four frames can be transmitted or received in one round of transmission or reception. Also, the SPFC bits specify the number of received data at which the SBI receive buffer full flag (SPRF) in the SBIi status register (SBiSR) is set. Table 18.6 and the figure below show the frame configurations that can be stored in the SBiDR register and examples of combinations of settings for transmission and reception. If combinations of settings other than those shown in the examples are made, subsequent operation is not guaranteed.

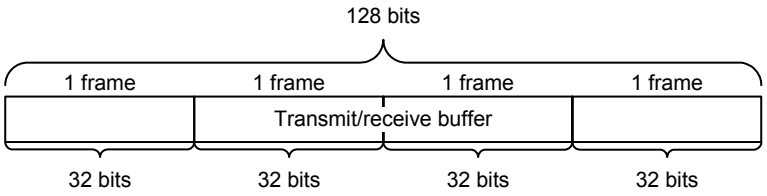


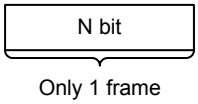
Table 18.6 Frame Settings by All Relevant Bits

Setting	SPB	SPSLN	SPFC	Number of frames for transfer	Number of frames at which SPRF is set to 1 and SPRF is cleared to 0
1-1	N	000	00	1	1
1-2	N	000	01	2	2
1-3	N	000	10	3	3
1-4	N	000	11	4	4
2-1	N, M	001	01	2	2
2-2	N, M	001	11	4	4
3	N, M, O	010	10	3	3
4	N, M, O, P	011	11	4	4
5	N, M, O, P, Q	100	00	5	1
6	N, M, O, P, Q, R	101	00	6	1
7	N, M, O, P, Q, R, S	110	00	7	1
8	N, M, O, P, Q, R, S, T	111	00	8	1

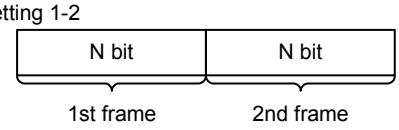
Note: N, M, O, P, Q, R, S, T: Data length that can be specified by SPB

When each activation is done in the above settings 1-1 to 8, data can be transmitted/received as follow.

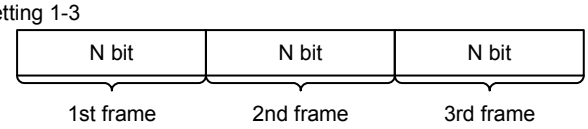
Setting 1-1



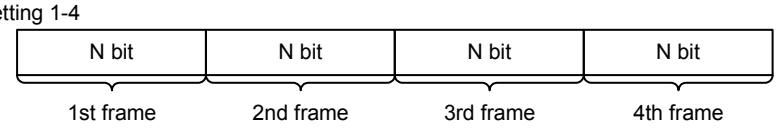
Setting 1-2



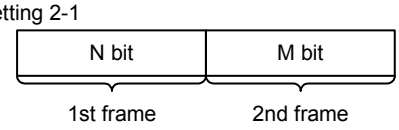
Setting 1-3



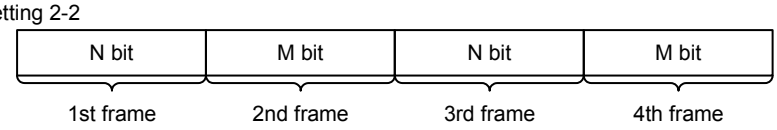
Setting 1-4



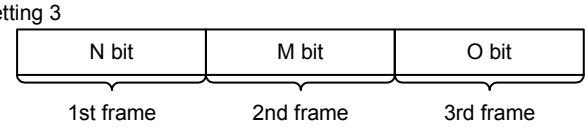
Setting 2-1



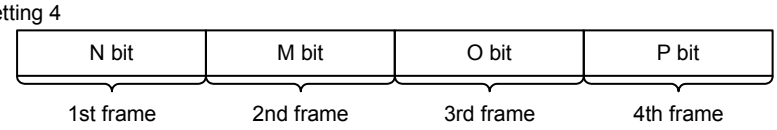
Setting 2-2



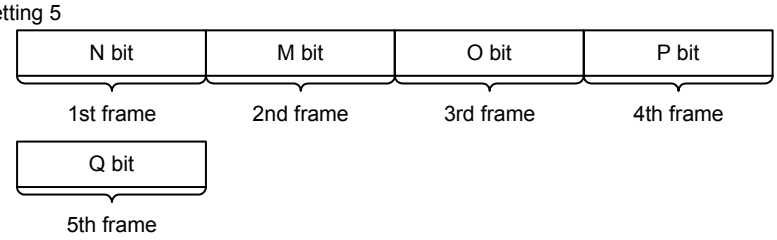
Setting 3

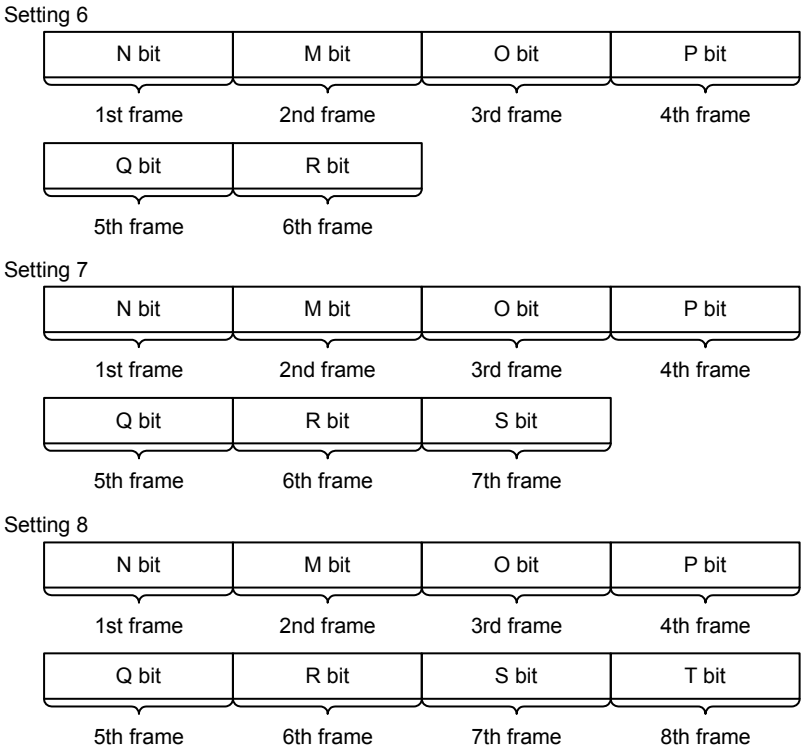


Setting 4



Setting 5




Setting 6
N bit
M bit
O bit
P bit
1st frame
2nd frame
3rd frame
4th frame
Q bit
R bit
5th frame
6th frame
Setting 7
N bit
M bit
O bit
P bit
1st frame
2nd frame
3rd frame
4th frame
Q bit
R bit
S bit
5th frame
6th frame
7th frame
Setting 8
N bit
M bit
O bit
P bit
1st frame
2nd frame
3rd frame
4th frame
Q bit
R bit
S bit
T bit
5th frame
6th frame
7th frame
8th frame

18.2.10 SBIi Clock Delay Register (SBiCKD) (i = 0 to 3)

Address SB0CKD: H'FF62 000C, SB1CKD: H'FF62 010C, SB2CKD: H'FF62 020C, SB3CKD: H'FF62 030C

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	SCKDL [2:0]		
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R
b2 to b0	SCKDL [2:0]	RSPCK Delay Setting Bits	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

The SBIi clock delay register (SBiCKD) sets a period from the beginning of SSL signal assertion to RSPCK oscillation (RSPCK delay) when the SCKDEN bit in the SBIi command register k (SBiCMDk) is 1. If the contents of the SBiCKD register is changed by the CPU while the MSTR and SPE bits in the SBIi control register 0 (SBiCR0) are 1 with the SBI function in master mode enabled, operation cannot be guaranteed.
When using the SBI in slave mode, set B'000 to the SCKDL bits.

SCKDL Bit

These bits set an RSPCK delay value when the SCKDEN bit in the SBiCMDk register is 1.

18.2.11 SBIi Slave Select Negate Delay Register (SBiSSLND) (i = 0 to 3)

Address SB0SSLND: H'FF62 000D, SB1SSLND: H'FF62 010D, SB2SSLND: H'FF62 020D, SB3SSLND: H'FF62 030D

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	SLNDL [2:0]		
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R
b2 to b0	SLNDL [2:0]	SSL Negation Delay Setting Bits	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

The SBIi slave select negate delay register (SBiSSLND) sets a period (SSL negation delay) from the transmission of a final RSPCK edge to the negation of the SSL signal during a serial transfer by the SBI in master mode. If the contents of the SBiSSLND register is changed by the CPU while the MSTR and SPE bits in the SBIi control register 0 (SBiCR0) are 1 with the SBI function in master mode enabled, operation cannot be guaranteed.
When using the SBI in slave mode, set B'000 to the SLNDL bits.

SLNDL Bit

These bits set an SSL negation delay value when the SBI is in master mode.

18.2.12 SBIi Next-Access Delay Register (SBiND) (i = 0 to 3)

Address SB0ND: H'FF62 000E, SB1ND: H'FF62 010E, SB2ND: H'FF62 020E, SB3ND: H'FF62 030E

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	SPNDL [2:0]		
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R
b2 to b0	SPNDL [2:0]	SBI Next-Access Delay Setting Bits	b2 b1 b0 0 0 0 : 1 RSPCK + 2 cycles of the peripheral bus clock B 0 0 1 : 2 RSPCK + 2 cycles of the peripheral bus clock B 0 1 0 : 3 RSPCK + 2 cycles of the peripheral bus clock B 0 1 1 : 4 RSPCK + 2 cycles of the peripheral bus clock B 1 0 0 : 5 RSPCK + 2 cycles of the peripheral bus clock B 1 0 1 : 6 RSPCK + 2 cycles of the peripheral bus clock B 1 1 0 : 7 RSPCK + 2 cycles of the peripheral bus clock B 1 1 1 : 8 RSPCK + 2 cycles of the peripheral bus clock B	R/W

The SBIi next-access delay register (SBiND) sets a non-active period (next-access delay) after termination of a serial transfer when the SPNDEN bit in the SBIi command register k (SBiCMDk) is 1. If the contents of the SBiND register is changed by the CPU while the MSTR and SPE bits in the SBIi control register 0 (SBiCR0) are 1 with the SBI function in master mode enabled, operation cannot be guaranteed.

When using the SBI in slave mode, set B'000 to the SPNDL bits.

SPNDL Bit

These bits set a next-access delay when the SPNDEN bit in SBiCMDk is 1.

18.2.13 SBIi Control Register 1 (SBiCR1) (i = 0 to 3)

Address SB0CR2: H'FF62 000F, SB1CR2: H'FF62 010F, SB2CR2: H'FF62 020F, SB3CR2: H'FF62 030F

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	PTE	SPIIE	SPOE	SPPE
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R
b3	PTE	Parity Self-diagnosis Bit	0: Self-diagnosis function of the parity circuit disabled 1: Self-diagnosis function of the parity circuit enabled	R/W
b2	SPIIE	SBI Idle Interrupt Enable Bit	0: Idle interrupt request generation disabled 1: Idle interrupt request generation enabled	R/W
b1	SPOE	Parity Mode Bit	0: Selects the even parity for transmission/reception. 1: Selects the odd parity for transmission/reception.	R/W
b0	SPPE	Parity Enable Bit	0: The parity bit is not appended to the transmit data and the parity check is not executed in the receive data. 1: When the TXMD bit is 0, the parity bit is appended to the transmit data and the parity check is executed in the receive data. When the TXMD bit is 1, the parity bit is appended to the transmit data and the parity check is not executed in the receive data.	R/W

If the setting values of bits SPPE and SPOE are changed while the SBI function is enabled by setting the SPE bit to 1, subsequent operations are not guaranteed.

PTE Bit

This bit enables the self-diagnosis of the parity circuit to confirm that parity functions work correctly.

SPIIE Bit

This bit detects the SBI which is in the idle state, and enables/disables the SBI idle interrupt request generation when the IDLNF flag in the SBIi status register (SBiSR) is set to 0.

SPOE Bit

When selecting even parity, set this bit so that the combined parity bit and transmit/receive character totals an even number of 1s.

Likewise, when selecting odd parity, set this bit so that the combined parity bit and transmit/receive character totals an odd number of 1s.

The SPOE bit is enabled only when the SPPE bit in the SBiCR1 register is 1.

SPPE Bit

This bit selects whether parity functions are enabled or disabled.

When the communication operating mode select bit (TXMD) in the SBIi control register 0 (SBiCR0) is 0 and the SPPE bit is 1, the parity bit is appended to the transmit data and the parity check is executed in the receive data.

When the TXMD bit is 1 and the SPPE bit is 1, the parity bit is appended to the transmit data and the parity check is not executed in receive data.

18.2.14 SBIi Control Register k (SBiCMDk) (i = 0 to 3; k = 0 to 7)

Address

SB0CMD0: H'FF62 0010, SB0CMD1: H'FF62 0012, SB0CMD2: H'FF62 0014, SB0CMD3: H'FF62 0016,
SB0CMD4: H'FF62 0018, SB0CMD5: H'FF62 001A, SB0CMD6: H'FF62 001C, SB0CMD7: H'FF62 001E
SB1CMD0: H'FF62 0110, SB1CMD1: H'FF62 0112, SB1CMD2: H'FF62 0114, SB1CMD3: H'FF62 0116,
SB1CMD4: H'FF62 0118, SB1CMD5: H'FF62 011A, SB1CMD6: H'FF62 011C, SB1CMD7: H'FF62 011E
SB2CMD0: H'FF62 0210, SB2CMD1: H'FF62 0212, SB2CMD2: H'FF62 0214, SB2CMD3: H'FF62 0216,
SB2CMD4: H'FF62 0218, SB2CMD5: H'FF62 021A, SB2CMD6: H'FF62 021C, SB2CMD7: H'FF62 021E
SB3CMD0: H'FF62 0310, SB3CMD1: H'FF62 0312, SB3CMD2: H'FF62 0314, SB3CMD3: H'FF62 0316,
SB3CMD4: H'FF62 0318, SB3CMD5: H'FF62 031A, SB3CMD6: H'FF62 031C, SB3CMD7: H'FF62 031E

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB[3:0]				SSLKP	SSLA[2:0]			BRDV[1:0]		CPOL	CPHA
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
b15	SCKDEN	RSPCK Delay Setting Enable Bit	0: An RSPCK delay of 1 RSPCK 1: An RSPCK delay is equal to the SBiCKD register settings.	R/W
b14	SLNDEN	SSL Negation Delay Setting Enable Bit	0: An SSL negation delay of 1 RSPCK 1: An SSL negation delay is equal to the SBiSSLND register settings.	R/W
b13	SPNDEN	SBI Next-Access Delay Enable Bit	0: A next-access delay of 1 RSPCK + 2 cycles of the peripheral bus clock B 1: A next-access delay is equal to the SBiND register settings.	R/W
b12	LSBF	SBI LSB First Bit	0: MSB first 1: LSB first	R/W
b11 to b8	SPB[3:0]	SBI Data Length Specification Bits 0	b11 b10 b9 b8 0 1 0 0 to 0 1 1 1: 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 0 0 1 0 , 0 0 1 1: 32 bits	R/W
b7	SSLKP	SSL Signal Level Keeping Bit	0: Negates all SSL signals upon completion of transfer. 1: Keeps the SSL signal level from the end of the transfer until the beginning of the next access.	R/W
b6 to b4	SSLA[2:0]	SSL Signal Assertion Setting Bits	b6 b5 b4 0 0 0 : SSLi0 0 0 1 : SSLi1 0 1 0 : SSLi2 0 1 1 : SSLi3 1 0 0 to 1 1 1: Setting prohibited	R/W
b3, b2	BRDV[1:0]	Bit Rate Division Setting Bits	b3 b2 0 0 : Select the base bit rate 0 1 : Select the base bit rate divided by 2 1 0 : Select the base bit rate divided by 4 1 1 : Select the base bit rate divided by 8	R/W
b1	CPOL	RSPCK Polarity Setting Bit	0: RSPCK = 0 when idle 1: RSPCK = 1 when idle	R/W
b0	CPHA	RSPCK Phase Setting Bit	0: Data sampling on odd edge, data variation on even edge 1: Data variation on odd edge, data sampling on even edge	R/W

Each channel has eight SBIi command registers k (SBiCMDk) are used to set a transfer format for the SBI in master mode. Some of the bits in the SBiCMD0 register is used to set a transfer mode for the SBI in slave mode. The SBI in master mode sequentially references the SBiCMDk register according to the settings in bits the SPSLN bits in the SBIi sequence control register (SBiSCR), and executes the serial transfer that is set in the referenced SBiCMDk register.
Before the SBiCMDk register is referred and the data to be transmitted are set, set the SBiCMDk register with the SPTEF flag in the SBIi status register (SBiSR) being 1.
The SBiCMDk register that is referenced by the SBI in master mode can be checked by means of the SPCP bits in the SBIi sequence status register (SBiSSR). When the SBI function in slave mode is enabled, operation cannot be guaranteed if the value set in the SBiCMDk register is changed by the CPU.

SCKDEN Bit

Sets the period from the point when the SBI in master mode activates the SSL signal until the RSPCK starts oscillation (RSPCK delay). If the SCKDEN bit is 0, the SBI sets the RSPCK delay to 1 RSPCK. If the SCKDEN bit is 1, the SBI starts the oscillation of RSPCK at an RSPCK delay in compliance with SBIi clock delay register (SBiCKD) settings.
To use the SBI in slave mode, the SCKDEN bit should be set to 0.

SLNDEN Bit

Sets the period (SSL negation delay) from the time the master mode SBI stops RSPCK oscillation until the SBI sets the SSL signal inactive.
If the SLNDEN bit is 0, the SBI sets the RSPCK delay to 1 RSPCK. If the SLNDEN bit is 1, the SBI negates the SSL signal at an SSL negation delay in compliance with SBIi slave select negation delay register (SBiSSLND) settings.
To use the SBI in slave mode, the SLNDEN bit should be set to 0.

SPNDEN Bit

Sets the period from the time the SBI in master mode terminates a serial transfer and sets the SSL signal inactive until the SBI enables the SSL signal assertion for the next access (next-access delay). If the SPNDEN bit is 0, the SBI sets the next-access delay to 1 RSPCK + 2 cycles of the peripheral bus clock B. If the SPNDEN bit is 1, the SBI inserts a next-access delay in compliance with SBIi next-access delay register (SBiND) settings.
To use the SBI in slave mode, the SPNDEN bit should be set to 0.

LSBF Bit

Sets the data format of the SBI in master mode or slave mode to MSB first or LSB first.

SPB Bit

These bits set a transfer data length for the SBI in master mode or slave mode.

SSLKP Bit

When the SBI in master mode performs a serial transfer, this bit specifies whether the SSL signal level for the current command is to be kept or negated between the SSLi negation timing associated with the current command and the SSLi assertion timing associated with the next command.
To use the SBI in slave mode, the SSLKP bit should be set to 0.

SSLA Bit

These bits control the SSL signal assertion when the SBI performs serial transfers in master mode. Setting the SSLA bits controls the assertion for the signals SSLi3 to SSLi0. When an SSL signal is asserted, its polarity is determined by the set value in the corresponding SBiSSLP (SBIi slave select polarity register). When the SSLA bits are set to B'000 or B'100 to B'111 in multi-master mode, serial transfers are performed with all the SLL signals in the negated state (as SSLi0 acts as input). Also, when the SSLA bits are set to between 100 and 111 in single-master mode, serial transfers are performed with all the SLL signals being in the negated state.
When using the SBI in slave mode, set B'000 to the SSLA bits.

BRDV Bit

These bits are used to determine the bit rate. A bit rate is determined by combinations of the BRDV bits and the settings in the SBIi bit rate register (SBiBR). The settings in the SBiBR register determine the base bit rate. The settings in the BRDV bits are used to select a bit rate which is obtained by dividing the base bit rate by 1, 2, 4, or 8. In the SBiCMDk register, different the BRDV bits settings can be specified. This permits the execution of serial transfers at a different bit rate for each command.

CPOL Bit

Sets an RSPCK polarity of the SBI in master or slave mode. Data communications between SBI modules require the same RSPCK polarity setting between the modules.

CPHA Bit

Sets an RSPCK phase of the SBI in master or slave mode. Data communications between SBI modules require the same RSPCK phase setting between the modules.

18.3 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.

18.3.1 Overview of SBI Operations

The SBI is capable of synchronous serial transfers in slave mode (SBI operation), single-master mode (SBI operation), multi-master mode (SBI operation), slave mode (clock synchronous operation), and master mode (clock synchronous operation). A particular mode of the SBI can be selected by using the MSTR, MODFEN, and SPMS bits in the SBIi control register 0 (SBiCR0). Table 18.7 lists Relationship between SBI Modes and SBiCR0 and Description of Each Mode.

Table 18.7 Relationship between SBI Modes and SBiCR0 and Description of Each Mode

Mode	Slave (SBI Operation)	Single-Master (SBI Operation)	Multi-Master (SBI 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
RSPCK signal	Input	Output	Output/Hi-Z	Input	Output
MOSIi signal	Input	Output	Output/Hi-Z	Input	Output
MISOi signal	Output/Hi-Z	Input	Input	Output	Input
SSLi0 signal	Input	Output	Input	Hi-Z	Hi-Z
SSLi1 to SSLi3 signals	Hi-Z	Output	Output/Hi-Z	Hi-Z	Hi-Z
Output pin mode	CMOS/ open-drain	CMOS/ open-drain	CMOS/ open-drain	CMOS/ open-drain	CMOS/ open-drain
SSL polarity modification function	Supported	Supported	Supported	—	—
Transfer rate	Up to f(PBB)/8	Up to f(PBB)/2	Up to f(PBB)/2	Up to f(PBB)/8	Up to f(PBB)/2
Clock source	RSPCK input	On-chip baud rate generator	On-chip baud rate generator	RSPCK input	On-chip baud rate generator
Clock polarity	Two	Two	Two	Two	Two
Clock phase	Two	Two	Two	One (CPHA = 1)	One (CPHA = 1)
First transfer bit	MSB/LSB	MSB/LSB	MSB/LSB	MSB/LSB	MSB/LSB
Transfer data length	8 to 32 bits	8 to 32 bits	8 to 32 bits	8 to 32 bits	8 to 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	Transmit buffer is written when SPTEF = 1	Transmit buffer is written when SPTEF = 1	RSPCK oscillation	Transmit buffer is written when SPTEF = 1
Sequence control	Not supported	Supported	Supported	Not supported	Supported
Transmit buffer empty detection	Supported	Supported	Supported	Supported	Supported
Receive buffer full detection	Supported *1	Supported *1	Supported *1	Supported *1	Supported *1
Overrun error detection	Supported *1	Supported *1	Supported *1	Supported *1	Supported *1
Parity error detection	Supported *1*2	Supported *1*2	Supported *1*2	Supported *1*2	Supported *1*2
Mode fault error detection	Supported (MODFEN = 1)	Not supported	Supported	Not supported	Not supported

Notes: 1. When the TXMD bit in the SBiCR0 register is 1, the receive buffer full detection, the overrun error detection, and the parity error detection are not performed.
2. When the SPPE bit in the SBiCR1 register is 0, the parity error detection is not performed.

18.3.2 Controlling SBI Pins

According to the MSTR, MODFEN, and SPMS bits in the SBIi control register 0 (SBiCR0) and the SPOM bit in the SBIi pin control register (SBiPCR), the SBI can automatically switch pin directions and output modes. Table 18.8 shows Relationship between Pin States and Bit Settings.

Table 18.8 Relationship between Pin States and Bit Settings

Mode	Pin	Pin State *1	
		SPOM = 0	SPOM = 1
Single-master mode (SBI operation) (MSTR = 1, MODFEN = 0, SPMS = 0)	RSPCK	CMOS output	Open-drain output
	SSLi0 to SSLi3	CMOS output	Open-drain output
	MOSIi	CMOS output	Open-drain output
	MISOi	Input	Input
Multi-master mode (SBI operation) (MSTR = 1, MODFEN = 1, SPMS = 0)	RSPCK *2	CMOS output/Hi-Z	Open-drain output/Hi-Z
	SSLi0	Input	Input
	SSLi1 to SSLi3 *2	CMOS output/Hi-Z	Open-drain output/Hi-Z
	MOSIi *2	CMOS output/Hi-Z	Open-drain output/Hi-Z
	MISOi	Input	Input
Slave mode (SBI operation) (MSTR = 0, SPMS = 0)	RSPCK	Input	Input
	SSLi0	Input	Input
	SSLi1 to SSLi 3	Hi-Z	Hi-Z
	MOSIi	Input	Input
	MISOi *3	CMOS output/Hi-Z	Open-drain output/Hi-Z
Master mode (Clock synchronous operation) (MSTR = 1, MODFEN = 0, SPMS = 1)	RSPCK	CMOS output	Open-drain output
	SSLi0 to SSLi3 *4	Hi-Z	Hi-Z
	MOSIi	CMOS output	Open-drain output
	MISOi	Input	Input
Slave mode (Clock synchronous operation) (MSTR = 0, SPMS = 1)	RSPCK	Input	Input
	SSLi0 to SSLi3 *4	Hi-Z	Hi-Z
	MOSIi	Input	Input
	MISOi	CMOS output	Open-drain output

Notes: 1. SBI settings are not indicated in the multiplex pins for which the SBI function is not selected.
2. When SSLi0 is at the active level, the pin state is Hi-Z.
3. When SSLi0 is at the non-active level or the SPE bit in the SBiCR0 register is cleared (= 0), the pin state is Hi-Z.
4. In clock synchronous operation, SSLi0 to SSLi3 are available for use as I/O port pins.

The SBI in single-master mode (SBI operation) or multi-master mode (SBI operation) determines MOSIi signal values during the SSL negation period * according to MOIFE and MOIFV bit settings in the SBiPCR register, as shown in Table 18.9.

Note: * Including the SSL retention period during a burst transfer

Table 18.9 MOSI Signal Value Determination during SSL Negation Period

MOIFE bit	MOIFV bit	MOSIi Signal Value during SSL Negation Period
0	0	Final data from previous transfer
0	1	
1	0	Always 0
1	1	Always 1

18.3.3 SBI System Configuration Example

(1) Single Master/Single Slave (with This LSI Acting as Master)

Figure 18.2 shows a single-master/single-slave SBI system configuration example when this LSI is used as a master. In the single-master/single-slave configuration, the SSLi0 to SSLi3 output of this LSI (master) are not used. The SSL input of the SBI slave is fixed to the low level, and the SBI slave is always maintained in a select state *.
This LSI (master) always drives the RSPCK and MOSIi. The SBI slave always drives the MISO.

Note: * In the transfer format corresponding to the case where the CPHA bit in the SBIi control register (SBiCR0) is 0, there are slave devices for which the SSL signal cannot be fixed to the active level. In situations where the SSL signal cannot be fixed, the SSL output of this LSI should be connected to the SSL input of the slave device.

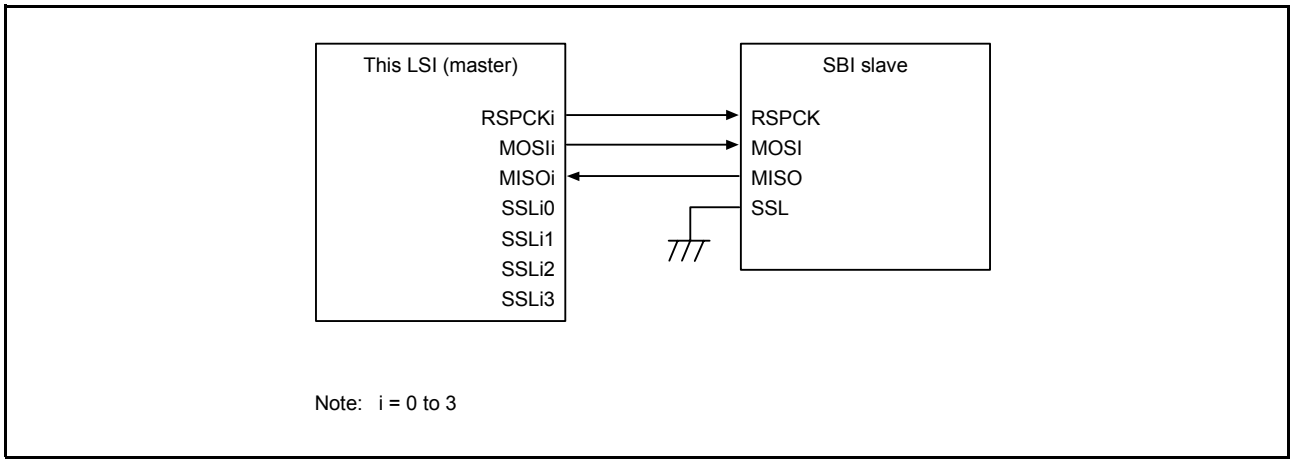


Figure 18.2 Single-Master/Single-Slave Configuration Example (This LSI = Master)

(2) Single Master/Single Slave (with This LSI Acting as Slave)

Figure 18.3 shows a single-master/single-slave SBI system configuration example when this LSI is used as a slave. When this LSI is to operate as a slave, the SSLi0 pin is used as SSL input. The SBI master always drives the RSPCK and MOSI. This LSI (slave) always drives the MISOi *.

In the single-slave configuration in which the CPHA bit in the SBIi command register k (SBiCMDk) is set to 1, the SSLi0 input of this LSI (slave) is fixed to the 0 level, this LSI (slave) is always maintained in a selected state, and in this manner it is possible to execute serial transfer (Figure 18.4).

Note: * When SSLi0 is at the non-active level, the pin state is Hi-Z.

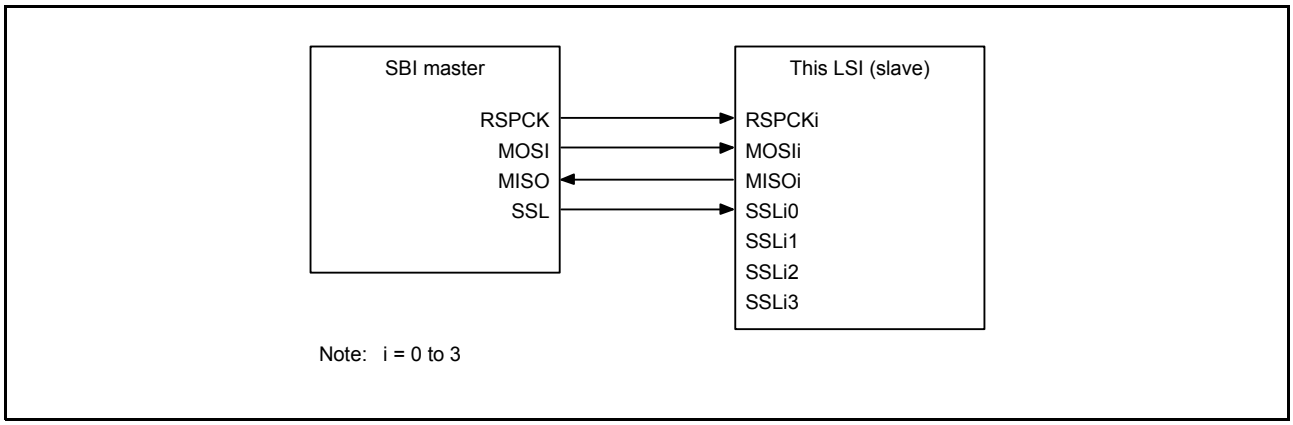


Figure 18.3 Single-Master/Single-Slave Configuration Example (This LSI = Slave)

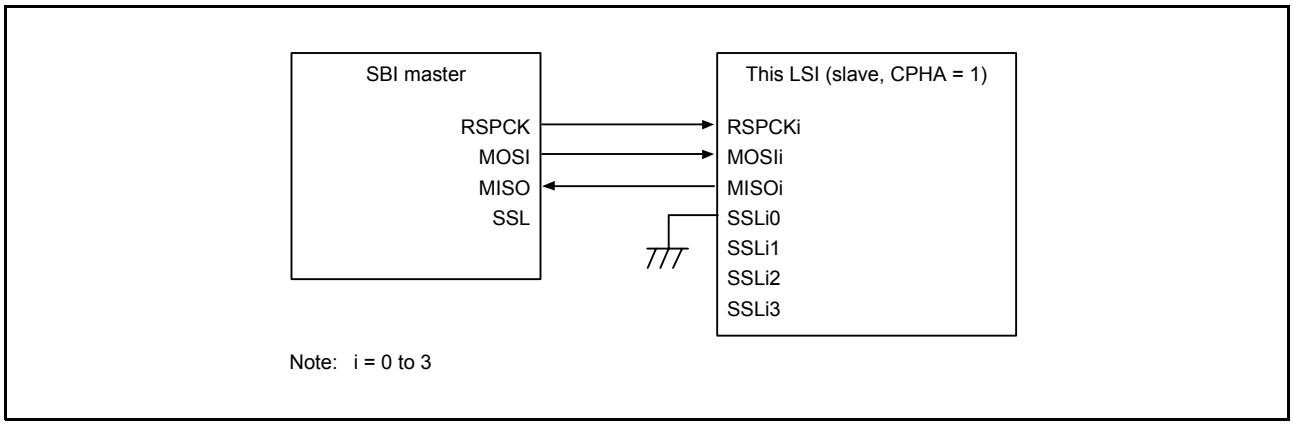


Figure 18.4 Single-Master/Single-Slave Configuration Example (This LSI = Slave, CPHA = 1)

(3) Single Master/Multi-Slave (with This LSI Acting as Master)

Figure 18.5 shows a single-master/multi-slave SBI system configuration example when this LSI is used as a master. In the example of Figure 18.5, the SBI system is comprised of this LSI (master) and four slaves (SBI slave 0 to SBI slave 3).

The RSPCK and MOSIi outputs of this LSI (master) are connected to the RSPCK and MOSI inputs of SBI slave 0 to SBI slave 3. The MISO outputs of SBI slave 0 to SBI slave 3 are all connected to the MISOi input of this LSI (master). SSLi0 to SSLi3 outputs of this LSI (master) are connected to the SSL inputs of SBI slave 0 to SBI slave 3, respectively.

This LSI (master) always drives RSPCK, MOSIi, and SSLi0 to SSLi3. Out of the SBI slave 0 to SBI slave 3, the slave that receives low-level input into the SSL input drives MISO.

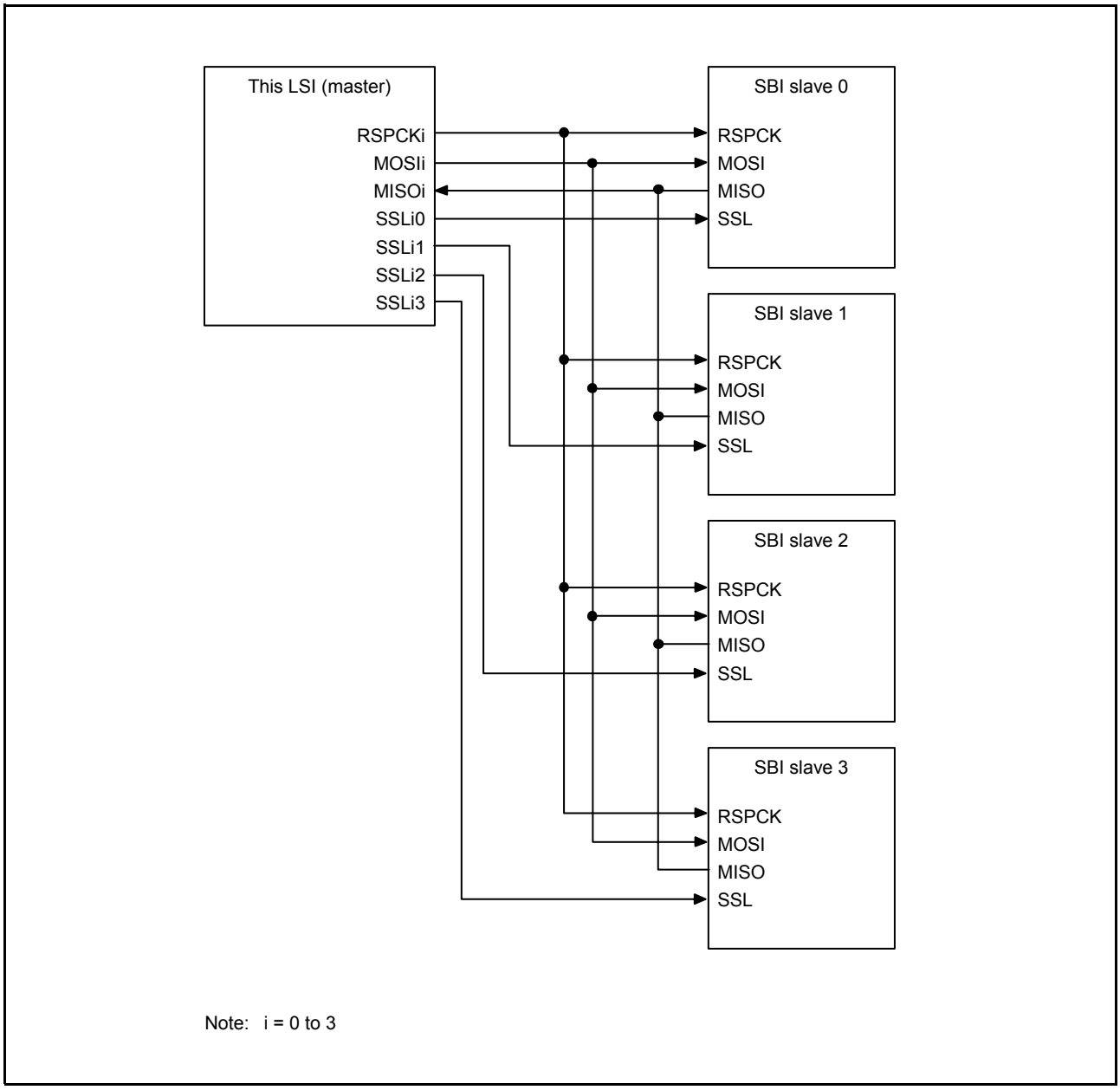


Figure 18.5 Single-Master/Multi-Slave Configuration Example (This LSI = Master)

(4) Single Master/Multi-Slave (with This LSI Acting as Slave)

Figure 18.6 shows a single-master/multi-slave SBI system configuration example when this LSI is used as a slave. In the example of Figure 18.6, the SBI system is comprised of an SBI master and two LSIs (slave X and slave Y).

The RSPCK and MOSIi outputs of the SBI master are connected to the RSPCK and MOSIi inputs of the LSIs (slave X and slave Y). The MISO outputs of the LSIs (slave X and slave Y) are all connected to the MISO input of the SBI master. SSLX and SSLY outputs of the SBI master are connected to the SSLi0 inputs of the LSIs (slave X and slave Y), respectively.

The SBI master always drives RSPCK, MOSI, SSLX, and SSLY. Out of the LSIs (slave X and slave Y), the slave that receives low-level input into the SSLi0 input drives MISOi.

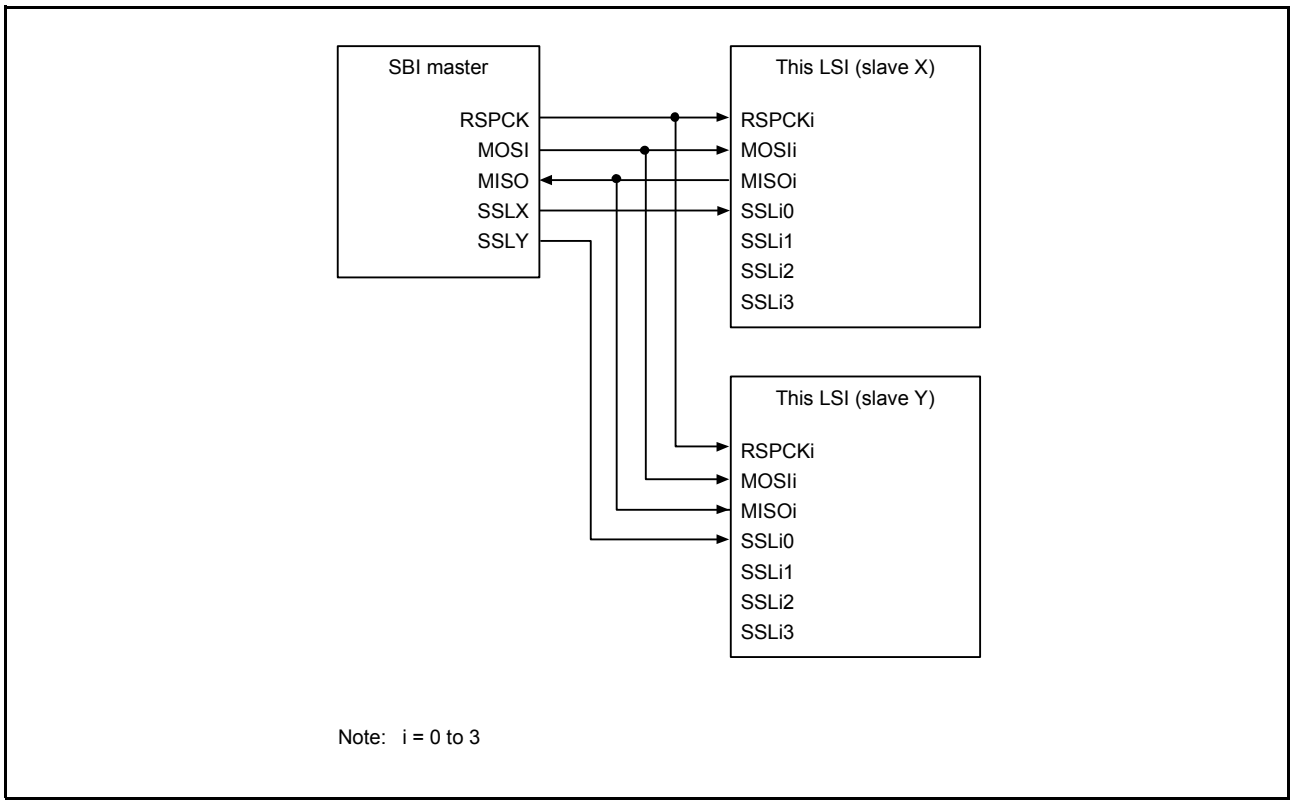


Figure 18.6 Single-Master/Multi-Slave Configuration Example (This LSI = Master)

(5) Multi-Master/Multi-Slave (with This LSI Acting as Master)

Figure 18.7 shows a multi-master/multi-slave SBI system configuration example when this LSI is used as a master. In the example of Figure 18.7, the SBI system is comprised of two LSIs (master X, master Y) and two SBI slaves (SBI slave 1, SBI slave 2).

The RSPCK and MOSIi outputs of this LSI (master X, master Y) are connected to the RSPCK and MOSI inputs of SBI slaves 1 and 2. The MISOi outputs of SBI slaves 1 and 2 are connected to the MISOi inputs of this LSI (master X, master Y). Any generic port Y output from this LSI (master X) is connected to the SSLi0 input of this LSI (master Y). Any generic port X output of this LSI (master Y) is connected to the SSLi0 input of this LSI (master X). The SSLi1 and SSLi2 outputs of this LSI (master X, master Y) are connected to the SSL inputs of the SBI slaves 1 and 2. In this configuration example, because the system can be comprised solely of SSLi0 input, and SSLi1 and SSLi2 outputs for slave connections, SSLi3 output of this LSI is not required.

This LSI drives RSPCK, MOSIi, SSLi1, and SSLi2 when the SSLi0 input level is 1. When the SSLi0 input level is 0, this LSI detects a mode fault error, sets RSPCK, MOSIi, SSLi1, and SSLi2 to Hi-Z, and releases the SBI bus right to the other master. Out of the SBI slaves 1 and 2, the slave that receives a level-0 input into the SSL input drives MISO.

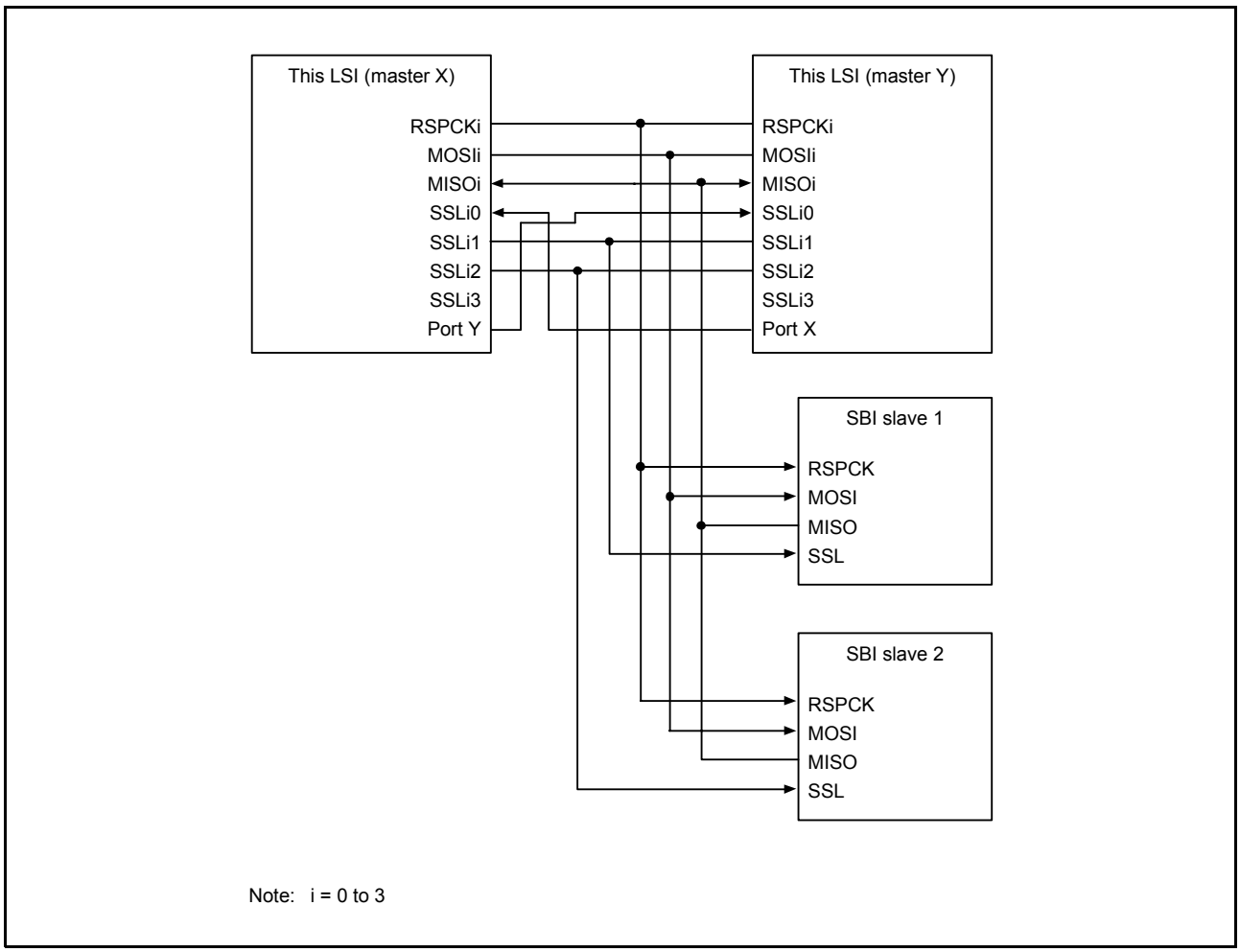


Figure 18.7 Multi-Master/Multi-Slave Configuration Example (This LSI = Master)

(6) Master (Clock Synchronous Operation)/Slave (Clock Synchronous Operation) (with This LSI Acting as Master)

Figure 18.8 shows a master (clock synchronous operation)/slave (clock synchronous operation) SBI system configuration example when this LSI is used as a master. In the master (clock synchronous operation)/slave (clock synchronous operation) configuration, SSLi0 to SSLi3 of this LSI (master) are not used.
This LSI (master) always drives the RSPCK and MOSIi. The SBI slave always drives the MISO.

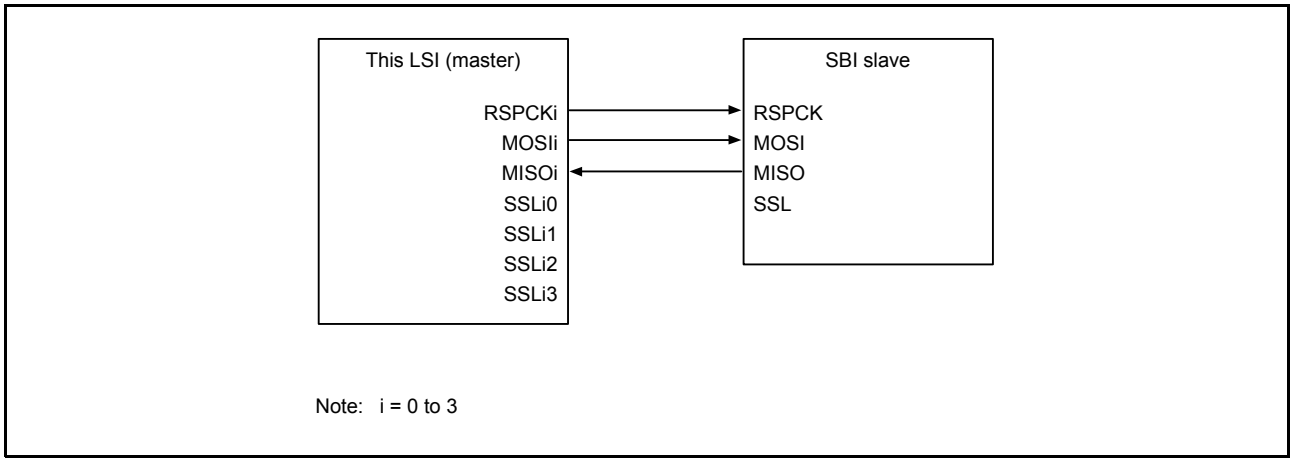


Figure 18.8 Master (Clock Synchronous Operation)/Slave (Clock Synchronous Operation) Configuration Example (This LSI = Master)

(7) Master (Clock Synchronous Operation)/Slave (Clock Synchronous Operation) (with This LSI Acting as Master)

Figure 18.9 shows a master (clock synchronous operation)/slave (clock synchronous operation) SBI system configuration example when this LSI is used as a master. When this LSI is to operate as a slave (clock synchronous operation), this LSI (slave) always drives the MISOi and the SBI master always drives the RSPCK and MOSI.
Only in the single-slave configuration in which the CPHA bit in the SBIi command register k (SBiCMDk) is set to 1, this LSI (slave) can execute serial transfer.

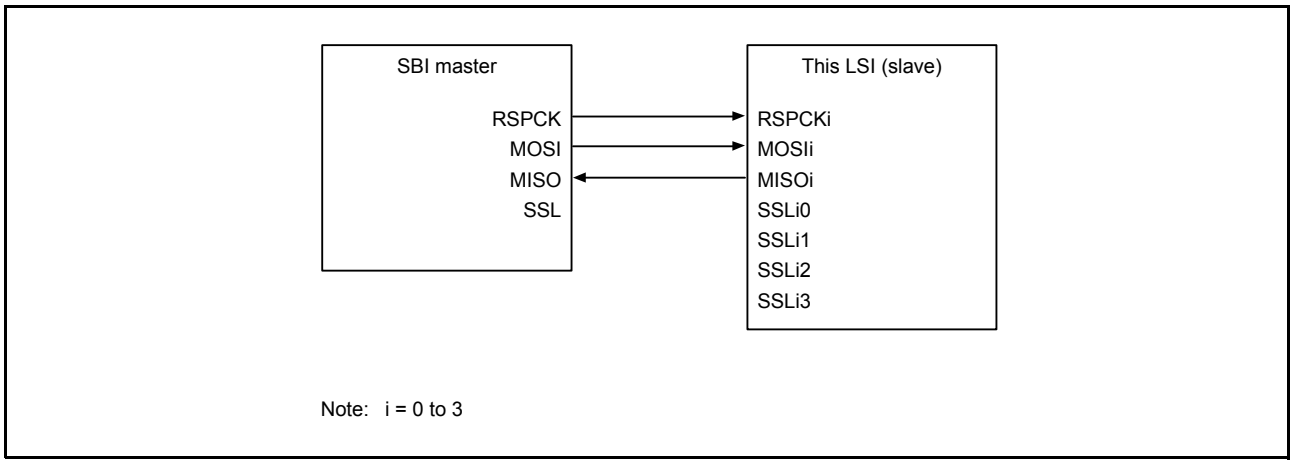


Figure 18.9 Master (Clock Synchronous Operation)/Slave (Clock Synchronous Operation) Configuration Example (This LSI = Slave, CPHA = 1)

18.3.4 Transfer Format

(1) CPHA = 0

Figure 18.10 shows a sample transfer format for the serial transfer of 8-bit data when the CPHA bit in the SBIi command register k (SBiCMDk) is 0. Note that clock synchronous operation (the SPMS bit in the SBIi control register 0 (SBiCR0) is 1) is not guaranteed when the CPHA bit is 0. In Figure 18.10, RSPCK (CPOL = 0) indicates the RSPCK signal waveform when the CPOL bit in the SBiCMDk register is 0; RSPCK (CPOL = 1) indicates the RSPCK signal waveform when the CPOL bit is 1. The sampling timing represents the timing at which the SBI fetches serial transfer data into the shift register. The input/output directions of the signals depend on the SBI settings. For details, refer to section 18.3.2, Controlling SBI Pins.

When the CPHA bit is 0, the driving of valid data to the MOSIi and MISOi signals commences at an SSL signal assertion timing. The first RSPCK signal change timing that occurs after the SSL signal assertion becomes the first transfer data fetch timing. After this timing, data is sampled at every 1 RSPCK cycle. The change timing for MOSIi and MISOi signals is always 1/2 RSPCK cycle after the transfer data fetch timing. The settings in the CPOL bit do not affect the RSPCK signal operation timing; they only affect the signal polarity.

t1 denotes a period from an SSL signal assertion to RSPCK oscillation (RSPCK delay). t2 denotes a period from the cessation of RSPCK oscillation to an SSL signal negation (SSL negation delay). t3 denotes a period in which SSL signal assertion is suppressed for the next transfer after the end of serial transfer (next-access delay). t1, t2, and t3 are controlled by a master device running on the SBI system. For a description of t1, t2, and t3 when the SBI of this LSI is in master mode, refer to section 18.3.10, SBI Operations (1), Master Mode Operation.

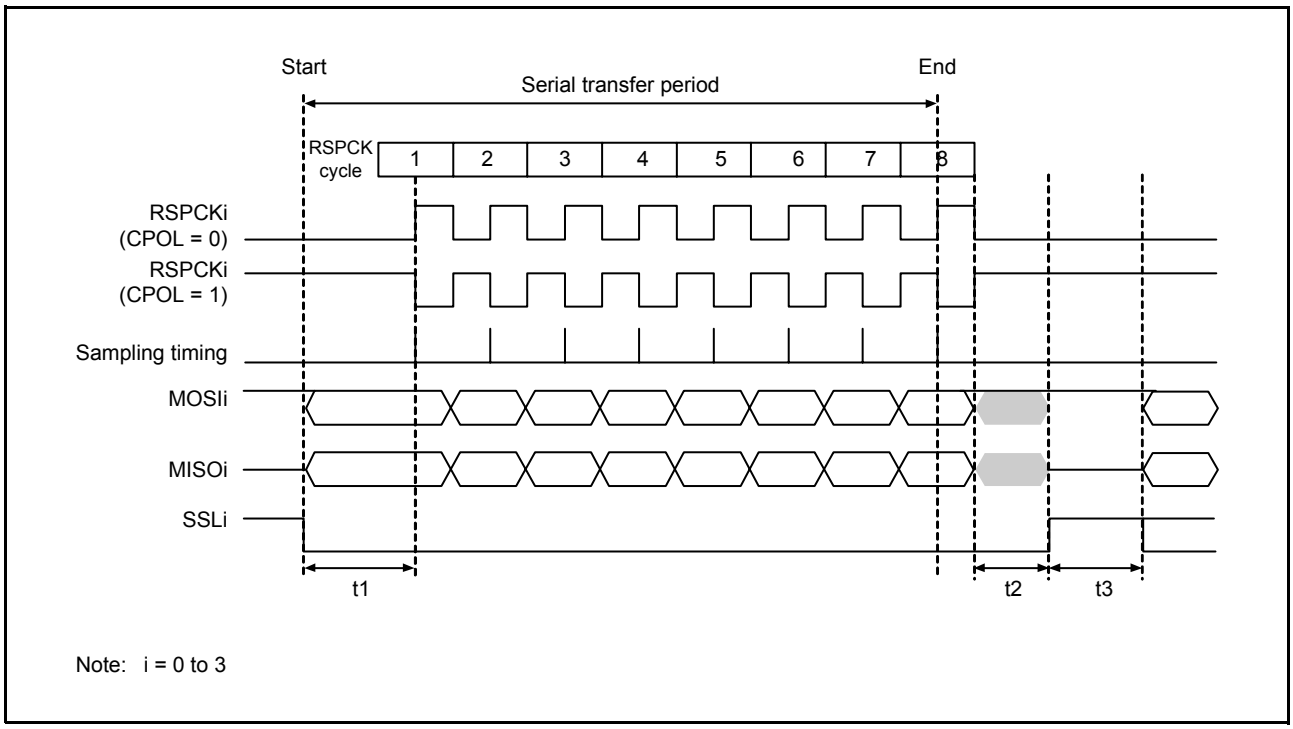


Figure 18.10 SBI Transfer Format (CPHA = 0)

(2) CPHA = 1

Figure 18.11 shows a sample transfer format for the serial transfer of 8-bit data when the CPHA bit in the SBIi command register k (SBiCMDk) is 1. However, when the SPMS bit in the SBIi control register 0 (SBiCR0) is 1, the SSL signals are not used, and only the three signals RSPCK, MOSIi, and MISOi handle communications. In Figure 18.11, RSPCK (CPOL = 0) indicates the RSPCK signal waveform when the CPOL bit in the SBiCMDk register is 0; RSPCK (CPOL = 1) indicates the RSPCK signal waveform when the CPOL bit is 1. The sampling timing represents the timing at which the SBI fetches serial transfer data into the shift register. The input/output directions of the signals depend on the SBI modes (master or slave). For details, refer to section 18.3.2, Controlling SBI Pins.

When the CPHA bit is 1, the driving of invalid data to the MOSIi and MISOi signals commences at an SSL signal assertion timing. The driving of valid data to the MOSIi and MISOi signals commences at the first RSPCK signal change timing that occurs after the SSL signal assertion. After this timing, data is updated at every 1 RSPCK cycle. The transfer data fetch timing is always 1/2 RSPCK cycle after the data update timing. The settings in the CPOL bit do not affect the RSPCK signal operation timing; they only affect the signal polarity.

t1, t2, and t3 are the same as those in the case of CPHA = 0. For a description of t1, t2, and t3 when the SBI of this LSI is in master mode, refer to section 18.3.10, SBI Operations (1), Master Mode Operation.

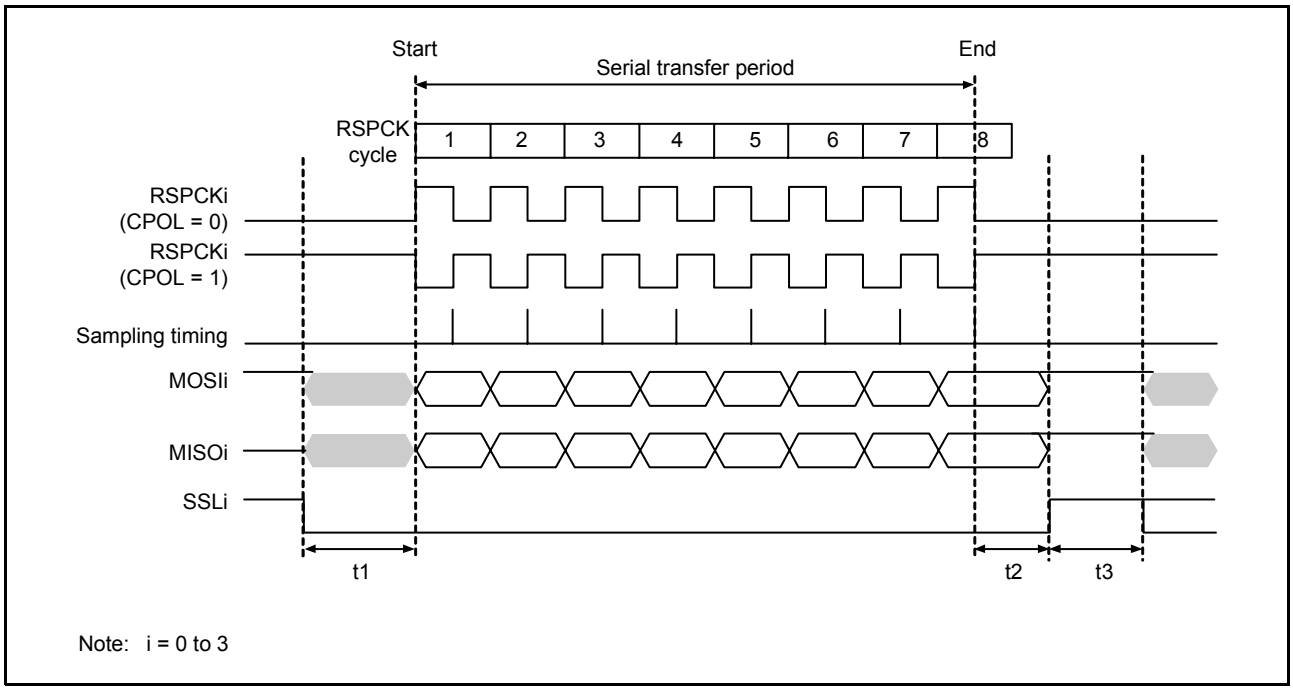


Figure 18.11 SBI Transfer Format (CPHA = 1)

18.3.5 Data Format

The SBI's data format depends on the settings in the SBIi command register k (SBiCMDk) and the parity enable bit (SPPE) in the SBIi control register 1 (SBiCR1). Irrespective of MSB/LSB first, the SBI treats the range from the LSB of the SBIi data register (SBiDR) to the assigned data length as transfer data.

(1) MSB First Transfer (32-Bit Data)

1. When Parity Function is Disabled (SPPE = 0)

Figure 18.12 shows the operation of the SBIi data register (SBiDR) and the shift register when the SBI performs a 32-bit data length MSB-first data transfer with the parity function disabled.

T31 to T00 is written to the transmit buffer of the SBiDR register. If the SPTEF flag in the SBIi status register (SBiSR) is 0 and the shift register is empty, the SBI copies the data in the transmit buffer of the SBiDR register to the shift register, and fully populates the shift register. When serial transfer starts, the SBI outputs data from the MSB (bit 31) of the shift register, and shifts in the data from the LSB (bit 0) of the shift register.

When the RSPCK cycle required for the serial transfer of 32 bits has passed, data R31 to R00 is stored in the shift register. In this state and the full-duplex synchronous serial communication (TXMD = 0), the SBI copies the data from the shift register to the receive buffer of the SBiDR register, and empties the shift register.

If another serial transfer is started before the transmit buffer of the SBiDR register is written received data R31 to R00 is shifted out from the shift register.

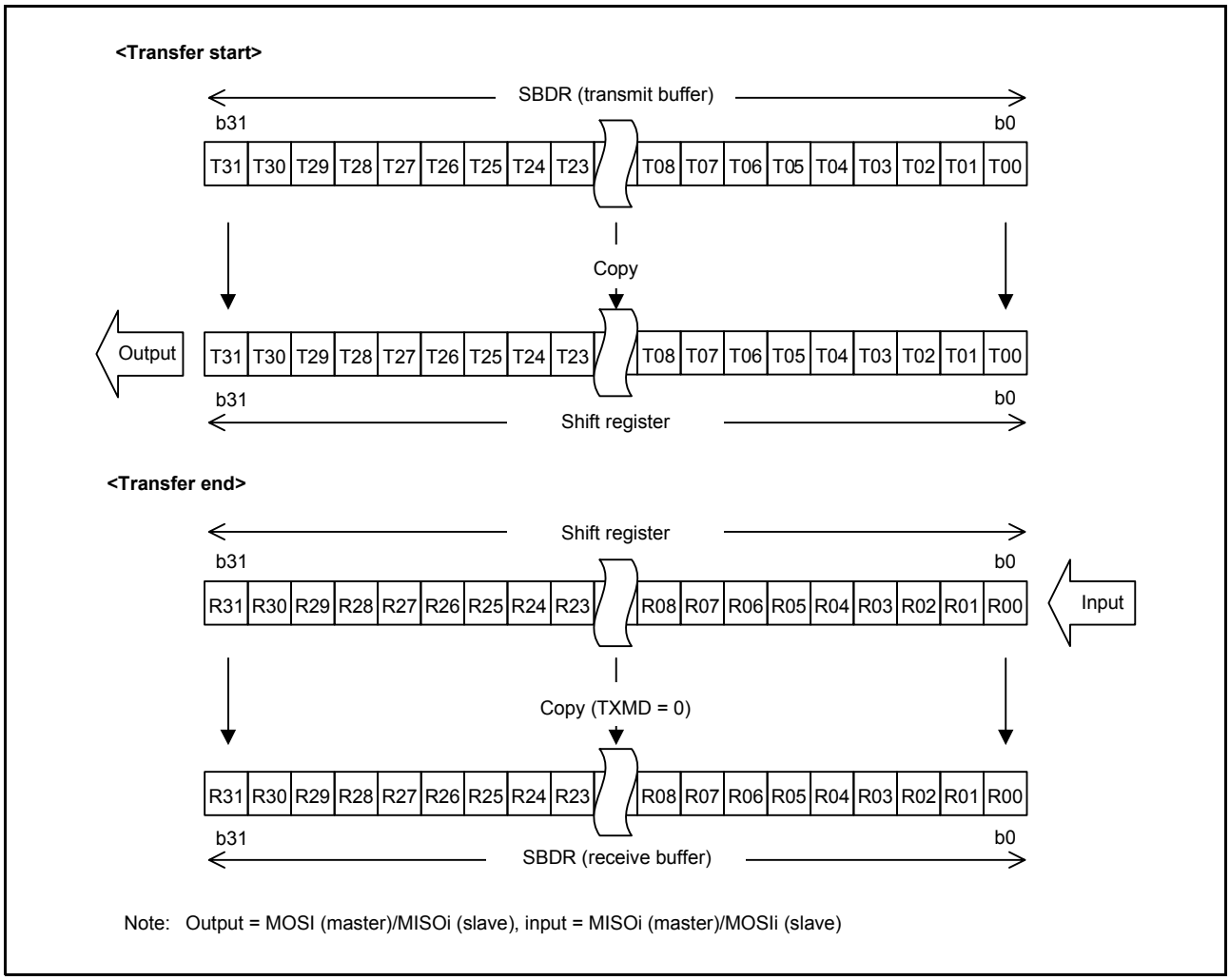


Figure 18.12 MSB First Transfer (32-Bit Data/Parity Function Disabled)

2. When Parity Function is Enabled (SPPE = 1)

Figure 18.13 shows the operation of the SBIi data register (SBiDR) and the shift register when the SBI performs a 32-bit data length MSB-first data transfer with the parity function enabled.

T31 to T00 is written to the transmit buffer of the SBiDR register. If the SPTEF flag in the SBIi status register (SBiSR) is 0 and the shift register is empty, the SBI converts the data T00 that stored in the transmit buffer of the SBiDR register to the parity bit (P). The SBI copies the data to which the parity bit (P) is appended to the shift register, and fully populates the shift register. When serial transfer starts, the SBI outputs data from the MSB (bit 31) of the shift register, and shifts in the data from the LSB (bit 0) of the shift register.

When the RSPCK cycle required for the serial transfer of 32 bits has passed, data R31 to P is stored in the shift register. In this state and the full-duplex synchronous serial communication (TXMD = 0), the SBI copies the data from the shift register to the receive buffer of the SBiDR register, and empties the shift register.

If another serial transfer is started before the transmit buffer of the SBiDR register is written received data R31 to P is shifted out from the shift register.

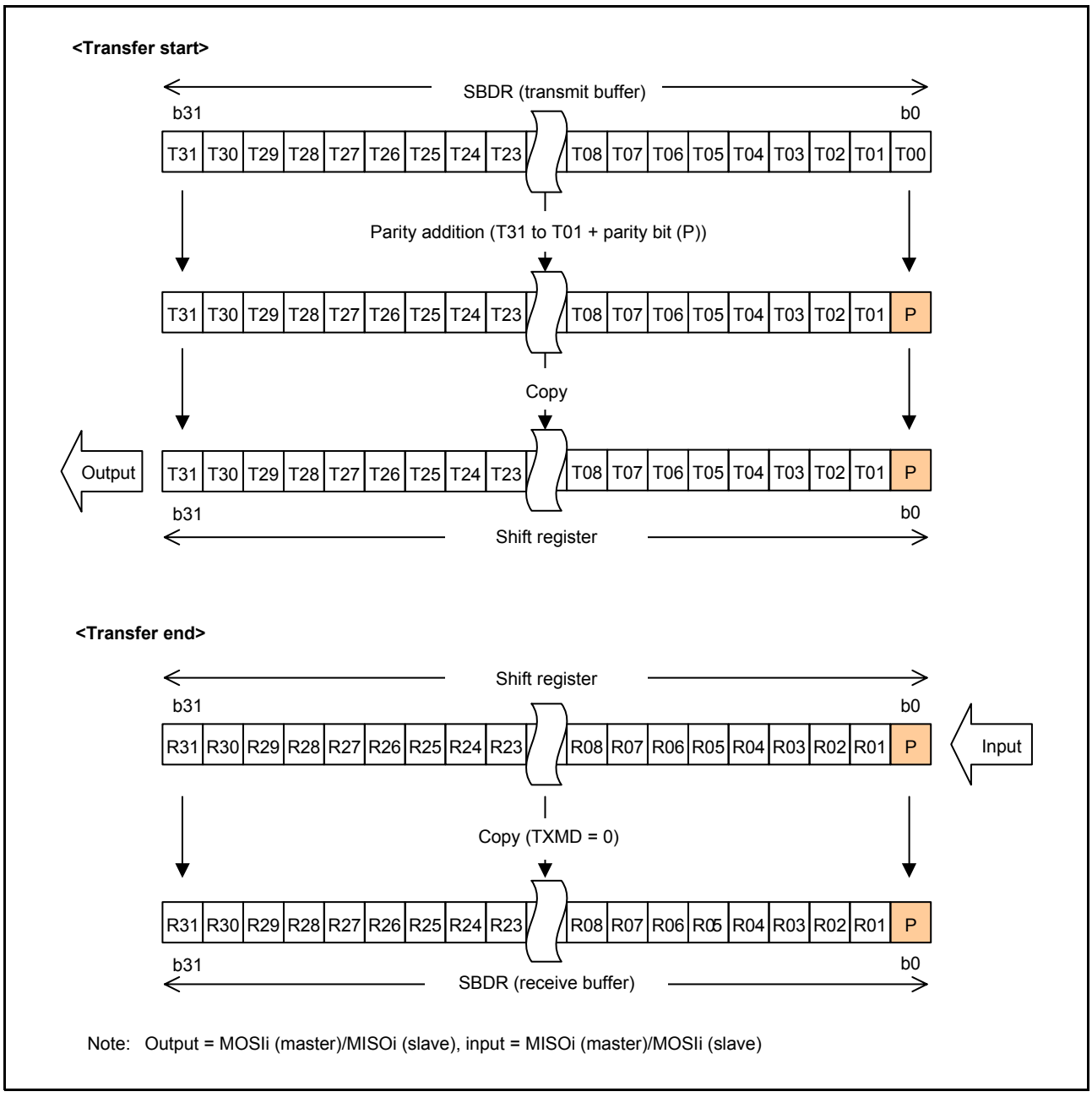


Figure 18.13 MSB First Transfer (32-Bit Data/Parity Function Enabled)

(2) MSB First Transfer (24-Bit Data)

1. When Parity Function is Disabled (SPPE = 0)

Figure 18.14 shows the operation of the SBIi data register (SBiDR) and the shift register when the SBI performs a 24-bit data length MSB-first data transfer with the parity function disabled (as an example of operation in the transfer of data with lengths other than 32 bits).

T31 to T00 is written to the transmit buffer of the SBiDR register. If the SPTEF flag in the SBIi status register (SBiSR) is 0 and the shift register is empty, the SBI copies the data in the transmit buffer of the SBiDR register to the shift register, and fully populates the shift register. When serial transfer starts, the SBI outputs data from bit 23 of the shift register, and shifts in the data from the LSB (bit 0) of the shift register. When the RSPCK cycle required for the serial transfer of 24 bits has passed, received data R23 to R00 is stored in bits 23 to 0 of the shift register. After completion of the serial transfer, data that existed before the transfer is retained in bits 31 to 24 in the shift register. In this state and the full-duplex synchronous serial communication (TXMD = 0), the SBI copies the data from the shift register to the receive buffer of the SBiDR register, and empties the shift register.

If another serial transfer is started before the transmit buffer of the SBiDR register is written received data R23 to R00 is shifted out from the shift register.

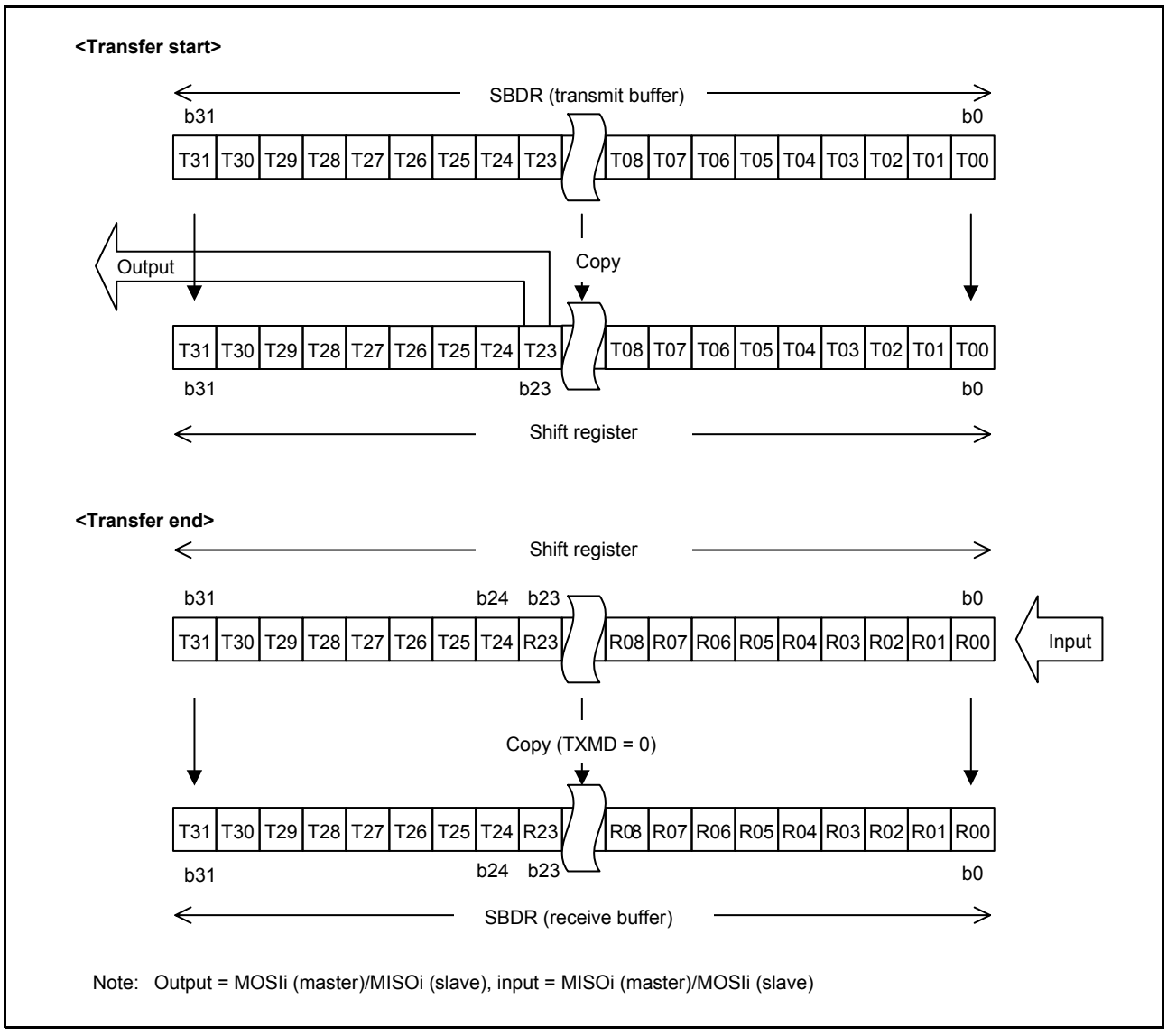


Figure 18.14 MSB First Transfer (24-Bit Data/Parity Function Disabled)

2. When Parity Function is Enabled (SPPE=1)

Figure 18.15 shows the operation of the SBIi data register (SBiDR) and the shift register when the SBI performs a 24-bit data length MSB-first data transfer with the parity function enabled (as an example of operation in the transfer of data with lengths other than 32 bits).

T31 to T00 is written to the transmit buffer of the SBiDR register. If the SPTEF flag in the SBIi status register (SBiSR) is 0 and the shift register is empty, the SBI converts the data T00 that stored in the transmit buffer of the SBiDR register to the parity bit (P). The SBI copies the data to which the parity bit (P) is appended to the shift register, and fully populates the shift register. When serial transfer starts, the SBI outputs data from bit 23 of the shift register, and shifts in the data from the LSB (bit 0) of the shift register. When the RSPCK cycle required for the serial transfer of 24 bits has passed, received data R23 to P is stored in bits 23 to 0 of the shift register. After completion of the serial transfer, data that existed before the transfer is retained in bits 31 to 24 in the shift register. In this state and the full-duplex synchronous serial communication (TXMD = 0), the SBI copies the data from the shift register to the receive buffer of the SBiDR register, and empties the shift register.

If another serial transfer is started before the transmit buffer of the SBiDR register is written received data R23 to P is shifted out from the shift register.

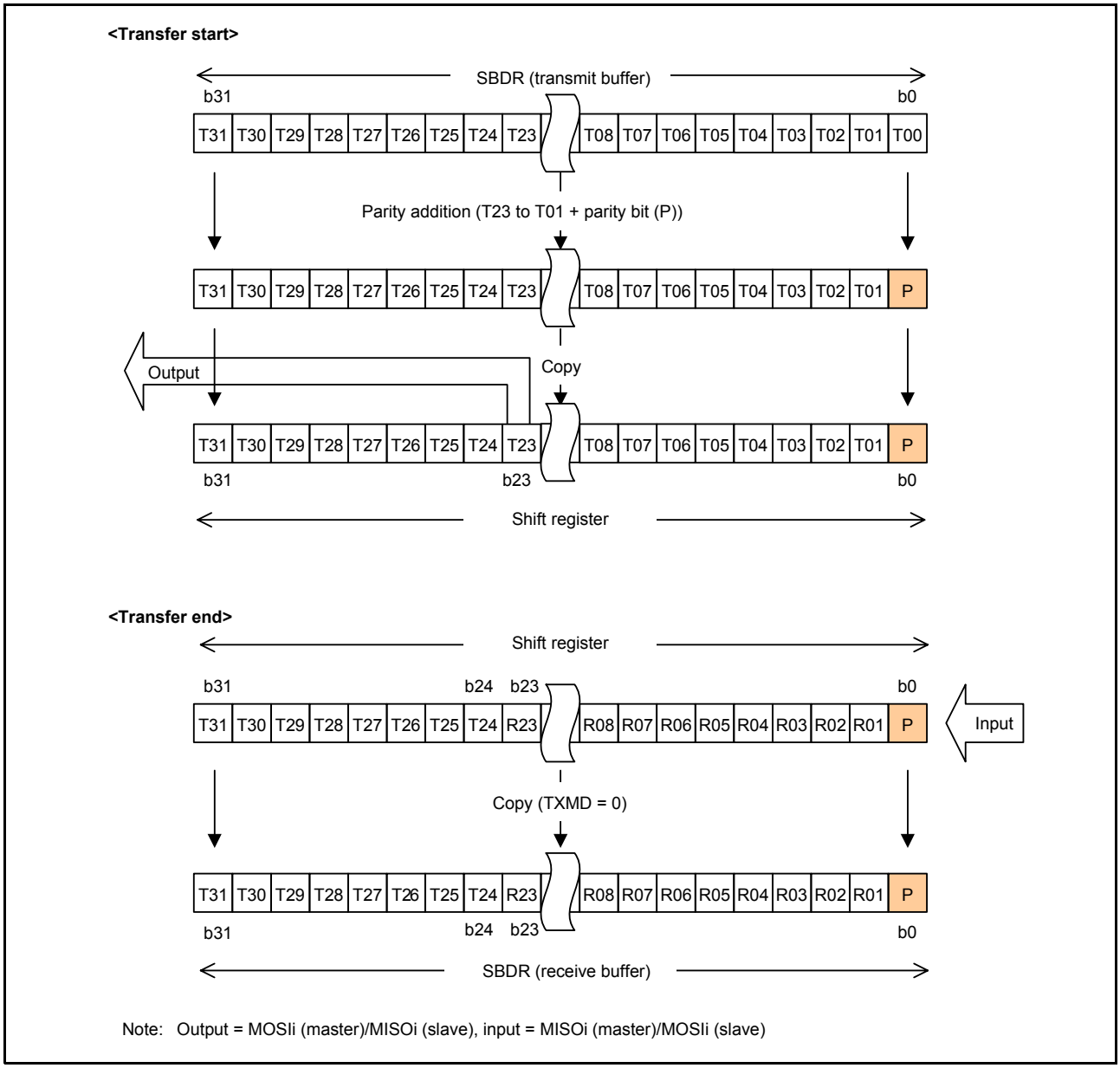


Figure 18.15 MSB First Transfer (24-Bit Data/Parity Function Enabled)

(3) LSB First Transfer (32-Bit Data)

1. When Parity Function is Disabled (SPPE = 0)

Figure 18.16 shows the operation of the SBIi data register (SBiDR) and the shift register when the SBI performs a 32-bit data length LSB-first data transfer with the parity function disabled.

The CPU or the DMAC writes T31 to T00 to the transmit buffer of the SBiDR register. If the SPTEF flag in the SBIi status register (SBiSR) is 0 and the shift register is empty, the SBI reverses the order of the bits of the data in the transmit buffer of the SBiDR register, copies it to the shift register, and fully populates the shift register. When serial transfer starts, the SBI outputs data from the MSB (bit 31) of the shift register, and shifts in the data from the LSB (bit 0) of the shift register. When the RSPCK cycle required for the serial transfer of 32 bits has passed, data R00 to R31 is stored in the shift register. In this state and the full-duplex synchronous serial communication (TXMD = 0), the SBI copies the data, in which the order of the bits is reversed, from the shift register to the receive buffer of the SBiDR register, and empties the shift register.

If another serial transfer is started before the transmit buffer of the SBiDR register is written received data R00 to R31 is shifted out from the shift register.

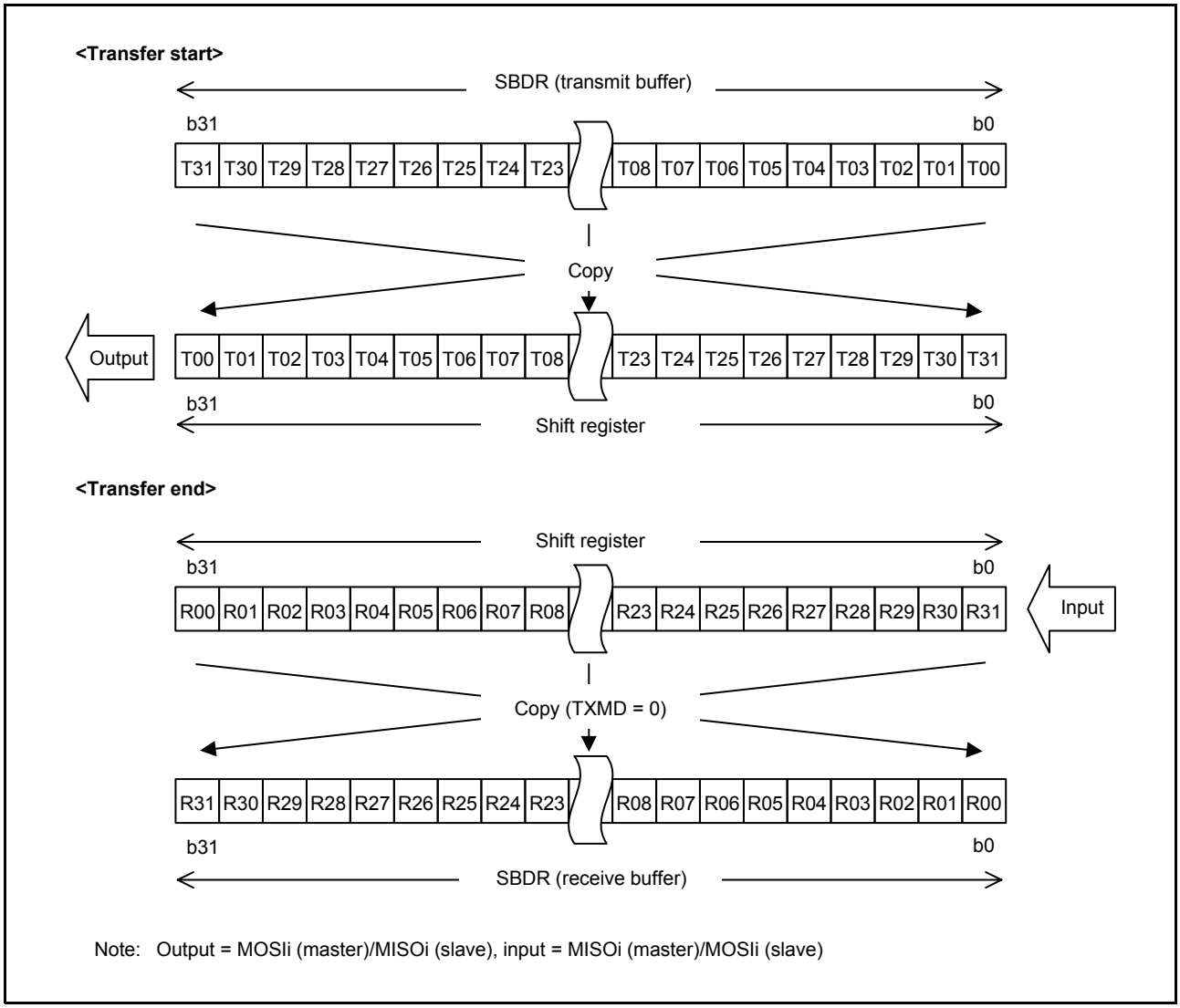


Figure 18.16 LSB First Transfer (32-Bit Data/Parity Function Disabled)

2. When Parity Function is Enabled (SPPE = 1)

Figure 18.17 shows the operation of the SBIi data register (SBiDR) and the shift register when the SBI performs a 32-bit data length LSB-first data transfer with the parity function enabled.

The CPU or the DMAC writes T31 to T00 to the transmit buffer of the SBiDR register. The SBI converts the data T31 that stored in the transmit buffer of the SBiDR register to the parity bit (P). If the SPTEF flag in the SBIi status register (SBiSR) is 0 and the shift register is empty, the SBI reverses the order of the bits of the data to which the parity bit (P) is appended, copies it to the shift register, and fully populates the shift register. When serial transfer starts, the SBI outputs data from the MSB (bit 31) of the shift register, and shifts in the data from the LSB (bit 0) of the shift register. When the RSPCK cycle required for the serial transfer of 32 bits has passed, data R00 to P is stored in the shift register. In this state and the full-duplex synchronous serial communication (TXMD = 0), the SBI copies the data, in which the order of the bits is reversed, from the shift register to the receive buffer of the SBiDR register, and empties the shift register.

If another serial transfer is started before the transmit buffer of the SBiDR register is written received data R00 to R31 is shifted out from the shift register.

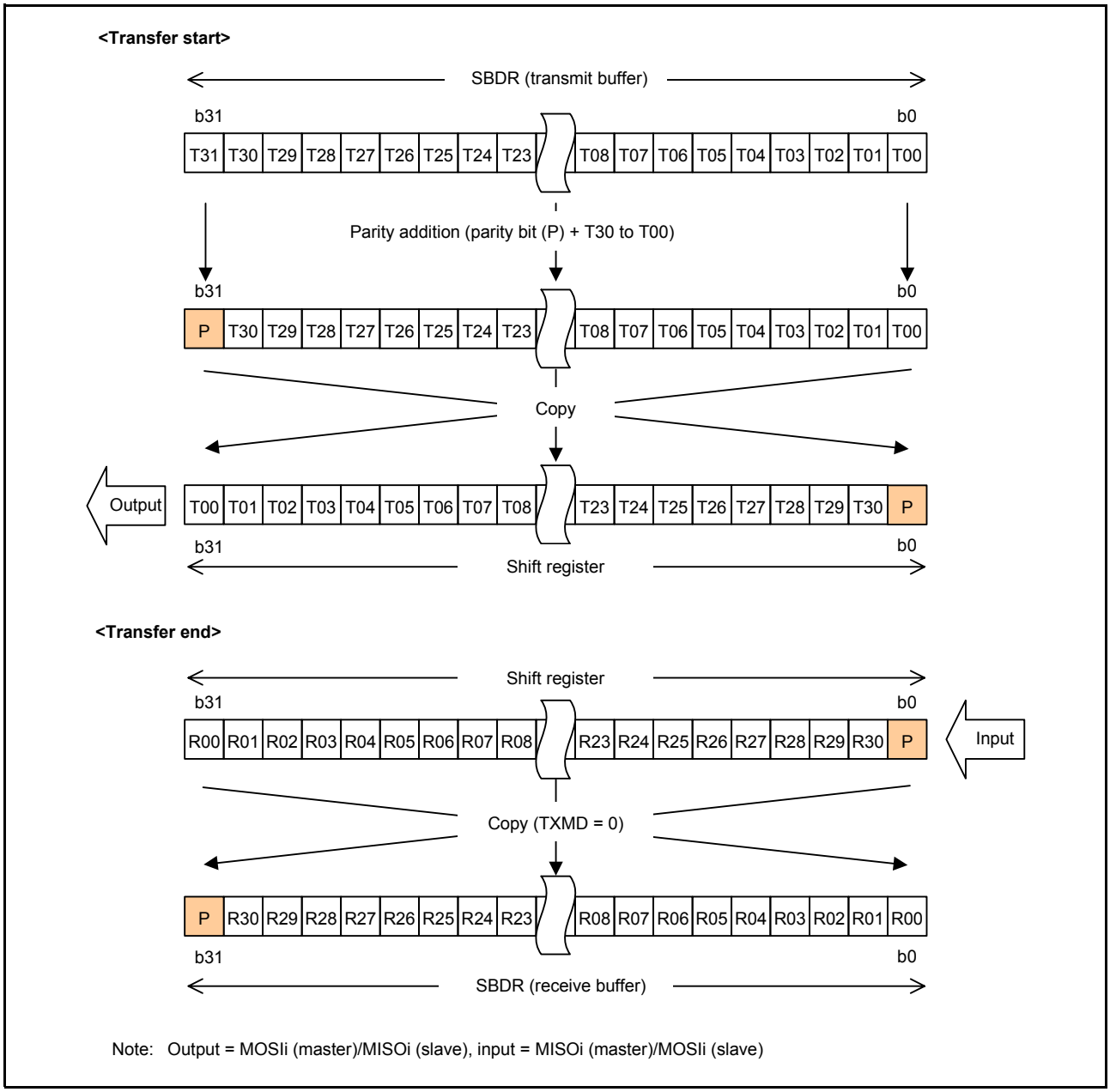


Figure 18.17 LSB First Transfer (32-Bit Data/Parity Function Enabled)

(4) LSB First Transfer (24-Bit Data)

1. When Parity Function is Disabled (SPPE = 0)

Figure 18.18 shows the operation of the SBIi data register (SBiDR) and the shift register when the SBI performs a 24-bit data length LSB-first data transfer with the parity function disabled (as an example of operation in the transfer of data with lengths other than 32 bits).

T31 to T00 is written to the transmit buffer of the SBiDR register. If the SPTEF flag in the SBIi status register (SBiSR) is 0 and the shift register is empty, the SBI reverses the order of the bits of the data in the transmit buffer of the SBiDR register, copies it to the shift register, and fully populates the shift register. When serial transfer starts, the SBI outputs data from the MSB (bit 31) of the shift register, and shifts in the data from bit 8 of the shift register. When the RSPCK cycle required for the serial transfer of 24 bits has passed, received data R00 to R23 is stored in bits 31 to 8 of the shift register. After completion of the serial transfer, data that existed before the transfer is retained in bits 7 to 0 of the shift register. In this state and the full-duplex synchronous serial communication (TXMD = 0), the SBI copies the data, in which the order of the bits is reversed, from the shift register to the receive buffer of the SBiDR register, and empties the shift register.

If another serial transfer is started before the transmit buffer of the SBiDR register is written received data R00 to R23 is shifted out from the shift register.

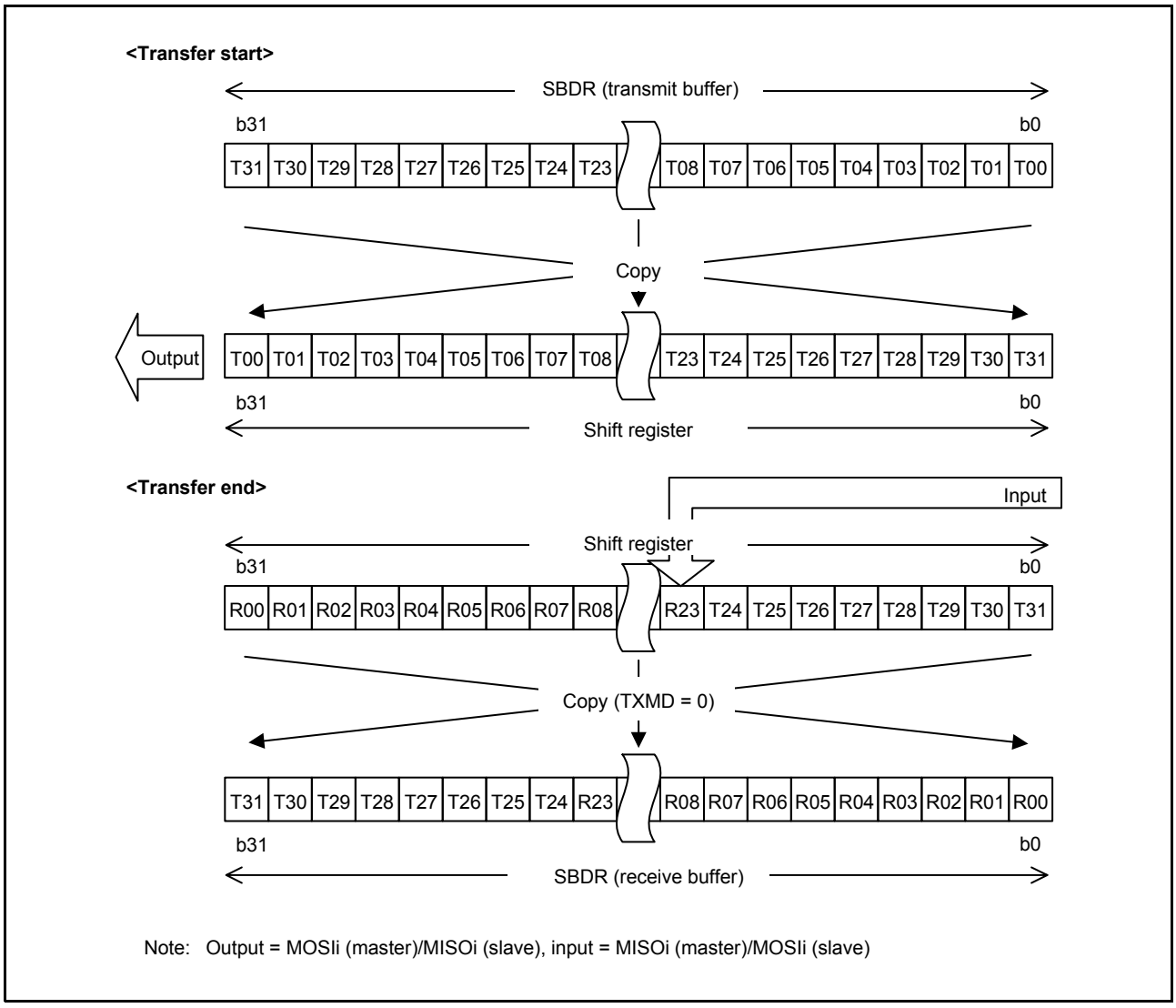


Figure 18.18 LSB First Transfer (24-Bit Data/Parity Function Disabled)

2. When Parity Function is Enabled (SPPE=1)

Figure 18.19 shows the operation of the SBIi data register (SBiDR) and the shift register when the SBI performs a 24-bit data length LSB-first data transfer with the parity function enabled (as an example of operation in the transfer of data with lengths other than 32 bits).

T31 to T00 is written to the transmit buffer of the SBiDR register. The SBI converts the data T23 that stored in the transmit buffer of the SBiDR register to the parity bit (P). If the SPTEF flag in the SBIi status register (SBiSR) is 0 and the shift register is empty, the SBI reverses the order of the bits of the data to which the parity bit (P) is appended, copies it to the shift register, and fully populates the shift register. When serial transfer starts, the SBI outputs data from the MSB (bit 31) of the shift register, and shifts in the data from bit 8 of the shift register. When the RSPCK cycle required for the serial transfer of 24 bits has passed, received data R00 to P is stored in bits 31 to 8 of the shift register. After completion of the serial transfer, data that existed before the transfer is retained in bits 7 to 0 of the shift register. In this state and the full-duplex synchronous serial communication (TXMD = 0), the SBI copies the data, in which the order of the bits is reversed, from the shift register to the receive buffer of the SBiDR register, and empties the shift register.

If another serial transfer is started before the transmit buffer of the SBiDR register is written received data R00 to R23 is shifted out from the shift register.

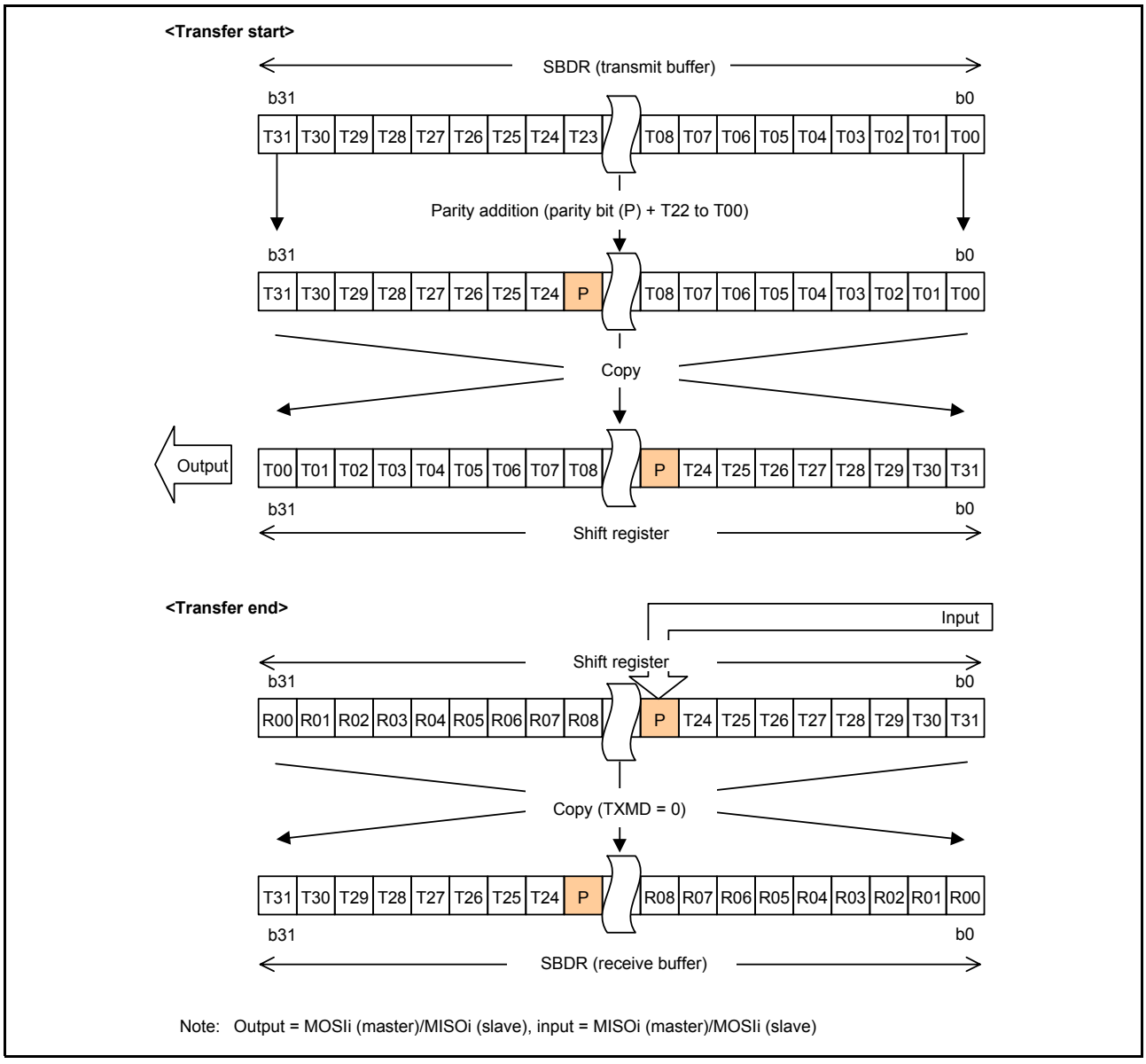


Figure 18.19 LSB First Transfer (24-Bit Data/Parity Function Enabled)

18.3.6 Communication Operating Mode

The full-duplex synchronous serial communication and the transmit-only operation are selected by setting the communication operating mode select bit (TXMD) in the SBIi control register 0 (SBiCR0).
The SBiDR register access described in Figure 18.20 and Figure 18.21 indicates the access condition to the SBIi data register (SBiDR), where the I and the W denote an idle cycle and a write cycle, respectively.

(1) Full-duplex Synchronous Serial Communication (TXMD = 0)

Figure 18.20 shows an operation example when the communication operating mode select bit (TXMD) in the SBIi control register 0 (SBiCR0) is set to 0. The SBI performs an 8-bit serial transfer in the example of Figure 18.20 where the SPFC bits in the SBIi data control register 0 (SBiDCR) are set to 00, and the CPHA bit in the SBIi command register k (SBiCMDk) is set to 1, and the CPOL bit is set to 0. The numbers given under the RSPCK waveform represent the numbers of the RSPCK cycles (i.e., the numbers of transferred bits).

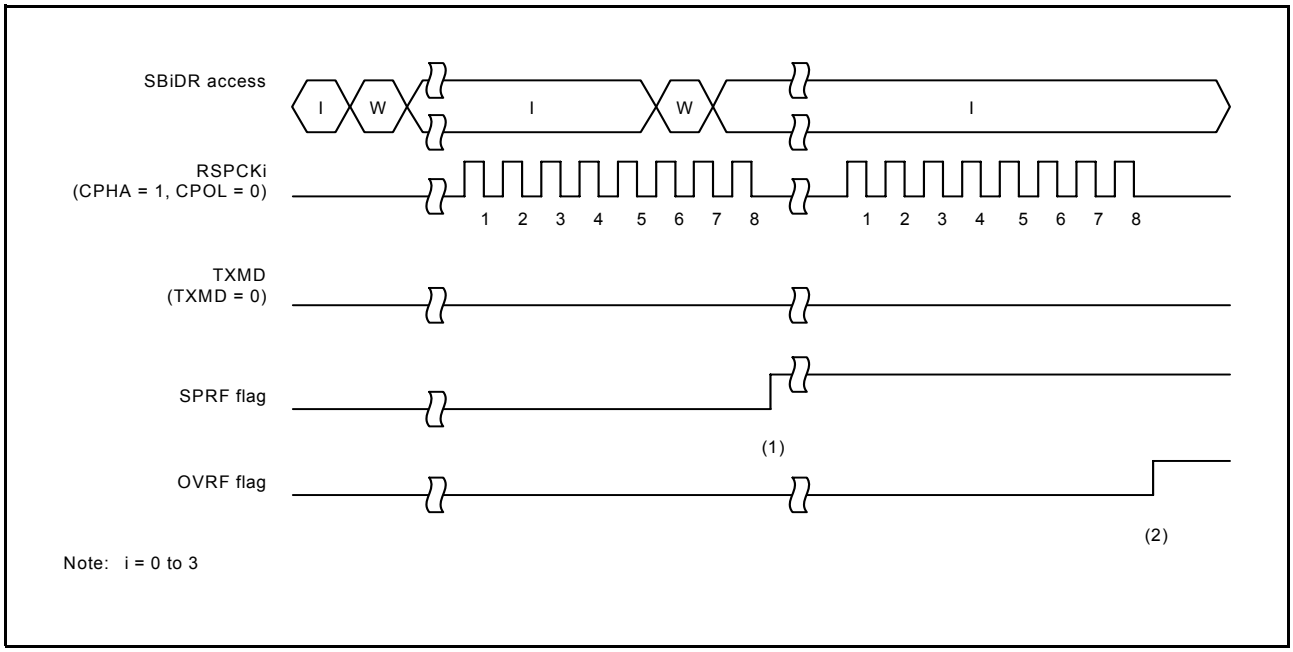


Figure 18.20 Operation Example when TXMD = 0

The operation of the flags at timings shown in steps (1) and (2) in the figure is described below.

(1) When the serial transfer ends with the receive buffer of the SBiDR register being empty, the SBI sets the SPRF flag to 1, and copies the receive data in the shift register to the receive buffer.
(2) When the serial transfer ends with the receive buffer of the SBiDR register having the previous receive data, the SBI sets the OVRF flag to 1, and discards the receive data in the shift register.

In the full-duplex synchronous serial communication (TXMD = 0), the transmit data are transmitted and the receive data are received. Thus, flags SPRF and OVRF turn to 1 at the timing of (1) or (2).

(2) Transmit-only Operation (TXMD = 1)

Figure 18.21 shows an operation example when the communication operating mode select bit (TXMD) in the SBIi control register 0 (SBiCR0) is set to 1. The SBI performs an 8-bit serial transfer in the example of Figure 18.21 where the SPFC bits in the SBIi data control register (SBiDCR) are set to 00, and the CPHA bit in the SBIi command register k (SBiCMDk) is set to 1, and the CPOL bit is set to 0. The numbers given under the RSPCK waveform represent the numbers of the RSPCK cycles (i.e., the numbers of transferred bits).

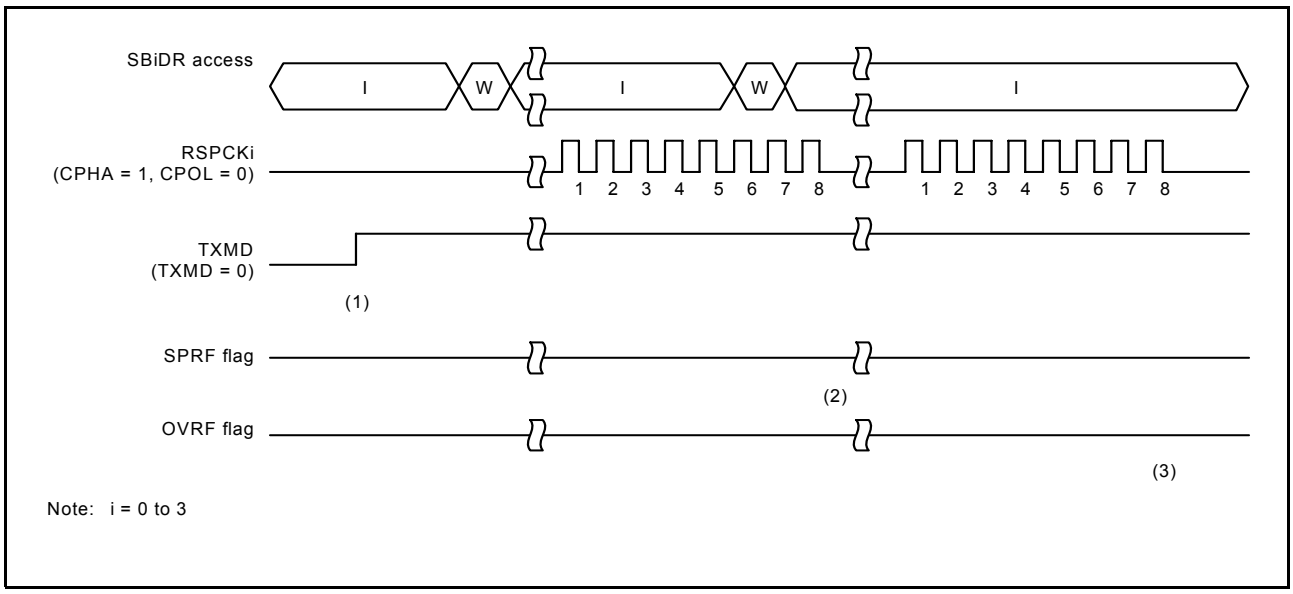


Figure 18.21 Operation Example when TXMD = 1

The operation of the flags at timings shown in steps (1), (2), and (3) in the figure is described below.

(1) Switch to the transmit-only operation (TXMD = 1) after making sure that flags SPRF and OVRF are 0.

(2) When the serial transfer ends with the receive buffer of the SBiDR register being empty, the SPRF flag retains 0 and the data in the shift register is not copied to the receive buffer in the transmit-only operation (TXMD = 1).

(3) Even after the serial transfer ends, the OVRF flag retains 0 and the data in the shift register is not copied to the receive buffer because of no data existed in the receive buffer of the SBiDR register.

In the transmit-only operation (TXMD = 1), the transmit data are transmitted and the receive data are not received. Thus, flags SPRF and OVRF get to retain 0 at the timing of (1), (2), or (3).

18.3.7 Transmit Buffer Empty/Receive Buffer Full Flags

Figure 18.22 shows an example of operation of the SBI transmit buffer empty flag (SPTEF) and the SBI receive buffer full flag in the SBIi status register (SBiSR). The SBiDR access shown in Figure 18.22 indicates the condition of access to the SBIi data register (SBiDR), where I denotes an idle cycle, W a write cycle, and R a read cycle. In the example of Figure 18.22, the SBI performs an 8-bit serial transfer in which the SPFC bit in the SBIi data control register (SBiDCR) are 00, and the CPHA bit in the SBIi command register k (SBiCMDk) is 1, and the CPOL bit is 0. The numbers given under the RSPCK waveform represent the number of RSPCK cycles (i.e., the number of transferred bits).

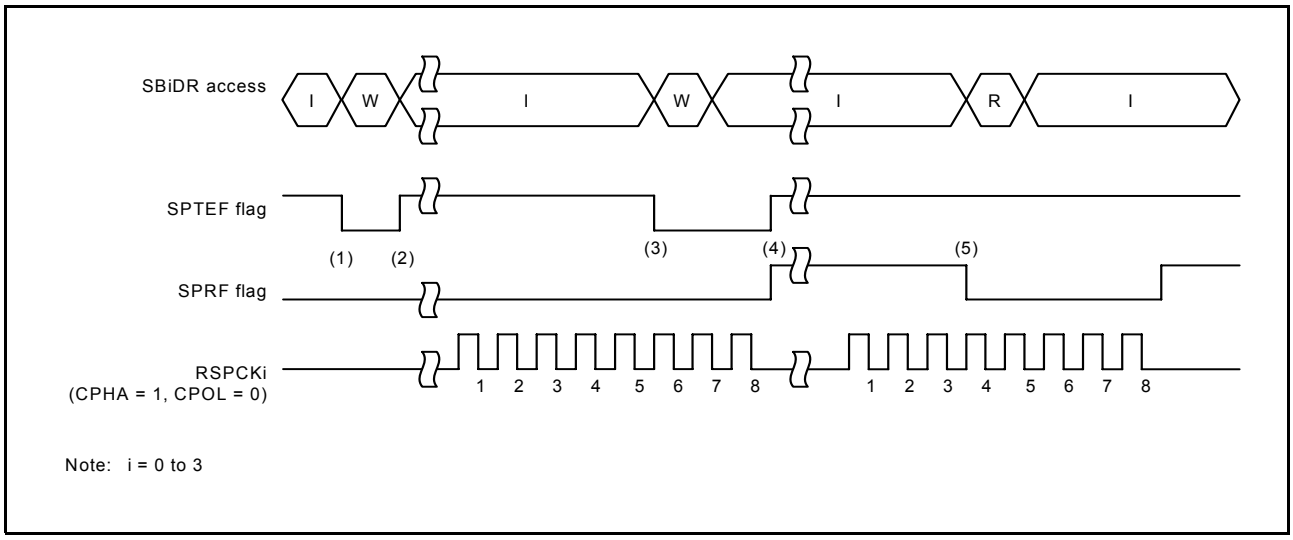


Figure 18.22 SPTEF and SPRF Flag Operation Example

The operation of the flags at timings shown in steps (1) to (5) in the figure is described below.

(1) When transmit data is written to the SBiDR register when the transmit buffer of the SBiDR register is empty, the SBI sets the SPTEF flag to 0, and writes data to the transmit buffer, with no change in the SPRF flag.

(2) If the shift register is empty, the SBI sets the SPTEF flag to 1, and copies the data in the transmit buffer to the shift register, with no change in the SPRF flag. How a serial transfer is started depends on the mode of the SBI. For details, refer to section 18.3.10, SBI Operations, and section 18.3.11, Clock Synchronous Operation.

(3) When transmit data is written to the SBiDR register with the transmit buffer of the SBiDR register being empty, the SBI sets the SPTEF flag to 0, and writes data to the transmit buffer, while the SPRF flag remains unchanged. Because the data being transferred serially is stored in the shift register, the SBI does not copy the data in the transmit buffer to the shift register.

(4) When the serial transfer ends with the receive buffer of the SBiDR register being empty, the SBI sets the SPRF flag to 1, and copies the receive data in the shift register to the receive buffer. Because the shift register becomes empty upon completion of serial transfer, when the transmit buffer had been full before the serial transfer ended, the SBI sets the SPTEF flag to 1, and copies the data in the transmit buffer to the shift register. Even when received data is not copied from the shift register to the receive buffer in an overrun error status, upon completion of the serial transfer the SBI determines that the shift register is empty, and as a result data transfer from the transmit buffer to the shift register is enabled.

(5) When the SBiDR register is read with the receive buffer being full, the SBI sets the SPRF flag to 0, and sends the data in the receive buffer to the bus inside the chip.

If the SBiDR register is written to when the SPTEF flag is 0, the SBI does not update the data in the transmit buffer. When writing to the SBiDR register, make sure that the SPTEF flag is 1. The status that the SPTEF flag is 1 can be checked by reading the SBiSR register or by using an SBI transmit interrupt. To use an SBI transmit interrupt, set the SPTIE bit in the SBiCR0 register to 1.

If the SBI is disabled (the SPE bit in the SBiCR0 register being 0), the SPTEF flag is initialized to 1. For this reason, setting the SPTIE bit to 1 when the SBI is disabled generates an SBI transmit interrupt.

When serial transfer ends with the SPRF flag being 1, the SBI does not copy data from the shift register to the receive buffer, and detects an overrun error (refer to section 18.3.8, Error Detection). To prevent a receive data overrun error, set the SPRF flag to 0 before the serial transfer ends. The status that the SPRF flag is 1 can be checked by either reading the SBiSR register or by using an SBI receive interrupt. To use an SBI receive interrupt, set the SPRIE bit in the SBiCR0 register to 1.

18.3.8 Error Detection

In the normal SBI serial transfer, the data written from the SBIi data register (SBiDR) to the transmit buffer is serially transmitted, and the serially received data can be read from the receive buffer of the SBiDR register. If access is made to the SBiDR register, depending on the status of the transmit buffer/receive buffer or the status of the SBI 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 SBI detects the event as an overrun error, a parity error, or a mode fault error. Table 18.10 shows Relationship between Non-Normal Transfer Operations and SBI Error Detection Function.

Table 18.10 Relationship between Non-Normal Transfer Operations and SBI Error Detection Function

	Occurrence Condition	SBI Operation	Error Detection
A	The SBiDR register is written when the transmit buffer is full.	Keeps the contents of the transmit buffer. Missing write data.	None
B	Serial transfer is started in slave mode when transmit data is still not loaded on the shift register.	Data received in previous serial transfer is serially transmitted.	None
C	The SBiDR register is read when the receive buffer is empty.	Previously received serial data is output.	None
D	Serial transfer terminates when the receive buffer is full.	Keeps the contents of the receive buffer. Missing serial receive data.	Overrun error
E	The erroneous parity bit is received with the parity function being enabled during the full-duplex synchronous serial communication.	Asserts the parity error flag.	Parity error
F	The SSLi0 input signal is asserted when the serial transfer is idle in multi-master mode.	Driving of the RSPCK, MOSIi, SSLi1 to SSLi3 output signals stopped. SBI disabled.	Mode fault error
G	The SSLi0 input signal is asserted during serial transfer in multi-master mode.	Serial transfer suspended. Missing send/receive data. Driving of the RSPCK, MOSIi, SSLi1 to SSLi3 output signals stopped. SBI disabled.	Mode fault error
H	The SSLi0 input signal is negated during serial transfer in slave mode.	Serial transfer suspended. Missing send/receive data. Driving of the MISOi output signal stopped. SBI disabled.	Mode fault error

On operation A shown in Table 18.10, the SBI does not detect an error. To prevent data omission during the writing to the SBiDR register, write operations to the SBiDR register should be executed when the SPTEF flag in the SBIi status register (SBiSR) is 1.

Likewise, the SBI does not detect an error on operation B. In a serial transfer that was started before the shift register was updated, the SBI sends the data that was received in the previous serial transfer, and does not treat the operation indicated in B as an error. Note that the received data from the previous serial transfer is retained in the receive buffer of the SBiDR register, thus it can be correctly read (if the SBiDR register is not read before the end of the serial transfer, an overrun error may result).

Similarly, the SBI does not detect an error on operation C. To prevent extraneous data from being read, the SBiDR register read operation should be executed when the SPRF flag in the SBiSR register is 1.
An overrun error shown in D is described in (1) Overrun Error and a parity error shown in E is described in (2) Parity Error for details. A mode fault error shown in F to H is described in (3) Mode Fault Error. On operations of the SPTEF and SPRF flags in the SBiSR register, refer to section 18.3.7, Transmit Buffer Empty/Receive Buffer Full Flags.

(1) Overrun Error

If serial transfer ends when the receive buffer of the SBIi data register (SBiDR) is full, the SBI detects an overrun error, and sets the OVRF flag in the SBiSR register to 1. When the OVRF flag is 1, the SBI does not copy data from the shift register to the receive buffer so that the data prior to the occurrence of the error is retained in the receive buffer. To reset the OVRF flag in the SBiSR register to 0, either perform a reset, or write a 0 to the OVRF flag after the CPU has read the SBiSR register with the OVRF flag set to 1.
Figure 18.23 shows an example of operation of the SPRF and OVRF flags in the SBiSR register. The SBiSR and SBiDR accesses shown in Figure 18.23 indicates the condition of accesses to the SBiSR and SBiDR registers, respectively, where I denotes an idle cycle, W a write cycle, and R a read cycle. In the example of Figure 18.23, the SBI performs an 8-bit serial transfer in which the CPHA bit in the SBIi command register k (SBiCMDk) is 1, and the CPOL bit is 0. The numbers given under the RSPCK waveform represent the number of RSPCK cycles (i.e., the number of transferred bits).

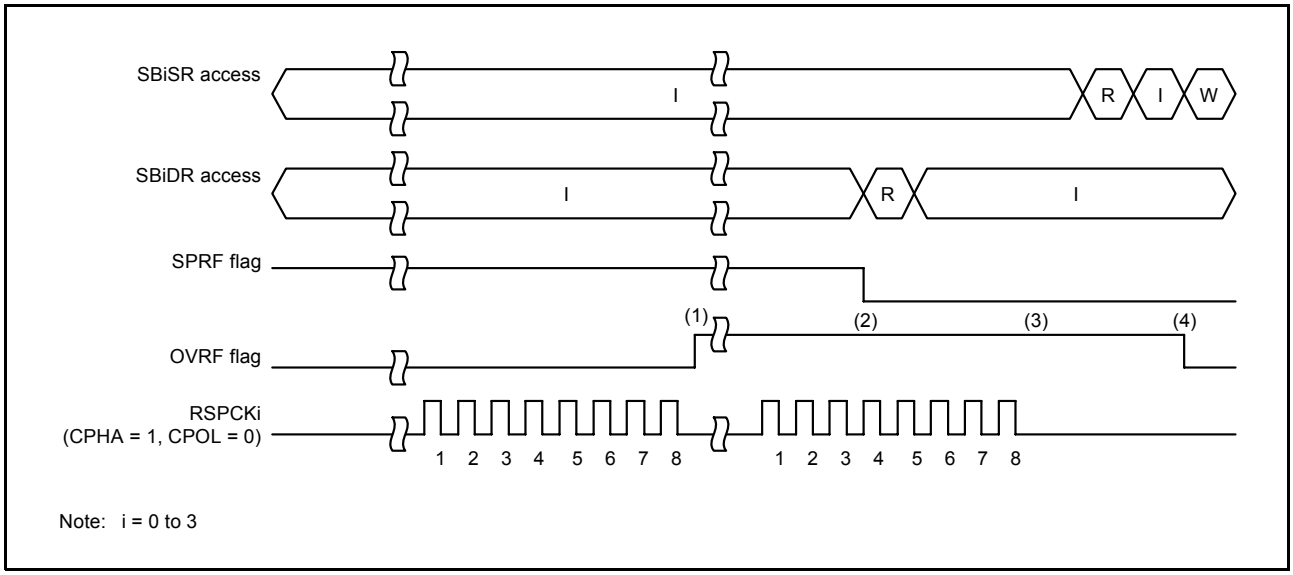


Figure 18.23 SPRF and OVRF Flag Operation Example

The operation of the flags at the timing shown in steps (1) to (4) in the figure is described below.

(1) If a serial transfer terminates with the SPRF flag being 1 (receive buffer full), the SBI detects an overrun error, and sets the OVRF flag to 1. The SBI does not copy the data in the shift register to the receive buffer. Even if the SPPE bit is 1, the detection of parity errors is not performed. In master mode, the SBI copies the value of the pointer to the SBIi command register k (SBiCMDk) to SPECM bits in the SBIi sequence status register (SBiSSR).

(2) When the SBiDR register is read, the SBI sets the SPRF flag to 0, and outputs the data in the receive buffer to an internal bus. The receive buffer becoming empty does not clear the OVRF flag.

(3) If the serial transfer terminates with the OVRF flag being 1 (an overrun error), the SBI keeps the SPRF flag at 0 and does not update it. Likewise, the SBI does not copy the data in the shift register to the receive buffer. Even if the SPPE bit is 1, the detection of parity errors is not performed. When in master mode, the SBI does not update SPECM bits of the SBiSSR register. If, in an overrun error state, the SBI does not copy the received data from the shift register to the receive buffer, upon termination of the serial transfer, the SBI determines that the shift register is empty; in this manner, data transfer is enabled from the transmit buffer to the shift register.

(4) If the CPU writes a 0 to the OVRF flag after reading the SBiSR register when the OVRF flag is 1, the SBI clears the OVRF flag.

The occurrence of an overrun can be checked either by reading the SBiSR register or by using an SBI error interrupt and reading the SBiSR register. When using an SBI error interrupt, set the SPEIE bit in the SBIi control register 0 (SBiCR0) to 1. When executing a serial transfer without using an SBI error interrupt, measures should be taken to ensure the early detection of overrun errors, such as reading the SBiSR register immediately after the SBiDR register is read. When the SBI is run in master mode, the pointer value to the SBiCMDk register can be checked by reading the SPECM bits of the SBiSSR register.

If an overrun error occurs and the OVRF flag is set to 1, normal reception operations cannot be performed until such time as the OVRF flag is cleared. The OVRF flag is cleared to 0 under the following conditions:

- After reading the SBiSR register in a condition in which the OVRF flag is set to 1, the CPU writes a 0 to the OVRF flag.
- The reset is performed.

(2) Parity error

After the full-duplex synchronous serial communication is performed and the transfer ends with the TXMD bit in the SBIi control register 0 (SBiCR0) being 0 and the SPPE bit in the SBIi control register 1 (SBiCR1) being 1, the determination of parity errors is executed. The SBI sets the PERF flag in the SBIi status register (SBiSR) to 1 after detecting a parity error.

While the OVRF flag is 1, the detection of parity errors to the receive data is not performed because the SBI does not copy the data in the shift register to the receive buffer. To set the PERF flag in the SBiSR register to 0, a reset needs to be performed or 0 needs to be written to the PERF flag after the CPU reads the SBiSR register in which the PERF flag is set to 1.

Figure 18.24 shows the operation of flags OVRF and PERF in the SBiSR register. The SBiSR access described in Figure 18.24 indicates the access condition to the SBiSR register, where the I and the W denote an idle state and a write cycle, respectively. In the example of Figure 18.24, the full-duplex synchronous serial communication is performed while the TXMD bit in the SBIi control register 0 (SBiCR0) is 0 and the SPPE bit in the SBIi control register 1 (SBiCR1) is 1. The SBI performs an 8-bit serial transfer while the CPHA bit in the SBIi command register k (SBiCMDk) is set to 1, and the CPOL bit is set to 0. The numbers given under the RSPCK waveform represent the numbers of the RSPCK cycles (i.e., the numbers of transferred bits).

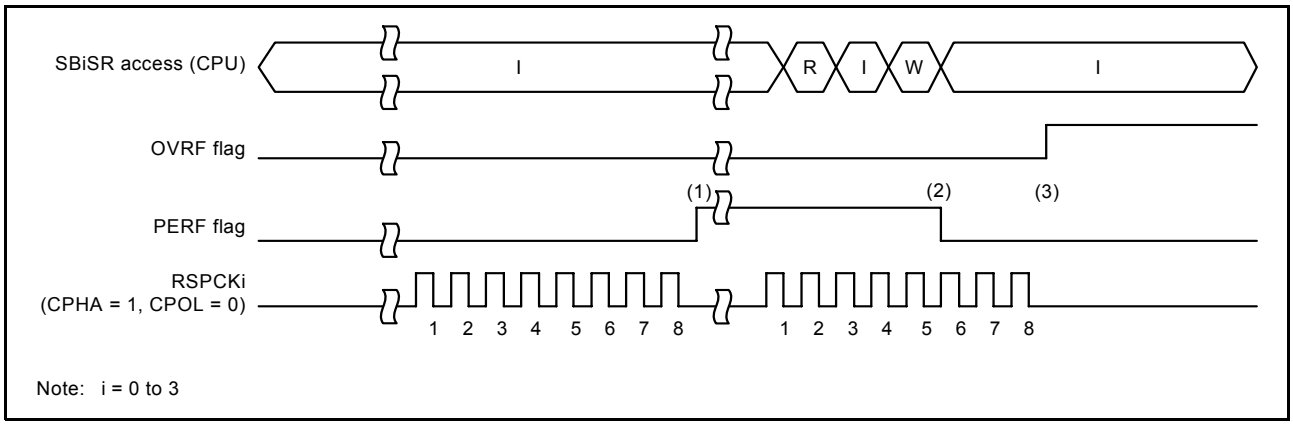


Figure 18.24 PERF Flag Operation Example

The operation of the flags at timings shown in steps (1), (2), and (3) in the figure is described below.

(1) When the serial transfer ends with the SBI not detecting an overrun error, the data in the shift register is copied to the receive buffer. At this time the SBI detects a parity error by determining the receive data, and then sets the PERF flag to 1. In master mode, the pointer value to the SBIi command register k (SBiCMDk) is copied to the SPECM bits in the SBIi sequence status register (SBiSSR).

(2) When reading the SBiSR register and then writing 0 to the PERF flag with the PERF flag being 1, the SBI clears the OVRF flag.

(3) When the SBI detects an overrun error and the serial transfer ends, the data in the shift register is not copied to the receive buffer. At this time the SBI does not detect a parity error.

The occurrence of a parity error can be confirmed by the read from the SBiSR register, or the SBI error interrupt and the read from the SBiSR register. When using the SBI error interrupt, set the SPEIE bit in the SBIi control register 0 (SBiCR0) to 1. When executing the serial transfer without using the SBI error interrupt, reading the SBiSR register, etc. need to be done to detect the occurrence of a parity error in an early stage. When the SBI is used in master mode, reading the SPECM bits in the SBiSSR register allows to verify the pointer value to the SBiCMDk register when an error occurs.

The PERF flag is cleared to 0 under the following conditions:

- After reading the SBiSR register in a condition where the PERF flag is set to 1, the CPU writes a 0 to the PERF flag.
- The reset is performed.

(3) Mode Fault Error

The SBI operates in multi-master mode when the MSTR bit in the SBIi control register 0 (SBiCR0) is 1, the SPMS bit is 0, and the MODFEN bit is also 1. If the active level is input with respect to the SSLi0 input signal of the SBI in multi-master mode, the SBI detects a mode fault error irrespective of the status of the serial transfer, and sets the MODF flag in the SBIi status register (SBiSR) to 1. Upon detecting the mode fault error, the SBI copies the value of the pointer to the SBIi command register k (SBiCMDk) to the SPECM bits in the SBI sequence status register (SBiSSR). The active level of the SSLi0 signal is determined by the SSL0P bit in the SBIi slave select polarity register (SBiSSLP).

When the MSTR bit is 0, the SBI operates in slave mode. The SBI detects a mode fault error if the MODFEN bit in the SBI in slave mode is 1, and the SPMS bit is 0, and if the SSLi0 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 SBI stops driving of the output signals and clears the SPE bit in the SBiCR0 register to 0 (refer to section 18.3.9, Initializing SBI). When the SPE bit is set to 0, the SBI disables the SBI function. In the case of multi-master configuration, detection of a mode fault error is used to stop driving of the output signals and SBI function, which allows the master right to be released.

The occurrence of a mode fault error can be checked either by reading the SBiSR register or by using an SBI error interrupt and reading the SBiSR register. When using an SBI error interrupt, set the SPEIE bit in the SBIi control register 0 (SBiCR0) to 1. To detect a mode fault error without using an SBI error interrupt, it is necessary to poll the SBiSR register. When using the SBI in master mode, reading the SPECM bits of the SBiSSR register allows to verify the value of the pointer to the SBiCMDk register when an error occurs.
When the MODF flag is 1, the SBI ignores the writing of the value 1 to the SPE bit by the CPU. To enable the SBI function after the detection of a mode fault error, the MODF flag must be set to 0. The MODF flag is cleared to 0 under the following conditions:

- After reading the SBiSR register in a condition where the MODF flag is set to 1, the CPU writes a 0 to the MODF flag.
- The reset is performed.

18.3.9 Initializing SBI

If the CPU writes a 0 to the SPE bit in the SBIi control register 0 (SBiCR0) or the SBI clears the SPE bit to 0 because of the detection of a mode fault error, the SBI disables the SBI function, and initializes a part of the module function. When a reset is generated, the SBI initializes all of the module function. The following describes initialization by the clearing of the SPE bit and initialization by a reset.

(1) Initialization by Clearing the SPE Bit

When the SPE bit in the SBiCR0 register is cleared, the SBI 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 SBI
- Initializing the SPTEF flag in the SBIi status register (SBiSR)

Initialization by the clearing of the SPE bit does not initialize the control bits of the SBI. For this reason, the SBI can be started in the same transfer mode as prior to the initialization if the CPU resets the value 1 to the SPE bit.
The SPRF, OVRF, and MODF flags in the SBiSR register are not initialized, nor is the value of the SBIi sequence status register (SBiSSR) initialized. For this reason, even after the SBI is initialized, data from the receive buffer can be read in order to check the status of error occurrence during an SBI transfer.
The SPTEF flag in the SBiSR register is initialized to 1. Therefore, if the SPTIE bit in the SBiCR0 register is set to 1 after SBI initialization, an SBI transmit interrupt is generated. When the SBI is initialized by the CPU, in order to disable any SBI transmit interrupt, a 0 should be written to the SPTIE bit simultaneously with the writing of a 0 to the SPE bit. To disable any SBI transmit interrupt after a mode fault error is detected, use an error handling routine to write a 0 to the SPTIE bit.

(2) Reset

The initialization by a reset completely initializes the SBI through the initialization of all bits for controlling the SBI, initialization of the status bits, and initialization of data registers, in addition to the requirements described in section 18.3.9 (1), Initialization by Clearing the SPE Bit.

18.3.10 SBI Operations

(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 18.3.8, Error Detection). When operating in single-master mode, the SBI does not detect mode fault errors whereas the SBI running in multi-master mode does detect mode fault errors. This section explains operations that are common to single-/multi-master modes.

(1-1) Starting a Serial Transfer

The SBI updates the data in the transmit buffer when data is written to the SBIi data register (SBiDR) with the SPTEF flag in the SBIi status register (SBiSR) set to 1. If the shift register is empty in a condition where the SPTEF flag has been cleared to 0 due to the writing of 0 either after the writing to the SBiDR register or by the writing of 0 after the value 1 is read from the SPTEF flag by the CPU, the SBI copies the data in the transmit buffer to the shift register and starts a serial transfer. Upon copying transmit data to the shift register, the SBI 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 from the CPU.

For details on the SBI transfer format, refer to section 18.3.4, Transfer Format. The polarity of the SSL output signal depends on the setting value of the SBIi slave select polarity register (SBiSSLP).

(1-2) Terminating a Serial Transfer

Irrespective of the CPHA bit in the SBIi command register k (SBiCMDk), the SBI terminates the serial transfer after transmitting an RSPCK edge corresponding to the final sampling timing. If the SPRF flag in the SBIi status register (SBiSR) is 0 and free space is available in the receive buffer, upon termination of serial transfer the SBI copies data from the shift register to the receive buffer of the SBIi data register (SBiDR).

It should be noted that the final sampling timing varies depending on the bit length of transfer data. In master mode, the SBI data length depends on the settings in the SPB bits in the SBiCMDk register. For details on the SBI transfer format, refer to section 18.3.4, Transfer Format.

(1-3) Sequence Control

The transfer format that is employed in master mode is determined by the SBIi sequence control register (SBiSCR), SBIi command register k (SBiCMDk), the SBIi bit rate register (SBiBR), the SBIi clock delay register (SBiCKD), the SBIi slave select negation delay register (SBiSSLND), and the SBIi next-access delay register (SBiND).

The SBiSCR register is a register used to determine the sequence configuration for serial transfers that are executed by a master mode SBI. The following items are set in the SBiCMDk registers: SSL output signal value, MSB/LSB first, data length, some of the bit rate settings, RSPCK polarity/phase, whether the SBiCKD register is to be referenced, whether the SBiSSLND register is to be referenced, and whether the SBiND register is to be referenced. The SBiBR register holds some of the bit rate settings; the SBiCKD register, an SBI clock delay value; the SBiSSLND register, an SSL negation delay; and the SBiND register, a next-access delay value.

According to the sequence length that is assigned to the SBiSCR register, the SBI makes up a sequence comprised of a part or all of the SBiCMDk registers. The SBI contains a pointer to the SBiCMDk register that makes up the sequence. The CPU can check the value of this pointer by reading the SPCP bits in the SBIi sequence status register (SBiSSR).

When the SPE bit in the SBIi control register 0 (SBiCR0) is set to 1 and the SBI function is enabled, the SBI loads the pointer to the commands in the SBiCMD0 register, and incorporates the SBiCMD0 register settings into the transfer format at the beginning of serial transfer. The SBI 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 SBI sets the pointer in the SBiCMD0 register, and in this manner the sequence is executed repeatedly.

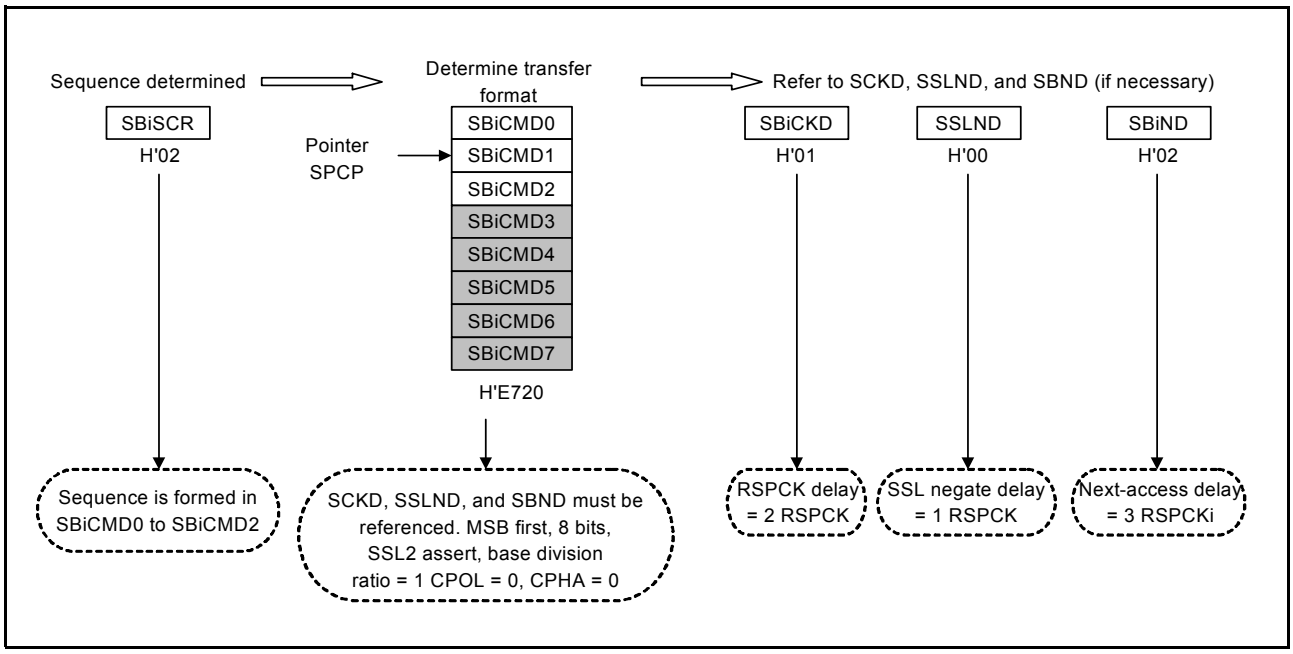


Figure 18.25 Determination Procedure of Serial Transfer Mode in Master Mode

(1-4) Burst Transfer

If the SSLKP bit in the SBIi command register k (SBiCMDk) that the SBI references during the current serial transfer is 1, the SBI keeps the SSL signal level during the serial transfer until the beginning of the SSL signal assertion for the next serial transfer. If the SSL signal level for the next serial transfer is the same as the SSL signal level for the current serial transfer, the SBI can execute continuous serial transfers while keeping the SSL signal assertion status (burst transfer).

Figure 18.26 shows an example of an SSL signal operation for the case where a burst transfer is implemented using the SBiCMD0 and SPCMD1 registers settings. The text below explains the SBI operations (1) to (7) as shown in Figure 18.26. It should be noted that the polarity of the SSL output signal depends on the settings in the SBIi slave select polarity register (SBiSSLP).

(1) Based on the SBiCMD0 register, the SBI asserts the SSL signal and inserts RSPCK delays.
(2) The SBI executes serial transfers according to the SBiCMD0 register.
(3) The SBI inserts SSL negation delays.
(4) Because the SSLKP bit in the SBiCMD0 register is 1, the SBI keeps the SSL signal value on the SBiCMD0 register. This period is sustained, at the shortest, for a period equal to the next-access delay of the SBiCMD0 register. If the shift register is empty after the passage of a minimum period, this period is sustained until such time as the transmit data is stored in the shift register for another transfer.
(5) Based on the SBiCMD1 register, the SBI asserts the SSL signal and inserts RSPCK delays.
(6) The SBI executes serial transfers according to the SBiCMD1 register.
(7) Because the SSLKP bit in the SBiCMD1 register is 0, the SBI negates the SSL signal. In addition, a next-access delay is inserted according to the SBiCMD1 register.

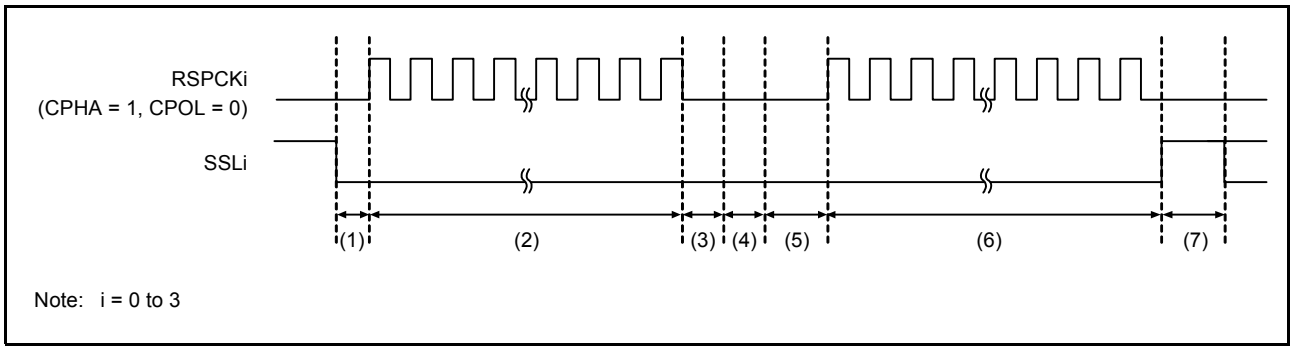


Figure 18.26 Example of Burst Transfer Operation using SSLKP Bit

If the SSL signal settings in the SBiCMDk register in which 1 is assigned to the SSLKP bit are different from the SSL signal output settings in the SBiCMDk register to be used in the next transfer, the SBI switches the SSL signal status to SSL signal assertion ((5) in Figure 18.26) corresponding to the command for the next transfer. Note that if such an SSL signal switching occurs, the slaves that drive the MISOi signal compete, and the collision of signal levels may occur. The SBI in master mode references the SSL signal operation within the module for the case where the SSLKP bit is not used. Even when the CPHA bit in the SBiCMDk register is 0, the SBI can accurately start serial transfers by asserting the SSL signal for the next transfer. For this reason, burst transfers in master mode can be executed irrespective of CPHA bit settings (refer to section 18.3.10, SBI Operations).

(1-5) RSPCK Delay (t1)

The RSPCK delay value of the SBI in master mode depends on SCKDEN bit settings in the SBIi command register k (SBiCMDk) and on SBIi clock delay register (SBiCKD) settings. The SBI determines the SBiCMDk register to be referenced during serial transfer by pointer control, and determines an RSPCK delay value during serial transfer by using the SCKDEN bit in the selected the SBiCMDk register and the SBiCKD registers, as shown in Table 18.11. For a definition of RSPCK delay, refer to section 18.3.4, Transfer Format.

Table 18.11 Relationship among SCKDEN and SBiCKD Settings and RSPCK Delay Values

SCKDEN Bit	SBiCKD Register	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

(1-6) SSL Negation Delay (t2)

The SSL negation delay value of the SBI in master mode depends on SLNDEN bit settings in the SBIi command register k (SBiCMDk) and on the SBIi slave select negate delay register (SBiSSLND) settings. The SBI determines the SBiCMDk register to be referenced during serial transfer by pointer control, and determines an SSL negation delay value during serial transfer by using the SLNDEN bit in the selected the SBiCMDk register and the SBiSSLND register, as shown in Table 18.12. For a definition of SSL negation delay, refer to section 18.3.4, Transfer Format.

Table 18.12 Relationship among SBiSSLND Settings and SSL Negation Delay Values

SLNDEN Bit	SBiSSLND Register	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

(1-7) Next-Access Delay (t3)

The next-access delay value of the SBI in master mode depends on SPNDEN bit settings in the SBIi command register k (SBiCMDk) and the SBIi next-access delay register (SBiND) settings. The SBI determines the SBiCMDk register to be referenced during serial transfer by pointer control, and determines a next-access delay value during serial transfer by using the SPNDEN bit in the selected the SBiCMDk and SPND registers, as shown in Table 18.13. For a definition of next-access delay, refer to section 18.3.4, Transfer Format.

Table 18.13 Relationship among SPNDEN and SBiND Settings and Next-Access Delay Values

SPNDEN Bit	SBiND Register	Next-Access Delay Value
0	000 to 111	1 RSPCK + 2 cycles of the peripheral bus clock B
1	000	1 RSPCK + 2 cycles of the peripheral bus clock B
	001	2 RSPCK + 2 cycles of the peripheral bus clock B
	010	3 RSPCK + 2 cycles of the peripheral bus clock B
	011	4 RSPCK + 2 cycles of the peripheral bus clock B
	100	5 RSPCK + 2 cycles of the peripheral bus clock B
	101	6 RSPCK + 2 cycles of the peripheral bus clock B
	110	7 RSPCK + 2 cycles of the peripheral bus clock B
	111	8 RSPCK + 2 cycles of the peripheral bus clock B

(1-8) Initialization Flowchart

Figure 18.27 is a flowchart illustrating an example of initialization in SBI operation when the SBI is used in master mode. For a description of how to set up an interrupt controller, the DMAC, and input/output ports, refer to the descriptions given in the individual blocks.

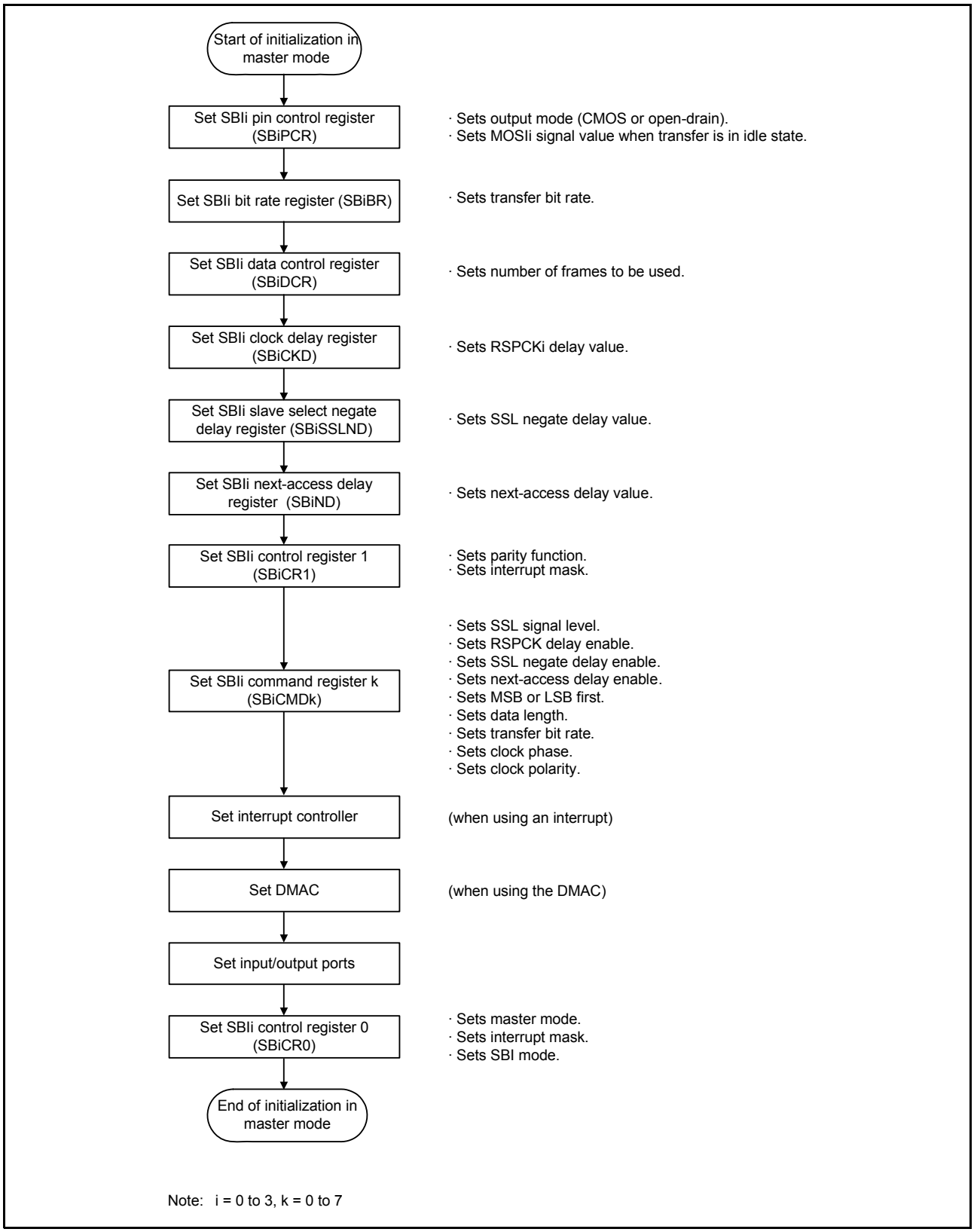
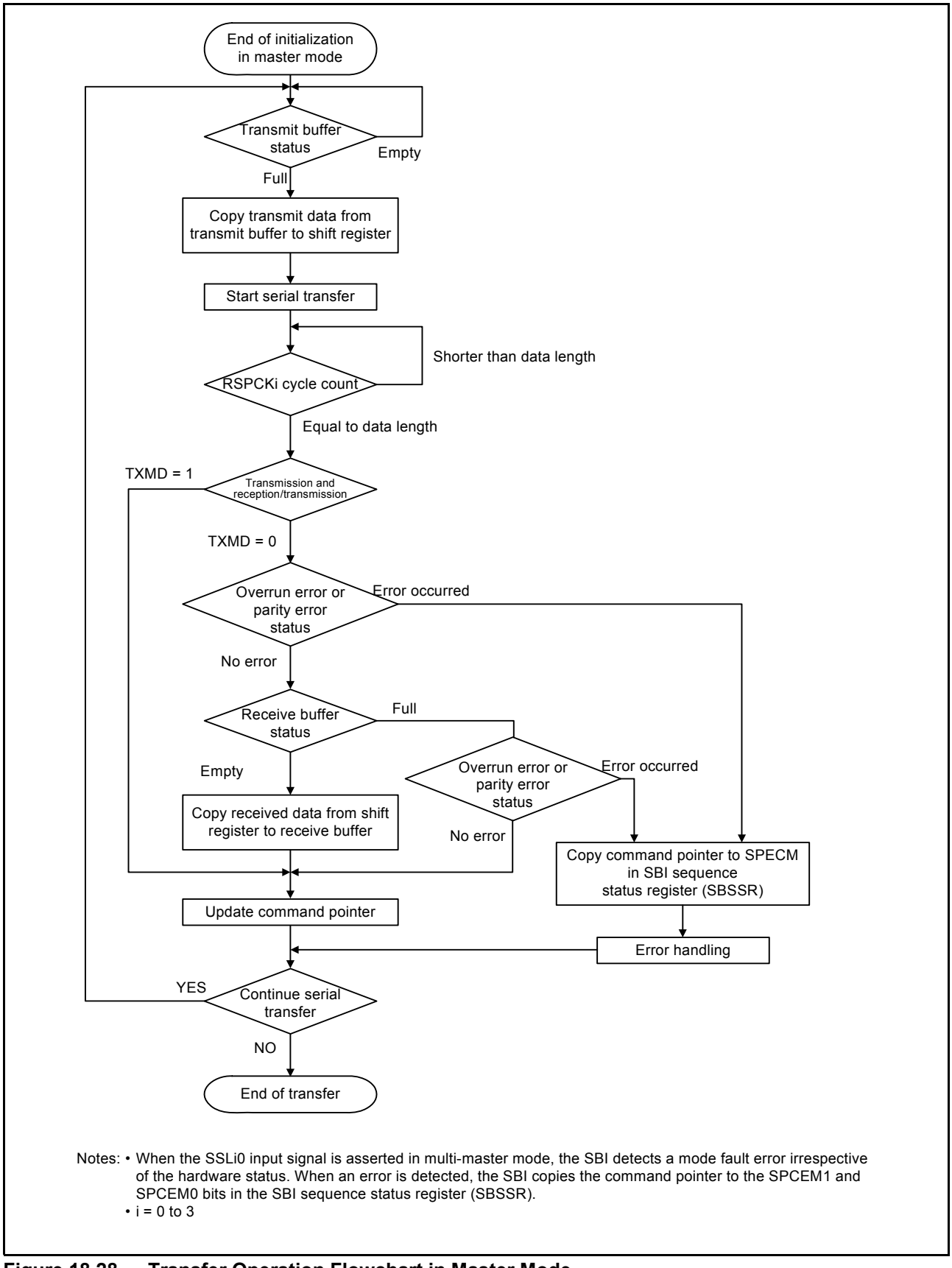


Figure 18.27 Example of Initialization Flowchart in Master Mode

(1-9) Transfer Operation Flowchart

Figure 18.28 is a flowchart illustrating a transfer in SBI operation when the SBI is used in master mode.



Notes: • When the SSLi0 input signal is asserted in multi-master mode, the SBI detects a mode fault error irrespective of the hardware status. When an error is detected, the SBI copies the command pointer to the SPCEM1 and SPCEM0 bits in the SBI sequence status register (SBSSR).
• i = 0 to 3

Figure 18.28 Transfer Operation Flowchart in Master Mode

(2) Slave Mode Operation

(2-1) Starting a Serial Transfer

If the CPHA bit in SBIi command register 0 (SBiCMD0) is 0, when detecting an SSLi0 input signal assertion, the SBI needs to start driving valid data to the MISOi output signal. For this reason, when the CPHA bit is 0, the assertion of the SSLi0 input signal triggers the start of a serial transfer.

If the CPHA bit is 1, when detecting the first RSPCK edge in an SSLi0 signal asserted condition, the SBI needs to start driving valid data to the MSOi signal. For this reason, when the CPHA bit is 1, the first RSPCK edge in an SSLi0 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 SBI changes the status of the shift register to “full”, so that data cannot be copied from the transmit buffer to the shift register when serial transfer is in progress. If the shift register was full before the serial transfer started, the SBI leaves the status of the shift register unchanged, in the full state.

Irrespective of CPHA bit settings, the timing at which the SBI starts driving MISOi output signals is the SSLi0 signal assertion timing. The data which is output by the SBI is either valid or invalid, depending on CPHA bit settings.

For details on the SBI transfer format, refer to section 18.3.4, Transfer Format. The polarity of the SSLi0 input signal depends on the setting of the SSL0P bit in the SBIi slave select polarity register (SBiSSLP).

(2-2) Terminating a Serial Transfer

Irrespective of the CPHA bit in SBI command register 0 (SBiCMD0), the SBI terminates the serial transfer after detecting an RSPCK edge corresponding to the final sampling timing. When the SPRF flag in the SBIi status register (SBiSR) is 0 and free space is available in the receive buffer, upon termination of serial transfer the SBI copies received data from the shift register to the receive buffer of the SBIi data register (SBiDR). Irrespective of the value of the SPRF flag, upon termination of a serial transfer the SBI changes the status of the shift register to “empty”. A mode fault error occurs if the SBI detects an SSLi0 input signal negation from the beginning of serial transfer to the end of serial transfer (refer to section 18.3.8, Error Detection).

The final sampling timing changes depending on the bit length of the transfer data. In slave mode, the SBI data length depends on the settings in the SPB bits in the SBiCMD0 register. The polarity of the SSLi0 input signal depends on the setting in the SSL0P bit in the SBIi slave select polarity register (SBiSSLP). For details on the SBI transfer format, refer to section 18.3.4, Transfer Format.

(2-3) Notes on Single-Slave Operations

If the CPHA bit in SBI command register 0 (SBiCMD0) is 0, the SBI starts serial transfers when it detects the assertion edge for an SSLi0 input signal. In the type of configuration shown in Figure 18.4 as an example, if the SBI is used in single-slave mode, the SSLi0 signal is always fixed at active state. Therefore, when the CPHA bit is set to 0, the SBI cannot correctly start a serial transfer. To correctly execute send/receive operation by the SBI in a configuration in which the SSLi0 input signal is fixed at active state, the CPHA bit should be set to 1. If there is a need for setting the CPHA bit to 0, the SSLi0 input signal should not be fixed.

(2-4) Burst Transfer

If the CPHA bit in SBIi command register 0 (SBiCMD0) is 1, continuous serial transfer (burst transfer) can be executed while retaining the assertion state for the SSLi0 input signal. If the CPHA bit is 1, the period from the first RSPCK edge to the sampling timing for the reception of the final bit in an SSLi0 signal active state corresponds to a serial transfer period. Even when the SSLi0 input signal remains at the active level, the SBI can accommodate burst transfers because it can detect the start of access.

If the CPHA bit is 0, for the reason given in 18.3.10 (2) (2-3), Notes on Single-Slave Operations, second and subsequent serial transfers during the burst transfer cannot be executed correctly.

(2-5) Initialization Flowchart

Figure 18.29 is a flowchart illustrating an example of initialization in SBI operation when the SBI is used in slave mode. For a description of how to set up an interrupt controller, the DMAC, and input/output ports, refer to the descriptions given in the individual blocks.

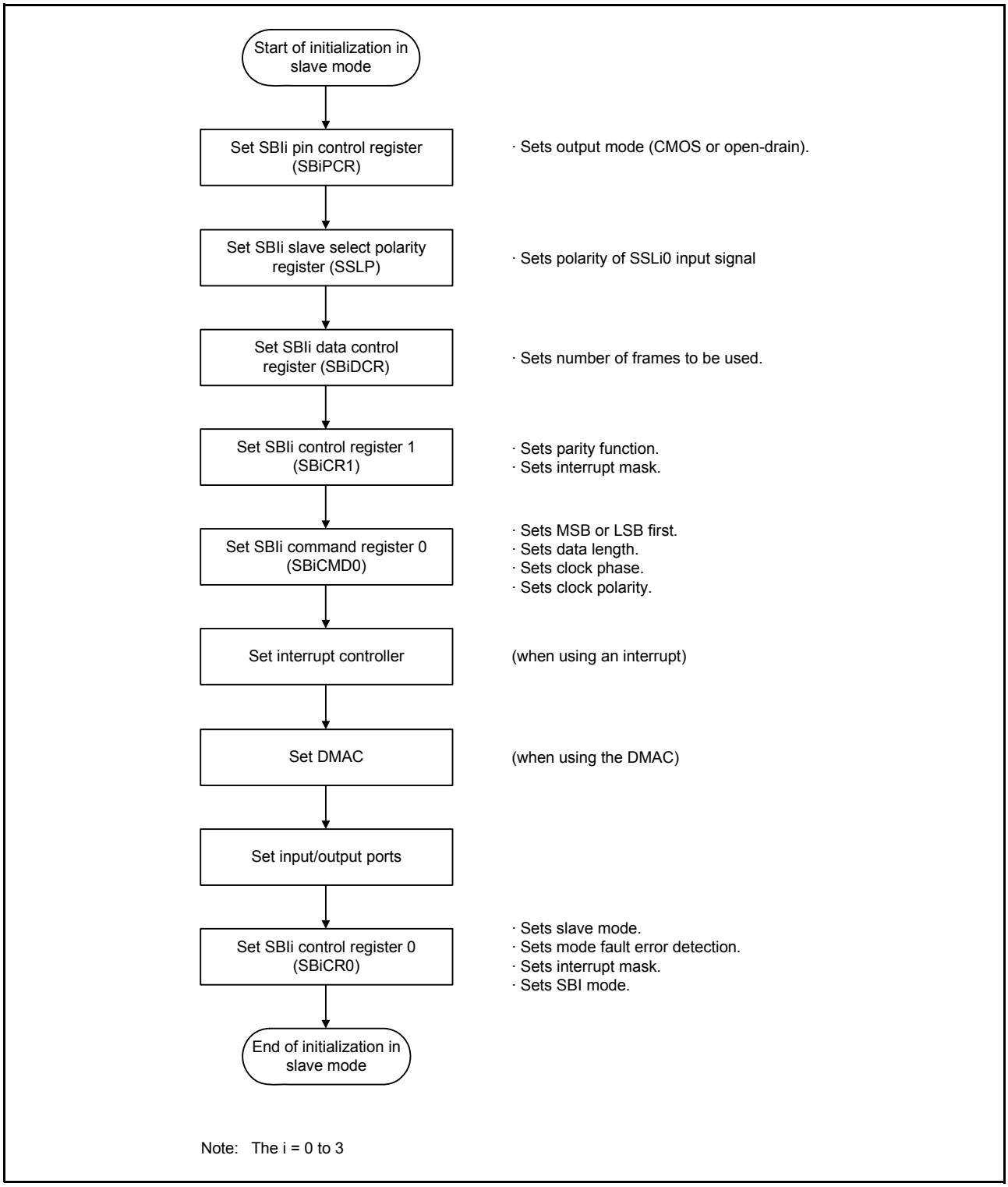


Figure 18.29 Example of Initialization Flowchart in Slave Mode

(2-6) Transfer Operation Flowchart (CPHA = 0)

Figure 18.30 is a flowchart illustrating a transfer in SBI operation when the SBI is used in slave mode with the CPHA bit in the SBIi command register 0 (SBiCMD0) set to 0.

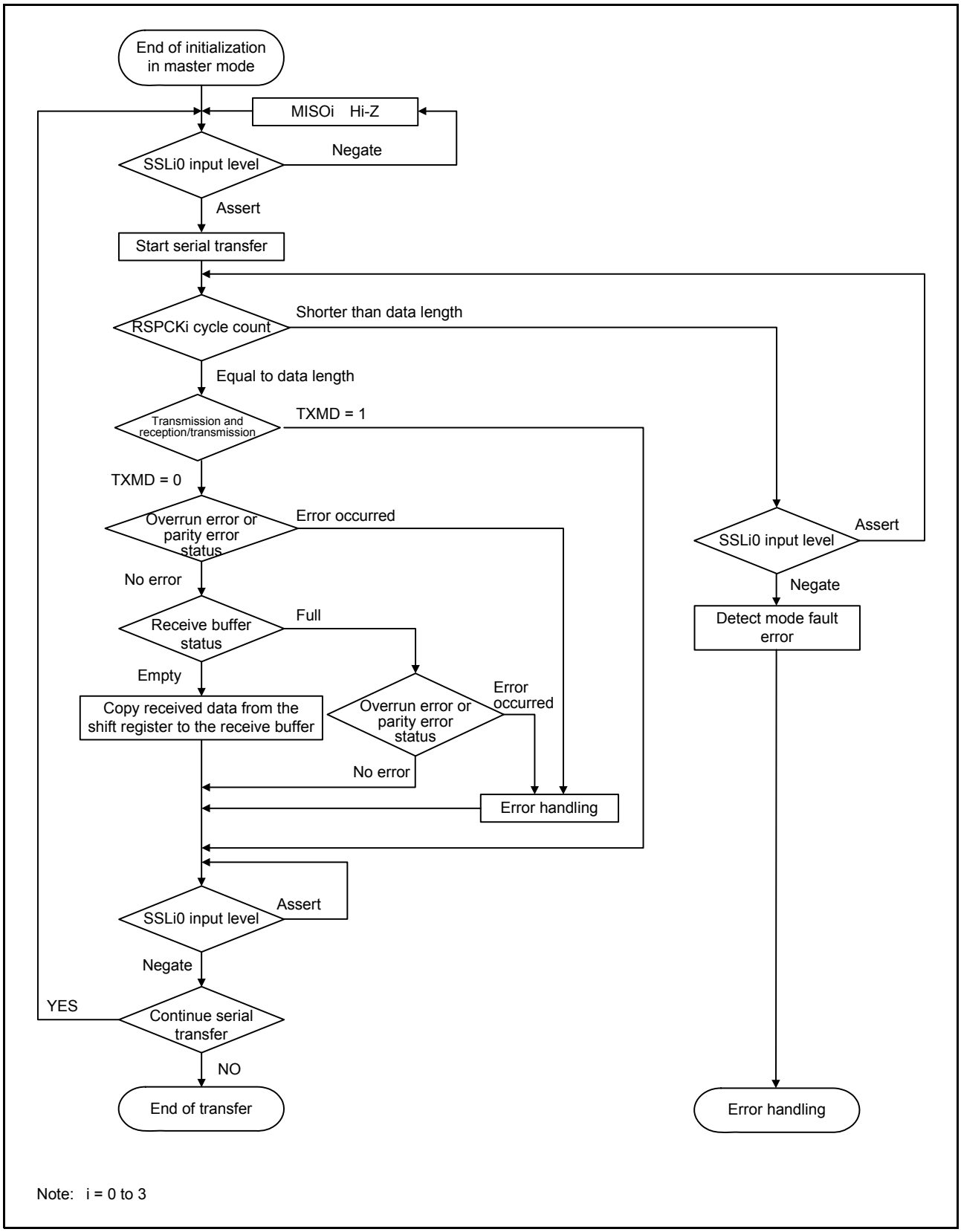


Figure 18.30 Transfer Operation Flowchart in Slave Mode (CPHA = 0)

(2-7) Transfer Operation Flowchart (CPHA = 1)

Figure 18.31 is a flowchart illustrating a transfer in SBI operation when the SBI is used in slave mode with the CPHA bit in the SBIi command register 0 (SBiCMD0) set to 1, respectively.

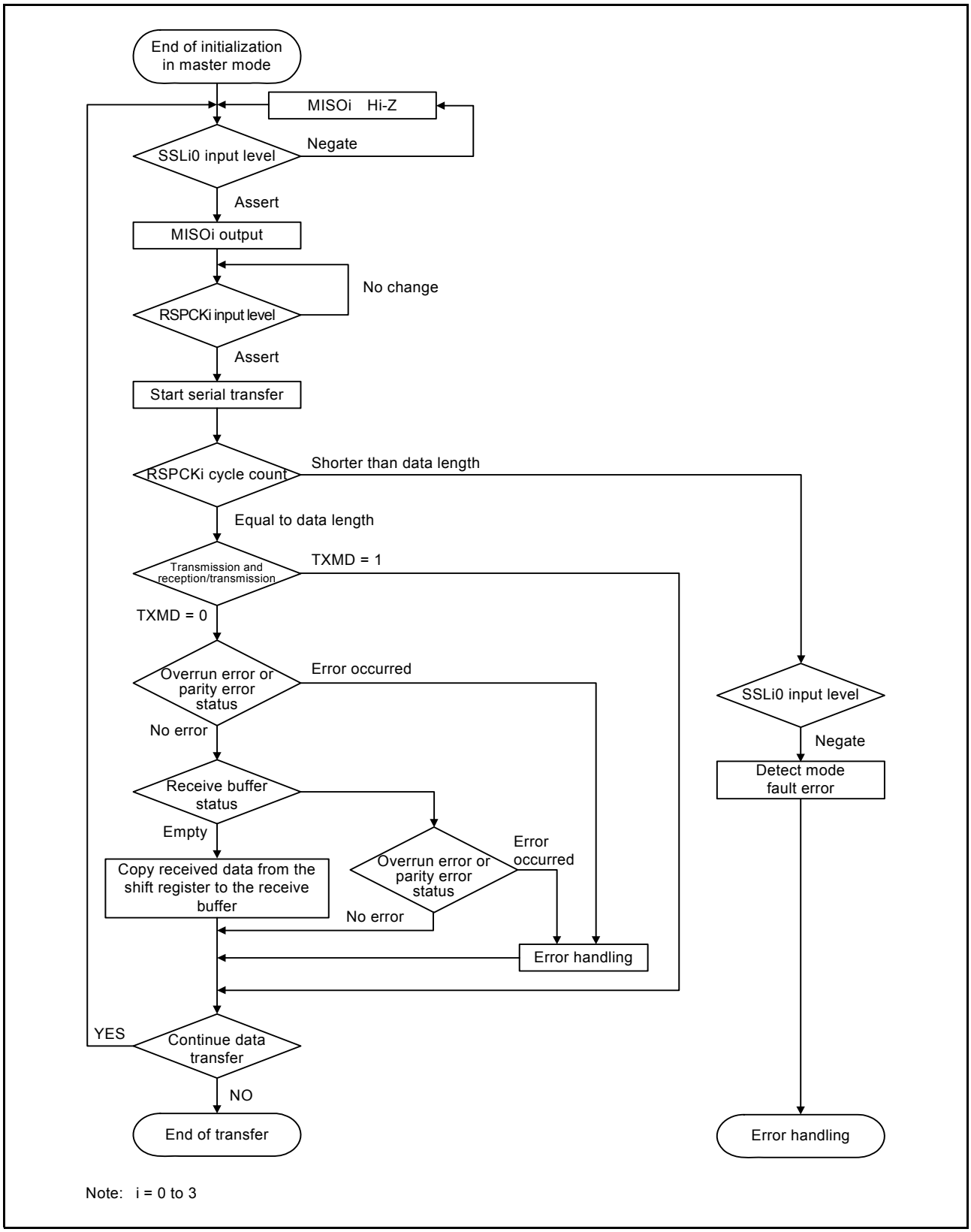


Figure 18.31 Transfer Operation Flowchart in Slave Mode (CPHA = 1)

18.3.11 Clock Synchronous Operation

Setting the SPMS bit in the SBIi control register 0 (SBiCR0) to 1 selects clock synchronous operation of the SBI. In clock synchronous operation, the SSLi pins are not used, and the three pins of RSPCK, MOSIi, and MISOi handle communications. The SSLi pins are available as I/O port pins.

Although clock synchronous operation does not require use of the SSLi pins, operation of the module is the same as in SBI operation. That is, in both master and slave operations, communications can be performed with the same flow as in SBI operation. However, mode fault errors are not detected because the SSLi pins are not used.

Furthermore, operation is not guaranteed if clock synchronous operation proceeds when the CPHA bit in the SBIi command register k (SBiCMDk) is set to 0.

(1) Master Mode Operation

(1-1) Starting Serial Transfer

The SBI updates the data in the transmit buffer when data is written to the SBIi data register (SBiDR) with the SPTEF flag in the SBIi status register (SBiSR) set to 1. If the shift register is empty in a condition where the SPTEF flag has been cleared to 0 due to the writing of 0 either after the writing to the SBiDR register or by the writing of 0 after the value 1 is read from the SPTEF flag by the CPU, the SBI copies the data in the transmit buffer to the shift register and starts a serial transfer. Upon copying transmit data to the shift register, the SBI 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 from the CPU.

For details on the SBI transfer format, refer to section 18.3.4, Transfer Format. However, communications are performed without using the SSLi0 output signal in the clock synchronous operation.

(1-2) Terminating a Serial Transfer

The SBI terminates the serial transfer after transmitting an RSPCK edge corresponding to the sampling timing. If the SPRF flag in the SBIi status register (SBiSR) is 0 and free space is available in the receive buffer, upon termination of serial transfer, the SBI copies data from the shift register to the receive buffer of the SBIi data register (SBiDR).

It should be noted that the final sampling timing varies depending on the bit length of transfer data. In master mode, the SBI data length depends on the settings in the SPB bits in the SBIi command register k (SBiCMDk). For details on the SBI transfer format, refer to section 18.3.4, Transfer Format. However, communications are performed without using the SSLi0 output signal in the clock synchronous operation.

(1-3) Sequence Control

The transfer format employed in master mode is determined by the SBIi sequence control register (SBiCMDk), SBIi command registers k (SBiCMDk), the SBIi bit rate register (SBiBR), the SBIi clock delay register (SBiCKD), the SBIi slave select negation delay register (SBiSSLND), and the SBIi next-access delay register (SBiND). Although the SSL signals are not output in clock synchronous operation, these settings are valid.

The SBiSCR register is a register used to determine the sequence configuration for serial transfers that are executed by a master mode SBI. The following items are set in the SBiCMD0 to SBiCMD7 registers: SSL output signal value, MSB/LSB first, data length, some of the bit rate settings, RSPCK polarity/phase, whether the SBiCKD register is to be referenced, whether the SBiSSLND register is to be referenced, and whether the SBiND register is to be referenced. the SBiBR register holds some of the bit rate settings; the SBiCKD register, an SBI clock delay value; the SBiSSLND register, an SSL negation delay; and the SBiND register, a next-access delay value.

According to the sequence length that is assigned to the SBiSCR register, the SBI makes up a sequence comprised of a part or all of the SBiCMD0 to SBiCMD7 registers. The SBI contains a pointer to the SBiCMDk register that makes up the sequence. The CPU can check the value of this pointer by reading the SPCP bits in the SBIi sequence status register (SBiSSR). When the SPE bit in the SBIi control register 0 (SBiCR0) is set to 1 and the SBI function is

enabled, the SBI loads the pointer to the commands in the SBiCMD0 register, and incorporates the SBiCMD0 register settings into the transfer format at the beginning of serial transfer. The SBI 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 SBI sets the pointer in the SBiCMD0 register, and in this manner the sequence is executed repeatedly.

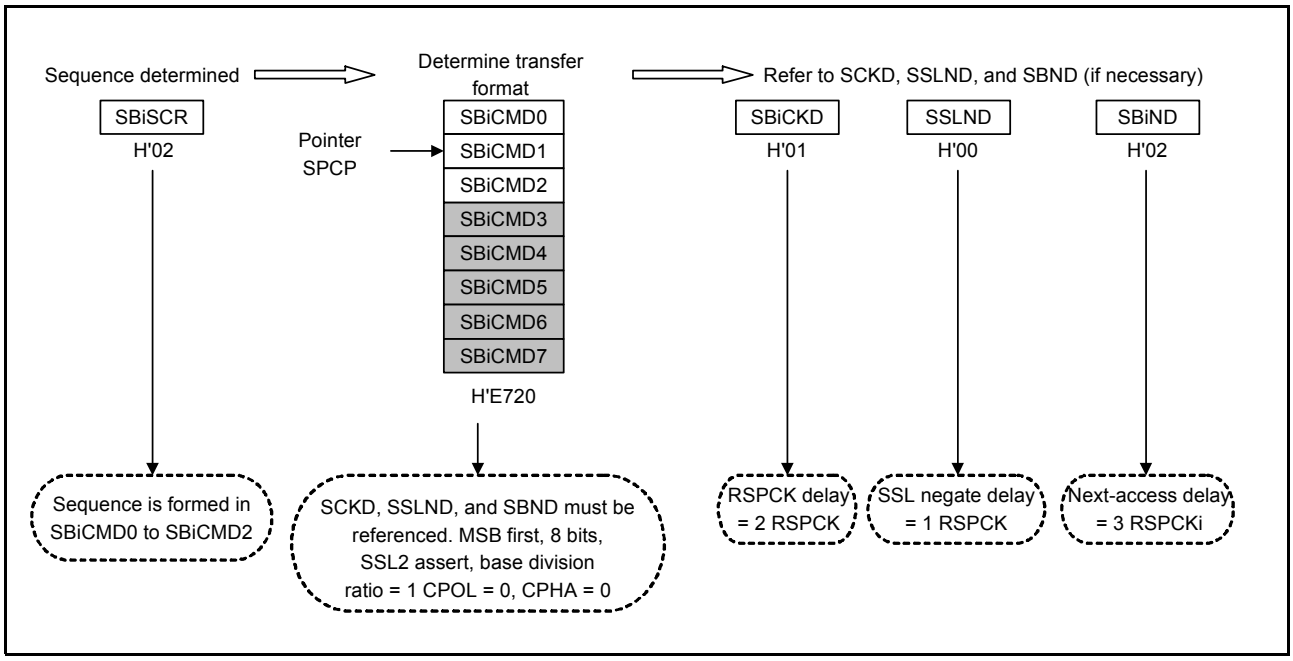


Figure 18.32 Determination Procedure of Serial Transfer Mode in Master Mode

(1-4) Initialization Flowchart

Figure 18.33 is a flowchart illustrating an example of initialization in clock synchronous operation when the SBI is used in master mode. For a description of how to set up an interrupt controller, the DMAC, and input/output ports, refer to the descriptions given in the individual blocks.

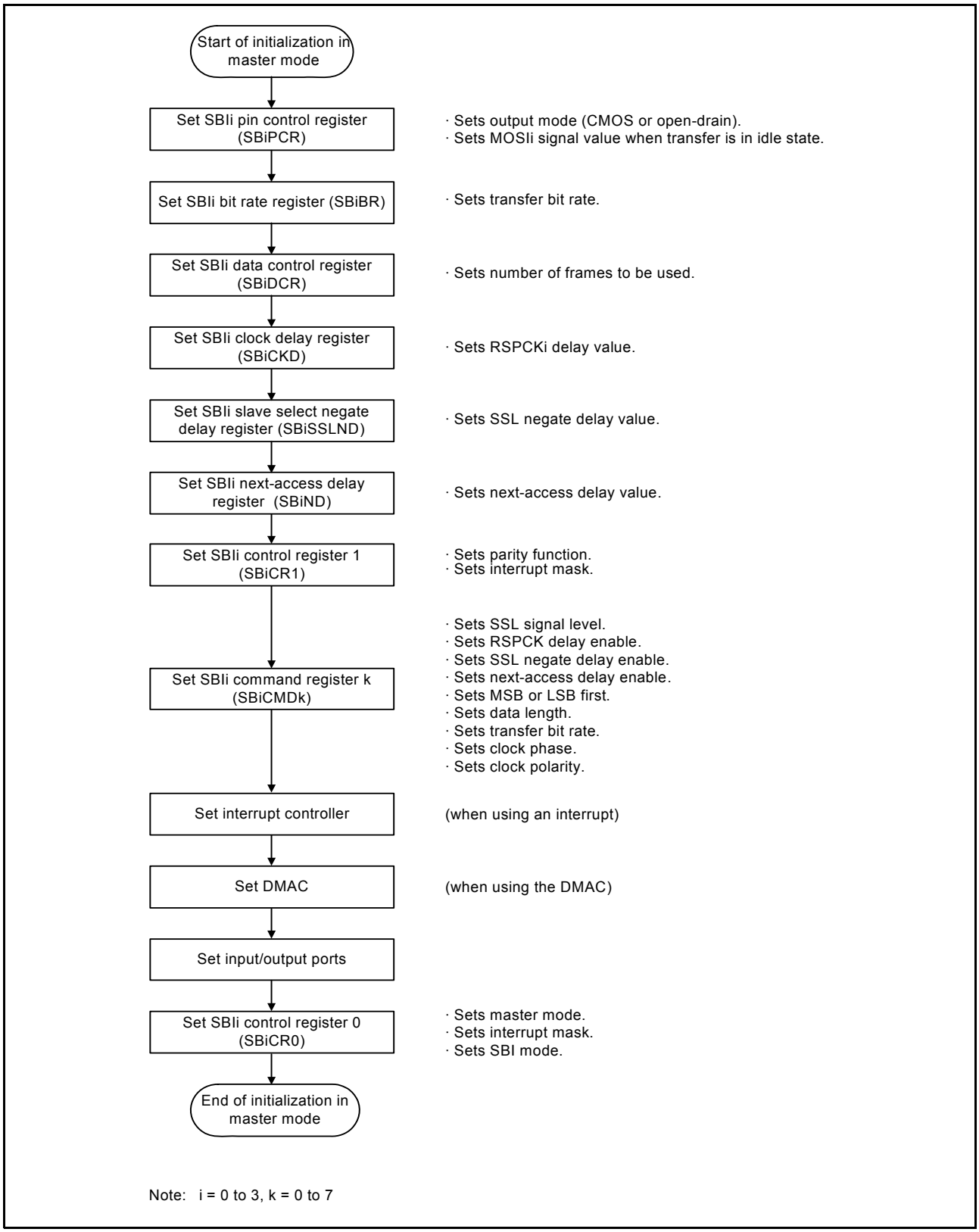


Figure 18.33 Example of Initialization Flowchart in Master Mode

(1-5) Transfer Operation Flowchart

Figure 18.34 is a flowchart illustrating a transfer in clock synchronous operation when the SBI is used in master mode.

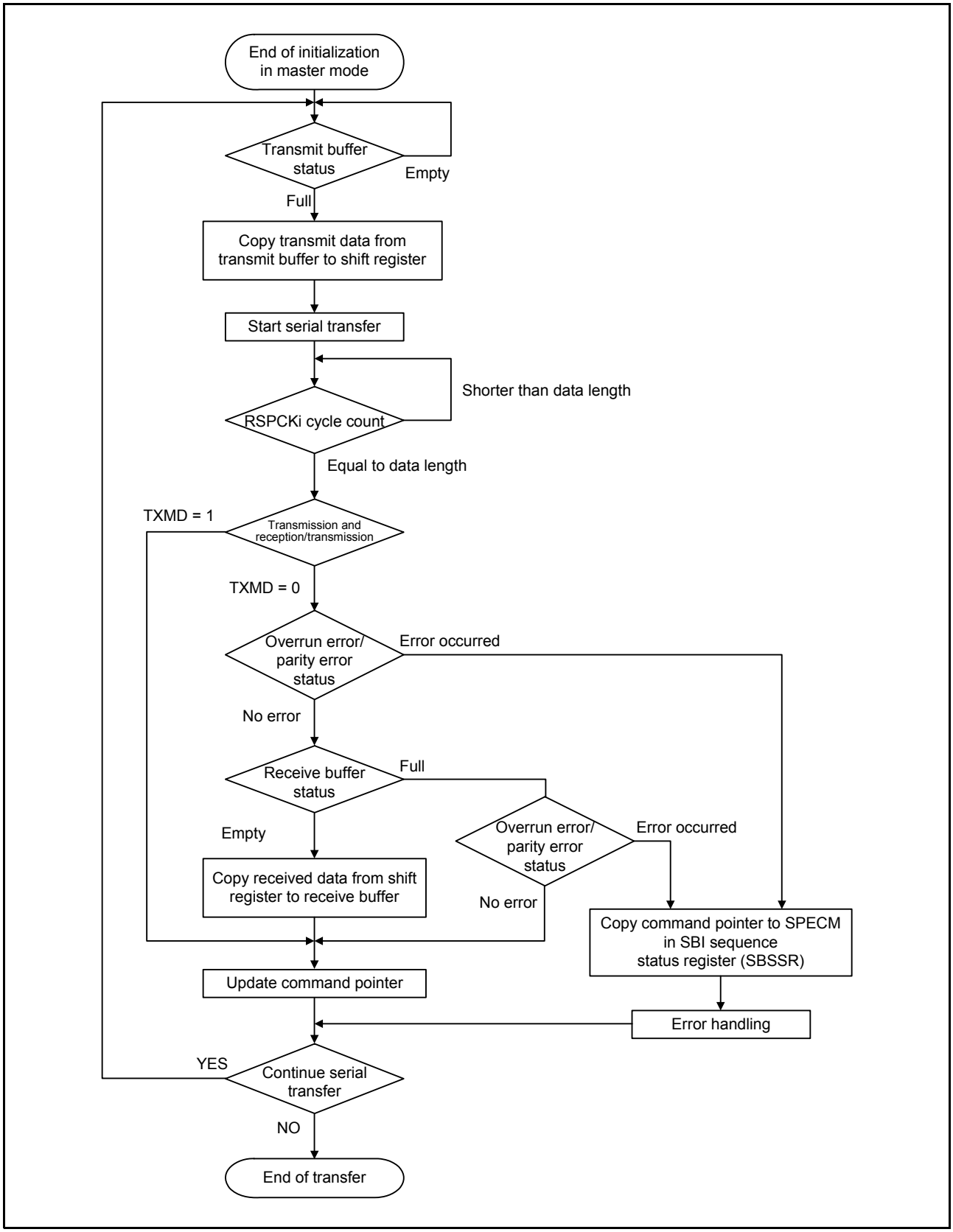


Figure 18.34 Transfer Operation Flowchart in Master Mode

(2) Slave Mode Operation

(2-1) Starting a Serial Transfer

When the SPMS bit in the SBIi control register 0 (SBiCR0) is 1, the first RSPCK edge triggers the start of a serial transfer in the SBI.

When detecting the start of a serial transfer in a condition in which the shift register is empty, the SBI changes the status of the shift register to “full”, so that data cannot be copied from the transmit buffer to the shift register when serial transfer is in progress. If the shift register was full before the serial transfer started, the SBI keeps the status of the shift register unchanged, in the full state.

When the SPMS bit is 1, the SBI always drives the MISOi output signal. For details on the SBI transfer format, refer to section 18.3.4, Transfer Format. However, the SSLi0 input signal is not used in the clock synchronous operation.

(2-2) Terminating a Serial Transfer

The SBI terminates the serial transfer after detecting an RSPCK edge corresponding to the final sampling timing. When the SPRF flag in the SBIi status register (SBiSR) is 0 and free space is available in the receive buffer, upon termination of serial transfer the SBI copies received data from the shift register to the receive buffer of the SBIi data register k (SBiDR). Irrespective of the value of the SPRF flag, upon termination of a serial transfer the SBI changes the status of the shift register to “empty”. The final sampling timing changes depending on the bit length of the transfer data. In slave mode, the SBI data length depends on the settings in the SPB bits in the SBiCMD0 register. For details on the SBI transfer format, refer to section 18.3.4, Transfer Format.

(2-3) Initialization Flowchart

Figure 18.35 is a flowchart illustrating an example of initialization in clock synchronous operation when the SBI is used in slave mode. For a description of how to set up an interrupt controller, the DMAC, and input/output ports, refer to the descriptions given in the individual blocks.

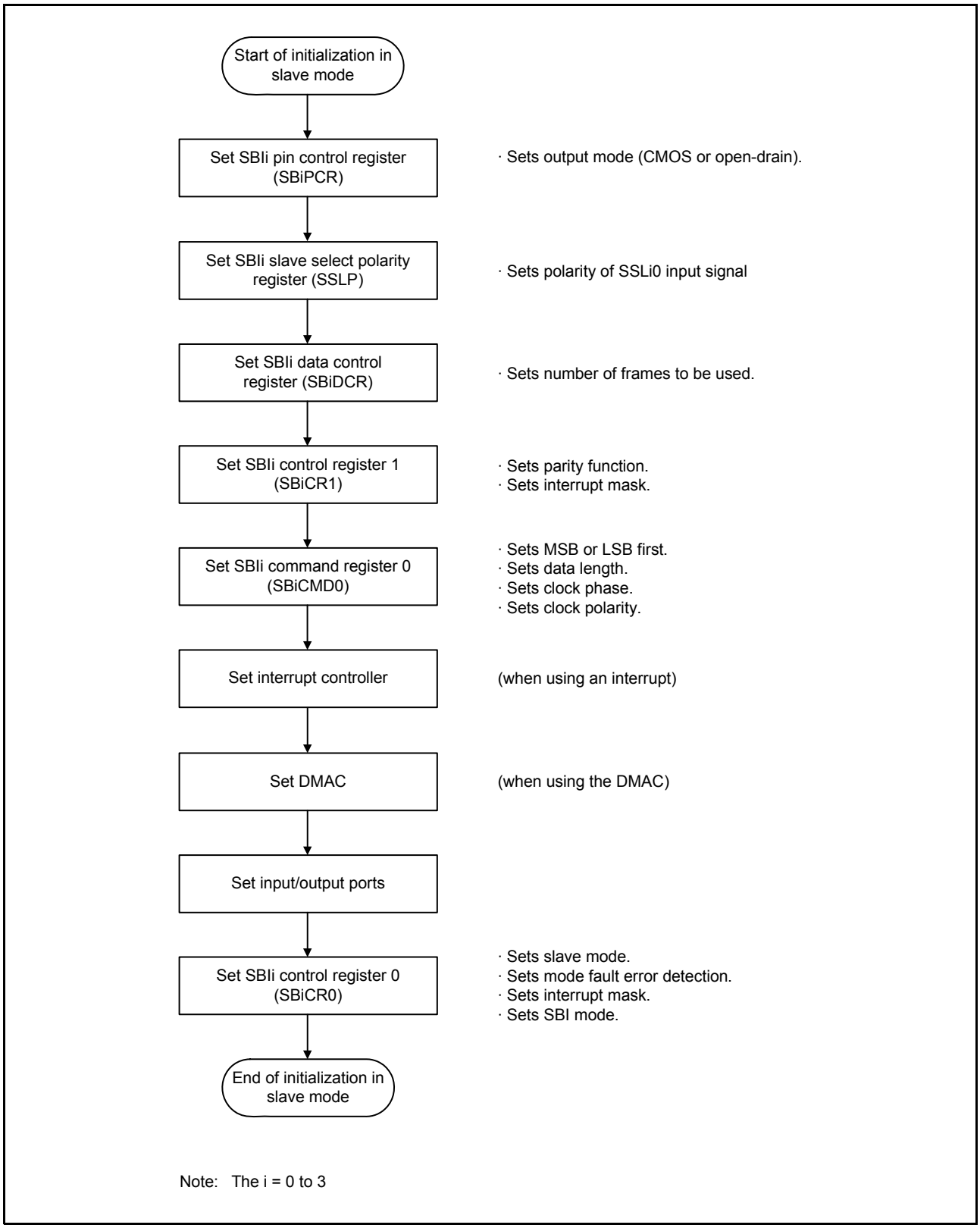


Figure 18.35 Example of Initialization Flowchart in Slave Mode

(2-4) Transfer Operation Flowchart

Figure 18.36 is a flowchart illustrating a transfer in clock synchronous operation when the SBI is used in master mode.

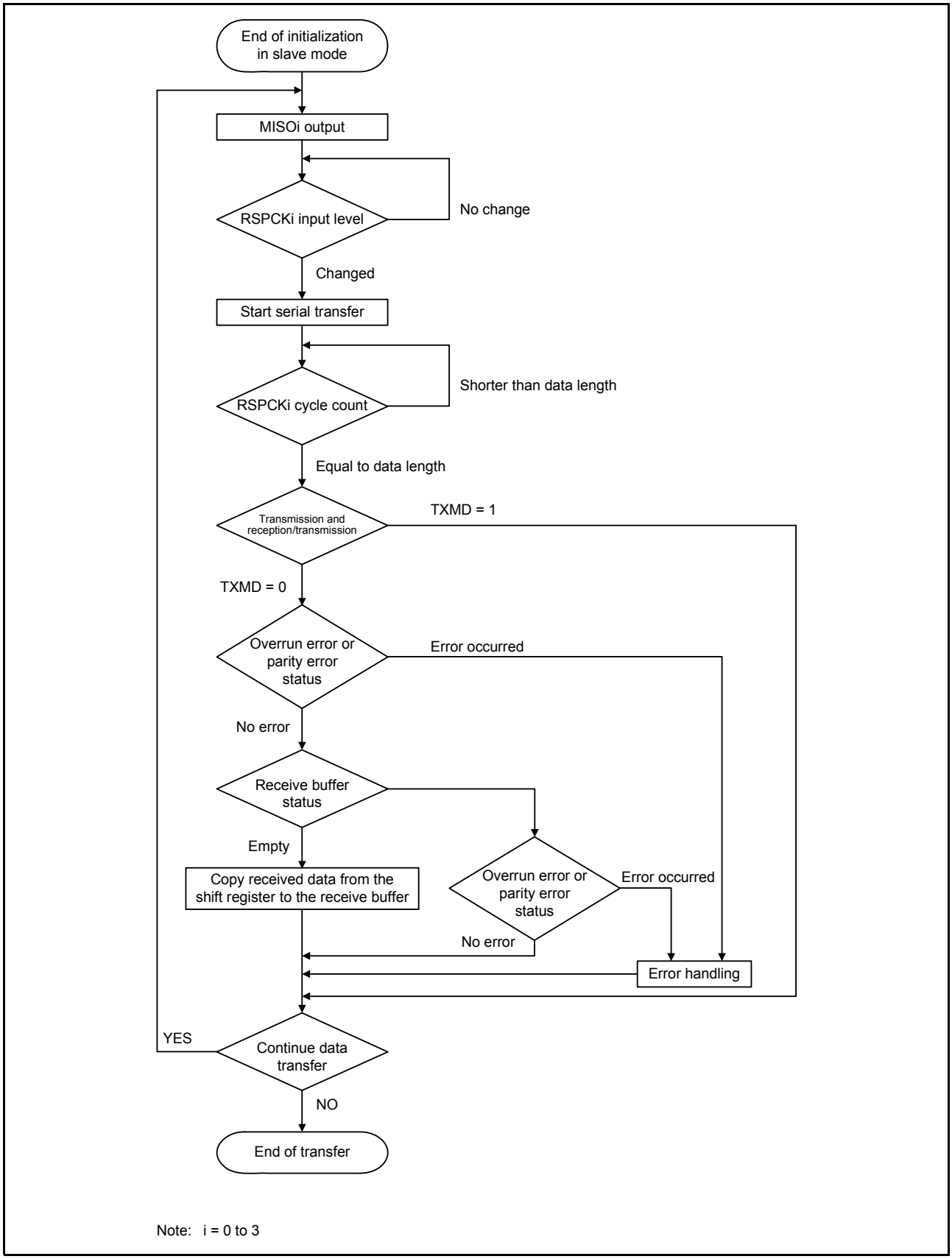


Figure 18.36 Transfer Operation Flowchart in Slave Mode (CPHA = 1)

18.3.12 Error Handling

Figure 18.37 to Figure 18.39 show the error handling for the SBI. The following error handling is used to return from the error state after an error occurred in master or slave mode.

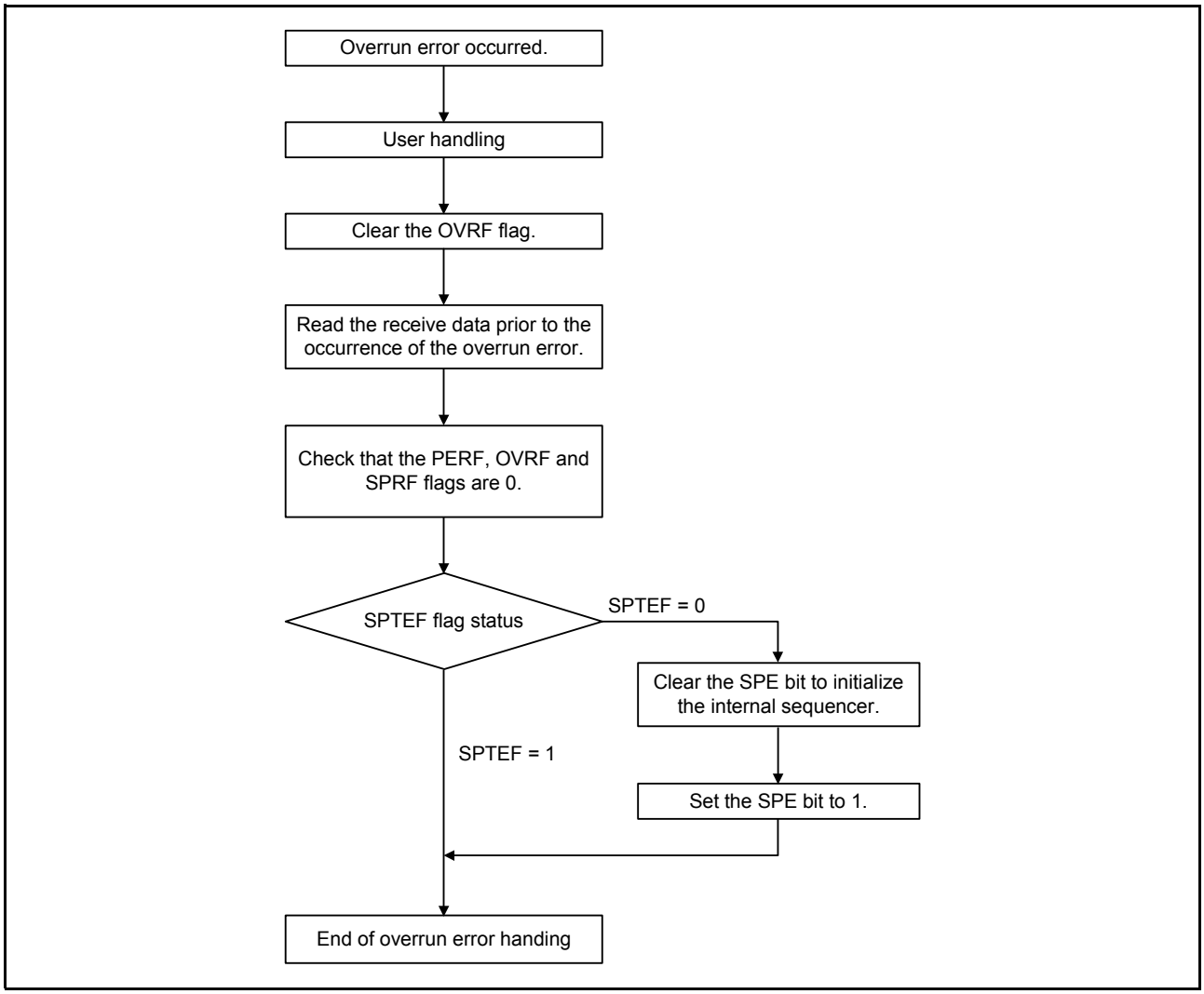


Figure 18.37 Error Handling (Overrun Error)

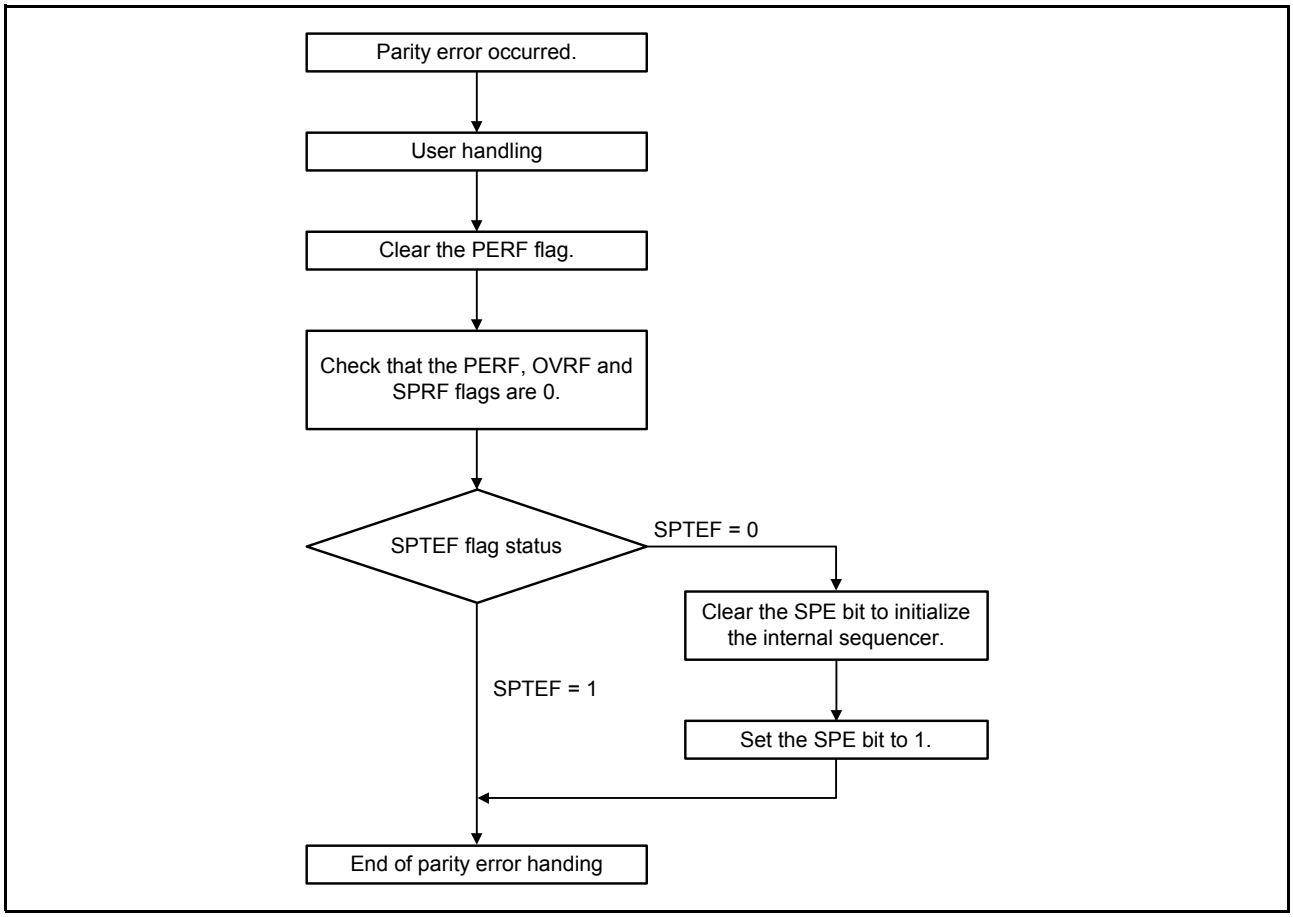


Figure 18.38 Error Handling (Parity error)

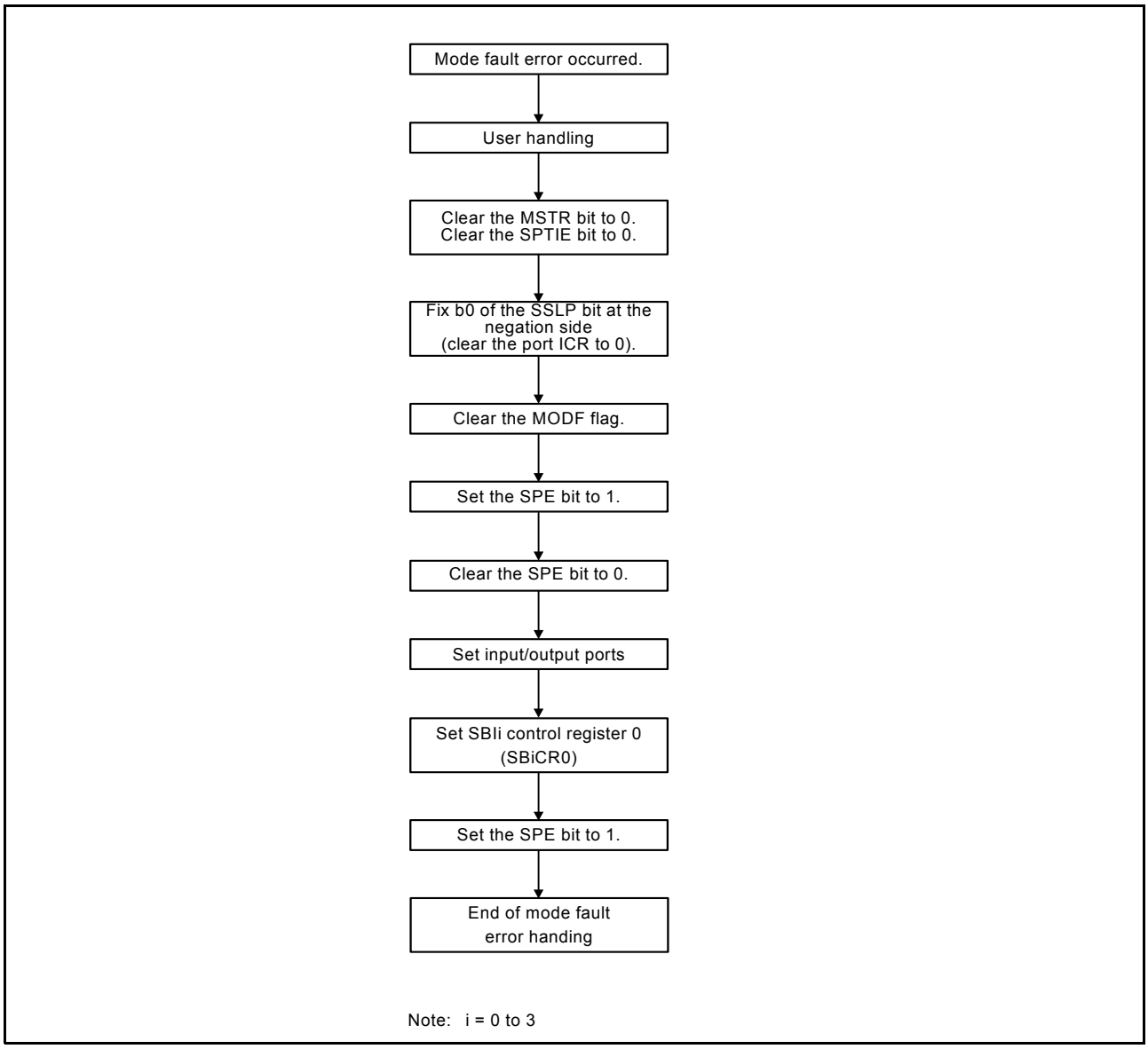


Figure 18.39 Error Handling (Mode Fault Error)

18.3.13 Loopback Mode

When the CPU writes 1 to the SPLP2 or SPLP bit in the SBIi pin control register (SBiPCR), the SBI shuts off the path between the MISOi pin and the shift register, and between the MOSIi pin and the shift register, and connects the input path and the output path (reversed) of the shift register.

When a serial transfer is executed in loopback mode, the transmit data or the inverted transmit data for the SBI becomes the received data for the SBI.

Table 18.14 lists the relation between the setting of bits SPLP2 and SPLP and the receive data.

Table 18.14 Setting of Bits SPLP2 and SPLP and Receive Data

SPLP2	SPLP	Receive Data
0	0	Input data from the MOSIi or MISOi pin
0	1	Transmit data inverted
1	0	Transmit data
1	1	Transmit data

Figure 18.40 shows the configuration of the shift register input/output paths for the case where the SBI in master mode is set in loopback mode (SPLP2 = 0, SPLP = 1).

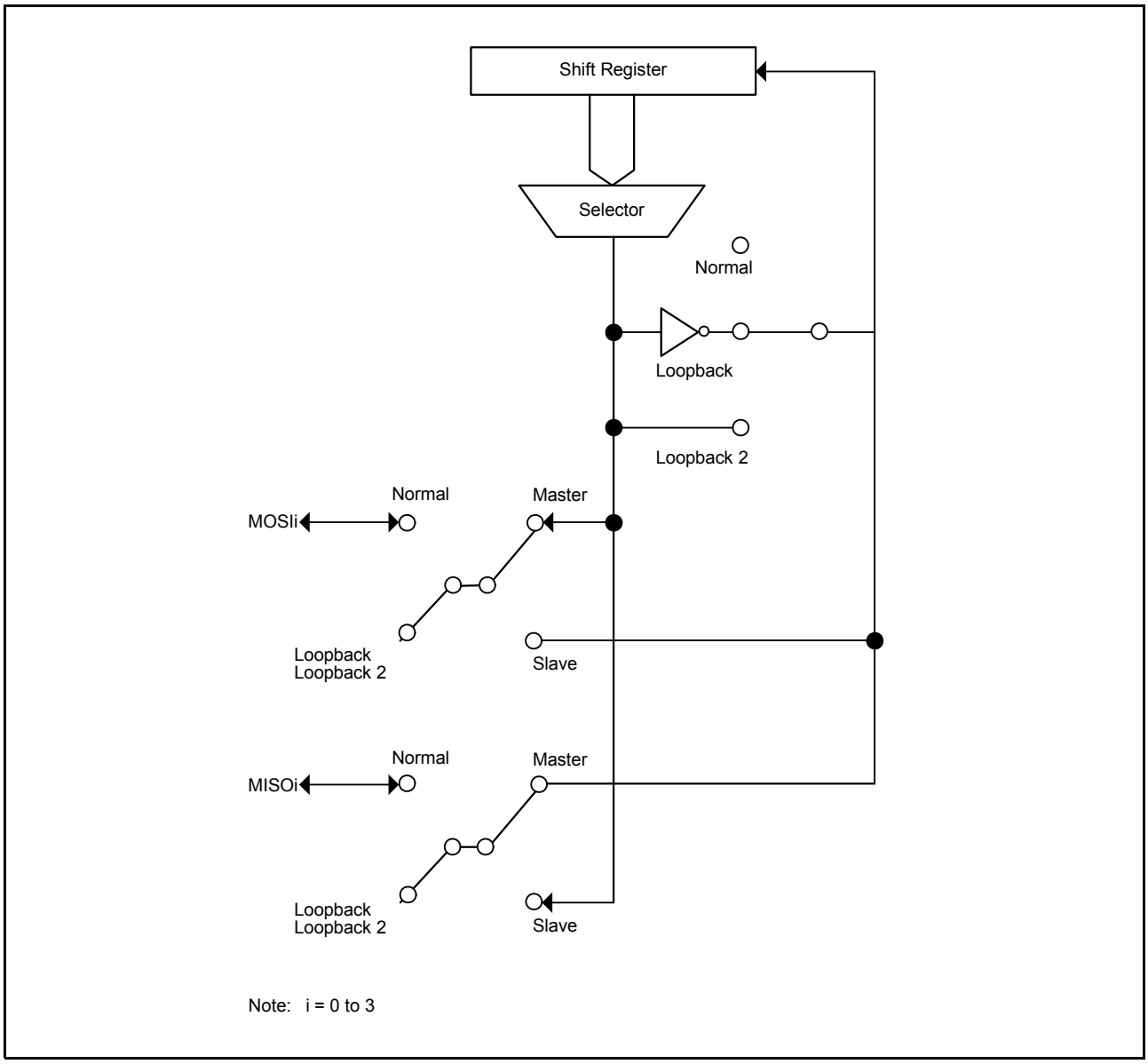


Figure 18.40 Configuration of Shift Register Input/Output Paths in Loopback Mode (Master Mode)

18.3.14 Self-diagnosis by Parity Function

The parity circuit is made up of the parity addition parts for the transmit data and the error detection parts for the receive data. To detect failures at the parity addition parts and the error detection parts of the parity circuit, perform the self-diagnosis of the parity circuit following the flowchart shown in Figure 18.41.

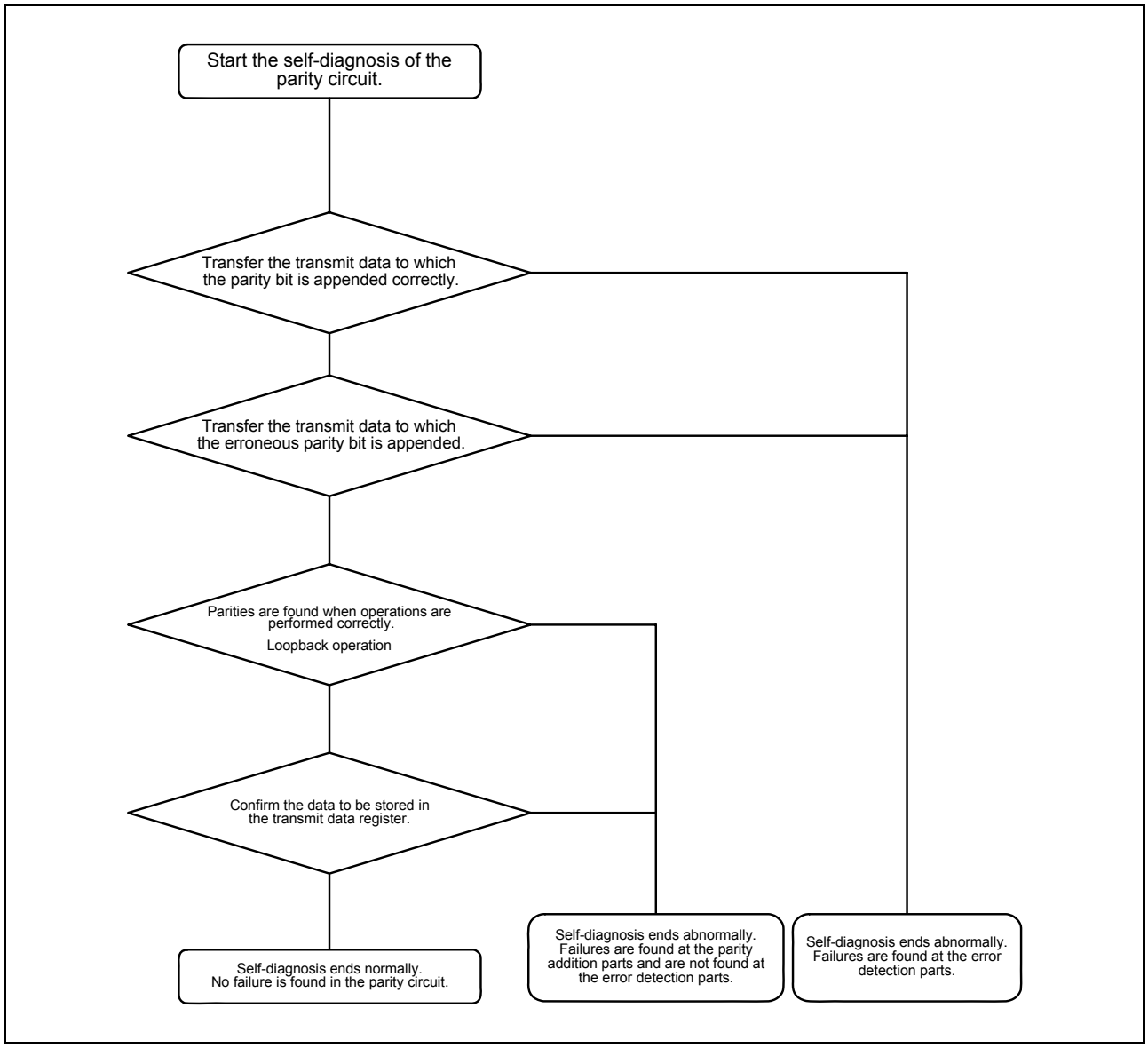


Figure 18.41 Self-diagnosis Flowchart of Parity Circuit

18.3.15 Interrupt sources

The SBI has interrupt sources of SBIi receive interrupt (receive buffer full), SBIi transmit interrupt (transmit buffer empty), SBIi error interrupt (mode fault, overrun, and parity error), and SBIi idle interrupt. In addition, the DMAC can be activated by the receive buffer full or transmit buffer empty interrupt for data transfer.

Table 18.15 shows Interrupt Sources of SBI.

When any of the interrupt conditions in Table 18.15 is met, an interrupt is generated. Clear the interrupt sources with data transfer by the CPU or DMAC.

Table 18.15 Interrupt Sources of SBI

Interrupt Source	Condition	DMAC Activation
SBIi receive interrupt (Receive buffer full)	When the SPRIE bit is 1 and the SPRF flag is 1	Possible
SBIi transmit interrupt (Transmit buffer empty)	When the SPTIE bit is 1 and the SPTEF flag is 1	Possible
SBIi error interrupt (Mode fault)	When the SPEIE bit is 1 and the MODF flag is 1	Not possible
SBIi error interrupt (Overrun)	When the SPEIE bit is 1 and the OVRF flag is 1	Not possible
SBIi error interrupt (parity error)	When the SPEIE bit is 1 and the PERF flag is 1	Not possible
SBIi idle interrupt (SBI idle)	When the SPIIE bit is 1 and the IDINF flag is 0	Not possible

Note: i = 0 to 3

19. LIN

The SH72A2 group and the SH72A0 group implement a LIN (Local Interconnect Network), compliant with LIN Specification Package Revisions 1.3, 2.0, and 2.1, which transmits and receives frames, and judges errors automatically. This LSI has two channels (LIN2 and LIN3) in the SH72A2 group and one channel (LIN2) in the SH72A0 group. * This chapter describes the LIN having 12 channels.

Table 19.1 lists the LIN Specifications. Figure 19.1 shows the LIN Block Diagram.

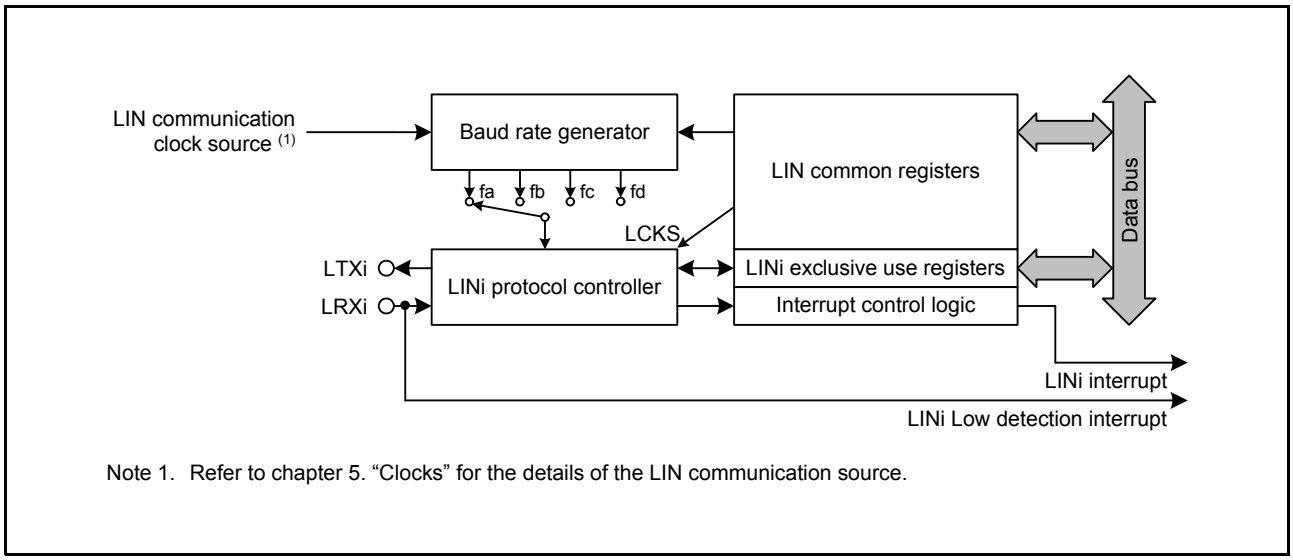
In this chapter, the variable i in LINi and Li represents a value of 0 to 11.

Note: * The SH72A2 group does not have channels other than LIN2 and LIN3.
The SH72A0 group does not have channels other than LIN2.
The functions of the channels not in the groups cannot be used.
Some products of the SH72A0 group do not have a LIN. For the details, see section 1, Overview.

Table 19.1 LIN Specifications

Item	Specifications
Protocols	LIN Specification Package Revisions 1.3, 2.0, and 2.1
Channel	12 channels (LIN master)
Variable frame composition	• Break dominant: 13 to 28 Tbits • Break delimiter: 1 to 4 Tbits • Interbyte space (Header): 0 to 7 Tbits (space between Sync field and ID field)*1 • Response space: 0 to 7 Tbits*1 • Interbyte space: 0 to 3 Tbits (space between bytes in response space) • Wake-up: 1 to 16 Tbits
Check sum	Automatic calculation in both transmission and reception Selectable either classic check sum or enhanced check sum (changeable at each frame)
Data byte(s) in response field	Variable from 0 to 8
Frame transmission mode	Selectable either non-frame separate mode or frame separate mode; the former mode is to transmit a header and response by a single start command and the latter is to separately transmit them by respective start command
Wake-up transmission/reception	Available in LIN wake-up mode • Wake-up transmission (1 to 16 Tbits) • Wake-up reception - Input signal Low time counting - Input signal Low detection
Operational status	• Frame/wake-up transmit completion • Header transmit completion • Frame/wake-up receive completion*2 • Data 1 receive completion • Error detection • Operation mode (LIN reset mode, LIN wake-up mode, LIN operation mode, and LIN self-test mode)
Error status	• Bit error • Check sum error • Frame timeout error • Physical bus error • Framing error
Baud rate	Selectable LIN-specified value generated by the baud rate generator
Test mode	Self-test mode for user evaluation
Interrupt function	• Frame/wake-up transmit completion • Frame/wake-up receive completion*2 • Error detection • LINi Low detection (input signal Low detection)

Notes: 1. The interbyte space (Header) has the same value as the response space since the both spaces are set in the same register.
2. The wake-up reception indicates input signal Low time counting.



Note 1. Refer to chapter 5. “Clocks” for the details of the LIN communication source.

Figure 19.1 LIN Block Diagram

- LTXi, LRXi: LIN I/O pins
- Baud rate generator: Generates a communication clock for LIN
- LINi common registers and LINi exclusive use registers: Registers for the LIN
- Interrupt control logic: Controls the interrupt request (LINi interrupt) generated by the LIN

Table 19.2 lists the LIN I/O Pins.

Table 19.2 LIN I/O Pins

Pin Name	I/O	Description
LRX0 to LRX11	Input	Input pins for the LIN communication function
LTX0 to LTX11	Output	Output pins for the LIN communication function

19.1 Registers

Table 19.3 to Table 19.9 list the LIN registers.

Table 19.3 LIN Registers (1)

Register Name	Symbol	After Reset	Address	Access Size
LIN wake-up baud rate select register	LWBR	H'00	H'FF61 0001	8
LIN baud rate prescaler 0 register	LBRP0	H'00	H'FF61 0002	8/16
LIN baud rate prescaler 1 register	LBRP1	H'00	H'FF61 0003	8/16
LIN self-test control register	LSTC	H'00	H'FF61 0004	8
LIN0 mode register	L0MD	H'00	H'FF61 0008	8/16/32
LIN1 mode register	L1MD	H'00	H'FF61 0028	8/16/32
LIN2 mode register	L2MD	H'00	H'FF61 0048	8/16/32
LIN3 mode register	L3MD	H'00	H'FF61 0068	8/16/32
LIN4 mode register	L4MD	H'00	H'FF61 0088	8/16/32
LIN5 mode register	L5MD	H'00	H'FF61 00A8	8/16/32
LIN6 mode register	L6MD	H'00	H'FF61 00C8	8/16/32
LIN7 mode register	L7MD	H'00	H'FF61 00E8	8/16/32
LIN8 mode register	L8MD	H'00	H'FF61 0108	8/16/32
LIN9 mode register	L9MD	H'00	H'FF61 0128	8/16/32
LIN10 mode register	L10MD	H'00	H'FF61 0148	8/16/32
LIN11 mode register	L11MD	H'00	H'FF61 0168	8/16/32
LIN0 break field setting register	L0BRK	H'00	H'FF61 0009	8/16/32
LIN1 break field setting register	L1BRK	H'00	H'FF61 0029	8/16/32
LIN2 break field setting register	L2BRK	H'00	H'FF61 0049	8/16/32
LIN3 break field setting register	L3BRK	H'00	H'FF61 0069	8/16/32
LIN4 break field setting register	L4BRK	H'00	H'FF61 0089	8/16/32
LIN5 break field setting register	L5BRK	H'00	H'FF61 00A9	8/16/32
LIN6 break field setting register	L6BRK	H'00	H'FF61 00C9	8/16/32
LIN7 break field setting register	L7BRK	H'00	H'FF61 00E9	8/16/32
LIN8 break field setting register	L8BRK	H'00	H'FF61 0109	8/16/32
LIN9 break field setting register	L9BRK	H'00	H'FF61 0129	8/16/32
LIN10 break field setting register	L10BRK	H'00	H'FF61 0149	8/16/32
LIN11 break field setting register	L11BRK	H'00	H'FF61 0169	8/16/32
LIN0 space setting register	L0SPC	H'00	H'FF61 000A	8/16/32
LIN1 space setting register	L1SPC	H'00	H'FF61 002A	8/16/32
LIN2 space setting register	L2SPC	H'00	H'FF61 004A	8/16/32
LIN3 space setting register	L3SPC	H'00	H'FF61 006A	8/16/32
LIN4 space setting register	L4SPC	H'00	H'FF61 008A	8/16/32
LIN5 space setting register	L5SPC	H'00	H'FF61 00AA	8/16/32
LIN6 space setting register	L6SPC	H'00	H'FF61 00CA	8/16/32
LIN7 space setting register	L7SPC	H'00	H'FF61 00EA	8/16/32
LIN8 space setting register	L8SPC	H'00	H'FF61 010A	8/16/32
LIN9 space setting register	L9SPC	H'00	H'FF61 012A	8/16/32
LIN10 space setting register	L10SPC	H'00	H'FF61 014A	8/16/32

Table 19.4 LIN Registers (2)

Register Name	Symbol	After Reset	Address	Access Size
LIN11 space setting register	L11SPC	H'00	H'FF61 016A	8/16/32
LIN0 wake-up setting register	L0WUP	H'00	H'FF61 000B	8/16/32
LIN1 wake-up setting register	L1WUP	H'00	H'FF61 002B	8/16/32
LIN2 wake-up setting register	L2WUP	H'00	H'FF61 004B	8/16/32
LIN3 wake-up setting register	L3WUP	H'00	H'FF61 006B	8/16/32
LIN4 wake-up setting register	L4WUP	H'00	H'FF61 008B	8/16/32
LIN5 wake-up setting register	L5WUP	H'00	H'FF61 00AB	8/16/32
LIN6 wake-up setting register	L6WUP	H'00	H'FF61 00CB	8/16/32
LIN7 wake-up setting register	L7WUP	H'00	H'FF61 00EB	8/16/32
LIN8 wake-up setting register	L8WUP	H'00	H'FF61 010B	8/16/32
LIN9 wake-up setting register	L9WUP	H'00	H'FF61 012B	8/16/32
LIN10 wake-up setting register	L10WUP	H'00	H'FF61 014B	8/16/32
LIN11 wake-up setting register	L11WUP	H'00	H'FF61 016B	8/16/32
LIN0 interrupt enable register	L0IE	H'00	H'FF61 000C	8/16
LIN1 interrupt enable register	L1IE	H'00	H'FF61 002C	8/16
LIN2 interrupt enable register	L2IE	H'00	H'FF61 004C	8/16
LIN3 interrupt enable register	L3IE	H'00	H'FF61 006C	8/16
LIN4 interrupt enable register	L4IE	H'00	H'FF61 008C	8/16
LIN5 interrupt enable register	L5IE	H'00	H'FF61 00AC	8/16
LIN6 interrupt enable register	L6IE	H'00	H'FF61 00CC	8/16
LIN7 interrupt enable register	L7IE	H'00	H'FF61 00EC	8/16
LIN8 interrupt enable register	L8IE	H'00	H'FF61 010C	8/16
LIN9 interrupt enable register	L9IE	H'00	H'FF61 012C	8/16
LIN10 interrupt enable register	L10IE	H'00	H'FF61 014C	8/16
LIN11 interrupt enable register	L11IE	H'00	H'FF61 016C	8/16
LIN0 error detection enable register	L0EDE	H'00	H'FF61 000D	8/16
LIN1 error detection enable register	L1EDE	H'00	H'FF61 002D	8/16
LIN2 error detection enable register	L2EDE	H'00	H'FF61 004D	8/16
LIN3 error detection enable register	L3EDE	H'00	H'FF61 006D	8/16
LIN4 error detection enable register	L4EDE	H'00	H'FF61 008D	8/16
LIN5 error detection enable register	L5EDE	H'00	H'FF61 00AD	8/16
LIN6 error detection enable register	L6EDE	H'00	H'FF61 00CD	8/16
LIN7 error detection enable register	L7EDE	H'00	H'FF61 00ED	8/16
LIN8 error detection enable register	L8EDE	H'00	H'FF61 010D	8/16
LIN9 error detection enable register	L9EDE	H'00	H'FF61 012D	8/16
LIN10 error detection enable register	L10EDE	H'00	H'FF61 014D	8/16
LIN11 error detection enable register	L11EDE	H'00	H'FF61 016D	8/16
LIN0 control register	L0C	H'00	H'FF61 000E	8
LIN1 control register	L1C	H'00	H'FF61 002E	8
LIN2 control register	L2C	H'00	H'FF61 004E	8
LIN3 control register	L3C	H'00	H'FF61 006E	8
LIN4 control register	L4C	H'00	H'FF61 008E	8
LIN5 control register	L5C	H'00	H'FF61 00AE	8

Table 19.5 LIN Registers (3)

Register Name	Symbol	After Reset	Address	Access Size
LIN6 control register	L6C	H'00	H'FF61 00CE	8
LIN7 control register	L7C	H'00	H'FF61 00EE	8
LIN8 control register	L8C	H'00	H'FF61 010E	8
LIN9 control register	L9C	H'00	H'FF61 012E	8
LIN10 control register	L10C	H'00	H'FF61 014E	8
LIN11 control register	L11C	H'00	H'FF61 016E	8
LIN0 transmission control register	L0TC	H'00	H'FF61 0010	8/16/32
LIN1 transmission control register	L1TC	H'00	H'FF61 0030	8/16/32
LIN2 transmission control register	L2TC	H'00	H'FF61 0050	8/16/32
LIN3 transmission control register	L3TC	H'00	H'FF61 0070	8/16/32
LIN4 transmission control register	L4TC	H'00	H'FF61 0090	8/16/32
LIN5 transmission control register	L5TC	H'00	H'FF61 00B0	8/16/32
LIN6 transmission control register	L6TC	H'00	H'FF61 00D0	8/16/32
LIN7 transmission control register	L7TC	H'00	H'FF61 00F0	8/16/32
LIN8 transmission control register	L8TC	H'00	H'FF61 0110	8/16/32
LIN9 transmission control register	L9TC	H'00	H'FF61 0130	8/16/32
LIN10 transmission control register	L10TC	H'00	H'FF61 0150	8/16/32
LIN11 transmission control register	L11TC	H'00	H'FF61 0170	8/16/32
LIN0 mode status register	L0MST	H'00	H'FF61 0011	8/16/32
LIN1 mode status register	L1MST	H'00	H'FF61 0031	8/16/32
LIN2 mode status register	L2MST	H'00	H'FF61 0051	8/16/32
LIN3 mode status register	L3MST	H'00	H'FF61 0071	8/16/32
LIN4 mode status register	L4MST	H'00	H'FF61 0091	8/16/32
LIN5 mode status register	L5MST	H'00	H'FF61 00B1	8/16/32
LIN6 mode status register	L6MST	H'00	H'FF61 00D1	8/16/32
LIN7 mode status register	L7MST	H'00	H'FF61 00F1	8/16/32
LIN8 mode status register	L8MST	H'00	H'FF61 0111	8/16/32
LIN9 mode status register	L9MST	H'00	H'FF61 0131	8/16/32
LIN10 mode status register	L10MST	H'00	H'FF61 0151	8/16/32
LIN11 mode status register	L11MST	H'00	H'FF61 0171	8/16/32
LIN0 status register	L0ST	H'00	H'FF61 0012	8/16/32
LIN1 status register	L1ST	H'00	H'FF61 0032	8/16/32
LIN2 status register	L2ST	H'00	H'FF61 0052	8/16/32
LIN3 status register	L3ST	H'00	H'FF61 0072	8/16/32
LIN4 status register	L4ST	H'00	H'FF61 0092	8/16/32
LIN5 status register	L5ST	H'00	H'FF61 00B2	8/16/32
LIN6 status register	L6ST	H'00	H'FF61 00D2	8/16/32
LIN7 status register	L7ST	H'00	H'FF61 00F2	8/16/32
LIN8 status register	L8ST	H'00	H'FF61 0112	8/16/32
LIN9 status register	L9ST	H'00	H'FF61 0132	8/16/32
LIN10 status register	L10ST	H'00	H'FF61 0152	8/16/32
LIN11 status register	L11ST	H'00	H'FF61 0172	8/16/32
LIN0 error status register	L0EST	H'00	H'FF61 0013	8/16/32

Table 19.6 LIN Registers (4)

Register Name	Symbol	After Reset	Address	Access Size
LIN1 error status register	L1EST	H'00	H'FF61 0033	8/16/32
LIN2 error status register	L2EST	H'00	H'FF61 0053	8/16/32
LIN3 error status register	L3EST	H'00	H'FF61 0073	8/16/32
LIN4 error status register	L4EST	H'00	H'FF61 0093	8/16/32
LIN5 error status register	L5EST	H'00	H'FF61 00B3	8/16/32
LIN6 error status register	L6EST	H'00	H'FF61 00D3	8/16/32
LIN7 error status register	L7EST	H'00	H'FF61 00F3	8/16/32
LIN8 error status register	L8EST	H'00	H'FF61 0113	8/16/32
LIN9 error status register	L9EST	H'00	H'FF61 0133	8/16/32
LIN10 error status register	L10EST	H'00	H'FF61 0153	8/16/32
LIN11 error status register	L11EST	H'00	H'FF61 0173	8/16/32
LIN0 response field set register	L0RFC	H'00	H'FF61 0014	8/16
LIN1 response field set register	L1RFC	H'00	H'FF61 0034	8/16
LIN2 response field set register	L2RFC	H'00	H'FF61 0054	8/16
LIN3 response field set register	L3RFC	H'00	H'FF61 0074	8/16
LIN4 response field set register	L4RFC	H'00	H'FF61 0094	8/16
LIN5 response field set register	L5RFC	H'00	H'FF61 00B4	8/16
LIN6 response field set register	L6RFC	H'00	H'FF61 00D4	8/16
LIN7 response field set register	L7RFC	H'00	H'FF61 00F4	8/16
LIN8 response field set register	L8RFC	H'00	H'FF61 0114	8/16
LIN9 response field set register	L9RFC	H'00	H'FF61 0134	8/16
LIN10 response field set register	L10RFC	H'00	H'FF61 0154	8/16
LIN11 response field set register	L11RFC	H'00	H'FF61 0174	8/16
LIN0 ID buffer register	L0IDB	Undefined	H'FF61 0015	8/16
LIN1 ID buffer register	L1IDB	Undefined	H'FF61 0035	8/16
LIN2 ID buffer register	L2IDB	Undefined	H'FF61 0055	8/16
LIN3 ID buffer register	L3IDB	Undefined	H'FF61 0075	8/16
LIN4 ID buffer register	L4IDB	Undefined	H'FF61 0095	8/16
LIN5 ID buffer register	L5IDB	Undefined	H'FF61 00B5	8/16
LIN6 ID buffer register	L6IDB	Undefined	H'FF61 00D5	8/16
LIN7 ID buffer register	L7IDB	Undefined	H'FF61 00F5	8/16
LIN8 ID buffer register	L8IDB	Undefined	H'FF61 0115	8/16
LIN9 ID buffer register	L9IDB	Undefined	H'FF61 0135	8/16
LIN10 ID buffer register	L10IDB	Undefined	H'FF61 0155	8/16
LIN11 ID buffer register	L11IDB	Undefined	H'FF61 0175	8/16
LIN0 check sum buffer register	L0CBR	Undefined	H'FF61 0016	8
LIN1 check sum buffer register	L1CBR	Undefined	H'FF61 0036	8
LIN2 check sum buffer register	L2CBR	Undefined	H'FF61 0056	8
LIN3 check sum buffer register	L3CBR	Undefined	H'FF61 0076	8
LIN4 check sum buffer register	L4CBR	Undefined	H'FF61 0096	8
LIN5 check sum buffer register	L5CBR	Undefined	H'FF61 00B6	8
LIN6 check sum buffer register	L6CBR	Undefined	H'FF61 00D6	8
LIN7 check sum buffer register	L7CBR	Undefined	H'FF61 00F6	8

Table 19.7 LIN Registers (5)

Register Name	Symbol	After Reset	Address	Access Size
LIN8 check sum buffer register	L8CBR	Undefined	H'FF61 0116	8
LIN9 check sum buffer register	L9CBR	Undefined	H'FF61 0136	8
LIN10 check sum buffer register	L10CBR	Undefined	H'FF61 0156	8
LIN11 check sum buffer register	L11CBR	Undefined	H'FF61 0176	8
LIN0 data 1 buffer register	L0DB1	Undefined	H'FF61 0018	8/16/32
LIN0 data 2 buffer register	L0DB2	Undefined	H'FF61 0019	8/16/32
LIN0 data 3 buffer register	L0DB3	Undefined	H'FF61 001A	8/16/32
LIN0 data 4 buffer register	L0DB4	Undefined	H'FF61 001B	8/16/32
LIN0 data 5 buffer register	L0DB5	Undefined	H'FF61 001C	8/16/32
LIN0 data 6 buffer register	L0DB6	Undefined	H'FF61 001D	8/16/32
LIN0 data 7 buffer register	L0DB7	Undefined	H'FF61 001E	8/16/32
LIN0 data 8 buffer register	L0DB8	Undefined	H'FF61 001F	8/16/32
LIN1 data 1 buffer register	L1DB1	Undefined	H'FF61 0038	8/16/32
LIN1 data 2 buffer register	L1DB2	Undefined	H'FF61 0039	8/16/32
LIN1 data 3 buffer register	L1DB3	Undefined	H'FF61 003A	8/16/32
LIN1 data 4 buffer register	L1DB4	Undefined	H'FF61 003B	8/16/32
LIN1 data 5 buffer register	L1DB5	Undefined	H'FF61 003C	8/16/32
LIN1 data 6 buffer register	L1DB6	Undefined	H'FF61 003D	8/16/32
LIN1 data 7 buffer register	L1DB7	Undefined	H'FF61 003E	8/16/32
LIN1 data 8 buffer register	L1DB8	Undefined	H'FF61 003F	8/16/32
LIN2 data 1 buffer register	L2DB1	Undefined	H'FF61 0058	8/16/32
LIN2 data 2 buffer register	L2DB2	Undefined	H'FF61 0059	8/16/32
LIN2 data 3 buffer register	L2DB3	Undefined	H'FF61 005A	8/16/32
LIN2 data 4 buffer register	L2DB4	Undefined	H'FF61 005B	8/16/32
LIN2 data 5 buffer register	L2DB5	Undefined	H'FF61 005C	8/16/32
LIN2 data 6 buffer register	L2DB6	Undefined	H'FF61 005D	8/16/32
LIN2 data 7 buffer register	L2DB7	Undefined	H'FF61 005E	8/16/32
LIN2 data 8 buffer register	L2DB8	Undefined	H'FF61 005F	8/16/32
LIN3 data 1 buffer register	L3DB1	Undefined	H'FF61 0078	8/16/32
LIN3 data 2 buffer register	L3DB2	Undefined	H'FF61 0079	8/16/32
LIN3 data 3 buffer register	L3DB3	Undefined	H'FF61 007A	8/16/32
LIN3 data 4 buffer register	L3DB4	Undefined	H'FF61 007B	8/16/32
LIN3 data 5 buffer register	L3DB5	Undefined	H'FF61 007C	8/16/32
LIN3 data 6 buffer register	L3DB6	Undefined	H'FF61 007D	8/16/32
LIN3 data 7 buffer register	L3DB7	Undefined	H'FF61 007E	8/16/32
LIN3 data 8 buffer register	L3DB8	Undefined	H'FF61 007F	8/16/32
LIN4 data 1 buffer register	L4DB1	Undefined	H'FF61 0098	8/16/32
LIN4 data 2 buffer register	L4DB2	Undefined	H'FF61 0099	8/16/32
LIN4 data 3 buffer register	L4DB3	Undefined	H'FF61 009A	8/16/32
LIN4 data 4 buffer register	L4DB4	Undefined	H'FF61 009B	8/16/32
LIN4 data 5 buffer register	L4DB5	Undefined	H'FF61 009C	8/16/32
LIN4 data 6 buffer register	L4DB6	Undefined	H'FF61 009D	8/16/32
LIN4 data 7 buffer register	L4DB7	Undefined	H'FF61 009E	8/16/32

Table 19.8 LIN Registers (6)

Register Name	Symbol	After Reset	Address	Access Size
LIN4 data 8 buffer register	L4DB8	Undefined	H'FF61 009F	8/16/32
LIN5 data 1 buffer register	L5DB1	Undefined	H'FF61 00B8	8/16/32
LIN5 data 2 buffer register	L5DB2	Undefined	H'FF61 00B9	8/16/32
LIN5 data 3 buffer register	L5DB3	Undefined	H'FF61 00BA	8/16/32
LIN5 data 4 buffer register	L5DB4	Undefined	H'FF61 00BB	8/16/32
LIN5 data 5 buffer register	L5DB5	Undefined	H'FF61 00BC	8/16/32
LIN5 data 6 buffer register	L5DB6	Undefined	H'FF61 00BD	8/16/32
LIN5 data 7 buffer register	L5DB7	Undefined	H'FF61 00BE	8/16/32
LIN5 data 8 buffer register	L5DB8	Undefined	H'FF61 00BF	8/16/32
LIN6 data 1 buffer register	L6DB1	Undefined	H'FF61 00D8	8/16/32
LIN6 data 2 buffer register	L6DB2	Undefined	H'FF61 00D9	8/16/32
LIN6 data 3 buffer register	L6DB3	Undefined	H'FF61 00DA	8/16/32
LIN6 data 4 buffer register	L6DB4	Undefined	H'FF61 00DB	8/16/32
LIN6 data 5 buffer register	L6DB5	Undefined	H'FF61 00DC	8/16/32
LIN6 data 6 buffer register	L6DB6	Undefined	H'FF61 00DD	8/16/32
LIN6 data 7 buffer register	L6DB7	Undefined	H'FF61 00DE	8/16/32
LIN6 data 8 buffer register	L6DB8	Undefined	H'FF61 00DF	8/16/32
LIN7 data 1 buffer register	L7DB1	Undefined	H'FF61 00F8	8/16/32
LIN7 data 2 buffer register	L7DB2	Undefined	H'FF61 00F9	8/16/32
LIN7 data 3 buffer register	L7DB3	Undefined	H'FF61 00FA	8/16/32
LIN7 data 4 buffer register	L7DB4	Undefined	H'FF61 00FB	8/16/32
LIN7 data 5 buffer register	L7DB5	Undefined	H'FF61 00FC	8/16/32
LIN7 data 6 buffer register	L7DB6	Undefined	H'FF61 00FD	8/16/32
LIN7 data 7 buffer register	L7DB7	Undefined	H'FF61 00FE	8/16/32
LIN7 data 8 buffer register	L7DB8	Undefined	H'FF61 00FF	8/16/32
LIN8 data 1 buffer register	L8DB1	Undefined	H'FF61 0118	8/16/32
LIN8 data 2 buffer register	L8DB2	Undefined	H'FF61 0119	8/16/32
LIN8 data 3 buffer register	L8DB3	Undefined	H'FF61 011A	8/16/32
LIN8 data 4 buffer register	L8DB4	Undefined	H'FF61 011B	8/16/32
LIN8 data 5 buffer register	L8DB5	Undefined	H'FF61 011C	8/16/32
LIN8 data 6 buffer register	L8DB6	Undefined	H'FF61 011D	8/16/32
LIN8 data 7 buffer register	L8DB7	Undefined	H'FF61 011E	8/16/32
LIN8 data 8 buffer register	L8DB8	Undefined	H'FF61 011F	8/16/32
LIN9 data 1 buffer register	L9DB1	Undefined	H'FF61 0138	8/16/32
LIN9 data 2 buffer register	L9DB2	Undefined	H'FF61 0139	8/16/32
LIN9 data 3 buffer register	L9DB3	Undefined	H'FF61 013A	8/16/32
LIN9 data 4 buffer register	L9DB4	Undefined	H'FF61 013B	8/16/32
LIN9 data 5 buffer register	L9DB5	Undefined	H'FF61 013C	8/16/32
LIN9 data 6 buffer register	L9DB6	Undefined	H'FF61 013D	8/16/32
LIN9 data 7 buffer register	L9DB7	Undefined	H'FF61 013E	8/16/32
LIN9 data 8 buffer register	L9DB8	Undefined	H'FF61 013F	8/16/32
LIN10 data 1 buffer register	L10DB1	Undefined	H'FF61 0158	8/16/32
LIN10 data 2 buffer register	L10DB2	Undefined	H'FF61 0159	8/16/32

Table 19.9 LIN Registers (7)

Register Name	Symbol	After Reset	Address	Access Size
LIN10 data 3 buffer register	L10DB3	Undefined	H'FF61 015A	8/16/32
LIN10 data 4 buffer register	L10DB4	Undefined	H'FF61 015B	8/16/32
LIN10 data 5 buffer register	L10DB5	Undefined	H'FF61 015C	8/16/32
LIN10 data 6 buffer register	L10DB6	Undefined	H'FF61 015D	8/16/32
LIN10 data 7 buffer register	L10DB7	Undefined	H'FF61 015E	8/16/32
LIN10 data 8 buffer register	L10DB8	Undefined	H'FF61 015F	8/16/32
LIN11 data 1 buffer register	L11DB1	Undefined	H'FF61 0178	8/16/32
LIN11 data 2 buffer register	L11DB2	Undefined	H'FF61 0179	8/16/32
LIN11 data 3 buffer register	L11DB3	Undefined	H'FF61 017A	8/16/32
LIN11 data 4 buffer register	L11DB4	Undefined	H'FF61 017B	8/16/32
LIN11 data 5 buffer register	L11DB5	Undefined	H'FF61 017C	8/16/32
LIN11 data 6 buffer register	L11DB6	Undefined	H'FF61 017D	8/16/32
LIN11 data 7 buffer register	L11DB7	Undefined	H'FF61 017E	8/16/32
LIN11 data 8 buffer register	L11DB8	Undefined	H'FF61 017F	8/16/32

19.1.1 LIN Wake-up Baud Rate Select Register (LWBR)

Address H'FF61 0001

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	LWBR0
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R
b0	LWBR0	Wake-up Baud Rate Select Bit	0: When LIN 1.3 used 1: When LIN 2.0 and 2.1 used	R/W

This register should be set in LIN reset mode.

LWBR0 Bit

When LIN Specification Package Revision 1.3 is used, set the LWBR0 bit in the LWBR register to 0. This allows the input signal Low time to be measured 2.5 or more Tbits of fLIN. When LIN Specification Package Revisions 2.0 and 2.1 are used, set the LWBR0 bit to 1. This causes fa to be selected as a LIN system clock (fLIN) in LIN wake-up mode regardless of the LCKS bit in the L0MD register (the LCKS bit does not change) and allows the input signal Low time to be measured 125 µs or more.

19.1.2 LIN Baud Rate Prescaler 0 Register (LBRP0)

Address H'FF61 0002

	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	0	0	0	0	0	0	0	0

Bit	Description	Setting Range	R/W
b7 to b0	The peripheral clock is divided by (N + 1) (N = setting value (0 to 255)) by the baud rate prescaler	H'00 to H'FF	R/W

This register should be set in LIN reset mode.

19.1.3 LIN Baud Rate Prescaler 1 Register (LBRP1)

Address H'FF61 0003

	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	0	0	0	0	0	0	0	0

Bit	Description	Setting Range	R/W
b7 to b0	The peripheral clock is divided by (M + 1) (M = setting value (0 to 255)) by the baud rate prescaler	H'00 to H'FF	R/W

This register should be set in LIN reset mode.

19.1.4 LIN Self-test Control Register (LSTC)

Address H'FF61 0004

	b7	b6	b5	b4	b3	b2	b1	b0
								LSTM
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b1	When H'A7, H'58, and H'01 are consecutively written, the LIN enters LIN self-test mode. These bits are read as 0.			R/W
b0	LSTM	Self-test Mode Bit	0: Non-LIN self-test mode 1: LIN self-test mode	R/W

This register should be set in LIN reset mode.
This register is used to unlock the protection of LIN self-test mode.
When H'A7, H'58, and H'01 are consecutively written to this register, the LIN enters LIN self-test mode.
If consecutive writing is successful and the LIN has entered LIN self-test mode, the LSTM bit is set to 1.
Do not perform another writing while consecutive writing is in progress.
For the details of transition to LIN self-test mode, refer to section 19.11, LIN Self-test Mode.

LSTM Bit

This bit is set to 1 when the LIN enters LIN self-test mode.
For the details of how to exit LIN self-test mode, refer to section 19.11, LIN Self-test Mode.
Other than consecutively writing H'A7, H'58, and H'01 to the LSTC register, writing a 1 does not change the value of this bit.

19.1.5 LINi Mode Register (LiMD)

Address: L0MD: H'FF61 0008, L1MD: H'FF61 0028, L2MD: H'FF61 0048, L3MD: H'FF61 0068, L4MD: H'FF61 0088, L5MD: H'FF61 00A8, L6MD: H'FF61 00C8, L7MD: H'FF61 00E8, L8MD: H'FF61 0108, L9MD: H'FF61 0128, L10MD: H'FF61 0148, L11MD: H'FF61 0168

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	LCKS[1:0]		—	—
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R
b3, b2	LCKS[1:0]	LIN System Clock Select Bits	b3 b2 0 0 : fa 0 1 : fb 1 0 : fc 1 1 : fd	R/W
b1, b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

This register should be set in LIN reset mode.

LCKS Bit

This bit selects the clock to be input to the protocol controller.
When this bit is set to B'00, fa (clock generated by baud prescaler 0) is input to the protocol controller.
When this bit is set to B'01, fb (clock generated by baud prescaler 0 divided by 2) is input to the protocol controller.
When this bit is set to B'10, fc (clock generated by baud prescaler 0 divided by 8) is input to the protocol controller.
When this bit is set to B'11, fd (clock generated by baud prescaler 1 divided by 2) is input to the protocol controller.
When the LWBR0 bit in the LWBR register is set to 1 (when LIN Specification Package Revisions 2.0 and 2.1 are used) and moreover the L0MST register is set to H'01 (LIN wake-up mode), fa is input to the protocol controller regardless of the LCKS bit (the LCKS bit does not change).

19.1.6 LINi Break Field Setting Register (LiBRK)

Address: L0BRK: H'FF61 0009, L1BRK: H'FF61 0029, L2BRK: H'FF61 0049, L3BRK: H'FF61 0069, L4BRK: H'FF61 0089, L5BRK: H'FF61 00A9, L6BRK: H'FF61 00C9, L7BRK: H'FF61 00E9, L8BRK: H'FF61 0109, L9BRK: H'FF61 0129, L10BRK: H'FF61 0149, L11BRK: H'FF61 0169

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	BDT[1:0]		BLT[3:0]			
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R
b5, b4	BDT[1:0]	Break Delimiter Transmission Setting Bits	b5 b4 0 0 : 1 Tbit 0 1 : 2 Tbits 1 0 : 3 Tbits 1 1 : 4 Tbits	R/W
b3 to b0	BLT[3:0]	Break Transmission Setting Bits	b3 b2 b1 b0 0 0 0 0 : 13 Tbits 0 0 0 1 : 14 Tbits 0 0 1 0 : 15 Tbits : 1 1 1 0 : 27 Tbits 1 1 1 1 : 28 Tbits	R/W

This register should be set in LIN reset mode.
The length of one frame may exceed frame timeout time depending on the combination of setting values. Make sure to set appropriate values.

BDT Bit

This bit sets the (High) time pulse width for break delimiter in the transmit frame header.
A value from 1 Tbit to 4 Tbits can be set.

BLT Bit

This bit sets the (Low) time pulse width for break in the transmit frame header.
A value from 13 Tbits to 28 Tbits can be set.

19.1.7 LINi Space Setting Register (LiSPC)

Address: L0SPC: H'FF61 000A, L1SPC: H'FF61 002A, L2SPC: H'FF61 004A, L3SPC: H'FF61 006A, L4SPC: H'FF61 008A, L5SPC: H'FF61 00AA, L6SPC: H'FF61 00CA, L7SPC: H'FF61 00EA, L8SPC: H'FF61 010A, L9SPC: H'FF61 012A, L10SPC: H'FF61 014A, L11SPC: H'FF61 016A

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	IBS[1:0]		—	IBSH[2:0]		
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R
b5, b4	IBS[1:0]	Interbyte Space Setting Bits	b5 b4 0 0 : 0 Tbit 0 1 : 1 Tbit 1 0 : 2 Tbits 1 1 : 3 Tbits	R/W
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R
b2 to b0	IBSH[2:0]	Interbyte Space (Header)/ Response Space Setting Bits	b2 b1 b0 0 0 0 : 0 Tbit 0 0 1 : 1 Tbit 0 1 0 : 2 Tbits 0 1 1 : 3 Tbits 1 0 0 : 4 Tbits 1 0 1 : 5 Tbits 1 1 0 : 6 Tbits 1 1 1 : 7 Tbits	R/W

This register should be set in LIN reset mode.
This register is enabled only for transmission (header and response). In response reception, this register is disabled.
The length of one frame may exceed frame timeout time depending on the combination of setting values. Make sure to set appropriate values.

IBS Bit

This bit sets the space width of the interbyte space required for the transmit frame response.
A value from 0 Tbit to 3 Tbits can be set.

IBSH Bit

This bit sets the space width of the interbyte space (Header) and response space required for the transmit frame header.
A value from 0 Tbit to 7 Tbits can be set.
The values of the interbyte space (Header) and response space are the same.

19.1.8 LINi Wake-up Setting Register (LiWUP)

Address: L0WUP: H'FF61 000B, L1WUP: H'FF61 002B, L2WUP: H'FF61 004B, L3WUP: H'FF61 006B, L4WUP: H'FF61 008B, L5WUP: H'FF61 00AB, L6WUP: H'FF61 00CB, L7WUP: H'FF61 00EB, L8WUP: H'FF61 010B, L9WUP: H'FF61 012B, L10WUP: H'FF61 014B, L11WUP: H'FF61 016B

	b7	b6	b5	b4	b3	b2	b1	b0
	WUTL[3:0]				—	—	—	—
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b4	WUTL[3:0]	Wake-up Transmission Low Time Pulse Width Setting Bits	b7 b6 b5 b4 0 0 0 0 : 1 Tbit 0 0 0 1 : 2 Tbits 0 0 1 0 : 3 Tbits 0 0 1 1 : 4 Tbits : 1 1 0 0 : 13 Tbits 1 1 0 1 : 14 Tbits 1 1 1 0 : 15 Tbits 1 1 1 1 : 16 Tbits	R/W
b3 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

This register should be set in LIN reset mode.

WUTL Bit

This bit sets the Low time pulse width of wake-up transmission.
A value from 1 Tbit to 16 Tbits can be set.
When the LWBR0 bit in the LWBR register is set to 1 (when LIN Specification Package Revisions 2.0 and 2.1 are used), fa is selected as a LIN system clock (fLIN) regardless of the LCKS bit in the L0MD register (the LCKS bit does not change).

19.1.9 LINi Interrupt Enable Register (LiIE)

Address: L0IE: H'FF61 000C, L1IE: H'FF61 002C, L2IE: H'FF61 004C, L3IE: H'FF61 006C, L4IE: H'FF61 008C, L5IE: H'FF61 00AC, L6IE: H'FF61 00CC, L7IE: H'FF61 00EC, L8IE: H'FF61 010C, L9IE: H'FF61 012C, L10IE: H'FF61 014C, L11IE: H'FF61 016C

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	ERRIE	FRCIE	FTCIE
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R
b2	ERRIE	Error Detection Interrupt Enable Bit	0: Error detection interrupt disabled 1: Error detection interrupt enabled	R/W
b1	FRCIE	Frame/Wake-up Receive Completion Interrupt Enable Bit	0: Frame/wake-up receive completion interrupt disabled 1: Frame/wake-up receive completion interrupt enabled	R/W
b0	FTCIE	Frame/Wake-up Transmit Completion Interrupt Enable Bit	0: Frame/wake-up transmit completion interrupt disabled 1: Frame/wake-up transmit completion interrupt enabled	R/W

This register should be set in LIN reset mode.

ERRIE Bit

This bit enables or disables an interrupt when an error is detected.
When this bit is set to 1, a LINi interrupt is generated when the ERR bit in the LiST register becomes 1.
When this bit is set to 0, a LINi interrupt is not generated when the ERR bit becomes 1.
The sources to generate the LINi interrupt are the following interrupts: bit error, physical bus error, frame timeout error, framing error, and check sum error.
The detection of the above errors except check sum error can be enabled or disabled by the LiEDE register.

FRCIE Bit

This bit enables or disables an interrupt when a frame or wake-up frame reception (input signal Low time counting) is completed.
When this bit is set to 1, a LINi interrupt is generated when the FRC bit in the LiST register becomes 1.
When this bit is set to 0, a LINi interrupt is not generated when the FRC bit becomes 1.

FTCIE Bit

This bit enables or disables an interrupt when a frame or wake-up frame transmission is completed.
When this bit is set to 1, a LINi interrupt is generated when the FTC bit in the LiST register becomes 1.
When this bit is set to 0, a LINi interrupt is not generated when the FTC bit becomes 1.

19.1.10 LINi Error Detection Enable Register (LiEDE)

Address: L0EDE: H'FF61 000D, L1EDE: H'FF61 002D, L2EDE: H'FF61 004D, L3EDE: H'FF61 006D, L4EDE: H'FF61 008D, L5EDE: H'FF61 00AD, L6EDE: H'FF61 00CD, L7EDE: H'FF61 00ED, L8EDE: H'FF61 010D, L9EDE: H'FF61 012D, L10EDE: H'FF61 014D, L11EDE: H'FF61 016D

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	FERE	FTERE	PBERE	BERE
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R
b3	FERE	Framing Error Detection Enable Bit	0: Framing error detection disabled 1: Framing error detection enabled	R/W
b2	FTERE	Frame Timeout Error Detection Enable Bit	0: Frame timeout error detection disabled 1: Frame timeout error detection enabled	R/W
b1	PBERE	Physical Bus Error Detection Enable Bit	0: Physical bus error detection disabled 1: Physical bus error detection enabled	R/W
b0	BERE	Bit Error Detection Enable Bit	0: Bit error detection disabled 1: Bit error detection enabled	R/W

This register should be set in LIN reset mode.

FERE Bit

This bit enables or disables framing error detection.
When this bit is set to 1, a framing error is detected.
When this bit is set to 0, no framing error is detected.
The detection result when this bit is 1 is reflected in the FER bit in the LiEST register.
For the details of framing error, refer to section 19.9, Error Status.

FTERE Bit

This bit enables or disables frame timeout error detection.
When this bit is set to 1, a frame timeout error is detected.
When this bit is set to 0, no frame timeout error is detected.
The detection result when this bit is 1 is reflected in the FTER bit in the LiEST register.
For the details of frame timeout error, refer to section 19.9, Error Status.

PBERE Bit

This bit enables or disables physical bus error detection.
When this bit is set to 1, a physical bus error is detected.
When this bit is set to 0, no physical bus error is detected.
The detection result when this bit is 1 is reflected in the PBER bit in the LiEST register.
For the details of physical bus error, refer to section 19.9, Error Status.

BERE Bit

This bit enables or disables bit error detection.
When this bit is set to 1, a bit error is detected.
When this bit is set to 0, no bit error is detected.
The detection result when this bit is 1 is reflected in the BER bit in the LiEST register.
For the details of bit error, refer to section 19.9, Error Status.

19.1.11 LINi Control Register (LiC)

Address: L0C: H'FF61 000E, L1C: H'FF61 002E, L2C: H'FF61 004E, L3C: H'FF61 006E, L4C: H'FF61 008E, L5C: H'FF61 00AE, L6C: H'FF61 00CE, L7C: H'FF61 00EE, L8C: H'FF61 010E, L9C: H'FF61 012E, L10C: H'FF61 014E, L11C: H'FF61 016E

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	OM1	OM0
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b1	OM1	LIN Mode Select Bit	0: LIN wake-up mode 1: LIN operation mode	R/W
b0	OM0	LIN Reset Bit	0: LIN reset mode 1: Non-LIN reset mode	R/W

When the LIN exits LIN reset mode, this register should be set to H'01 to enter LIN wake-up mode and H'03 to enter LIN operation mode.
In LIN self-test mode, this register should be set to H'03 after the LIN enters LIN self-test mode.

OM1 Bit

This bit selects the LIN operational mode (LIN wake-up mode or LIN operation mode) when the LIN exits LIN reset mode.
When this bit is set to 0, the LIN enters LIN wake-up mode.
When this bit is set to 1, the LIN enters LIN operation mode.
This register is enabled only when the OMM0 bit in the LiMST register is set to 1.
This bit cannot be written to while the FTS bit in the LiTC register is 1.

OM0 Bit

This bit selects whether the LIN enters or exits in LIN reset mode.
When this bit is set to 0, the LIN enters LIN reset mode.
When this bit is set to 1, the LIN exits LIN reset mode.

19.1.12 LINi Transmission Control Register (LiTC)

Address: L0TC: H'FF61 0010, L1TC: H'FF61 0030, L2TC: H'FF61 0050, L3TC: H'FF61 0070, L4TC: H'FF61 0090, L5TC: H'FF61 00B0, L6TC: H'FF61 00D0, L7TC: H'FF61 00F0, L8TC: H'FF61 0110, L9TC: H'FF61 0130, L10TC: H'FF61 0150, L11TC: H'FF61 0170

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	RTS	FTS
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b1	RTS	Response Transmission Start Bit	0: Stop response transmission in frame separate mode 1: Start response transmission in frame separate mode	R/W
b0	FTS	Frame Transmission/ Wake-up Transmission and Reception Start Bit	0: Stop frame transmission/wake-up transmission and reception 1: Start frame transmission/wake-up transmission and reception	R/W

RTS Bit

Set this bit to 1 when a response transmission starts.

This bit is set to 1 during communication. It becomes 0 when there is no communication and when the LIN enters LIN reset mode.

This bit can be set to 1 only. It cannot be set to 0. This bit automatically becomes 0 when the transmission is completed.

Set this bit when the FSM bit in the LiRFC register is 1 (frame separate mode) and the FTS bit is 1 (frame transmission/wake-up transmission and reception started).

When setting the RTS bit to 1, write H'02 to the LINi transmission control register (LiTC) with a MOV instruction.

FTS Bit

Set this bit to 1 when a frame/wake-up transmission starts.

Set this bit to 1 to perform a wake-up reception (input signal Low time counting).

This bit is set to 1 during communication. It becomes to 0 when there is no communication and when the LIN enters LIN reset mode.

This bit can be set to 1 only. It cannot be set to 0. This bit automatically becomes 0 when the transmission is completed.

19.1.13 LINi Mode Status Register (LiMST)

Address: L0MST: H'FF61 0011, L1MST: H'FF61 0031, L2MST: H'FF61 0051, L3MST: H'FF61 0071, L4MST: H'FF61 0091, L5MST: H'FF61 00B1, L6MST: H'FF61 00D1, L7MST: H'FF61 00F1, L8MST: H'FF61 0111, L9MST: H'FF61 0131, L10MST: H'FF61 0151, L11MST: H'FF61 0171

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	OMM1	OMM0
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b1	OMM1	LINi Mode Status Monitor	0: LIN0 wake-up mode 1: LIN0 operation mode	R
b0	OMM0	LINi Reset Status Monitor	0: LIN0 reset mode 1: Non-LIN0 reset mode	R

Bits OMM0 and OMM1

These bits are used to check the current operational mode.

19.1.14 LINi Status Register (LiST)

Address: L0ST: H'FF61 0012, L1ST: H'FF61 0032, L2ST: H'FF61 0052, L3ST: H'FF61 0072, L4ST: H'FF61 0092, L5ST: H'FF61 00B2, L6ST: H'FF61 00D2, L7ST: H'FF61 00F2, L8ST: H'FF61 0112, L9ST: H'FF61 0132, L10ST: H'FF61 0152, L11ST: H'FF61 0172

	b7	b6	b5	b4	b3	b2	b1	b0
	HTRC	D1RC	—	—	ERR	—	FRC	FTC
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	HTRC	Header Transmit Completion Flag	0: Transmission uncompleted 1: Header transmission completed	R/W
b6	D1RC	Data 1 Receive Completion Flag	0: Reception uncompleted 1: Data 1 reception completed	R/W
b5, b4	—	Reserved	These bits are read as 0. The write value should be 0.	R
b3	ERR	Error Detection Flag	0: Error not detected 1: Error detected	R
b2	—	Reserved	This bit is read as 0. The write value should be 0.	R
b1	FRC	Frame/Wake-up Receive Completion Flag	0: Reception uncompleted 1: Frame or wake-up reception has been completed	R/W
b0	FTC	Frame/Wake-up Transmit Completion Flag	0: Transmission uncompleted 1: Frame or wake-up transmission has been completed	R/W

This register automatically becomes H'00 when the LIN enters LIN reset mode and when the next communication starts.

This register retains H'00 during LIN reset mode.

Do not write to this register while the FTS bit in the LiTC register is set to 1 (start frame transmission/wake-up transmission and reception).

HTRC Bit

This bit can be set to 0 only. Writing a 1 to this bit has no effect.
This bit becomes 1 when a header reception is completed, but an interrupt is not generated. To set this bit to 0 before the next communication starts, write a 0 in LIN operation mode.

D1RC Bit

This bit can be set to 0 only. Writing a 1 to this bit has no effect.
This bit becomes 1 when a data 1 reception is completed, but an interrupt is not generated. To set this bit to 0 before the next communication starts, write a 0 in LIN operation mode.

ERR Bit

This bit becomes 1 when an error is detected. An interrupt is generated if the ERRIE bit in the LiIE register is set to 1 (interrupt enabled) at this time. To set this bit to 0 before the next communication starts, write a 0 to bits BER, PBER, FTER, FER, and CSER in the LiEST register in LIN operation mode. The ERR bit becomes 0.

FRC Bit

This bit can be set to 0 only. Writing a 1 to this bit has no effect.
This bit becomes 1 when a frame or wake-up reception (input signal Low time counting) is completed. An interrupt is generated if the FRCIE bit in the LiIE register is set to 1 (interrupt enabled) at this time. To set this bit to 0 before the next communication starts, write a 0 in LIN operation mode.

FTC Bit

This bit can be set to 0 only. Writing a 1 to this bit has no effect.
This bit becomes 1 when a frame or wake-up transmission is completed. An interrupt is generated if the FTCIE bit in the LiIE register is set to 1 (interrupt enabled) at this time. To set this bit to 0 before the next communication starts, write a 0 in LIN operation mode.

19.1.15 LINi Error Status Register (LiEST)

Address: L0EST: H'FF61 0013, L1EST: H'FF61 0033, L2EST: H'FF61 0053, L3EST: H'FF61 0073, L4EST: H'FF61 0093, L5EST: H'FF61 00B3, L6EST: H'FF61 00D3, L7EST: H'FF61 00F3, L8EST: H'FF61 0113, L9EST: H'FF61 0133, L10EST: H'FF61 0153, L11EST: H'FF61 0173

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	CSER	—	FER	FTER	PBER	BER
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R
b5	CSER	Check Sum Error Flag	0: Check sum error not detected 1: Check sum error detected	R/W
b4	—	Reserved	This bit is read as 0. The write value should be 0.	R
b3	FER	Framing Error Flag	0: Framing error not detected 1: Framing error detected	R/W
b2	FTER	Frame Timeout Error Flag	0: Frame timeout error not detected 1: Frame timeout error detected	R/W
b1	PBER	Physical Bus Error Flag	0: Physical bus error not detected 1: Physical bus error detected	R/W
b0	BER	Bit Error Flag	0: Bit error not detected 1: Bit error detected	R/W

This register automatically becomes H'00 when the LIN enters LIN reset mode and when the next communication starts.

This register retains H'00 during LIN reset mode.

Do not write to this register while the FTS bit in the LiTC register is set to 1 (start frame transmission/wake-up transmission and reception).

CSER Bit

Only a 0 can be written to this bit. Writing a 1 to this bit has no effect.

This bit becomes 1 when a check sum error is detected. To set this bit to 0 before the next communication starts, write a 0 in LIN operation mode.

FER Bit

Only a 0 can be written to this bit. Writing a 1 to this bit has no effect.

This bit becomes 1 when a framing error is detected. To set this bit to 0 before the next communication starts, write a 0 in LIN operation mode.

FTER Bit

Only a 0 can be written to this bit. Writing a 1 to this bit has no effect.

This bit becomes 1 when a frame timeout error is detected. To set this bit to 0 before the next communication starts, write a 0 in LIN operation mode.

PBER Bit

Only a 0 can be written to this bit. Writing a 1 to this bit has no effect.

This bit becomes 1 when a physical bus error is detected. To set this bit to 0 before the next communication starts, write a 0 in LIN operation mode.

BER Bit

Only a 0 can be written to this bit. Writing a 1 to this bit has no effect.
This bit becomes 1 when a bit error is detected. To set this bit to 0 before the next communication starts, write a 0 in LIN wake-up mode or LIN operation mode.

19.1.16 LINi Response Field Setting Register (LiRFC)

Address: L0RFC: H'FF61 0014, L1RFC: H'FF61 0034, L2RFC: H'FF61 0054, L3RFC: H'FF61 0074, L4RFC: H'FF61 0094, L5RFC: H'FF61 00B4, L6RFC: H'FF61 00D4, L7RFC: H'FF61 00F4, L8RFC: H'FF61 0114, L9RFC: H'FF61 0134, L10RFC: H'FF61 0154, L11RFC: H'FF61 0174

	b7	b6	b5	b4	b3	b2	b1	b0
	—	FSM	CSM	RFT	RFDL[3:0]			
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R
b6	FSM	Frame Separate Mode Select Bit	0: Non-frame separate mode 1: Frame separate mode	R/W
b5	CSM	Check Sum Model Select Bit	0: Classic check sum 1: Enhanced check sum	R/W
b4	RFT	Response Field Transmit/ Receive Direction Setting Bit	0: Reception 1: Transmission	R/W
b3 to b0	RFDL[3:0]	Response Field Data Length Setting Bits	b3 b2 b1 b0 0 0 0 0 : 0 bytes + check sum 0 0 0 1 : 1 byte + check sum 0 0 1 0 : 2 bytes + check sum : 0 1 1 1 : 7 bytes + check sum 1 0 0 0 : 8 bytes + check sum Do not set these bits to values other than the above.	R/W

This register should be set when the FTS bit in the LiTC register is 0 (frame transmission/wake-up transmission and reception stopped).

FSM Bit

When this bit is set to 0, the LIN does not enter frame separate mode.
When this bit is set to 1, the LIN enters frame separate mode.
This bit does not give influence during response reception (when the RFT bit is set to 0).
Set this bit to 0 before the LIN enters LIN self-test mode.
For the details of frame separate mode, see section 19.6.1.1, Frame Separate Mode.

CSM Bit

This bit sets a check sum model.
When this bit is set to 0, a classic check sum model is selected.
When this bit is set to 1, an enhanced check sum model is selected.
When frame timeout error is used (the FTERE bit in the LiEDE register is set to 1), the period of time (as time out value) depends on the CSM bit setting.
For the details, see section 19.9, Error Status.

RFT Bit

When this bit is set to 0, reception is performed though the response field. In LIN wake-up mode, wake-up reception (input signal Low time counting) is performed.
When this bit is set to 1, transmission is performed through the response field. In LIN wake-up mode, wake-up transmission is performed.

RFDL Bits

These bits set a response field data length.
The data length can be set in the range of zero to eight bytes and does not include a check sum size.

19.1.17 LINi ID Buffer Register (LiIDB)

Address: L0IDB: H'FF61 0015, L1IDB: H'FF61 0035, L2IDB: H'FF61 0055, L3IDB: H'FF61 0075, L4IDB: H'FF61 0095, L5IDB: H'FF61 00B5, L6IDB: H'FF61 00D5, L7IDB: H'FF61 00F5, L8IDB: H'FF61 0115, L9IDB: H'FF61 0135, L10IDB: H'FF61 0155, L11IDB: H'FF61 0175

	b7	b6	b5	b4	b3	b2	b1	b0
	IDP		ID					
After Reset	x	x	x	x	x	x	x	x

Bit	Symbol	Bit Name	Description	R/W
b7, b6	IDP	Parity Setting Bits	Set the parity bit (P0 or P1) to be transmitted by ID field	R/W
b5 to b0	ID	ID Setting Bits	Set a value of 6-bit ID to be transmitted by ID field	R/W

This register should be set when the FTS bit in the LiTC register is 0 (frame transmission/wake-up transmission and reception stopped).

In LIN self-test mode, the settings are as follows.

- When the RFT bit = 1 (transmission):
  The inverted value of the transmitted value can be read. The value to be transmitted can be written before communication.
- When the RFT bit = 0 (reception):
  The inverted value of the received value can be read. The value to be received can be written before communication.

IDP Bits

These bits set the parity bit (P0 or P1) to be transmitted by the ID field of the LIN frame.
The parity is not automatically calculated.
Set a calculation value.
If a wrong value is set in the parity bit, the bit is transmitted without change.

ID Bits

These bits set a value of six-bit ID to be transmitted by the ID field of the LIN frame.

19.1.18 LINi Check Sum Buffer Register (LiCBR)

Address: L0CBR: H'FF61 0016, L1CBR: H'FF61 0036, L2CBR: H'FF61 0056, L3CBR: H'FF61 0076, L4CBR: H'FF61 0096, L5CBR: H'FF61 00B6, L6CBR: H'FF61 00D6, L7CBR: H'FF61 00F6, L8CBR: H'FF61 0116, L9CBR: H'FF61 0136, L10CBR: H'FF61 0156, L11CBR: H'FF61 0176

	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	x	x	x	x	x	x	x	x

Bit	Description	R/W
b7 to b0	Store the check sum data to be transmitted and received	R/W

This register should be set when the FTS bit in the LiTC register is 0 (frame transmission/wake-up transmission and reception stopped).

In LIN operation mode, the settings are as follows.

- When the RFT bit = 1 (transmission):
  The transmitted value can be read. Writing is disabled.
- When the RFT bit = 0 (reception):
  The received value can be read. Writing is disabled.

In LIN self-test mode, the settings are as follows.

- When the RFT bit = 1 (transmission):
  The inverted value of the transmitted value can be read. Writing is disabled.
- When the RFT bit = 0 (reception):
  The inverted value of the received value can be read. The value to be received can be written before communication. Writing to this register is disabled in LIN reset mode or LIN wake-up mode.

19.1.19 LINi Data n Buffer Register (LiDBn) (n = 1 to 8)

Address

L0DB1: H'FF61 0018, L0DB2: H'FF61 0019, L0DB3: H'FF61 001A, L0DB4: H'FF61 001B,
L0DB5: H'FF61 001C, L0DB6: H'FF61 001D, L0DB7: H'FF61 001E, L0DB8: H'FF61 001F,
L1DB1: H'FF61 0038, L1DB2: H'FF61 0039, L1DB3: H'FF61 003A, L1DB4: H'FF61 003B,
L1DB5: H'FF61 003C, L1DB6: H'FF61 003D, L1DB7: H'FF61 003E, L1DB8: H'FF61 003F,
L2DB1: H'FF61 0058, L2DB2: H'FF61 0059, L2DB3: H'FF61 005A, L2DB4: H'FF61 005B,
L2DB5: H'FF61 005C, L2DB6: H'FF61 005D, L2DB7: H'FF61 005E, L2DB8: H'FF61 005F,
L3DB1: H'FF61 0078, L3DB2: H'FF61 0079, L3DB3: H'FF61 007A, L3DB4: H'FF61 007B,
L3DB5: H'FF61 007C, L3DB6: H'FF61 007D, L3DB7: H'FF61 007E, L3DB8: H'FF61 007F,
L4DB1: H'FF61 0098, L4DB2: H'FF61 0099, L4DB3: H'FF61 009A, L4DB4: H'FF61 009B,
L4DB5: H'FF61 009C, L4DB6: H'FF61 009D, L4DB7: H'FF61 009E, L4DB8: H'FF61 009F,
L5DB1: H'FF61 00B8, L5DB2: H'FF61 00B9, L5DB3: H'FF61 00BA, L5DB4: H'FF61 00BB,
L5DB5: H'FF61 00BC, L5DB6: H'FF61 00BD, L5DB7: H'FF61 00BE, L5DB8: H'FF61 00BF,
L6DB1: H'FF61 00D8, L6DB2: H'FF61 00D9, L6DB3: H'FF61 00DA, L6DB4: H'FF61 00DB,
L6DB5: H'FF61 00DC, L6DB6: H'FF61 00DD, L6DB7: H'FF61 00DE, L6DB8: H'FF61 00DF,
L7DB1: H'FF61 00F8, L7DB2: H'FF61 00F9, L7DB3: H'FF61 00FA, L7DB4: H'FF61 00FB,
L7DB5: H'FF61 00FC, L7DB6: H'FF61 00FD, L7DB7: H'FF61 00FE, L7DB8: H'FF61 00FF,
L8DB1: H'FF61 0118, L8DB2: H'FF61 0119, L8DB3: H'FF61 011A, L8DB4: H'FF61 011B,
L8DB5: H'FF61 011C, L8DB6: H'FF61 011D, L8DB7: H'FF61 011E, L8DB8: H'FF61 011F,
L9DB1: H'FF61 0138, L9DB2: H'FF61 0139, L9DB3: H'FF61 013A, L9DB4: H'FF61 013B,
L9DB5: H'FF61 013C, L9DB6: H'FF61 013D, L9DB7: H'FF61 013E, L9DB8: H'FF61 013F,
L10DB1: H'FF61 0158, L10DB2: H'FF61 0159, L10DB3: H'FF61 015A, L10DB4: H'FF61 015B,
L10DB5: H'FF61 015C, L10DB6: H'FF61 015D, L10DB7: H'FF61 015E, L10DB8: H'FF61 015F,
L11DB1: H'FF61 0178, L11DB2: H'FF61 0179, L11DB3: H'FF61 017A, L11DB4: H'FF61 017B,
L11DB5: H'FF61 017C, L11DB6: H'FF61 017D, L11DB7: H'FF61 017E, L11DB8: H'FF61 017F

	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	x	x	x	x	x	x	x	x

Bit	Description	Setting Range	R/W
b7 to b0	Set the data to be transmitted or read the received data	H'00 to H'FF	R/W

This register should be set in the following states.

In response transmission

- The RFT bit in the LiRFC register = 1 (transmission)
- The FSM bit in the LiRFC register = 0 (non-frame separate mode)
- The FTS bit in the LiTC register = 0 (frame transmission/wake-up transmission and reception stopped)

or

- The RFT bit in the LiRFC register = 1 (transmission)
- The FSM bit in the LiRFC register = 1 (frame separate mode)
- The RTS bit in the LiTC register = 0 (response transmission stopped)

In response reception

The received data is overwritten.

When an error is detected, data before the reception is aborted is stored.

In LIN self-test mode, the settings are as follows.

- When the RFT bit = 1 (transmission):
  The inverted value of the transmitted value can be read. The value to be transmitted can be written before communication.
- When the RFT bit = 0 (reception):
  The inverted value of the received value can be read. The value to be received can be written before communication.

19.2 Operational Mode

The LIN has the following four operational modes:

- LIN reset mode
- LIN operation mode
- LIN wake-up mode
- LIN self-test mode

In LIN reset mode, the power consumption can be reduced since the clock is not provided to the LIN.
Figure 19.2 shows the transition processes between LIN operational mode, and Table 19.10 lists functions available in each mode.

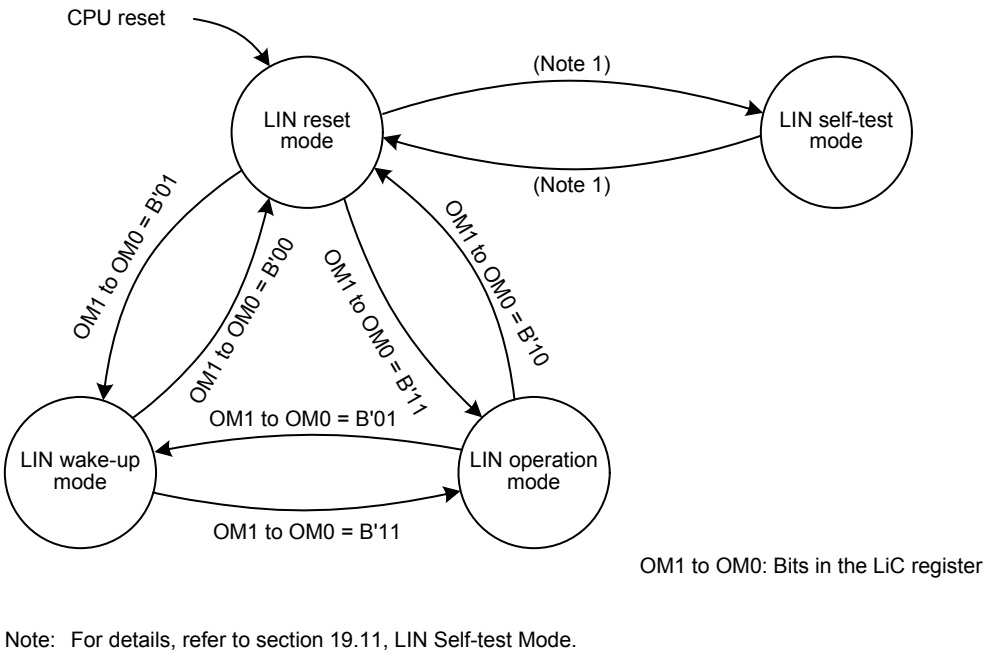


Figure 19.2 Operational Mode Transition

Table 19.10 Available Functions in Each Operational Mode

LIN Reset Mode	LIN Operation Mode	LIN Wake-up Mode	LIN Self-test Mode
LINi Low detection	Header transmission Response transmission Response reception Error detection LINi Low detection	Wake-up transmission Wake-up reception Error detection LINi Low detection	Self-test

The transition to LIN reset mode, LIN operation mode, and LIN wake-up mode can be checked by reading bits OMM1 to OMM0 in the LiMST register.
For the details of LIN self-test mode, refer to section 19.11, LIN Self-test Mode.

19.2.1 LIN Reset Mode

LIN reset mode is activated by setting bits OM1 to OM0 in the LiC register to B'00 or B'10 (LIN reset mode), and the transition to LIN reset mode can be checked when bits OMM1 to OMM0 in the LiMST register become B'00 or B'10 (LIN reset mode). In this mode, all the LIN functions are stopped as well as fLIN.
The LIN can enter LIN operation mode, LIN wake-up mode, or LIN self-test mode from LIN reset mode.

The following registers are initialized to their reset values after the LIN enters LIN reset mode. The registers retain their initial values during LIN reset mode.

- LiTC register
- LiST register
- LiEST register

The following registers retain their previous values after the LIN enters LIN reset mode.

- LWBR register
- LBRP0 register
- LBRP1 register
- LiMD register
- LiBRK register
- LiSPC register
- LiWUP register
- LiIE register
- LiEDE register
- LiRFC register
- LiCBR register
- LiIDB register
- LiDBn register

19.2.2 LIN Operation Mode

LIN operation mode is activated by setting bits OM1 to OM0 in the LiC register to B'11, and bits OMM1 to OMM0 in the LiMST register become B'11.

19.2.3 LIN Wake-up Mode

LIN wake-up mode is activated by setting bits OM1 to OM0 in the LiC register to B'01, and bits OMM1 to OMM0 in the LiMST register become B'01.

19.2.4 LIN Self-test Mode

LIN self-test mode is activated by writing to the LSTC register, and the transition to LIN self-test mode can be checked when the LSTM bit in the LSTC register is set to 1.

19.3 Operational Overview

19.3.1 Header Transmission

Figure 19.3 shows the Operation in Header Transmission, and Table 19.11 lists the Processing in Header Transmission.

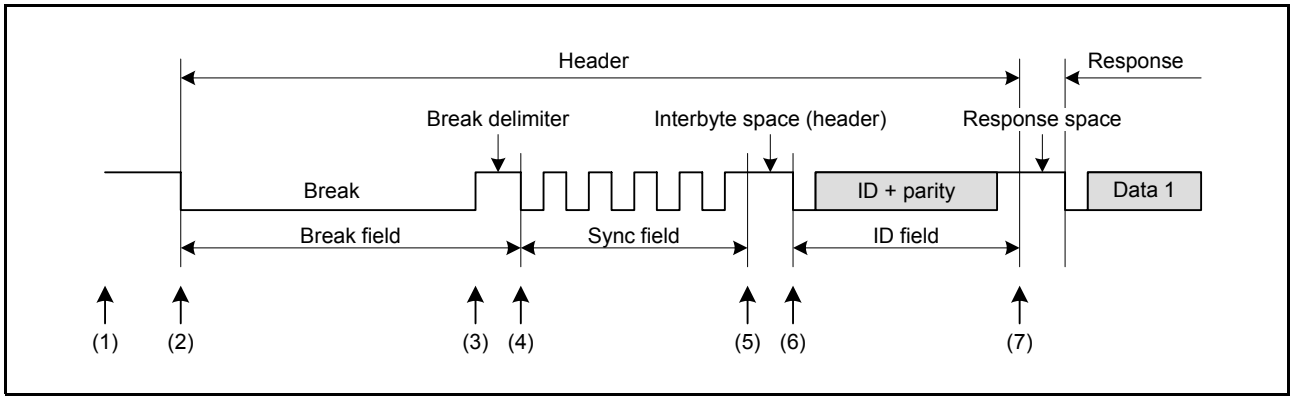


Figure 19.3 Operation in Header Transmission

Table 19.11 Processing in Header Transmission

	Software Processing	LIN Processing
(1)	• Set a baud rate (Refer to section 19.4, Baud Rate Generator.) • Set the following bits in the LiIE register: FTCIE bit to 1 (frame/wake-up transmit completion interrupt enabled); FRCIE bit to 1 (frame/wake-up receive completion interrupt enabled); ERRIE bit to 1 (error detection interrupt enabled) • Set bits OM1 to OM0 in the LiC register for LIN mode operation • Set the following bits in the LiBRK register: BLT bits for break dominant (13 to 28 Tbits); BDT bits for break delimiter (1 to 4 Tbits) • Set the following bits in the LiSPC register: IBSH bits for interbyte space (Header)/response space (0 to 7 Tbits); IBS bits for interbyte space (0 to 3 Tbits) • Set the following bits in the LiIDB register: ID bit for ID value; IDP bit for parity value • Set the following bits in the LiRFC register: RFDL bits for data length; RFT bit for response transmit/receive direction; CSM bit for check sum model • Set the transmit data	Wait for frame/wake-up transmission start by software (idle)
(2)	• Set the FTS bit in the LiTC register to 1 (frame transmission/wake-up transmission and reception started)	Transmit break dominant
(3)		Transmit break delimiter
(4)		Transmit sync field (55h)
(5)		Transmit interbyte space (Header)
(6)		Transmit ID field
(7)		• Set header transmit completion flag or error flag Transmit response space

19.3.2 Response Transmission

Figure 19.4 shows the Operation in Response Transmission, and Table 19.12 lists the Processing in Response Transmission.

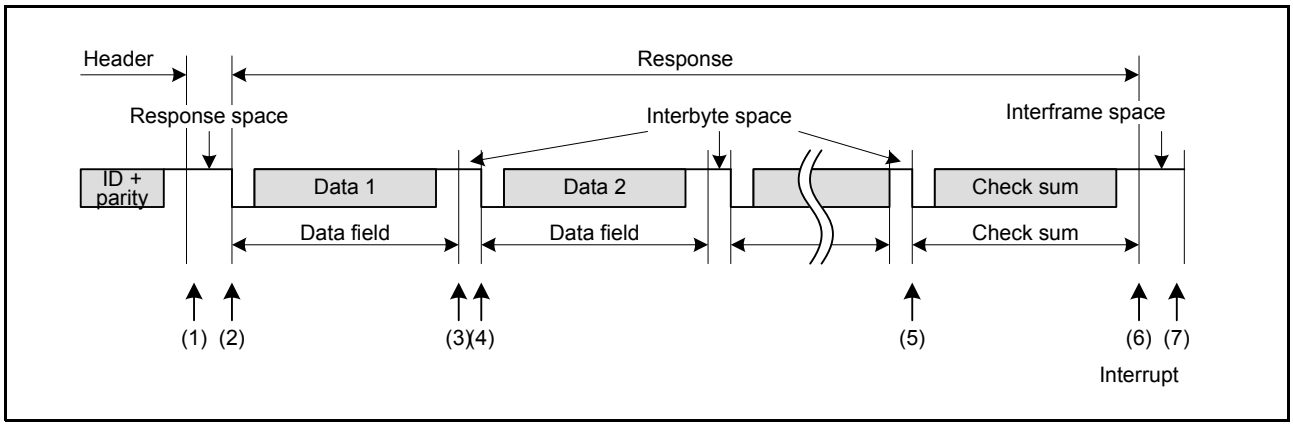


Figure 19.4 Operation in Response Transmission

Table 19.12 Processing in Response Transmission

	Software Processing	LIN Processing
(1)	• In frame separate mode Set the RTS bit in the LiTC register to 1 (response transmission started) • In non-frame separate mode Wait for generating an interrupt request	• In frame separate mode Transmit response space while waiting for response transmission enabled • In non-frame separate mode Go to (2) if the response space transmission has been completed
(2)	Wait for generating an interrupt request	Transmit Data 1
(3)		Transmit interbyte space
(4)		Transmit Data 2, then the next interbyte space Transmit Data 3, then the next interbyte space Repeat this processing for the data length specified by the RFDL bits in the L0RFC register. Go to (6) if an error occurs. : :
(5)		Transmit check sum
(6)		• Set frame/wake-up transmit completion flag or error flag • Set the following bits in the LiTC register: FTS bit = 0 (frame transmission/wake-up transmission and reception stopped); RTS bit = 0 (response transmission stopped)
(7)	Processing after communication Check the LiST register and set flags to 0	Idle

19.3.3 Response Reception

Figure 19.5 shows the Operation in Response Reception, and Table 19.13 lists the Processing in Response Reception.

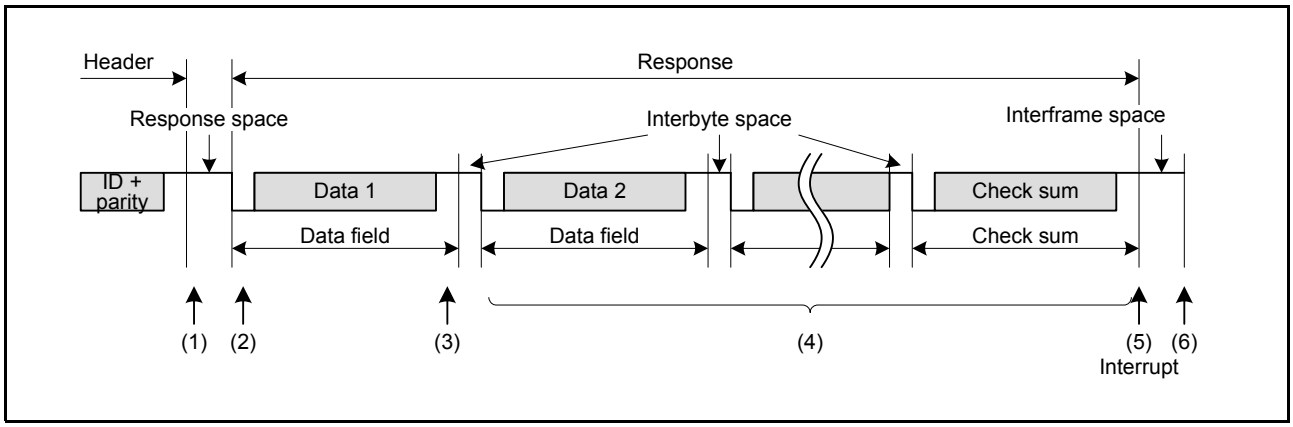


Figure 19.5 Operation in Response Reception

Table 19.13 Processing in Response Reception

	Software Processing	LIN Processing
(1)	Wait for generating an interrupt request (without processing)	Wait for detecting a start bit
(2)	Wait for generating an interrupt request	Receive Data 1 due to a start bit detected
(3)		• Set Data 1 receive completion flag
(4)		Receive Data 2 due to a start bit detected Receive Data 3 due to a start bit detected Repeat this processing for the data length specified by the RFDL bits in the L0RFC register. Stop it when any bit in the LiEST register becomes 1 (error detected). If an error occurs, check-sum judgement in (5) is not executed. ⋮ Receive check sum due to a start bit detected
(5)		• Judge the check sum • Set frame receive completion flag or error flag • Set the FTS bit in the LiTC register to 0 (frame transmission/wake-up transmission and reception stopped)
(6)	Processing after communication Read the received data Check the LiST register and set flags to 0	Idle

19.4 Baud Rate Generator

The LIN system clock (fLIN) is generated by the peripheral clock divided by the baud rate generator. This fLIN, divided by 16 is used as bit rate. The reciprocal of the bit rate is called bit time, expressed as “Tbit”.

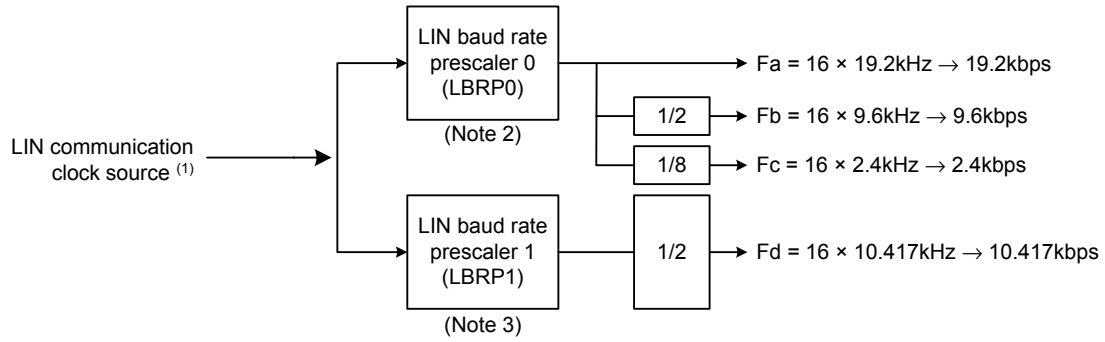
The LBRP0 register should be set so that fa is 307200 Hz (19200 bps × 16). Therefore,

fa = 19200 bps × 16,

fb = 9600 bps × 16, and

fc = 2400 bps × 16.

Then they are divided by 16 in the bit timing generator, which results in 19200 bps, 9600 bps, and 2400 bps, respectively. The baud rate of 10417 bps is generated by the bit setting of the LBRP1 register.



Note 1. Refer to chapter 5. “Clocks” for the details of the LIN communication source.
Note 2. Divide-by-(n+1) (n = 1 to 255) is set by the LBRP0 register.
Note 3. Divide-by-(m+1) (m = 1 to 255) is set by the LBRP1 register.

Figure 19.6 Baud Rate Generation Block Diagram

Table 19.14 and Table 19.15 list the baud rates (19200, 9600, 2400, and 10417 bps) generated by the peripheral clock frequency, and their error.

Table 19.14 Baud Rate Generation Example for 19200 bps, 9600 bps, and 2400 bps

Peripheral Clock	Baud Rate Generator 0 Divide-by-(N + 1)	Baud Rate to be Generated			Error (unit: %)
		fa (unit: bps)	fb (unit: bps)	fc (unit: bps)	
40 MHz	130	19230.77 (19200)	9615.38 (9600)	2403.85 (2400)	+0.16
25 MHz	81	19290.12 (19200)	9645.06 (9600)	2411.27 (2400)	+0.47
24 MHz	78	19230.77 (19200)	9615.38 (9600)	2403.85 (2400)	+0.16
20 MHz	65	19230.77 (19200)	9615.38 (9600)	2403.85 (2400)	+0.16
16 MHz	52	19230.77 (19200)	9615.38 (9600)	2403.85 (2400)	+0.16
12 MHz	39	19230.77 (19200)	9615.38 (9600)	2403.85 (2400)	+0.16
10 MHz	65	9615.38 (9600)	—	—	+0.16
8 MHz	26	19230.77 (19200)	9615.38 (9600)	2403.85 (2400)	+0.16
6 MHz	39	9615.38 (9600)	—	—	+0.16
	156	2403.85 (2400)			+0.16
5 MHz	130	2403.85 (2400)	—	—	+0.16
4 MHz	13	19230.77 (19200)	9615.38 (9600)	2403.85 (2400)	+0.16
2 MHz	13	9615.38 (9600)	—	—	+0.16
	52	2403.85 (2400)			+0.16

—: Corresponding baud rate cannot be generated

Table 19.15 Baud Rate Generation Example for 10417 bps

Peripheral Clock	Baud Rate Generator 1 Divide-by-(M + 1)	Baud Rate to be Generated fd (unit: bps)	Error (unit: %)
40 MHz	120	10416.67	-0.003
25 MHz	75	10416.67	-0.003
24 MHz	72	10416.67	-0.003
20 MHz	60	10416.67	-0.003
16 MHz	48	10416.67	-0.003
12 MHz	36	10416.67	-0.003
10 MHz	30	10416.67	-0.003
8 MHz	24	10416.67	-0.003
6 MHz	18	10416.67	-0.003
5 MHz	15	10416.67	-0.003
4 MHz	12	10416.67	-0.003
2 MHz	6	10416.67	-0.003

19.5 Data Transmission and Reception

19.5.1 Data Transmission

The LIN transmits one bit data per Tbit.

The transmitted data returns to the input pin for data reception via the LIN transceiver. Then the received data is compared to the transmitted data and the result is stored in the BER bit in the LiEST register (refer to section 19.9, Error Status). The timing to sample the received data is at the 13th clock (position of 81.25%) when 1 Tbit is generated by 16 fLIN.

Figure 19.7 shows the Data Transmission Timing.

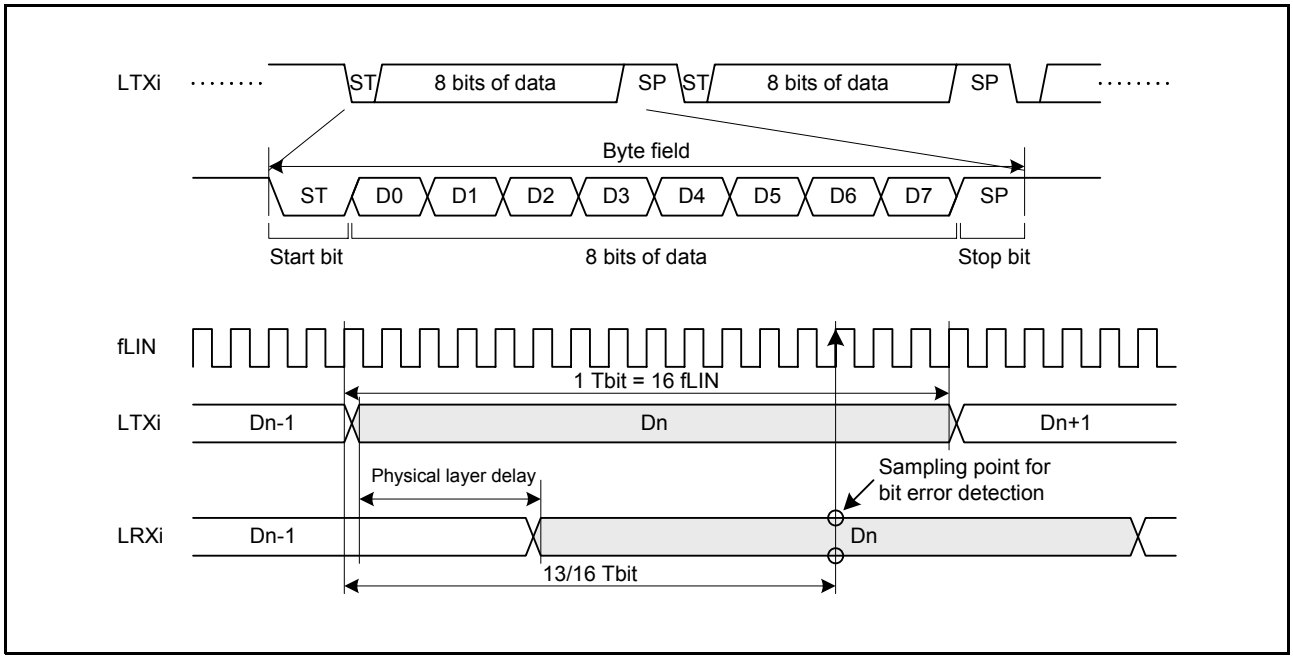


Figure 19.7 Data Transmission Timing

19.5.2 Data Reception

The data reception in the LIN requires an internal signal generated from the input signal at the LRXi pin by double-latch clocking scheme.

The byte field of this internal signal, called the synchronized LRXi, is synchronized with fLIN at the falling edge of the start bit. The start bit is verified if the synchronized LRXi signal is held low 0.5 Tbit after the falling edge is detected. Otherwise, that is, if the signal remains low after exiting reset mode or if it is held high on sampling, no start bit is found.

Once the start bit is detected, data bits are sampled every 1 Tbit.

Figure 19.8 shows the Data Reception Timing.

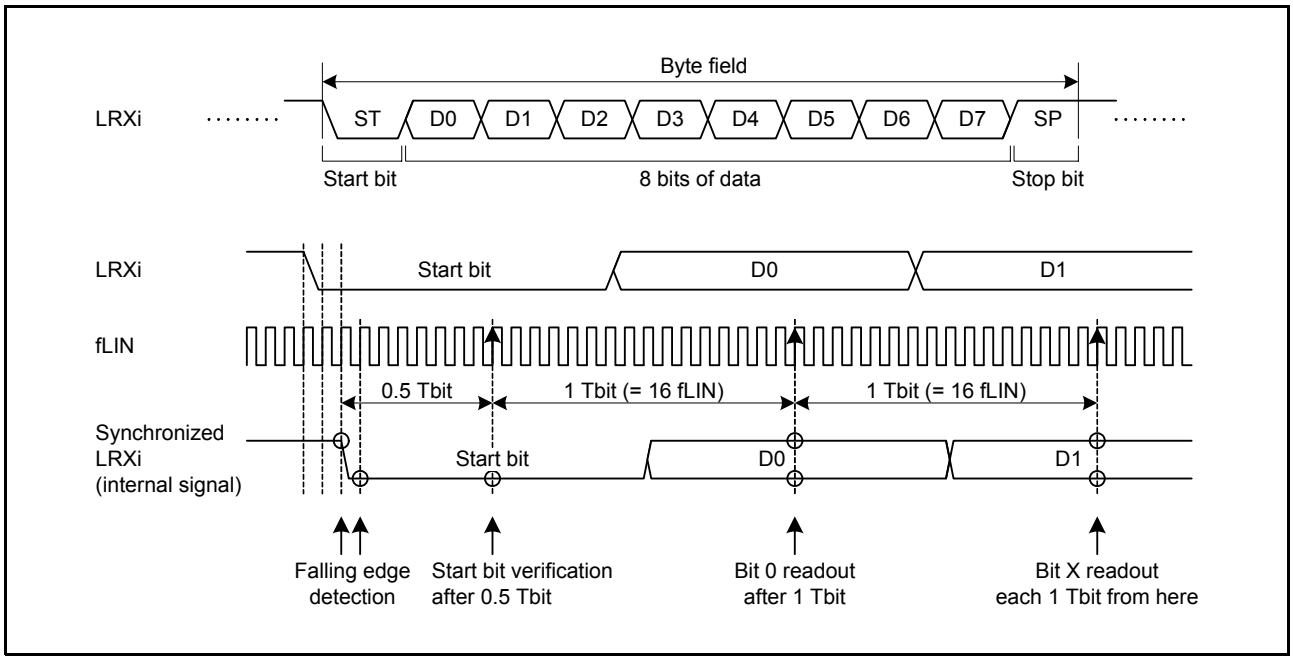


Figure 19.8 Data Reception Timing

19.6 Buffer Processing of Data to be Transmitted and Received Data

The buffer processing in consecutive data transmission and reception of the LIN is described below.

19.6.1 Transmission of LIN Frame

In 8-byte transmission, the contents stored in registers LiDB1 to LiDB8 are transmitted to Data 1 to Data 8 of the LIN frame in order.

In 4-byte transmission, the contents stored in registers LiDB1 to LiDB4 are transmitted to Data 1 to Data 4 of the LIN frame, and the contents stored in registers LiDB5 to LiDB8 are not transmitted. The transmitted check sum data is stored in the LiCB register.

Figure 19.9 shows LIN Transmission Processing and Buffers.

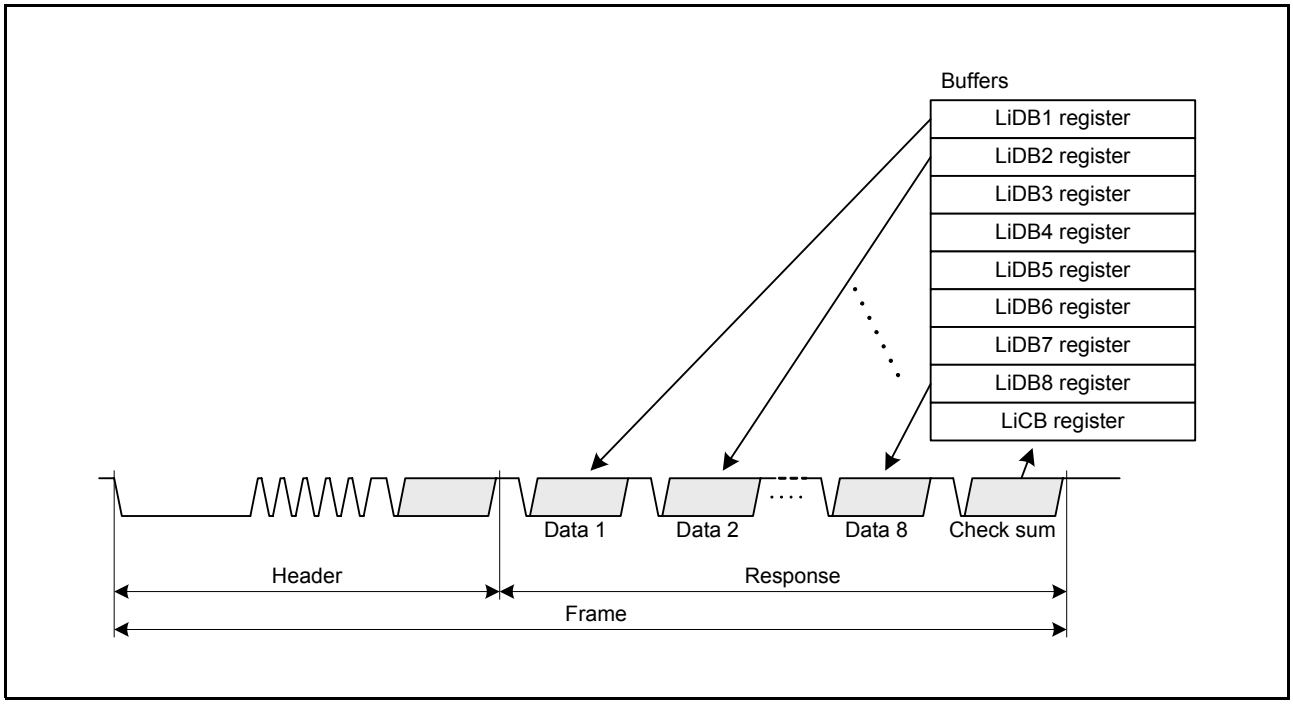


Figure 19.9 LIN Transmission Processing and Buffers

19.6.1.1 Frame Separate Mode

Frame separate mode is selected by setting the FSM bit in the LiRFC register to 1.

In frame separate mode, a header and response can be transmitted according to separate transmit start requests.

When frame separate mode is selected, the HTRC bit in the LiST register becomes 1 (header transmission completed) if the header transmission has been completed.

19.6.2 Reception of LIN Frame

In 8-byte reception, the contents of Data 1 to Data 8 of the LIN frame are stored in registers LiDB1 to LiDB8 in order in every completion of stop bit reception.

In 4-byte reception, the contents of Data 1 to Data 4 of the LIN frame are stored in registers LiDB1 to LiDB4 respectively, and no bits are stored in registers LiDB5 to LiDB8.

The received check-sum is stored in the LiCB register.

Figure 19.10 shows LIN Reception Processing and Buffers.

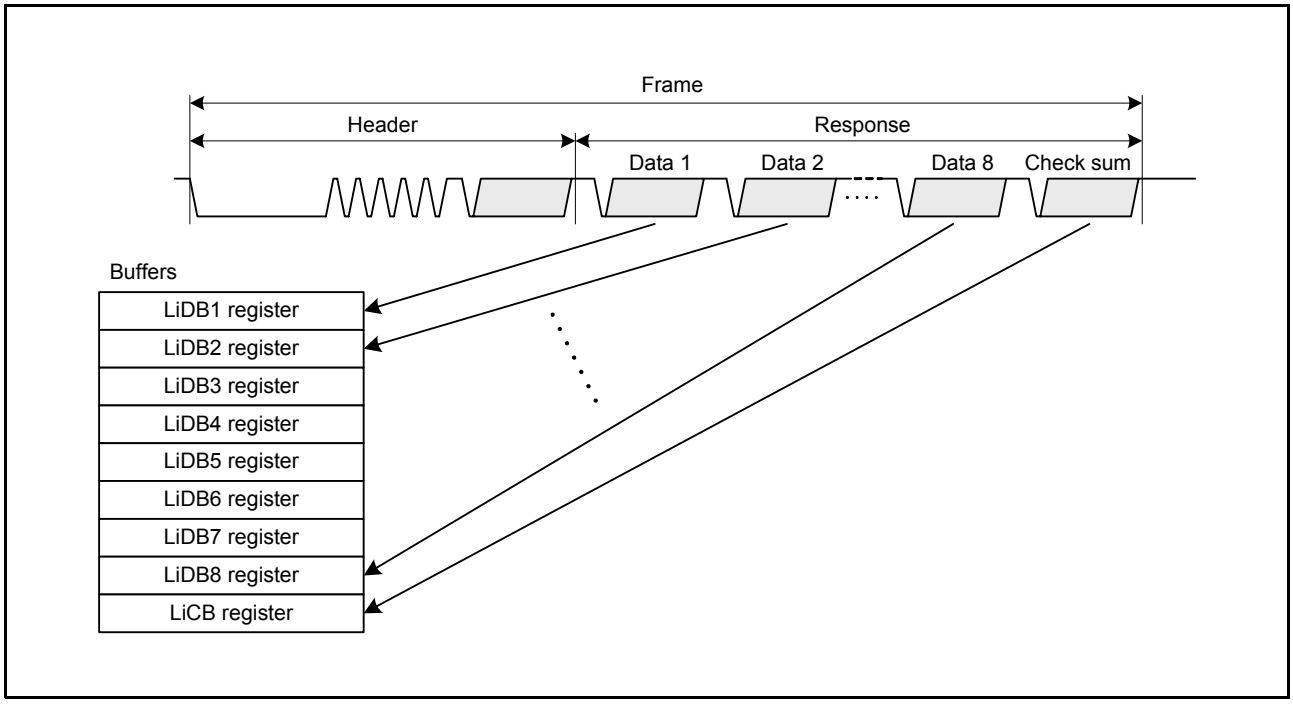


Figure 19.10 LIN Reception Processing and Buffers

19.6.2.1 Data 1 Reception

When the first byte of data reception is completed, the D1RC bit in the LiST register becomes 1 (data 1 reception completed).

19.7 Wake-up Transmission and Reception

Wake-up transmission and reception is available in LIN wake-up mode.

19.7.1 Operation of Wake-up Transmission

In LIN wake-up mode, if the RFT bit in the LiRFC register is set to 1 (transmission) and the FTS bit in the LiTC register is set to 1 (frame transmission/wake-up transmission and reception started), the wake-up signal is output at the output pin. Low time of the wake-up signal is set by the WUTL bits in the LiWUP register.
When low of wake-up is output without any bit error detected, the FTC bit in the LiST register becomes 1 (frame or wake-up transmission completed). If the FTCIE bit in the LiIE register is set to 1 (frame/wake-up transmit completion interrupt enabled), an interrupt request is generated.
When a bit error is detected, the operation is aborted and the BER bit in the LiEST register becomes 1 (bit error detected).
Figure 19.11 shows the Wake-up Transmission Timing.

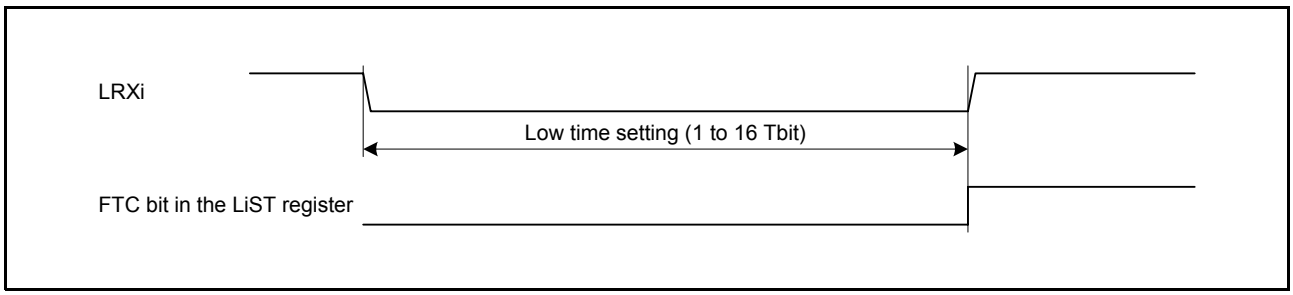


Figure 19.11 Wake-up Transmission Timing

19.7.2 Operation of Wake-up Reception

A wake-up is received by either of following two functions: LINi Low detection and input signal Low time counting.

The LINi Low detection is to asynchronously detect the falling edge of the input signal at the LRXi pin. When the rising edge of the input signal is detected, an interrupt request of LINi Low detection is generated.

The input signal Low time counting is to measure the low time of the input signal at the LRXi pin by counting at the same sampling timing as that of data reception. The input signal Low time can be measured 2.5 or more Tbits. When LIN Specification Package Revision 1.3 is used, set the LWBR0 bit in the LWBR register to 0. When LIN Specification Package Revisions 2.0 and 2.1 are used, set the LWBR0 bit to 1. This causes fa to be selected as a LIN system clock (fLIN) regardless of the LCKS bit in the L0MD register (the LCKS bit does not change).
To use this function, set the RFT bit in the LiRFC register to 0 (reception) and the FTS bit in the LiTC register to 1 (frame transmission/wake-up transmission and reception started).
When the Low time to be measured is reached, the FRC bit in the LiST register becomes 1 (frame or wake-up reception completed). If the FRCIE bit in the LiIE register is set to 1 (frame/wake-up receive completion interrupt enabled), an interrupt request is generated.

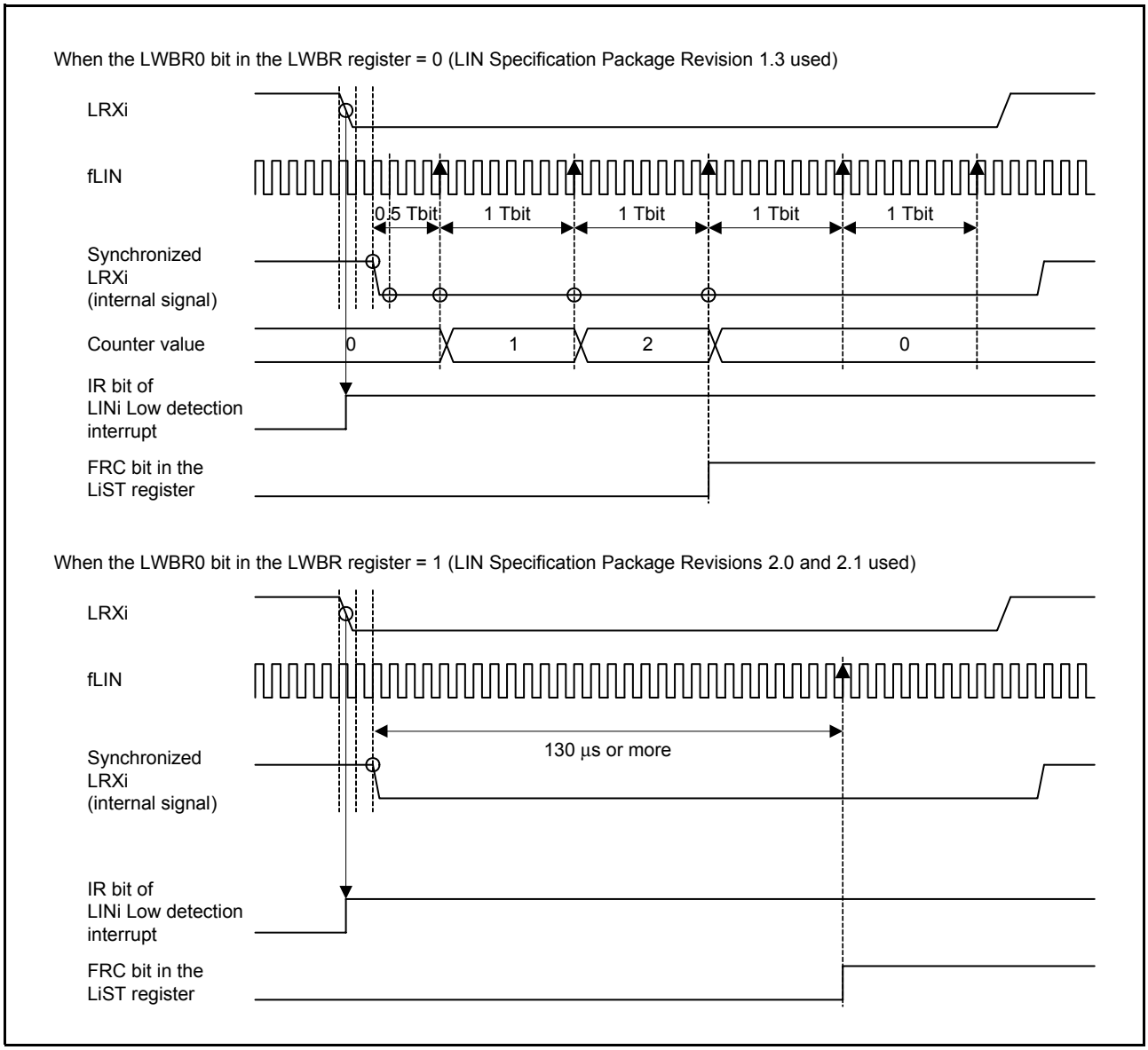


Figure 19.12 LINi Low Detection

During wake-up transmission, the LINi Low detection is workable, but the input signal Low time counting is not workable.

19.7.3 Low-Power Consumption Mode Control Using Input Signal Low Detection

The LINi Low detection function can be used as a factor to exit from standby mode, power down mode 0, or power down mode 1.

Figure 19.13 shows the Example of Processing before Transition to Standby Mode by Using LINi Low Detection Function. For the details of the transition to power down mode 0 or power down mode 1, see section 5.5, Power Control.

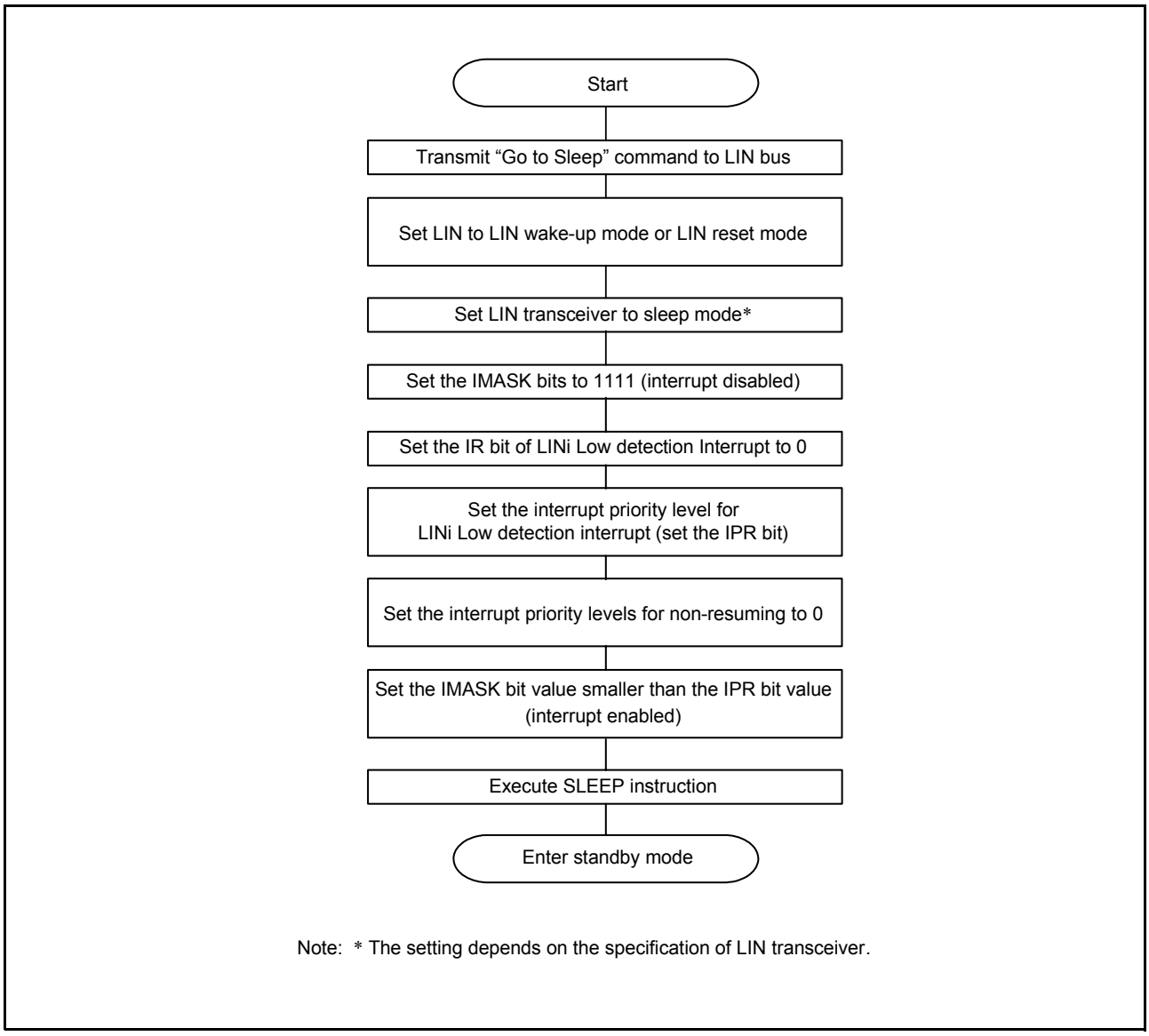


Figure 19.13 Example of Processing before Transition to Standby Mode by Using LINi Low Detection Function

19.7.4 Wake-up Collision

When the master and slave transmit wake-up signals simultaneously, a signal collision occurs on the LIN bus. However, no collision of wake-up signals is detected in the LIN.

19.8 Operational Status

The LIN detects seven types of operational status.

The status that can generate an interrupt request are: frame/wake-up transmit completion, frame/wake-up receive completion, and error detection.

Table 19.16 lists the types of Operational Status.

Table 19.16 Operational Status

Operational Status	Setting Condition	Clearing Condition	Detectable Operational Modes	Corresponding Bits
LIN mode	When LIN has entered LIN operation mode after the setting of the OM1 bit in the LiC register for LIN operation mode	When LIN has entered LIN wake-up mode after the setting of the OM1 bit in the LiC register for LIN wake-up mode	LIN operation mode LIN wake-up mode	OMM1 bit in the LiMST register
Reset	When LIN has exited LIN reset mode after the setting of the OM0 bit in the LiC register for non-LIN reset mode	When LIN has entered LIN reset mode after the setting of the OM0 bit in the LiC register for LIN reset mode	All modes	OMM0 bit in the LiMST register
Frame/wake-up transmit completion	When a response field or wake-up signal transmission has been successfully completed	When the next communication starts Clearing by software On transition to LIN reset mode	LIN operation mode LIN wake-up mode	FTC bit in the LiST register
Frame/wake-up receive completion	When a response field or wake-up signal reception has been successfully completed	When the next communication starts Clearing by software On transition to LIN reset mode	LIN operation mode LIN wake-up mode	FRC bit in the LiSRT register
Error detection	When any of bits BER, PBER, FTER, FER, CSER in the LiEST register becomes 1 (error detected)	When the next communication starts Clearing by software *1 On transition to LIN reset mode	LIN operation mode LIN wake-up mode	ERR bit in the LiST register
Data 1 receive completion	When the reception of the first one byte of a response frame has been completed in the bit setting as follows: RFT bit in the LiRFC register = 0 (reception) *2	When the next communication starts Clearing by software On transition to LIN reset mode	LIN operation mode	D1RC bit in the LiST register
Header transmit completion	If a header field transmission has been successfully completed	When the next communication starts Clearing by software On transition to LIN reset mode	LIN operation mode	HTRC bit in the LiST register

Notes: 1. By writing a 0 to bits BER, PEBER, FTER, FER, and CSER in the LiEST register in LIN operation mode, the ERR bit is set to 0.
2. This status is detected except when the RFDL bits in the LiRFC register are set to B'0000 (0 byte + check sum).

19.9 Error Status

19.9.1 Error Status Types

The LIN detects five types of error status. These error status can be checked by the bits in the LEST register. Table 19.17 lists the types of error status.

Table 19.17 Error Status Types

Error Status	Error Detecting Condition (clearing to 0 by software)	Error Detectable Operational Modes	Communication Processing	Detection Enabled/ Disabled	Corresponding Bits
Bit error	When the transmitted data does not match with that on LIN bus monitored by the pin for reception *1	LIN operation mode LIN wake-up mode	Abort	Selectable	BER bit in the LiEST register
Physical bus error	• If LIN bus detects high when the break field is transmitted • If LIN bus detects low when the break delimiter is transmitted • If LIN bus detects high when the wake-up is transmitted	LIN operation mode LIN wake-up mode	Abort	Selectable	PBER bit in the LiEST register
Frame timeout error	When the transmit/receive operation is not completed within a specified period of time *2	LIN operation mode	Abort	Selectable	FTER bit in the LiEST register
Framing error	When the stop bit of each data byte is low in response frame reception processing	LIN operation mode	Abort	Selectable	FER bit in the LiEST register
Check sum error	When the check sum judgement of response frame reception processing results in an error	LIN operation mode	—	Non-selectable	CSER bit in the LiEST register

Notes: 1. When a bit error is detected, the processing is aborted after the stop bit is transmitted. If it is detected in non-data space, such as a break field an interbyte space, or in wake-up transmission, the transmission is aborted immediately after the error bit is transmitted.

2. The period of time (as time out value) depends on the response field data length set by the RFDL bits in the LiRFC register and the check sum selected by the CSM bit in the LiRFC register as follows:

When classic check sum is selected (the CSM bit in the LiRFC register = 0)
Time out value = 49 + (data bytes + 1) ×14 [Tbit]
When enhanced check sum is selected (the CSM bit in the LiRFC register = 1)
Time out value = 48 + (data bytes + 1) ×14 [Tbit]

The above period of time will exceed the value of TFRAME_MAX shown in LIN Specification Package Revision 1.3 when classic check sum is selected, and the value of TFRAME_MAX shown in LIN Specification Package Revisions 2.0 and 2.1 when enhanced check sum is selected.

19.9.2 LIN Error Detection Targets

Figure 19.14 shows the area which the LIN monitors for error detection.

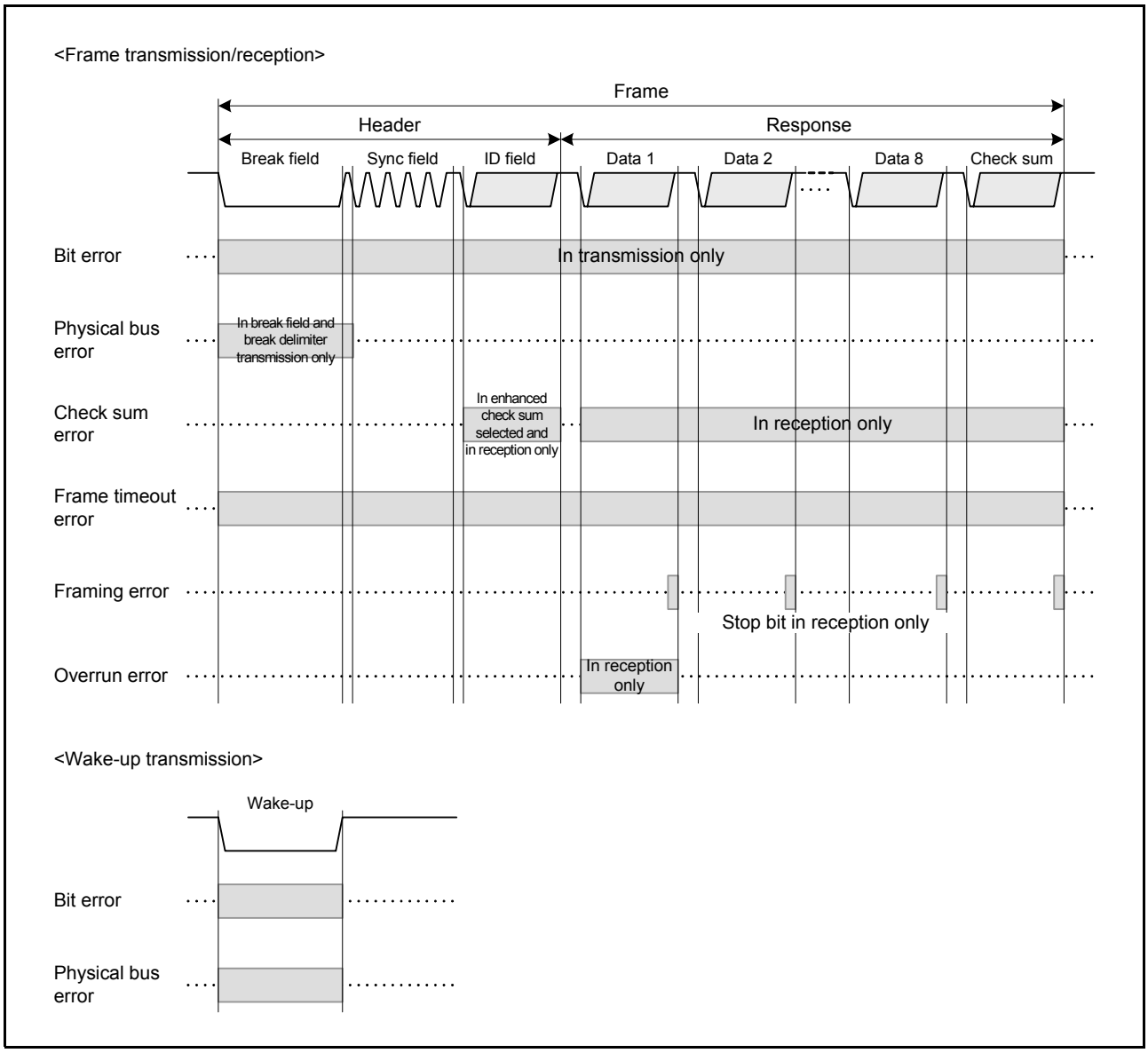


Figure 19.14 LIN Error Detection Targets

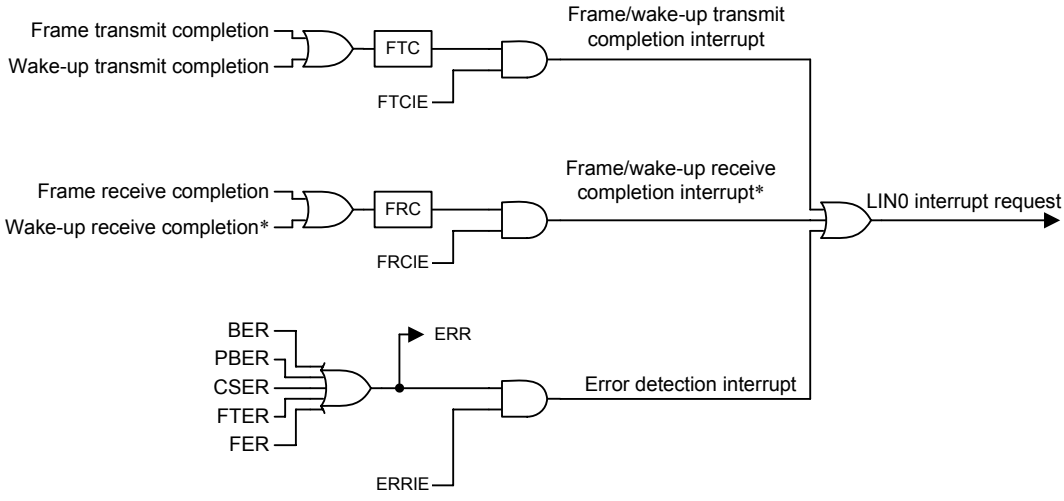
19.10 LIN Interrupt

The interrupts requests generated by the LIN are LINi interrupt and LINi Low detection interrupt. The channel has four sources for these interrupts as follows: frame/wake-up transmit completion, frame/wake-up receive (input signal Low time counting) completion, error detection, and LINi Low detection.

Among the four interrupt sources above, interrupt requests by the former three interrupt sources are aggregated by logical OR as interrupt request “LINi interrupt”. Each channel has an interrupt request generated by LINi Low detection.

The respective interrupt request is output when the corresponding flag in the LiST register becomes 1 while the corresponding bit in the LiIE register is set to 1 (interrupt enabled).

Figure 19.15 shows the block diagram of the LINi interrupt. For the details of the LINi Low detection interrupt, refer to section 8., Interrupt Controller (INTC).



FTC, FRC, and ERR: Bits in the LiST register
FTCIE, FRCIE, ERRIE: Bits in the LiIE register
BER, PBER, CSER, FTER, and FER: Bits in the LiEST register

Note: * The wake-up reception indicate input signal Low time counting.

Figure 19.15 LINi Interrupt Block Diagram

19.11 LIN Self-test Mode

The LIN has LIN self-test mode. Once the LIN enters LIN self-test mode, it is disconnected from the LIN bus, and the internal LTXi is looped back to the internal LRXi (loop back).

LIN self-test mode operates in the following states.

- In LIN self-test mode
- Wake-up is not supported
- Frame separate mode is not supported
- The baud rate generator is set to the fastest settings (LBRP0 register = H'00, LBRP1 register = H'00, LCKS bits = B'00)

Do not enter LIN wake-up mode.

Before entering LIN self-test mode, set the FSM bit in the LiRFC register to 0 (non-frame separate mode).

The baud rate setting is automatic in LIN self-test mode. When entering LIN reset mode from LIN self-test mode, the automatic setting is set back to the setting prior to LIN self-test mode.

Other settings are retained and remain valid when entering LIN self-test mode, and when entering reset mode from LIN self-test mode.

Registers LiST and LiEST remain functional, but bits BER, PBER, and FER in the LiEST register cannot be detect any errors at the loop back, and do not become 1.

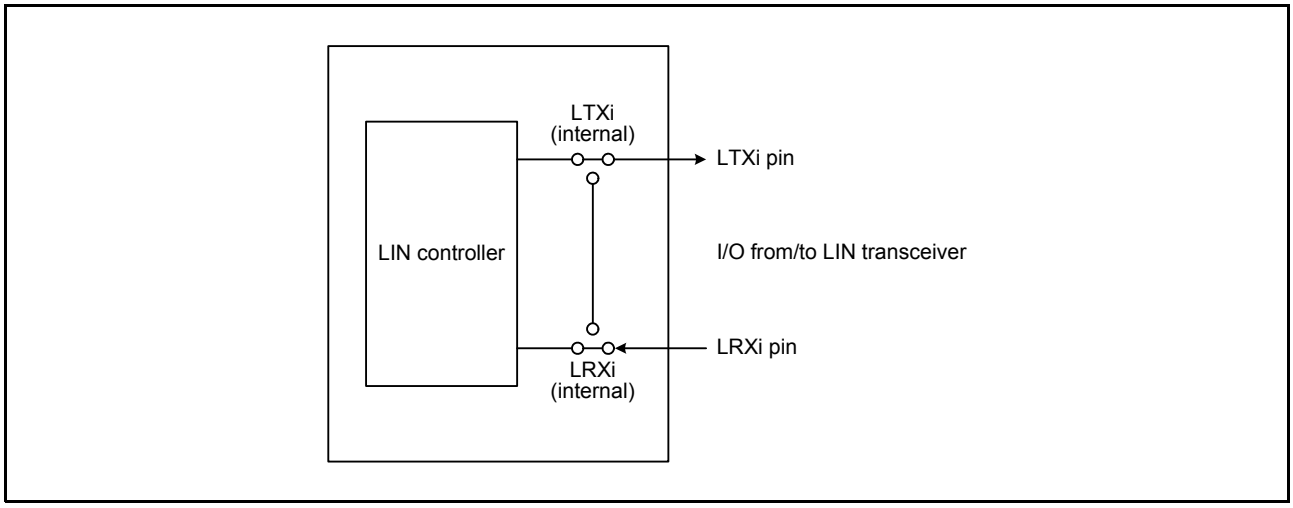


Figure 19.16 LIN Operation Mode Connection

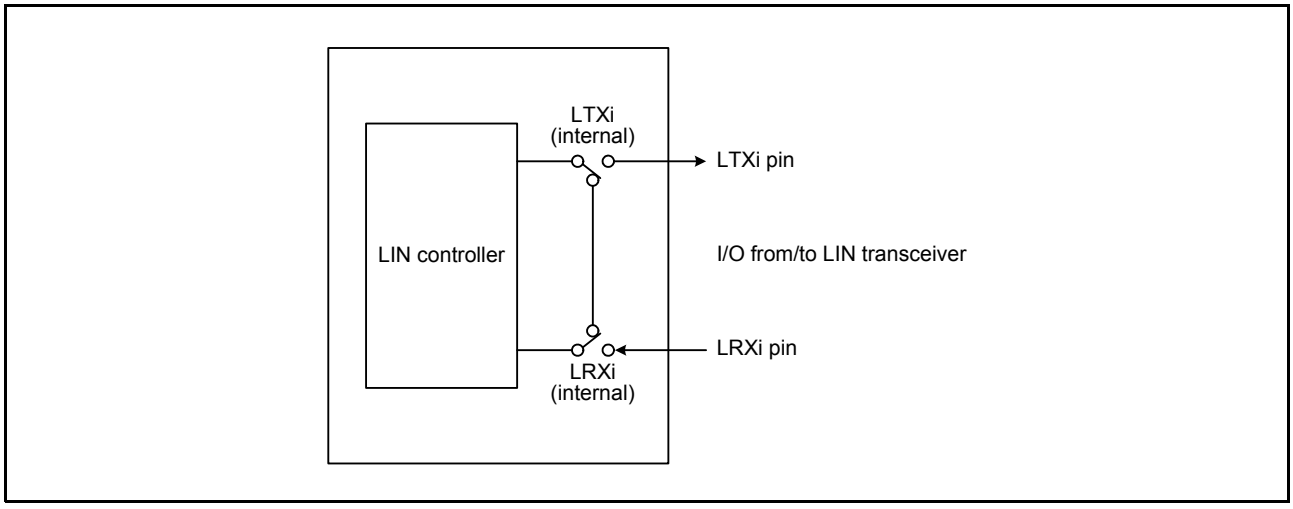


Figure 19.17 LIN Self-test Mode Connection

19.11.1 Entry into LIN Self-test Mode

To enter LIN self-test mode, a particular key sequence must be used. In this key sequence, the user has to do three consecutive writes to the LIN self-test control register as follows.

- Enter LIN reset mode.
- First write: LSTC register = 1010 0111 (H'A7)
- Second write: LSTC register = 0101 1000 (H'58)
- Third write: LSTC register = 0000 0001 (H'01)

If the first key are written twice, then the sequence is broken and must be restarted. If this sequence is broken by a write access to any other LIN registers, then it must be restarted.

LIN self-test mode does not support frame separate mode. Two tests can be carried out.

- LIN self-test mode (transmission): Header transmission and response transmission
- LIN self-test mode (reception): Header transmission and response reception

19.11.2 Transmission in LIN Self-test Mode

The following steps are to be performed to execute this test.

- Write B'11 to bits OM1 to OM0 in the LiC register, and then confirm bits OMM1 to OMM0 in the LiMST register are set to B'11.
- Set the RFC bit in the LiRFC register to 1 (transmission).
- Set the frame configuration for transmission.
- Set the FTS bit in the LiTC register to 1 (frame transmission/wake-up transmission and reception started).
- LIN self-test mode (transmission) is executed. Interrupt generation and update of status and error status is appropriately executed. Check-sum is automatically calculated by the LIN.
- If transmission is completed, the inverted value of the frame data after looped back is stored in registers LiIDB, LiCB, and LiDBn (stored as inverted value to compare the transmitted value and the looped back value).
- If transmission is not completed due to an error, the corresponding error flag is set.

19.11.3 Reception in LIN Self-test Mode

The following steps are to be performed to execute this test.

- Enter LIN self-test mode.
- Set the RFC bit in the LiRFC register to 0 (reception).
- Set the frame configuration for reception. Since check sum is not automatically calculated, store the calculated value. A check sum error can be tested by setting an erroneous calculation result in the check sum.
- Set the FTS bit in the LiTC register to 1 (frame transmission/wake-up transmission and reception started).
- LIN self-test mode (reception) is executed. Interrupt generation and update of status and error status is appropriately executed.
- If reception is completed, the inverted value of the frame data after looped back is stored in registers LiIDB, LiCB, and LiDBn (stored as inverted value to compare the set value and the received value after looped back).
- If reception is not completed due to an error, the corresponding error flag is set.

19.11.4 Exit from LIN Self-test Mode

To exit LIN self-test mode, perform the following steps.

- Enter LIN reset mode.
  (When bits OMM1 to OMM0 in the LiMST register are set to B'11, write B'11 to bits OM1 to OM0 in the LiC register, and then confirm bits OMM1 to OMM0 are set to B'11 before entering LIN reset mode.)

20. CAN Module

20.1 Overview

This MCU implements two channels (referred to as CAN0 and CAN1) of CAN (Controller Area Network) module that complies with the ISO11898-1 Specifications*. In this section, explanations for six channels of CAN module are given.

The CAN module transmits and receives both formats of messages, namely the standard identifier (11 bits) (identifier hereafter referred to as ID) and extended ID (29 bits).

Table 20.1 and Table 20.2 list the CAN module specifications and Figure 20.1 shows the CAN module block diagram. Connect the CAN bus transceiver externally.

Note: * Products of the SH72A2 and SH72A0 Groups do not inplement channels other than CAN0 and CAN1. Accordingly, functions for channels that are not implemented are not available.
Some products of the SH72A0 Group implement only a channel (CAN0) of CAN module. For details, see section 1, Overview.

Table 20.1 CAN Module Specifications (1)

Item	Overview
Protocol	• ISO11898-1 compliant
Bit rate	• Up to 1 Mbps
Message box	• 64 mailboxes: Two selectable mailbox modes Normal mailbox mode: Of the 64 mailboxes, 32 can be configured for either transmission or reception (and the other 32 are reception-only). FIFO mailbox mode: 24 mailboxes configurable as transmission or reception (and the other 32 are reception-only). 4 stages FIFO for transmission and 4 stages FIFO for reception
Reception	• Data frame and remote frame can be received. • Selectable receiving ID format (only standard ID, only extended ID or both ID) • Programmable one-shot reception function (enabled or disabled) • Selectable overwrite mode (message overwritten) or overrun mode (message discarded) • The reception complete interrupt can be enabled or disabled for each mailbox.
Acceptance Filter	• 8 acceptance masks (one mask every 4 mailboxes) • 2 acceptance masks (one mask every 16 mailboxes) • The mask can be enabled or disabled for each mailbox.
Transmission	• Data frame and remote frame can be transmitted. • Selectable transmitting ID format (only standard ID, only extended ID or both ID) • Programmable one-shot transmission function (enabled or disabled) • Selectable ID priority transmit mode or mailbox number priority transmit mode • Transmission request can be aborted (The completion of abort can be confirmed with a flag.) • The transmission complete interrupt can be enabled or disabled for each mailbox.
Mode transition for bus-off recovery	• Mode transition for the recovery from the bus-off state can be selected: ISO11898-1 compliant Automatic entry to CAN halt mode at bus-off entry Automatic entry to CAN halt mode at bus-off end Entry to CAN halt mode by a program Transition into error-active state by a program
Error status monitoring	• CAN bus errors (stuff error, form error, ACK error, CRC error, bit error, and ACK delimiter error) can be monitored. • Transition to error states can be detected (error-warning, error-passive, bus-off entry, and bus-off recovery). • The error counters can be read.
Time stamp function	• Time stamp function using a 16-bit counter • The reference clock can be selected from either 1-, 2-, 4- or 8-bit time periods.

Table 20.2 CAN Module Specifications (2)

Item	Overview
Interrupt sources	• 6 types: Reception complete Transmission complete Receive FIFO Transmit FIFO Error Wake-up
CAN sleep mode	• Current consumption can be reduced by stopping the CAN clock.
Software support unit	• 3 software support units: Acceptance filter support Mailbox search support (receive mailbox search, transmit mailbox search and message lost search) Channel search support
CAN clock source (fCAN)	• Peripheral bus clock B
Test mode	• 3 test modes available for user evaluation: Listen-only mode Self-test mode 0 (external loop back) Self-test mode 1 (internal loop back)

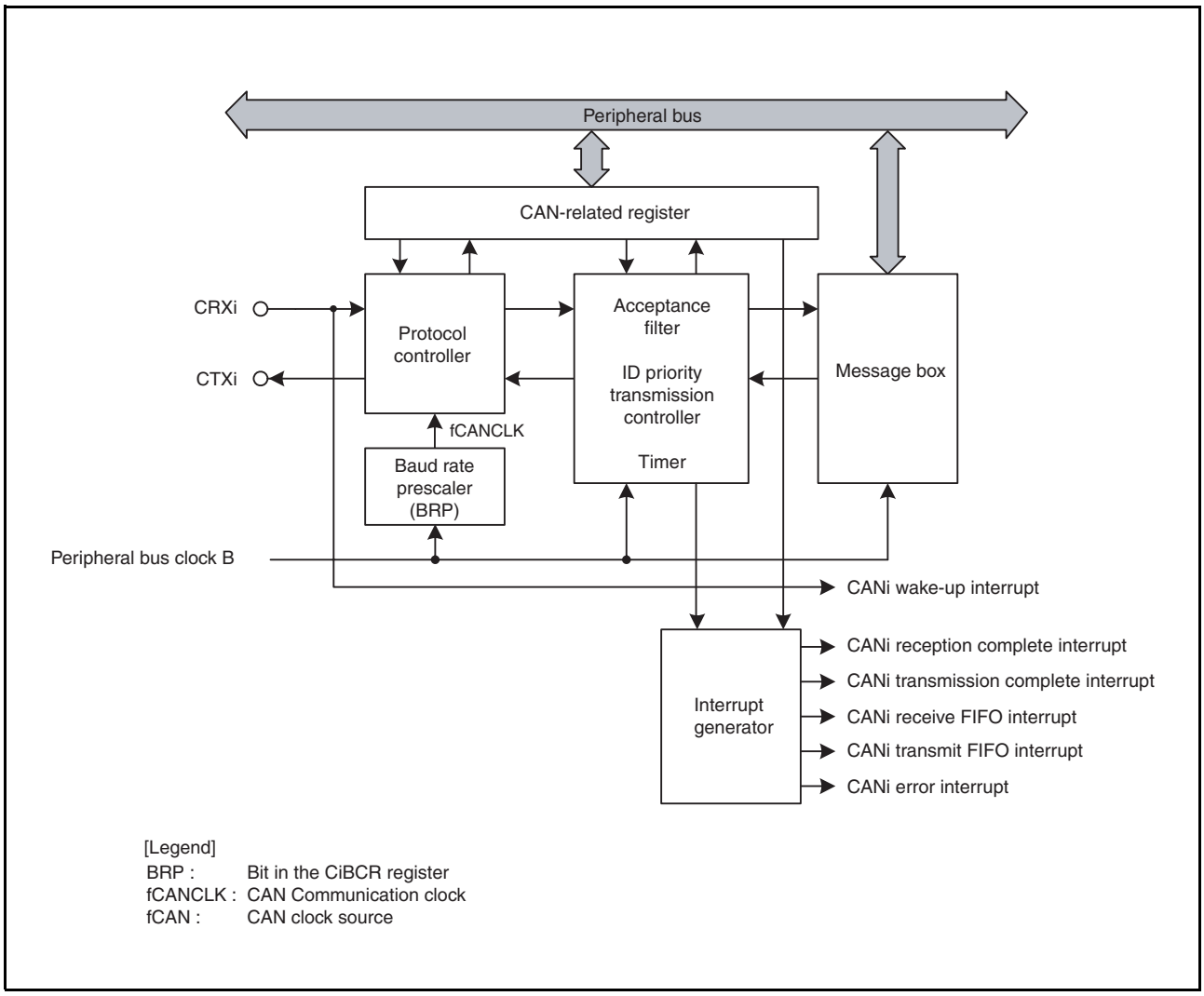


Figure 20.1 Block Diagram of CAN Module (i = 0 to 5)

- CRXi/CTXi (i = 0 to 5):
  CAN input/output pins
- Protocol controller:
  Handles CAN protocol processing such as bus arbitration, bit timing at transmission and reception, stuffing, and error handling, etc.
- Message box:
  Consists of 64 mailboxes which can be configured as either transmit or receive mailboxes. Each mailbox has an individual ID, data length code, a data field (8 bytes), and a time stamp.
- Acceptance filter:
  Performs filtering of received messages. Registers CiMKR0 to CiMKR9 are used for the filtering process.
- Timer:
  Used for the time stamp function. The timer value when storing a message into the mailbox is written as the time stamp value.
- Wake-up
  When a message is detected on the CAN bus, a CANi wake-up interruput request is generated.
- Interrupt generator:
  Generates the following five types of interrupts:
  CANi reception complete interrupt
  CANi transmission complete interrupt
  CANi receive FIFO interrupt
  CANi transmit FIFO interrupt
  CANi error interrupt

20.2 Input/Output Pins

Table 20.3 shows the I/O pins of the CAN module.

Table 20.3 I/O Pins of the CAN Module

Pin Name	I/O	Function
CRX0 to CRX5	Input	Pins for receiving data of the CAN communication function
CTX0 to CTX5	Output	Pins for transmitting data of the CAN communication function

20.3 Register Descriptions

Table 20.4 to Table 20.9 list the registers of the CAN module.

Table 20.4 List of CAN Module Registers (1)

Register Name	Abbreviation	After Reset	Address	Access Size
CAN0 Control Register	C0CTLR	H'0500	H'FF60 0540	8, 16, 32
CAN0 Bit Configuration Register	C0BCR	H'00 0000	H'FF60 0544	8, 16, 32
CAN0 Mask Register 0	C0MKR0	Undefined	H'FF60 0430	8, 16, 32
CAN0 Mask Register 1	C0MKR1	Undefined	H'FF60 0434	8, 16, 32
CAN0 Mask Register 2	C0MKR2	Undefined	H'FF60 0400	8, 16, 32
CAN0 Mask Register 3	C0MKR3	Undefined	H'FF60 0404	8, 16, 32
CAN0 Mask Register 4	C0MKR4	Undefined	H'FF60 0408	8, 16, 32
CAN0 Mask Register 5	C0MKR5	Undefined	H'FF60 040C	8, 16, 32
CAN0 Mask Register 6	C0MKR6	Undefined	H'FF60 0410	8, 16, 32
CAN0 Mask Register 7	C0MKR7	Undefined	H'FF60 0414	8, 16, 32
CAN0 Mask Register 8	C0MKR8	Undefined	H'FF60 0418	8, 16, 32
CAN0 Mask Register 9	C0MKR9	Undefined	H'FF60 041C	8, 16, 32
CAN0 FIFO Received ID Compare Register 0	C0FIDCR0	Undefined	H'FF60 0420	8, 16, 32
CAN0 FIFO Received ID Compare Register 1	C0FIDCR1	Undefined	H'FF60 0424	8, 16, 32
CAN0 Mask Invalid Register 0	C0MKIVLR0	Undefined	H'FF60 0438	8, 16, 32
CAN0 Mask Invalid Register 1	C0MKIVLR1	Undefined	H'FF60 0428	8, 16, 32
CAN0 Mailbox Registers 0 to 63	C0MB0 to C0MB63	Undefined	H'FF60 0000 to H'FF60 03F0	8, 16, 32
CAN0 Mailbox Interrupt Enable Register 0	C0MIER0	Undefined	H'FF60 043C	8, 16, 32
CAN0 Mailbox Interrupt Enable Register 1	C0MIER1	Undefined	H'FF60 042C	8, 16, 32
CAN0 Message Control Registers 0 to 63	C0MCTL0 to C0MCTL63	H'00	H'FF60 0500 to H'FF60 053F	8, 16, 32
CAN0 Receive FIFO Control Register	C0RFCR	H'80	H'FF60 0548	8, 16, 32
CAN0 Receive FIFO Pointer Control Register	C0RFPCR	—	H'FF60 0549	8, 16, 32
CAN0 Transmit FIFO Control Register	C0TFCR	H'80	H'FF60 054A	8, 16, 32
CAN0 Transmit FIFO Pointer Control Register	C0TFPCR	—	H'FF60 054B	8, 16, 32
CAN0 Status Register	C0STR	H'0500	H'FF60 0542	8, 16, 32
CAN0 Mailbox Search Mode Register	C0MSMR	H'00	H'FF60 0553	8, 16, 32
CAN0 Mailbox Search Status Register	C0MSSR	H'80	H'FF60 0552	8, 16, 32
CAN0 Channel Search Support Register	C0CSSR	—	H'FF60 0551	8, 16, 32
CAN0 Acceptance Filter Support Register	C0AFSR	Undefined	H'FF60 0556	8, 16, 32
CAN0 Error Interrupt Enable Register	C0EIER	H'00	H'FF60 054C	8, 16, 32
CAN0 Error Interrupt Factor Judge Register	C0EIFR	H'00	H'FF60 054D	8, 16, 32
CAN0 Receive Error Count Register	C0RECR	H'00	H'FF60 054E	8, 16, 32
CAN0 Transmit Error Count Register	C0TECR	H'00	H'FF60 054F	8, 16, 32
CAN0 Error Code Store Register	C0ECSR	H'00	H'FF60 0550	8, 16, 32
CAN0 Time Stamp Register	C0TSR	H'0000	H'FF60 0554	8, 16, 32
CAN0 Test Control Register	C0TCR	H'00	H'FF60 0558	8
CAN1 Control Register	C1CTLR	H'0500	H'FF60 0D40	8, 16, 32

Table 20.5 List of CAN Module Registers (2)

Register Name	Abbreviation	After Reset	Address	Access Size
CAN1 Bit Configuration Register	C1BCR	H'00 0000	H'FF60 0D44	8, 16, 32
CAN1 Mask Register 0	C1MKR0	Undefined	H'FF60 0C30	8, 16, 32
CAN1 Mask Register 1	C1MKR1	Undefined	H'FF60 0C34	8, 16, 32
CAN1 Mask Register 2	C1MKR2	Undefined	H'FF60 0C00	8, 16, 32
CAN1 Mask Register 3	C1MKR3	Undefined	H'FF60 0C04	8, 16, 32
CAN1 Mask Register 4	C1MKR4	Undefined	H'FF60 0C08	8, 16, 32
CAN1 Mask Register 5	C1MKR5	Undefined	H'FF60 0C0C	8, 16, 32
CAN1 Mask Register 6	C1MKR6	Undefined	H'FF60 0C10	8, 16, 32
CAN1 Mask Register 7	C1MKR7	Undefined	H'FF60 0C14	8, 16, 32
CAN1 Mask Register 8	C1MKR8	Undefined	H'FF60 0C18	8, 16, 32
CAN1 Mask Register 9	C1MKR9	Undefined	H'FF60 0C1C	8, 16, 32
CAN1 FIFO Received ID Compare Register 0	C1FIDCR0	Undefined	H'FF60 0C20	8, 16, 32
CAN1 FIFO Received ID Compare Register 1	C1FIDCR1	Undefined	H'FF60 0C24	8, 16, 32
CAN1 Mask Invalid Register 0	C1MKIVLR0	Undefined	H'FF60 0C38	8, 16, 32
CAN1 Mask Invalid Register 1	C1MKIVLR1	Undefined	H'FF60 0C28	8, 16, 32
CAN1 Mailbox Registers 0 to 63	C1MB0 to C1MB63	Undefined	H'FF60 0800 to H'FF60 0BF0	8, 16, 32
CAN1 Mailbox Interrupt Enable Register 0	C1MIER0	Undefined	H'FF60 0C3C	8, 16, 32
CAN1 Mailbox Interrupt Enable Register 1	C1MIER1	Undefined	H'FF60 0C2C	8, 16, 32
CAN1 Message Control Registers 0 to 63	C1MCTL0 to C1MCTL63	H'00	H'FF60 0D00 to H'FF60 0D3F	8, 16, 32
CAN1 Receive FIFO Control Register	C1RFCR	H'80	H'FF60 0D48	8, 16, 32
CAN1 Receive FIFO Pointer Control Register	C1RFPCR	—	H'FF60 0D49	8, 16, 32
CAN1 Transmit FIFO Control Register	C1TFCR	H'80	H'FF60 0D4A	8, 16, 32
CAN1 Transmit FIFO Pointer Control Register	C1TFPCR	—	H'FF60 0D4B	8, 16, 32
CAN1 Status Register	C1STR	H'0500	H'F60 0D42	8, 16, 32
CAN1 Mailbox Search Mode Register	C1MSMR	H'00	H'FF60 0D53	8, 16, 32
CAN1 Mailbox Search Status Register	C1MSSR	H'80	H'FF60 0D52	8, 16, 32
CAN1 Channel Search Support Register	C1CSSR	—	H'FF60 0D51	8, 16, 32
CAN1 Acceptance Filter Support Register	C1AFSR	Undefined	H'FF60 0D56	8, 16, 32
CAN1 Error Interrupt Enable Register	C1EIER	H'00	H'FF60 0D4C	8, 16, 32
CAN1 Error Interrupt Factor Judge Register	C1EIFR	H'00	H'FF60 0D4D	8, 16, 32
CAN1 Receive Error Count Register	C1RECR	H'00	H'FF60 0D4E	8, 16, 32
CAN1 Transmit Error Count Register	C1TECR	H'00	H'FF60 0D4F	8, 16, 32
CAN1 Error Code Store Register	C1ECSR	H'00	H'FF60 0D50	8, 16, 32
CAN1 Time Stamp Register	C1TSR	H'0000	H'FF60 0D54	8, 16, 32
CAN1 Test Control Register	C1TCR	H'00	H'FF60 0D58	8
CAN2 Control Register	C2CTLR	H'0500	H'FF60 1540	8, 16, 32
CAN2 Bit Configuration Register	C2BCR	H'00 0000	H'FF60 1544	8, 16, 32
CAN2 Mask Register 0	C2MKR0	Undefined	H'FF60 1430	8, 16, 32
CAN2 Mask Register 1	C2MKR1	Undefined	H'FF60 1434	8, 16, 32
CAN2 Mask Register 2	C2MKR2	Undefined	H'FF60 1400	8, 16, 32

Table 20.6 List of CAN Module Registers (3)

Register Name	Abbreviation	After Reset	Address	Access Size
CAN2 Mask Register 3	C2MKR3	Undefined	H'FF60 1404	8, 16, 32
CAN2 Mask Register 4	C2MKR4	Undefined	H'FF60 1408	8, 16, 32
CAN2 Mask Register 5	C2MKR5	Undefined	H'FF60 140C	8, 16, 32
CAN2 Mask Register 6	C2MKR6	Undefined	H'FF60 1410	8, 16, 32
CAN2 Mask Register 7	C2MKR7	Undefined	H'FF60 1414	8, 16, 32
CAN2 Mask Register 8	C2MKR8	Undefined	H'FF60 1418	8, 16, 32
CAN2 Mask Register 9	C2MKR9	Undefined	H'FF60 141C	8, 16, 32
CAN2 FIFO Received ID Compare Register 0	C2FIDCR0	Undefined	H'FF60 1420	8, 16, 32
CAN2 FIFO Received ID Compare Register 1	C2FIDCR1	Undefined	H'FF60 1424	8, 16, 32
CAN2 Mask Invalid Register 0	C2MKIVLR0	Undefined	H'FF60 1438	8, 16, 32
CAN2 Mask Invalid Register 1	C2MKIVLR1	Undefined	H'FF60 1428	8, 16, 32
CAN2 Mailbox Registers 0 to 63	C2MB0 to C2MB63	Undefined	H'FF60 1000 to H'FF60 13F0	8, 16, 32
CAN2 Mailbox Interrupt Enable Register 0	C2MIER0	Undefined	H'FF60 143C	8, 16, 32
CAN2 Mailbox Interrupt Enable Register 1	C2MIER1	Undefined	H'FF60 142C	8, 16, 32
CAN2 Message Control Registers 0 to 63	C2MCTL0 to C2MCTL63	H'00	H'FF60 1500 to H'FF60 153F	8, 16, 32
CAN2 Receive FIFO Control Register	C2RFCR	H'80	H'FF60 1548	8, 16, 32
CAN2 Receive FIFO Pointer Control Register	C2RFPCR	—	H'FF60 1549	8, 16, 32
CAN2 Transmit FIFO Control Register	C2TFCR	H'80	H'FF60 154A	8, 16, 32
CAN2 Transmit FIFO Pointer Control Register	C2TFPCR	—	H'FF60 154B	8, 16, 32
CAN2 Status Register	C2STR	H'0500	H'FF60 1542	8, 16, 32
CAN2 Mailbox Search Mode Register	C2MSMR	H'00	H'FF60 1553	8, 16, 32
CAN2 Mailbox Search Status Register	C2MSSR	H'80	H'FF60 1552	8, 16, 32
CAN2 Channel Search Support Register	C2CSSR	—	H'FF60 1551	8, 16, 32
CAN2 Acceptance Filter Support Register	C2AFSR	Undefined	H'FF60 1556	8, 16, 32
CAN2 Error Interrupt Enable Register	C2EIER	H'00	H'FF60 154C	8, 16, 32
CAN2 Error Interrupt Factor Judge Register	C2EIFR	H'00	H'FF60 154D	8, 16, 32
CAN2 Receive Error Count Register	C2RECR	H'00	H'FF60 154E	8, 16, 32
CAN2 Transmit Error Count Register	C2TECR	H'00	H'FF60 154F	8, 16, 32
CAN2 Error Code Store Register	C2ECSR	H'00	H'FF60 1550	8, 16, 32
CAN2 Time Stamp Register	C2TSR	H'0000	H'FF60 1554	8, 16, 32
CAN2 Test Control Register	C2TCR	H'00	H'FF60 1558	8
CAN3 Control Register	C3CTLR	H'0500	H'FF60 1D40	8, 16, 32
CAN3 Bit Configuration Register	C3BCR	H'00 0000	H'FF60 1D44	8, 16, 32
CAN3 Mask Register 0	C3MKR0	Undefined	H'FF60 1C30	8, 16, 32
CAN3 Mask Register 1	C3MKR1	Undefined	H'FF60 1C34	8, 16, 32
CAN3 Mask Register 2	C3MKR2	Undefined	H'FF60 1C00	8, 16, 32
CAN3 Mask Register 3	C3MKR3	Undefined	H'FF60 1C04	8, 16, 32
CAN3 Mask Register 4	C3MKR4	Undefined	H'FF60 1C08	8, 16, 32
CAN3 Mask Register 5	C3MKR5	Undefined	H'FF60 1C0C	8, 16, 32
CAN3 Mask Register 6	C3MKR6	Undefined	H'FF60 1C10	8, 16, 32
CAN3 Mask Register 7	C3MKR7	Undefined	H'FF60 1C14	8, 16, 32

Table 20.7 List of CAN Module Registers (4)

Register Name	Abbreviation	After Reset	Address	Access Size
CAN3 Mask Register 8	C3MKR8	Undefined	H'FF60 1C18	8, 16, 32
CAN3 Mask Register 9	C3MKR9	Undefined	H'FF60 1C1C	8, 16, 32
CAN3 FIFO Received ID Compare Register 0	C3FIDCR0	Undefined	H'FF60 1C20	8, 16, 32
CAN3 FIFO Received ID Compare Register 1	C3FIDCR1	Undefined	H'FF60 1C24	8, 16, 32
CAN3 Mask Invalid Register 0	C3MKIVLR0	Undefined	H'FF60 1C38	8, 16, 32
CAN3 Mask Invalid Register 1	C3MKIVLR1	Undefined	H'FF60 1C28	8, 16, 32
CAN3 Mailbox Registers 0 to 63	C3MB0 to C3MB63	Undefined	H'FF60 1800 to H'FF60 1BF0	8, 16, 32
CAN3 Mailbox Interrupt Enable Register 0	C3MIER0	Undefined	H'FF60 1C3C	8, 16, 32
CAN3 Mailbox Interrupt Enable Register 1	C3MIER1	Undefined	H'FF60 1C2C	8, 16, 32
CAN3 Message Control Registers 0 to 63	C3MCTL0 to C3MCTL63	H'00	H'FF60 1D00 to H'FF60 1D3F	8, 16, 32
CAN3 Receive FIFO Control Register	C3RFCR	H'80	H'FF60 1D48	8, 16, 32
CAN3 Receive FIFO Pointer Control Register	C3RFPCR	—	H'FF60 1D49	8, 16, 32
CAN3 Transmit FIFO Control Register	C3TFCR	H'80	H'FF60 1D4A	8, 16, 32
CAN3 Transmit FIFO Pointer Control Register	C3TFPCR	—	H'FF60 1D4B	8, 16, 32
CAN3 Status Register	C3STR	H'0500	H'FF60 1D42	8, 16, 32
CAN3 Mailbox Search Mode Register	C3MSMR	H'00	H'FF60 1D53	8, 16, 32
CAN3 Mailbox Search Status Register	C3MSSR	H'80	H'FF60 1D52	8, 16, 32
CAN3 Channel Search Support Register	C3CSSR	—	H'FF60 1D51	8, 16, 32
CAN3 Acceptance Filter Support Register	C3AFSR	Undefined	H'FF60 1D56	8, 16, 32
CAN3 Error Interrupt Enable Register	C3EIER	H'00	H'FF60 1D4C	8, 16, 32
CAN3 Error Interrupt Factor Judge Register	C3EIFR	H'00	H'FF60 1D4D	8, 16, 32
CAN3 Receive Error Count Register	C3RECR	H'00	H'FF60 1D4E	8, 16, 32
CAN3 Transmit Error Count Register	C3TECR	H'00	H'FF60 1D4F	8, 16, 32
CAN3 Error Code Store Register	C3ECSR	H'00	H'FF60 1D50	8, 16, 32
CAN3 Time Stamp Register	C3TSR	H'0000	H'FF60 1D54	8, 16, 32
CAN3 Test Control Register	C3TCR	H'00	H'FF60 1D58	8
CAN4 Control Register	C4CTLR	H'0500	H'FF60 2540	8, 16, 32
CAN4 Bit Configuration Register	C4BCR	H'00 0000	H'FF60 2544	8, 16, 32
CAN4 Mask Register 0	C4MKR0	Undefined	H'FF60 2430	8, 16, 32
CAN4 Mask Register 1	C4MKR1	Undefined	H'FF60 2434	8, 16, 32
CAN4 Mask Register 2	C4MKR2	Undefined	H'FF60 2400	8, 16, 32
CAN4 Mask Register 3	C4MKR3	Undefined	H'FF60 2404	8, 16, 32
CAN4 Mask Register 4	C4MKR4	Undefined	H'FF60 2408	8, 16, 32
CAN4 Mask Register 5	C4MKR5	Undefined	H'FF60 240C	8, 16, 32
CAN4 Mask Register 6	C4MKR6	Undefined	H'FF60 2410	8, 16, 32
CAN4 Mask Register 7	C4MKR7	Undefined	H'FF60 2414	8, 16, 32
CAN4 Mask Register 8	C4MKR8	Undefined	H'FF60 2418	8, 16, 32
CAN4 Mask Register 9	C4MKR9	Undefined	H'FF60 241C	8, 16, 32
CAN4 FIFO Received ID Compare Register 0	C4FIDCR0	Undefined	H'FF60 2420	8, 16, 32
CAN4 FIFO Received ID Compare Register 1	C4FIDCR1	Undefined	H'FF60 2424	8, 16, 32
CAN4 Mask Invalid Register 0	C4MKIVLR0	Undefined	H'FF60 2438	8, 16, 32

Table 20.8 List of CAN Module Registers (5)

Register Name	Abbreviation	After Reset	Address	Access Size
CAN4 Mask Invalid Register 1	C4MKIVLR1	Undefined	H'FF60 2428	8, 16, 32
CAN4 Mailbox Registers 0 to 63	C4MB0 to C4MB63	Undefined	H'FF60 2000 to H'FF60 23F0	8, 16, 32
CAN4 Mailbox Interrupt Enable Register 0	C4MIER0	Undefined	H'FF60 243C	8, 16, 32
CAN4 Mailbox Interrupt Enable Register 1	C4MIER1	Undefined	H'FF60 242C	8, 16, 32
CAN4 Message Control Registers 0 to 63	C4MCTL0 to C4MCTL63	H'00	H'FF60 2500 to H'FF60 253F	8, 16, 32
CAN4 Receive FIFO Control Register	C4RFCR	H'80	H'FF60 2548	8, 16, 32
CAN4 Receive FIFO Pointer Control Register	C4RFPCR	—	H'FF60 2549	8, 16, 32
CAN4 Transmit FIFO Control Register	C4TFCR	H'80	H'FF60 254A	8, 16, 32
CAN4 Transmit FIFO Pointer Control Register	C4TFPCR	—	H'FF60 254B	8, 16, 32
CAN4 Status Register	C4STR	H'0500	H'FF60 2542	8, 16, 32
CAN4 Mailbox Search Mode Register	C4MSMR	H'00	H'FF60 2553	8, 16, 32
CAN4 Mailbox Search Status Register	C4MSSR	H'80	H'FF60 2552	8, 16, 32
CAN4 Channel Search Support Register	C4CSSR	—	H'FF60 2551	8, 16, 32
CAN4 Acceptance Filter Support Register	C4AFSR	Undefined	H'FF60 2556	8, 16, 32
CAN4 Error Interrupt Enable Register	C4EIER	H'00	H'FF60 254C	8, 16, 32
CAN4 Error Interrupt Factor Judge Register	C4EIFR	H'00	H'FF60 254D	8, 16, 32
CAN4 Receive Error Count Register	C4RECR	H'00	H'FF60 254E	8, 16, 32
CAN4 Transmit Error Count Register	C4TECR	H'00	H'FF60 254F	8, 16, 32
CAN4 Error Code Store Register	C4ECSR	H'00	H'FF60 2550	8, 16, 32
CAN4 Time Stamp Register	C4TSR	H'0000	H'FF60 2554	8, 16, 32
CAN4 Test Control Register	C4TCR	H'00	H'FF60 2558	8
CAN5 Control Register	C5CTLR	H'0500	H'FF60 2D40	8, 16, 32
CAN5 Bit Configuration Register	C5BCR	H'00 0000	H'FF60 2D44	8, 16, 32
CAN5 Mask Register 0	C5MKR0	Undefined	H'FF60 2C30	8, 16, 32
CAN5 Mask Register 1	C5MKR1	Undefined	H'FF60 2C34	8, 16, 32
CAN5 Mask Register 2	C5MKR2	Undefined	H'FF60 2C00	8, 16, 32
CAN5 Mask Register 3	C5MKR3	Undefined	H'FF60 2C04	8, 16, 32
CAN5 Mask Register 4	C5MKR4	Undefined	H'FF60 2C08	8, 16, 32
CAN5 Mask Register 5	C5MKR5	Undefined	H'FF60 2C0C	8, 16, 32
CAN5 Mask Register 6	C5MKR6	Undefined	H'FF60 2C10	8, 16, 32
CAN5 Mask Register 7	C5MKR7	Undefined	H'FF60 2C14	8, 16, 32
CAN5 Mask Register 8	C5MKR8	Undefined	H'FF60 2C18	8, 16, 32
CAN5 Mask Register 9	C5MKR9	Undefined	H'FF60 2C1C	8, 16, 32
CAN5 FIFO Received ID Compare Register 0	C5FIDCR0	Undefined	H'FF60 2C20	8, 16, 32
CAN5 FIFO Received ID Compare Register 1	C5FIDCR1	Undefined	H'FF60 2C24	8, 16, 32
CAN5 Mask Invalid Register 0	C5MKIVLR0	Undefined	H'FF60 2C38	8, 16, 32
CAN5 Mask Invalid Register 1	C5MKIVLR1	Undefined	H'FF60 2C28	8, 16, 32
CAN5 Mailbox Registers 0 to 63	C5MB0 to C5MB63	Undefined	H'FF60 2800 to H'FF60 2BF0	8, 16, 32
CAN5 Mailbox Interrupt Enable Register 0	C5MIER0	Undefined	H'FF60 2C3C	8, 16, 32
CAN5 Mailbox Interrupt Enable Register 1	C5MIER1	Undefined	H'FF60 2C2C	8, 16, 32

Table 20.9 List of CAN Module Registers (6)

Register Name	Abbreviation	After Reset	Address	Access Size
CAN5 Message Control Registers 0 to 63	C5MCTL0 to C5MCTL63	H'00	H'FF60 2D00 to H'FF60 2D3F	8, 16, 32
CAN5 Receive FIFO Control Register	C5RFCR	H'80	H'FF60 2D48	8, 16, 32
CAN5 Receive FIFO Pointer Control Register	C5RFPCR	—	H'FF60 2D49	8, 16, 32
CAN5 Transmit FIFO Control Register	C5TFCR	H'80	H'FF60 2D4A	8, 16, 32
CAN5 Transmit FIFO Pointer Control Register	C5TFPCR	—	H'FF60 2D4B	8, 16, 32
CAN5 Status Register	C5STR	H'0500	H'FF60 2D42	8, 16, 32
CAN5 Mailbox Search Mode Register	C5MSMR	H'00	H'FF60 2D53	8, 16, 32
CAN5 Mailbox Search Status Register	C5MSSR	H'80	H'FF60 2D52	8, 16, 32
CAN5 Channel Search Support Register	C5CSSR	—	H'FF60 2D51	8, 16, 32
CAN5 Acceptance Filter Support Register	C5AFSR	Undefined	H'FF60 2D56	8, 16, 32
CAN5 Error Interrupt Enable Register	C5EIER	H'00	H'FF60 2D4C	8, 16, 32
CAN5 Error Interrupt Factor Judge Register	C5EIFR	H'00	H'FF60 2D4D	8, 16, 32
CAN5 Receive Error Count Register	C5RECR	H'00	H'FF60 2D4E	8, 16, 32
CAN5 Transmit Error Count Register	C5TECR	H'00	H'FF60 2D4F	8, 16, 32
CAN5 Error Code Store Register	C5ECSR	H'00	H'FF60 2D50	8, 16, 32
CAN5 Time Stamp Register	C5TSR	H'0000	H'FF60 2D54	8, 16, 32
CAN5 Test Control Register	C5TCR	H'00	H'FF60 2D58	8

20.3.1 CANi Control Register (CiCTLR) (i = 0 to 5)

Address C0CTLR: H'FF60 0540, C1CTLR: H'FF60 0D40, C2CTLR: H'FF60 1540, C3CTLR: H'FF60 1D40, C4CTLR: H'FF60 2540, C5CTLR: H'FF60 2D40

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	RBOC	BOM[1:0]		SLPM	CANM[1:0]		TSPS[1:0]		TSRC	TPM	MLM	IDFM[1:0]		MBM
After Reset	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b14	—	Reserved	These bits are read as 0. The write value should be 0.	R
b13	RBOC	Forcible Return from Bus-oFF Bit*1	0: Nothing occurred 1: Forcible return from bus-off*2	R/W
b12, b11	BOM[1:0]	Bus-off Recovery Mode Select Bit*3	b12b11 0 0: Normal mode (ISO11898-1 compliant) 0 1: Entry to CAN halt mode automatically at bus-off entry 1 0: Entry to CAN halt mode automatically at bus-off end 1 1: Entry to CAN halt mode (during bus-off recovery period) by a program request	R/W
b10	SLPM	CAN Sleep Mode Bit*4*5	0: Other than CAN sleep mode 1: CAN sleep mode	R/W
b9, b8	CANM[1:0]	CAN Operating Mode Select Bit*4	b9 b8 0 0: CAN operation mode 0 1: CAN reset mode 1 0: CAN halt mode 1 1: CAN reset mode (forcible transition)	R/W
b7, b6	TSPS[1:0]	Time Stamp Prescaler Select Bit*3	b7 b6 0 0: Every bit time 0 1: Every 2-bit time 1 0: Every 4-bit time 1 1: Every 8-bit time	R/W
b5	TSRC	Time Stamp Counter Reset Command Bit*6	0: Nothing occurred 1: Reset*2	R/W
b4	TPM	Transmission Priority Mode Select Bit*3	0: ID priority transmit mode 1: Mailbox number priority transmit mode	R/W
b3	MLM	Message Lost Mode Select Bit*3	0: Overwrite mode 1: Overrun mode	R/W
b2, b1	IDFM[1:0]	ID Format Mode Select Bit*3	b2 b1 0 0: Standard ID mode 0 1: Extended ID mode 1 0: Mixed ID mode 1 1: Do not use this combination	R/W
b0	MBM	CAN Mailbox Mode Select Bit*3	0: Normal mailbox mode 1: FIFO mailbox mode	R/W

Notes: 1. Set the RBOC bit to 1 in bus-off state.
2. Bits RBOC and TSRC are automatically set back to 0 after being set to 1. It should be read as 0.
3. Write to bits BOM, MBM, IDFM, MLM, TPM, and TSPS in CAN reset mode.
4. When bits CANM and SLPM are changed, check the CiSTR register to ensure that the mode has been switched. Don't change bits CANM and SLPM until the mode has been switched.
5. Write to the SLPM bit in CAN reset mode or CAN halt mode. When rewriting the SLPM bit, set only this bit to 0 or 1.
6. Set the TSRC bit to 1 in CAN operation mode.

RBOC Bit

When the RBOC bit is set to 1 (forcible return from bus-off) in bus-off state, the CAN module forcibly returns from the bus-off state. This bit is automatically set to 0. The error state changes from bus-off to error-active.

When the RBOC bit is set to 1, registers CiRECR and CiTECR are set to H'00 and the BOST bit in the CiSTR register is set to 0 (the CAN module is not in bus-off state). The other registers remain unchanged. No bus-off recovery interrupt request is generated by this recovery from the bus-off state. Use the RBOC bit only when the BOM bits = 00 (normal mode).

BOM Bits

The BOM bits are used to select bus-off recovery mode.

When the BOM bits are 00, the recovery from bus-off is compliant with ISO11898-1, i.e. the CAN module re-enters CAN communication (error-active state) after detecting 11 consecutive recessive bits 128 times. A bus-off recovery interrupt request is generated when recovering from bus-off.

When the BOM bits are 01, as soon as the CAN reaches the bus-off state, the CANM bits in the CiCTLR register are set to 10 (CAN halt mode) and the CAN enters CAN halt mode. No bus-off recovery interrupt request is generated when recovering from bus-off and registers CiTECR and CiRECR are set to H'00.

When the BOM bits are 10, the CANM bits are set to 10 as soon as the CAN module reaches the bus-off state. The CAN module enters CAN halt mode after the recovery from the bus-off state, i.e. after detecting 11 consecutive recessive bits 128 times. A bus-off recovery interrupt request is generated when recovering from bus-off and registers CiTECR and CiRECR are set to H'00.

When the BOM bits are 11, the CAN module enters CAN halt mode by setting the CANM bits to 10 while the CAN module is still in bus-off state. No bus-off recovery interrupt request is generated when recovering from bus-off and registers CiTECR and CiRECR are set to H'00. However, if the CAN module recovers from bus-off after detecting 11 consecutive recessive bits 128 times before the CANM bits are set to 10, a bus-off recovery interrupt request is generated.

If the CPU requests an entry to CAN reset mode at the same time as the CAN module attempts to enter CAN halt mode (at bus-off entry when the BOM bits are 01, or at bus-off end when the BOM bits are 10), then the CPU request to enter CAN reset mode has higher priority.

SLPM Bit

When the SLPM bit is set to 1, the CAN module enters CAN sleep mode.

When the SLPM bit is set to 0, the CAN module exits CAN sleep mode.

Refer to section 20.4, Operating Mode for detail.

CANM Bits

The CANM bits select one of the following modes for the CAN module: CAN operation mode, CAN reset mode or CAN halt mode. Refer to section 20.4, Operating Mode for detail.

CAN sleep mode is set by the SLPM bit.

When the CAN module enters CAN halt mode according to the setting of the BOM bits, the CANM bits are automatically set to 10.

TSPS Bits

The TSPS bits select the prescaler for the time stamp.

The reference clock for the time stamp can be selected to be either 1-, 2-, 4- or 8-bit time periods.

TSRC Bit

The TSRC bit is used to reset the time stamp counter.

When the TSRC bit is set to 1, the CiTSR register is set to H'0000. It is automatically set to 0.

TPM Bit

The TPM bit specifies the priority of modes when transmitting messages. ID priority transmit mode or mailbox number transmit mode can be selected.

All mailboxes are set for either ID priority transmission or mailbox number priority transmission.

When the TPM bit is 0, ID priority transmit mode is selected and transmission priority complies with the CAN bus arbitration rule, as defined in the ISO 11898-1 Specifications. In ID priority transmit mode, mailboxes [0] to [63] (in normal mailbox mode), and mailboxes [0] to [55] (in FIFO mailbox mode), and the transmit FIFO are compared for

the IDs of mailboxes configured for transmission. If two or more mailbox IDs are the same, the mailbox with the smaller number has higher priority.
Only the next message to be transmitted from the transmit FIFO is included in the transmission arbitration. If a transmit FIFO message is being transmitted, the next pending message within the transmit FIFO is included in the transmission arbitration.
When the TPM bit is 1, mailbox number priority transmit mode is selected and the transmit mailbox with the smallest mailbox number has the highest priority. In FIFO mailbox mode, the transmit FIFO has lower priority than normal mailboxes (mailboxes [0] to [55]).

MLM Bit

The MLM bit specifies the operation when a new message is captured in the unread mailbox.
Overwrite mode or overrun mode can be selected. All mailboxes (including the receive FIFO) are set to either overwrite mode or overrun mode.
When the MLM bit is 0, all mailboxes are set to overwrite mode and the new message overwrites the old message.
When this bit is 1, all mailboxes are set to overrun mode and the new message is discarded.

IDFM Bits

The IDFM bits specify the ID format.

00: All mailboxes (including FIFO mailboxes) handle only standard IDs.
01: All mailboxes (including FIFO mailboxes) handle only extended IDs.
10: All mailboxes (including FIFO mailboxes) handle both standard IDs and extended IDs. Standard IDs or extended IDs are specified by using the IDE bit in the corresponding mailbox in normal mail box mode. In FIFO mailbox mode, the IDE bit in the corresponding mailbox is used for mailboxes [0] to [55], the IDE bit in registers CiFIDCR0 and CiFIDCR1 is used for the receive FIFO, and the IDE bit in mailbox [56] is used for the transmit FIFO.
11: Do not use this combination.

MBM Bit

When the MBM bit is 0 (normal mailbox mode), mailboxes [0] to [63] are configured as transmit or receive mailboxes.
When the MBM bit is 1 (FIFO mailbox mode), mailboxes [0] to [55] are configured as transmit or receive mailboxes. Mailboxes [56] to [59] are configured as a transmit FIFO and mailboxes [60] to [63] as a receive FIFO.
Transmit data is written into mailbox [56] (mailbox [56] is a window mailbox for the transmit FIFO).
Receive data is read from mailbox [60] (mailbox [60] is a window mailbox for the receive FIFO).

Table 20.10 lists the Mailbox Configuration.

Table 20.10 Mailbox Configuration

Mailbox	MBM Bit = 0 (Normal mailbox mode)	MBM Bit = 1* (FIFO mailbox mode)
Mailboxes [0] to [55]	Normal mailbox	Normal mailbox
Mailboxes [56] to [59]		Transmit FIFO
Mailboxes [60] to [63]		Receive FIFO

Note: * Points 1 to 5 below should be considered when the MBM bit is set to 1.
- Transmit FIFO is controlled by the CiTFCR register.
  The CiMCTLj register of mailboxes [56] to [59] are disabled.
  Registers CiMCTL56 to CiMCTL59 cannot be used.
- Receive FIFO is controlled by the CiRFCR register.
  The CiMCTLj register of mailboxes [60] to [63] are disabled.
  Registers CiMCTL60 to CiMCTL63 cannot be used.
- Refer to the CiMIER1 register about the FIFO interrupts.
- The corresponding bits in the CiMKIVLR1 register for mailboxes [56] to [63] are disabled. Set 0 to these bits.
- Transmit/receive FIFOs can be used for both data frames and remote frames.

20.3.2 CANi Bit Configuration Register (CiBCR) (i = 0 to 5)

Address C0BCR: H'FF60 0544, C1BCR: H'FF60 0D44, C2BCR: H'FF60 1544, C3BCR: H'FF60 1D44, C4BCR: H'FF60 2544, C5BCR: H'FF60 2D44

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	TSEG1[3:0]				—	—	BRP[9:0]									
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
	—	—	SJW[1:0]		—	TSEG2[2:0]			—	—	—	—	—	—	—	—
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 b28	TSEG1[3:0]	Time Segment 1 Control Bits	b31 b30 b29 b28 0 0 0 0: Do not use this combination 0 0 0 1: Do not use this combination 0 0 1 0: Do not use this combination 0 0 1 1: 4Tq 0 1 0 0: 5Tq 0 1 0 1: 6Tq 0 1 1 0: 7Tq 0 1 1 1: 8Tq 1 0 0 0: 9Tq 1 0 0 1: 10Tq 1 0 1 0: 11Tq 1 0 1 1: 12Tq 1 1 0 0: 13Tq 1 1 0 1: 14Tq 1 1 1 0: 15Tq 1 1 1 1: 16Tq	R/W
b27, b26	—	Reserved	These bits are read as 0. The write value should be 0.	R
b25 to b16	BRP[9:0]	Prescaler Division Ratio Set Bit	If the setting value is P (0 to 1023), the baud rate prescaler divides fCAN by P + 1.	R/W
b15, b14	—	Reserved	These bits are read as 0. The write value should be 0.	R
b13, b12	SJW[1:0]	Resynchronization Jump Width Control Bit	b13 b12 0 0: 1Tq 0 1: 2Tq 1 0: 3Tq 1 1: 4Tq	R/W
b11	—	Reserved	This bit is read as 0. The write value should be 0.	R
b10 to b8	TSEG2[2:0]	Time Segment 2 Control Bit	b10 b9 b8 0 0 0: Do not use this combination 0 0 1: 2Tq 0 1 0: 3Tq 0 1 1: 4Tq 1 0 0: 5Tq 1 0 1: 6Tq 1 1 0: 7Tq 1 1 1: 8Tq	R/W
b7 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

For setting of a bit timing, refer to section 20.5, CAN Communication Speed Configuration.

Set the CiBCR register before entering CAN halt mode from CAN reset mode or CAN operation mode from CAN reset mode. After the setting is made once, this register can be written to in CAN reset mode or CAN halt mode.

TSEG1 Bits

The TSEG1 bits are used to specify the total length of the propagation time segment (PROP_SEG) and phase buffer segment 1 (PHASE_SEG1) with the value of Time Quantum (Tq).
A value from 4 to 16 time quanta can be set.

BRP Bits

The BRP bits are used to set the frequency of the CAN communication clock (fCANCLK).
One cycle of fCANCLK is a Time Quantum (Tq).

SJW Bits

The SJW bits are used to specify the resynchronization jump width with the value of Tq.
A value from 1 to 4 time quanta can be set.
Set the value smaller than or equal to that of the TSEG2 bits.

TSEG2 Bits

The TSEG2 bits are used to specify the length of phase buffer segment 2 (PHASE_SEG2) with the value of Tq.
A value from 2 to 8 time quanta can be set.
Set the value smaller than that of the TSEG1 bits.

20.3.3 CANi Mask Register k (CiMKRk) (i = 0 to 5; k = 0 to 9)

Address

C0MKR0: H'FF60 0430, C0MKR1: H'FF60 0434, C0MKR2: H'FF60 0400, C0MKR3: H'FF60 0404,
C0MKR4: H'FF60 0408, C0MKR5: H'FF60 040C, C0MKR6: H'FF60 0410, C0MKR7: H'FF60 0414,
C0MKR8: H'FF60 0418, C0MKR9: H'FF60 041C
C1MKR0: H'FF60 0C30, C1MKR1: H'FF60 0C34, C1MKR2: H'FF60 0C00, C1MKR3: H'FF60 0C04,
C1MKR4: H'FF60 0C08, C1MKR5: H'FF60 0C0C, C1MKR6: H'FF60 0C10, C1MKR7: H'FF60 0C14,
C1MKR8: H'FF60 0C18, C1MKR9: H'FF60 0C1C
C2MKR0: H'FF60 1430, C2MKR1: H'FF60 1434, C2MKR2: H'FF60 1400, C2MKR3: H'FF60 1404,
C2MKR4: H'FF60 1408, C2MKR5: H'FF60 140C, C2MKR6: H'FF60 1410, C2MKR7: H'FF60 1414,
C2MKR8: H'FF60 1418, C2MKR9: H'FF60 141C
C3MKR0: H'FF60 1C30, C3MKR1: H'FF60 1C34, C3MKR2: H'FF60 1C00, C3MKR3: H'FF60 1C04,
C3MKR4: H'FF60 1C08, C3MKR5: H'FF60 1C0C, C3MKR6: H'FF60 1C10, C3MKR7: H'FF60 1C14,
C3MKR8: H'FF60 1C18, C3MKR9: H'FF60 1C1C
C4MKR0: H'FF60 2430, C4MKR1: H'FF60 2434, C4MKR2: H'FF60 2400, C4MKR3: H'FF60 2404,
C4MKR4: H'FF60 2408, C4MKR5: H'FF60 240C, C4MKR6: H'FF60 2410, C4MKR7: H'FF60 2414,
C4MKR8: H'FF60 2418, C4MKR9: H'FF60 241C
C5MKR0: H'FF60 2C30, C5MKR1: H'FF60 2C34, C5MKR2: H'FF60 2C00, C5MKR3: H'FF60 2C04,
C5MKR4: H'FF60 2C08, C5MKR5: H'FF60 2C0C, C5MKR6: H'FF60 2C10, C5MKR7: H'FF60 2C14,
C5MKR8: H'FF60 2C18, C5MKR9: H'FF60 2C1C

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	SID[10:0]											EID[17:16]	
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
	EID[15:0]															
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
b31 to b29	—	Reserved	The reset value is undefined. The write value should be 0. These bits are read as 0 after 0 is written to.	R
b28 to b18	SID[10:0]	Standard ID Bit	0: Corresponding SID bit is not compared 1: Corresponding SID bit is compared	R/W
b17 to b0	EID[17:0]	Extended ID Bit	0: Corresponding EID bit is not compared 1: Corresponding EID bit is compared	R/W

For the masking function in FIFO mailbox mode, refer to section 20.7, Acceptance Filtering and Masking Function.
Write to registers CiMKR0 to CiMKR9 in CAN reset mode or CAN halt mode.

SID Bits

The SID bits are the filter mask bits corresponding to the CAN standard ID bits. The SID bits are used to receive both standard ID and extended ID messages.
When the SID bit is set to 0, the corresponding SID bit does not compare received ID with mailbox ID.
When the SID bit is set to 1, corresponding SID bit compares received ID with mailbox ID.

EID Bits

The EID bits are the filter mask bits for CAN extended ID bits.
The EID bits are used to receive extended ID messages.
When the EID bit is set to 0, corresponding EID bit does not compare received ID with mailbox ID.
When the EID bit is set to 1, corresponding EID bit compares received ID with mailbox ID.

20.3.4 CANi FIFO Received ID Compare Registers n (CiFIDCR0 and CiFIDCR1) (i = 0 to 5; n = 0, 1)

Address C0FIDCR0: H'FF60 0420, C1FIDCR0: H'FF60 0C20, C2FIDCR0: H'FF60 1420, C3FIDCR0: H'FF60 1C20, C4FIDCR0: H'FF60 2420, C5FIDCR0: H'FF60 2C20
C0FIDCR1: H'FF60 0424, C1FIDCR1: H'FF60 0C24, C2FIDCR1: H'FF60 1424, C3FIDCR1: H'FF60 1C24, C4FIDCR1: H'FF60 2424, C5FIDCR1: H'FF60 2C24

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IDE	RTR	—	SID[10:0]											EID[17:16]	
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
	EID[15:0]															
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
b31	IDE	ID Extension Bit*	0: Standard ID 1: Extended ID	R/W
b30	RTR	Remote Frame Request Bit	0: Data frame 1: Remote frame	R/W
b29	—	Reserved	The reset value is undefined. The write value should be 0. This bit is read as 0 after 0 is written to.	R
b28 to b18	SID[10:0]	Standard ID Bit	0: Corresponding SID bit is 0 1: Corresponding SID bit is 1	R/W
b17to b0	EID[17:0]	Extended ID Bit	0: Corresponding EID bit is 0 1: Corresponding EID bit is 1	R/W

Note: * The IDE bit is enabled when the IDFM bits in the CiCTLR register are 10 (mixed ID mode). When the IDFM bits are not 10, the IDE bit should be written with 0.

Registers CiFIDCR0 and CiFIDCR1 are enabled when the MBM bit in the CiCTLR register is set to 1 (FIFO mailbox mode). Bits EID, SID, RTR, and IDE in registers CiMB60 to CiMB63 are disabled.
For using these registers, refer to section 20.7, Acceptance Filtering and Masking Function.
Write to registers CiFIDCR0 and CiFIDCR1 in CAN reset mode or CAN halt mode.

IDE Bit

The IDE bit sets the ID format to standard ID or extended ID. The IDE bit is enabled when the IDFM bits in the CiCTLR register are 10 (mixed ID mode). When the IDFM bits are 10, the IDE bit specifies the following operation.

- When both IDE bits in registers CiFIDCR0 and CiFIDCR1 are set to 0, only standard ID frames can be received.
- When both IDE bits in registers CiFIDCR0 and CiFIDCR1 are set to 1, only extended ID frames can be received.
- When the IDE bits in registers CiFIDCR0 and CiFIDCR1 are set to 0 or 1 individually, both standard ID and extended ID frames can be received.

RTR Bit

The RTR bit sets the specified frames format of data frame or remote frame. The RTR bit specifies the following operation.

- When both RTR bits in registers CiFIDCR0 and CiFIDCR1 are set to 0, only data frames can be received.
- When both RTR bits in registers CiFIDCR0 and CiFIDCR1 are set to 1, only remote frames can be received.
- When the RTR bits in registers CiFIDCR0 and CiFIDCR1 are set to 0 or 1 individually, both data frames and remote frames can be received.

SID Bits

The SID bits set the standard ID of data frames and remote frames. The SID bits are used to receive both standard ID and extended ID messages.

EID Bits

The EID bits set the extended ID of data frames and remote frames. The EID bits are used to receive extended ID messages.

20.3.5 CANi Mask Invalid Registers n (CiMKIVLR0, CiMKIVLR1) (i = 0 to 5; n = 0, 1)

Address C0MKIVLR1: H'FF60 0428, C1MKIVLR1: H'FF60 0C28, C2MKIVLR1: H'FF60 1428, C3MKIVLR1: H'FF60 1C28, C4MKIVLR1: H'FF60 2428, C5MKIVLR1: H'FF60 2C28

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	MB63	MB62	MB61	MB60	MB59	MB58	MB57	MB56	MB55	MB54	MB53	MB52	MB51	MB50	MB49	MB48
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
	MB47	MB46	MB45	MB44	MB43	MB42	MB41	MB40	MB39	MB38	MB37	MB36	MB35	MB34	MB33	MB32
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
b31 to b0	MB63 to MB32	Mask invalid Bit	Bit 31 corresponds to mailbox 63 (MB63), and bit 0 corresponds to mailbox 32 (MB32)*. 0: Mask valid 1: Mask invalid	R/W

Note: * In FIFO mailbox mode, write 0 to bits 24 to 31.

Address C0MKIVLR0: H'FF60 0438, C1MKIVLR0: H'FF60 0C38, C2MKIVLR0: H'FF60 1438, C3MKIVLR0: H'FF60 1C38, C4MKIVLR0: H'FF60 2438, C5MKIVLR0: H'FF60 2C38

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	MB31	MB30	MB29	MB28	MB27	MB26	MB25	MB24	MB23	MB22	MB21	MB20	MB19	MB18	MB17	MB16
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
	MB15	MB14	MB13	MB12	MB11	MB10	MB9	MB8	MB7	MB6	MB5	MB4	MB3	MB2	MB1	MB0
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
b31 to b0	MB31 to MB0	Mask invalid Bit	Bit 31 corresponds to mailbox 31 (MB31), and bit 0 corresponds to mailbox 0 (MB0). 0: Mask valid 1: Mask invalid	R/W

Each bit in registers CiMKIVLR0 and CiMKIVLR1 corresponds to a mailbox. The correspondence between the bits and mailboxes is as follows:

- Bit 0 in the CiMKIVLR0 register corresponds to mailbox 0 (MB0).
- Bit 31 in the CiMKIVLR0 register corresponds to mailbox 31 (MB31).
- Bit 0 in the CiMKIVLR1 register corresponds to mailbox 32 (MB32).
- Bit 31 in the CiMKIVLR1 register corresponds to mailbox 63 (MB63).

When each bit is 1, the acceptance mask for the mailbox corresponding to the bit number is disabled. In this case, a receiving message is stored into mailbox only its ID matches bits SID and EID in the CiMBj register (j = 0 to 63). Write to registers CiMKIVLR0 and CiMKIVLR1 either in CAN reset mode or CAN halt mode.

20.3.6 CANi Mailbox Register j (CiMBj) (i = 0 to 5; j = 0 to 63)

Table 20.11 lists the CANi mailbox memory mapping and Table 20.12 lists the CAN data frame construction.
The value after reset of CANi Mailbox is undefined.
Write to the CiMBj register only when the associated CiMCTLj register (i = 0 to 5, j = 0 to 63) is H'00 and the corresponding mailbox is not processing an abort request.
Refer to Table 20.11 for detailed addresses.

Table 20.11 CANi Mailbox Memory Mapping (i = 0 to 5)

Address						Message Content
CAN0	CAN1	CAN2	CAN3	CAN4	CAN5	Memory Mapping
H'FF60 0000 + 16 × j + 0	H'FF60 0800 + 16 × j + 0	H'FF60 1000 + 16 × j + 0	H'FF60 1800 + 16 × j + 0	H'FF60 2000 + 16 × j + 0	H'FF60 2800 + 16 × j + 0	IDE, RTR, SID10 to SID6
H'FF60 0000 + 16 × j + 1	H'FF60 0800 + 16 × j + 1	H'FF60 1000 + 16 × j + 1	H'FF60 1800 + 16 × j + 1	H'FF60 2000 + 16 × j + 1	H'FF60 2800 + 16 × j + 1	SID5 to SID0, EID17, EID16
H'FF60 0000 + 16 × j + 2	H'FF60 0800 + 16 × j + 2	H'FF60 1000 + 16 × j + 2	H'FF60 1800 + 16 × j + 2	H'FF60 2000 + 16 × j + 2	H'FF60 2800 + 16 × j + 2	EID15 to EID8
H'FF60 0000 + 16 × j + 3	H'FF60 0800 + 16 × j + 3	H'FF60 1000 + 16 × j + 3	H'FF60 1800 + 16 × j + 3	H'FF60 2000 + 16 × j + 3	H'FF60 2800 + 16 × j + 3	EID7 to EID0
H'FF60 0000 + 16 × j + 4	H'FF60 0800 + 16 × j + 4	H'FF60 1000 + 16 × j + 4	H'FF60 1800 + 16 × j + 4	H'FF60 2000 + 16 × j + 4	H'FF60 2800 + 16 × j + 4	—
H'FF60 0000 + 16 × j + 5	H'FF60 0800 + 16 × j + 5	H'FF60 1000 + 16 × j + 5	H'FF60 1800 + 16 × j + 5	H'FF60 2000 + 16 × j + 5	H'FF60 2800 + 16 × j + 5	Data length code (DLC)
H'FF60 0000 + 16 × j + 6	H'FF60 0800 + 16 × j + 6	H'FF60 1000 + 16 × j + 6	H'FF60 1800 + 16 × j + 6	H'FF60 2000 + 16 × j + 6	H'FF60 2800 + 16 × j + 6	Data byte 0
H'FF60 0000 + 16 × j + 7 : H'FF60 0000 + 16 × j + 13	H'FF60 0800 + 16 × j + 7 : H'FF60 0800 + 16 × j + 13	H'FF60 1000 + 16 × j + 7 : H'FF60 1000 + 16 × j + 13	H'FF60 1800 + 16 × j + 7 : H'FF60 1800 + 16 × j + 13	H'FF60 2000 + 16 × j + 7 : H'FF60 2000 + 16 × j + 13	H'FF60 2800 + 16 × j + 7 : H'FF60 2800 + 16 × j + 13	Data byte 1 : Data byte 7
H'FF60 0000 + 16 × j + 14	H'FF60 0800 + 16 × j + 14	H'FF60 1000 + 16 × j + 14	H'FF60 1800 + 16 × j + 14	H'FF60 2000 + 16 × j + 14	H'FF60 2800 + 16 × j + 14	Time stamp upper byte
H'FF60 0000 + 16 × j + 15	H'FF60 0800 + 16 × j + 15	H'FF60 1000 + 16 × j + 15	H'FF60 1800 + 16 × j + 15	H'FF60 2000 + 16 × j + 15	H'FF60 2800 + 16 × j + 15	Time stamp lower byte

Table 20.12 CAN Data Frame Construction

SID10 to SID6	SID5 to SID0	EID17 to EID16	EID15 to EID8	EID7 to EID0	DLC3 to DLC0	DATA0	DATA1	• • •	DATA7

The previous value of each mailbox is retained unless a new message is received.

Address C0MB0 to C0MB63: H'FF60 0000 to H'FF60 03F0, C1MB0 to C1MB63: H'FF60 0800 to H'FF60 0BF0, C2MB0 to C2MB63: H'FF60 1000 to H'FF60 13F0, C3MB0 to C3MB63: H'FF60 1800 to H'FF60 1BF0, C4MB0 to C4MB63: H'FF60 2000 to H'FF60 23F0, C5MB0 to C5MB63: H'FF60 2800 to H'FF60 2BF0

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IDE	RTR	—	SID[10:0]											EID[17:16]	
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
	EID[15:0]															
After Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

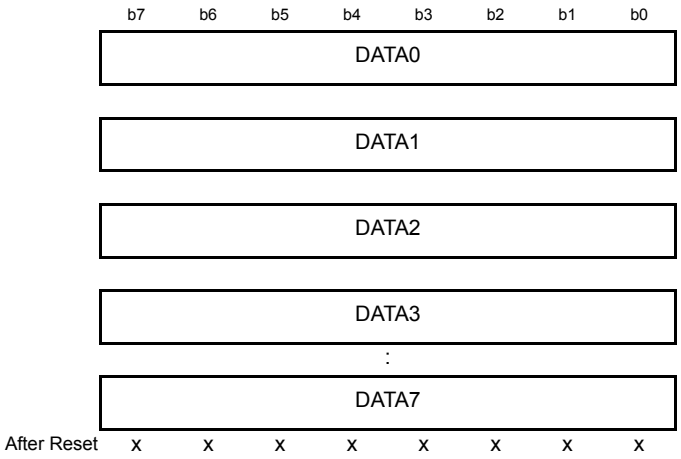
Bit	Symbol	Bit Name	Description	R/W
b31	IDE	ID Extension Bit*1	0: Standard ID 1: Extended ID	R/W
b30	RTR	Remote Frame Request Bit	0: Data frame 1: Remote frame	R/W
b29	—	Reserved	The reset value is undefined. The write value should be 0. This bit is read as 0 after 0 is written to.	R
b28 to b18	SID[10:0]	Standard ID Bits	0: Corresponding SID bit is 0 1: Corresponding SID bit is 1	R/W
b17 to b0	EID[17:0]	Extended ID Bits*2	0: Corresponding EID bit is 0 1: Corresponding EID bit is 1	R/W

Notes: 1. The IDE bit is enabled when the IDFM bits in the CiCTLR register are 10 (mixed ID mode). When the IDFM bits are not 10, it should be written with 0.
2. If the mailbox has received a standard ID message, the EID bits in the mailbox are undefined.

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	DLC[3:0]			
After Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

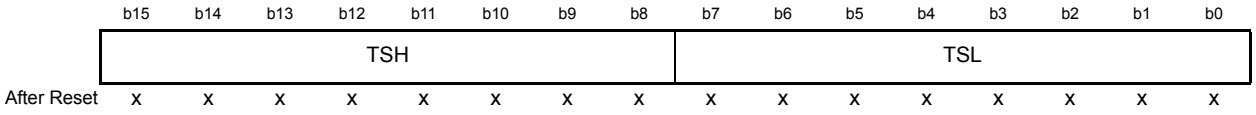
Bit	Symbol	Bit Name	Description	R/W
b15 to b4	—	Reserved	The reset value is undefined. The write value should be 0. These bits are read as 0 after 0 is written to.	R
b3 to b0	DLC[3:0]	Data Length Code Bits*	b3 b2 b1 b0 0 0 0 0: Data length = 0 bytes 0 0 0 1: Data length = 1 byte 0 0 1 0: Data length = 2 bytes 0 0 1 1: Data length = 3 bytes 0 1 0 0: Data length = 4 bytes 0 1 0 1: Data length = 5 bytes 0 1 1 0: Data length = 6 bytes 0 1 1 1: Data length = 7 bytes 1 x x x: Data length = 8 bytes Legend: x represents any value.	R/W

Note: * When a message with a DLC value lower than eight is received, data exceeding the DLC of the mailbox are undefined.



Bit	Symbol	Bit Name	Description	R/W
b7 to b0	DATA0 to DATA7	Data Bytes 0 to 7*1*2	H'00 to H'FF	R/W

Notes: 1. If the mailbox has received a message with n bytes less than 8 bytes, the values of DATAn to DATA7 in the mailbox are undefined.
2. If the mailbox has received a remote frame, the previous values of DATA0 to DATA7 in the mailbox are retained.



Bit	Symbol	Bit Name	Description	R/W
b15 to b8	TSH	Time Stamp Higher Byte	H'00 to H'FF	R/W
b7 to b0	TSL	Time Stamp Lower Byte	H'00 to H'FF	R/W

IDE Bit

The IDE bit sets the ID format of standard IDs or extended IDs.
The IDE bit is enabled when the IDFM bits in the CiCTLR register is 10 (mixed ID mode).
When the IDFM bits are 10, the IDE bit specifies the following operation.

- Receive mailbox receives only ID format specified by the IDE bit.
- Transmit mailbox transmits with ID format specified by the IDE bit.
- Receive FIFO mailbox receives messages with the standard ID, extended ID, or both IDs specified by the IDE bit in registers CiFIDCR0 and CiFIDCR1.
- Transmit FIFO mailbox transmits messages with the standard ID or extended ID specified by the IDE bit in the relevant transmitting message.

RTR Bit

The RTR bit sets the frame format of data frames or remote frames.
This bit specifies the following operation:

- Receive mailbox receives only frames with the format specified by the RTR bit.
- Transmit mailbox transmits according to the frame format specified by the RTR bit.
- Receive FIFO mailbox receives the data frame, remote frame, or both frames specified by the RTR bit in registers CiFIDCR0 and CiFIDCR1.
- Transmit FIFO mailbox transmits the data frame or remote frame specified by the RTR bit in the relevant transmitting message.

SID Bits

The SID bits set the standard ID of data frames and remote frames.
The SID bits are used to transmit or receive both standard ID and extended ID messages.

EID Bits

The EID bits set the extended ID of data frames and remote frames.
The EID bits are used to transmit or receive extended ID messages.

DLC Bits

The DLC bits are used to set the number of data bytes to be transmitted in a data frame. When data is requested using a remote frame, the number of data bytes to be requested is set.
When a data frame is received, the number of received data bytes is stored. When a remote frame is received, the number of requested data bytes is stored.

DATA0 to DATA7

DATA0 to DATA7 store the transmitted or received CAN message data. Transmission or reception starts from DATA0. The bit order on the CAN bus is MSB first, and transmission or reception starts from bit 7.

TSH, TSL

Bits TSL and TSH store the counter value of the time stamp when received messages are stored in the mailbox.

20.3.7 CANi Mailbox Interrupt Enable Registers n (CiMIER0 and CiMIER1) (i = 0 to 5; n = 0, 1)

Address C0MIER1: H'FF60 042C, C1MIER1: H'FF60 0C2C, C2MIER1: H'FF60 142C, C3MIER1: H'FF60 1C2C, C4MIER1: H'FF60 242C, C5MIER1: H'FF60 2C2C

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	MB63	MB62	MB61	MB60	MB59	MB58	MB57	MB56	MB55	MB54	MB53	MB52	MB51	MB50	MB49	MB48
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
	MB47	MB46	MB45	MB44	MB43	MB42	MB41	MB40	MB39	MB38	MB37	MB36	MB35	MB34	MB33	MB32
After Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

- Normal mailbox mode

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	MB63 to MB32	Interrupt Enable Bits	Bit 31 corresponds to mailbox 63 (MB63), and bit 0 corresponds to mailbox 32 (MB32). 0: Interrupt disabled 1: Interrupt enabled	R/W

- FIFO mailbox mode (CiMIER1 only)

Bit	Symbol	Bit Name	Description	R/W
b31, b30	—	Reserved	The read data is undefined. The write value should be 0.	R
b29	MB61	Receive FIFO Interrupt Generation Timing Control Bit*	Receive FIFO interrupt request is generated 0: Every time reception is completed 1: When receive FIFO becomes buffer warning by completion of reception	R/W
b28	MB60	Receive FIFO Interrupt Enable Bit	0: Interrupt disabled 1: Interrupt enabled	R/W
b27, b26	—	Reserved	The read data is undefined. The write value should be 0.	R
b25	MB57	Transmit FIFO Interrupt Generation Timing Control Bit	Transmit FIFO interrupt request is generated 0: Every time transmission is completed 1: When transmit FIFO becomes empty due to completion of transmission	R/W
b24	MB56	Transmit FIFO Interrupt Enable Bit	0: Interrupt disabled 1: Interrupt enabled	R/W
b23 to b0	MB55 to MB32	Interrupt Enable Bits	Bit 23 corresponds to mailbox 55 (MB55), and bit 0 corresponds to mailbox 32 (MB32). 0: Interrupt disabled 1: Interrupt enabled	R/W

Note: * No interrupt request is generated when the receive FIFO becomes buffer warning from full. “Buffer warning” indicates a state in which the third unread message is stored in the receive FIFO.

Address C0MIER0: H'FF60 043C, C1MIER0: H'FF60 0C3C, C2MIER0: H'FF60 143C, C3MIER0: H'FF60 1C3C, C4MIER0: H'FF60 243C, C5MIER0: H'FF60 2C3C

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	MB31	MB30	MB29	MB28	MB27	MB26	MB25	MB24	MB23	MB22	MB21	MB20	MB19	MB18	MB17	MB16
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
	MB15	MB14	MB13	MB12	MB11	MB10	MB9	MB8	MB7	MB6	MB5	MB4	MB3	MB2	MB1	MB0
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
b31 to b0	MB31 to MB0	Interrupt Enable Bits	Bit 31 corresponds to mailbox 31 (MB31), and bit 0 corresponds to mailbox 0 (MB0). 0: Interrupt disabled 1: Interrupt enabled	R/W

Interrupts can be enabled individually for each mailbox.

In normal mailbox mode (all bits) and in FIFO mailbox mode (bits 23 to 0 in the CiMIER1 register and all bits in the CiMIER0 register), each bit corresponds to the mailbox with the related number. These bits enable or disable transmission/reception complete interrupts for the corresponding mailboxes.

- Bit 0 in the CiMIER0 register corresponds to mailbox 0 (MB0).
- Bit 31 in the CiMIER0 register corresponds to mailbox 31 (MB31).
- Bit 0 in the CiMIER1 register corresponds to mailbox 32 (MB32).
- Bit 31 in the CiMIER1 register corresponds to mailbox 63 (MB63).

In FIFO mailbox mode, bits 29, 28, 25, and 24 of the CiMIER1 register specify whether transmit/receive FIFO interrupts are enabled/disabled and timing when interrupt requests are generated.

Write to registers CiMIER0 and CiMIER1 only when the associated CiMCTLj register (j = 0 to 63) is H'00 and the corresponding mailbox is not processing a transmission or reception abort request. In FIFO mailbox mode, change the bits in the CiMIER1 register for the associated FIFO only when:

- The TFE bit in the CiTFCR register is 0 and the TFEST bit is 1, and
- The RFE bit in the CiRFCR register is 0 and the RFEST bit is 1.

20.3.8 CANi Message Control Register j (CiMCTLj) (i = 0 to 5; j = 0 to 63)

Address C0MCTL0 to C0MCTL63: H'FF60 0500 to H'FF60 053F, C1MCTL0 to C1MCTL63: H'FF60 0D00 to H'FF60 0D3F, C2MCTL0 to C2MCTL63: H'FF60 1500 to H'FF60 153F, C3MCTL0 to C3MCTL63: H'FF60 1D00 to H'FF60 1D3F, C4MCTL0 to C4MCTL63: H'FF60 2500 to H'FF60 253F, C5MCTL0 to C5MCTL63: H'FF60 2D00 to H'FF60 2D3F

Registers CiMCTL32 to CiMCTL63

Transmit mailbox setting enabled (When the TRMREQ bit is 1 and the RECREQ bit is 0)

	b7	b6	b5	b4	b3	b2	b1	b0
	TRMREQ	RECREQ	—	ONESHOT	—	TRMABT	TRMACTIVE	SENTDATA
After Reset	0	0	0	0	0	0	0	0

Receive mailbox setting enabled (When the TRMREQ bit is 0 and the RECREQ bit is 1)

	b7	b6	b5	b4	b3	b2	b1	b0
	TRMREQ	RECREQ	—	ONESHOT	—	MSGLOST	INVALDATA	NEWDATA
After Reset	0	0	0	0	0	0	0	0

Registers CiMCTL0 to CiMCTL31

	b7	b6	b5	b4	b3	b2	b1	b0
	—	RECREQ	—	—	—	MSGLOST	INVALDATA	NEWDATA
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	TRMREQ	Transmit Mailbox Request Bit*2*4	0: Not configured for transmission 1: Configured for transmission	R/W
	—	Reserved (Registers CiMCTL0 to CiMCTL31)	This bit is read as 0. The write value should be 0.	R
b6	RECREQ	Receive Mailbox Request Bit*2*4*5	0: Not configured for reception 1: Configured for reception	R/W
b5	—	Reserved	This bit is read as 0. The write value should be 0.	R
b4	ONESHOT	One-shot Enable Bit*3	0: One-shot reception or one-shot transmission disabled 1: One-shot reception or one-shot transmission enabled	R/W
	—	Reserved (Registers CiMCTL0 to CiMCTL31)	This bit is read as 0. The write value should be 0.	R
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R
b2	TRMABT	Transmission Abort Complete Flag (Transmit mailbox setting enabled)*1*2	0: Transmission has started, transmission abort failed because transmission is completed, or transmission abort is not requested 1: Transmission abort is completed	R/W
	MSGLOST	Message Lost Flag (Receive mailbox setting enabled)*1*2	0: Message is not overwritten or overrun 1: Message is overwritten or overrun	R/W
b1	TRMACTIVE	Transmission-in-Progress Status Flag (Transmit mailbox setting enabled)	0: Transmission is pending or transmission is not requested 1: From acceptance of transmission request to completion of transmission, or generation of error/arbitration lost	R
	INVALDATA	Reception-in-Progress Status Flag (Receive mailbox setting enabled)	0: Message valid 1: Message being updated	R
b0	SENTDATA	Transmission Complete Flag (Transmit mailbox setting enabled)*1*2	0: Transmission is not completed (pending) 1: Transmission is completed (success)	R/W
	NEWDATA	Reception Complete Flag (Receive mailbox setting enabled)*1*2	0: No data has been received or 0 is written to the NEWDATA bit 1: A new message is being stored or has been stored to the mailbox	R/W

Notes: 1. Write 0 only. Writing 1 has no effect.
2. When writing 0 to bits NEWDATA, SENTDATA, MSGLOST, TRMABT, RECREQ, and TRMREQ by a program, use the MOV

instruction to ensure that only the specified bit is set to 0 and the other bits are set to 1.
3. To enter one-shot receive mode, write 1 to the ONESHOT bit at the same time as setting the RECREQ bit to 1.
   To exit one-shot receive mode, write 0 to the ONESHOT bit after writing 0 to the the RECREQ bit and confirming it has been set to 0.
   To enter one-shot transmit mode, write 1 to the ONESHOT bit at the same time as setting the TRMREQ bit to 1.
   To exit one-shot transmit mode, write 0 to the ONESHOT bit after the message has been transmitted or aborted.
4. Do not set both the RECREQ and TRMREQ bits to 1.
5. When setting the RECREQ bit to 0, set bits MSGLOST, NEWDATA, and RECREQ to 0 simultaneously.

Write to the CiMCTLj register in CAN operation mode or CAN halt mode.
Do not use registers CiMCTL56 to CiMCTL63 in FIFO mailbox mode.

TRMREQ Bit

The TRMREQ bit selects transmit modes shown in Table 20.17.
When TRMREQ bit is set to 1, the corresponding mailbox is configured for transmission of a data frame or a remote frame.
When TRMREQ bit is set to 0, the corresponding mailbox is not configured for transmission of a data frame or a remote frame.
If the TRMREQ bit is changed from 1 to 0 to cancel the corresponding transmission request, either the TRMABT or SENTDATA bit is set to 1.
When setting the TRMREQ bit to 1, do not set the RECREQ bit to 1.
To change the configuration of a mailbox from reception to transmission, first abort the reception and then set bits NEWDATA and MSGLOST to 0 before changing to transmission.

RECREQ Bit

The RECREQ bit selects receive modes shown in Table 20.17.
When the RECREQ bit is set to 1, the corresponding mailbox is configured for reception of a data frame or a remote frame.
When the RECREQ bit is set to 0, the corresponding mailbox is not configured for reception of a data frame or a remote frame.
Due to hardware protection the RECREQ bit cannot be set to 0 by writing 0 by a program during the following period.

- Hardware protection is started

From the acceptance filter procedure. (the beginning of CRC field)

- Hardware protection is released

- For the mailbox that is specified to receive the incoming message, after the received data is stored into the mailbox or a CAN bus error occurs. (i.e. a maximum period of hardware protection is from the beginning of CRC field to the end of the 7th bit of EOF.)
- For the other mailboxes, after the acceptance filter procedure.
- If no mailbox is specified to receive the message, after the acceptance filter procedure.

When setting the RECREQ bit to 1, do not set 1 to the TRMREQ bit.
To change the configuration of a mailbox from transmission to reception, first abort the transmission and then set bits SENTDATA and TRMABT to 0 before changing to reception.

ONESHOT Bit

The ONESHOT bit can be used in the following two ways, receive mode and transmit mode:

- One-Shot Receive Mode

When the ONESHOT bit is set to 1 in receive mode (RECREQ bit = 1 and TRMREQ bit = 0), the mailbox receives a message only one time. The mailbox does not behave as a receive mailbox after having received a message one time. The behavior of bits NEWDATA and INVALDATA is the same as in normal receive mode. In one-shot receive mode, the MSGLOST bit is not set to 1.

To set the ONESHOT bit to 0, first write 0 to the RECREQ bit and ensure that it has been set to 0.

- One-Shot Transmit Mode

When the ONESHOT bit is set to 1 in transmit mode (RECREQ bit = 0 and TRMREQ bit = 1), the CAN module transmits a message only one time.

The CAN module does not transmit the message again if a CAN bus error or CAN bus arbitration lost occurs. When transmission is completed, the SENTDATA bit is set to 1. If transmission is not completed due to a CAN bus error or CAN bus arbitration lost, the TRMABT bit is set to 1.

Set the ONESHOT bit to 0 after the SENTDATA or TRMABT bit is set to 1.

TRMABT Bit

The TRMABT bit is set to 1 in the following cases:

- Following a transmission abort request, when the transmission abort is completed before starting transmission.
- Following a transmission abort request, when the CAN module detects CAN bus arbitration lost or a CAN bus error.
- In one-shot transmit mode (RECREQ bit = 0, TRMREQ bit = 1, and ONESHOT bit = 1), when the CAN module detects CAN bus arbitration lost or a CAN bus error.

The TRMABT bit is not set to 1 when data transmission is completed. In this case, the SENTDATA bit is set to 1.

The TRMABT bit is set to 0 by writing 0 by a program.

MSGLOST Bit

The MSGLOST bit is set to 1 when the mailbox is overwritten or overrun by a new received message while the NEWDATA bit is 1.

The MSGLOST bit is set to 1 at the end of the 6th bit of EOF.

The MSGLOST bit is set to 0 by writing 0 by a program.

In both overwrite and overrun modes, the MSGLOST bit is not set to 0 by writing 0 by a program during the 5 peripheral bus clock B cycles following the 6th bit of EOF.

TRMACTIVE Bit

The TRMACTIVE bit is set to 1 when the corresponding mailbox of the CAN module begins transmitting a message.

The TRMACTIVE is set to 0 when the CAN module has lost CAN bus arbitration, a CAN bus error occurs, or data transmission is completed.

INVALDATA Bit

After the completion of a message reception, the INVALDATA bit is set to 1 while the received message is being updated into the corresponding mailbox.

The INVALDATA bit is set to 0 immediately after the message has been stored. If the mailbox is read while the INVALDATA bit is 1, the data is undefined.

SENTDATA Bit

The SENTDATA bit is set to 1 when data transmission from the corresponding mailbox is completed.

The SENTDATA bit is set to 0 by writing 0 by a program.

To set the SENTDATA bit to 0, first set the TRMREQ bit to 0.

Bits SENTDATA and TRMREQ cannot be set to 0 simultaneously.

To transmit a new message from the corresponding mailbox, set the SENTDATA bit to 0.

NEWDATA Bit

The NEWDATA bit is set to 1 when a new message is being stored or has been stored to the mailbox. The timing for setting this bit to 1 is simultaneous with the INVALDATA bit.
The NEWDATA bit is set to 0 by writing 0 by a program.
The NEWDATA bit is not set to 0 by writing 0 by a program while the related INVALDATA bit is 1.

20.3.9 CANi Receive FIFO Control Register (CiRFCR) (i = 0 to 5)

Address C0RFCR: H'FF60 0548, C1RFCR: H'FF60 0D48, C2RFCR: H'FF60 1548, C3RFCR: H'FF60 1D48, C4RFCR: H'FF60 2548, C5RFCR: H'FF60 2D48

	b7	b6	b5	b4	b3	b2	b1	b0
	RFEST	RFWST	RFFST	RFMLF	RFUST [2:0]			RFE
After Reset	1	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	RFEST	Receive FIFO Empty Status Flag	0: Unread message in receive FIFO 1: No unread message in receive FIFO	R
b6	RFWST	Receive FIFO Buffer Warning Status Flag	0: Receive FIFO is not buffer warning 1: Receive FIFO is buffer warning (3 unread messages)	R
b5	RFFST	Receive FIFO Full Status Flag	0: Receive FIFO is not full 1: Receive FIFO is full (4 unread messages)	R
b4	RFMLF	Receive FIFO Message Lost Flag*1	0: No receive FIFO message lost has occurred 1: Receive FIFO message lost has occurred	R/W
b3 to b1	RFUST [2:0]	Receive FIFO Unread Message Number Status Flag	b3 b2 b1 0 0 0: No unread message 0 0 1: 1 unread message 0 1 0: 2 unread messages 0 1 1: 3 unread messages 1 0 0: 4 unread messages 1 0 1: Reserved 1 1 0: Reserved 1 1 1: Reserved	R
b0	RFE	Receive FIFO Enable Bit*2	0: Receive FIFO disabled 1: Receive FIFO enabled	R/W

Notes: 1. Write 0 only. Writing 1 has no effect.
2. Write 0 to the RFE bit at the same time as the RFMLF bit.

Write to the CiRFCR register in CAN operation mode or CAN halt mode.

RFEST Bit

The RFEST bit is set to 1 (no unread message in receive FIFO) when the number of unread messages in the receive FIFO is 0. The RFEST bit is set to 1 when the RFE bit is set to 0. The RFEST bit is set to 0 (unread message in receive FIFO) when the number of unread messages in the receive FIFO is one or more.

RFWST Bit

The RFWST bit is set to 1 (receive FIFO is buffer warning) when the number of unread messages in the receive FIFO is 3. The RFWST bit is 0 (receive FIFO is not buffer warning) when the number of unread messages in the receive FIFO is less than 3 or equal to 4. The RFWST bit is set to 0 when the RFE bit is 0.

RFFST Bit

The RFFST bit is set to 1 (receive FIFO is full) when the number of unread messages in the receive FIFO is 4. The RFFST bit is 0 (receive FIFO is not full) when the number of unread messages in the receive FIFO is less than 4. The RFFST bit is set to 0 when the RFE bit is 0.

RFMLF Bit

The RFMLF bit is set to 1 (receive FIFO message lost has occurred) when the receive FIFO receives a new message and the receive FIFO is full. The timing for setting this bit to 1 is at the end of the 6th bit of EOF.
The RFMLF bit is set to 0 by writing 0 by a program.
In both overwrite and overrun modes, the RFMLF bit cannot be set to 0 (receive FIFO message lost has not occurred) by writing 0 by a program due to hardware protection during the five cycles of peripheral bus clock B following the 6th bit of EOF, if the receive FIFO is full and determined to receive the message.

RFUST Bits

The RFUST bit indicates the number of unread messages in the receive FIFO.
The value of the RFUST bit is initialized to 000 when the RFE bit is set to 0.

RFE Bit

When the RFE bit is set to 1, the receive FIFO is enabled.
When this bit is set to 0, the receive FIFO is disabled for reception and becomes empty (RFEST bit = 1).
Do not set this bit to 1 in normal mailbox mode (MBM bit in the CiCTLR register = 0).
Due to hardware protection, the RFE bit is not set to 0 by writing 0 by a program during the following period:

- The hardware protection is started
  - From the acceptance filter procedure (the beginning of CRC field)
- The hardware protection is released
  - If the receive FIFO is specified to receive the incoming message, after the received data is stored into the receive FIFO or a CAN bus error occurs. (i.e. maximum period of hardware protection is from the beginning of CRC field to the end of 7th bit of EOF.)
  - If the receive FIFO is not specified to receive the message, after the acceptance filter procedure.

Figure 20.2 shows the receive FIFO mailbox operation.

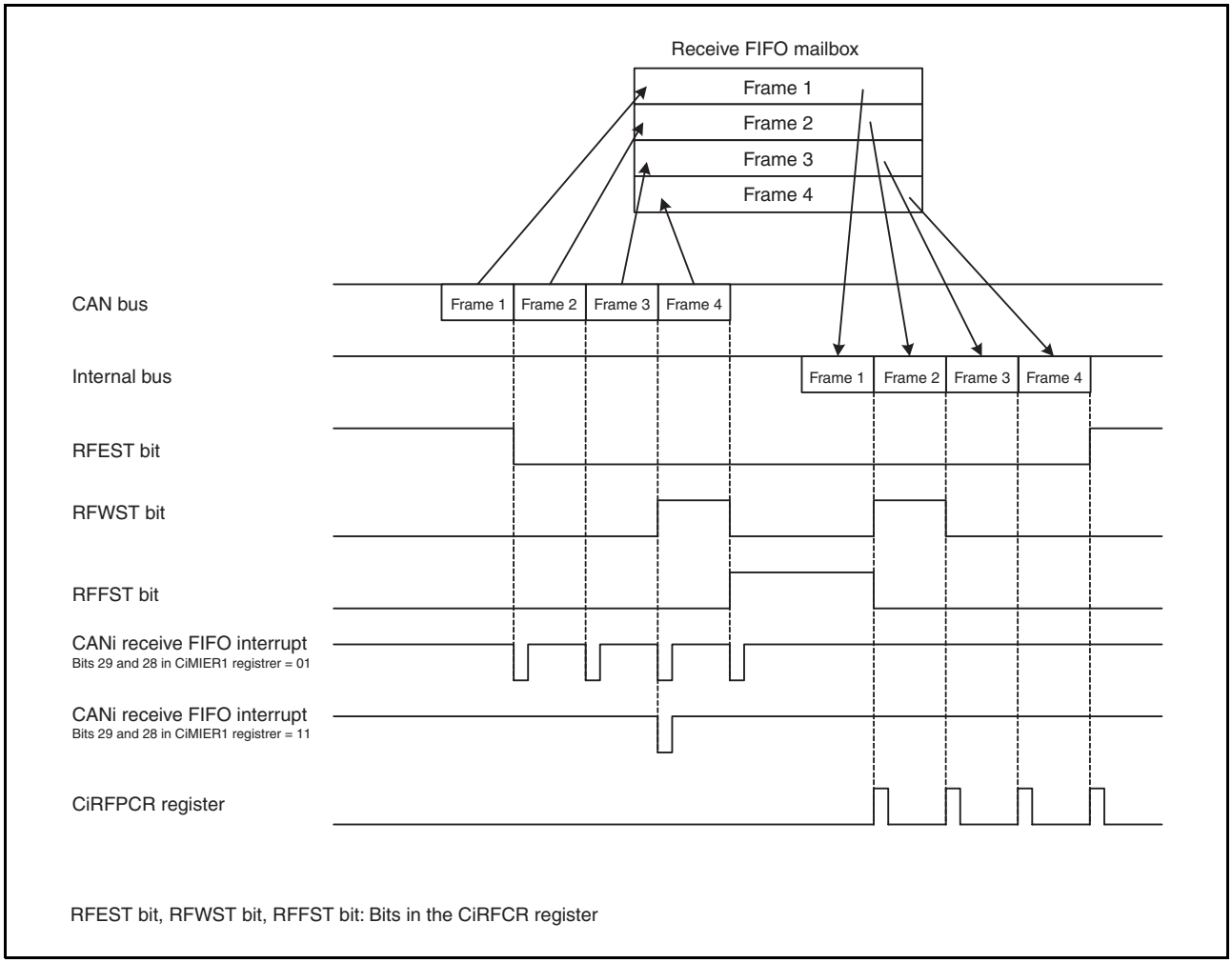


Figure 20.2 Receive FIFO Mailbox Operation (Bits 29 and 28 in CiMIER1 Register = 01 and 11) (i = 0 to 5)

20.3.10 CANi Receive FIFO Pointer Control Register (CiRFPCR) (i = 0 to 5)

Address C0RFPCR: H'FF60 0549, C1RFPCR: H'FF60 0D49, C2RFPCR: H'FF60 1549, C3RFPCR: H'FF60 1D49, C4RFPCR: H'FF60 2549, C5RFPCR: H'FF60 2D49

	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	—	—	—	—	—	—	—	—

Bit	Description	R/W
b7 to b0	The CPU-side pointer for the receive FIFO is incremented by writing H'FF	W

When the receive FIFO is not empty, write H'FF to the CiRFPCR register by a program to increment the CPU-side pointer for the receive FIFO to the next mailbox location.

Do not write to the CiRFPCR register when the RFE bit in the CiRFCR register is 0 (receive FIFO disabled).

Both the CAN-side pointer and the CPU-side pointer are incremented when a new message is received and the RFFST bit is 1 (receive FIFO is full) in overwrite mode. When the RFMLF bit is 1 in this condition, the CPU-side pointer cannot be incremented by writing to the CiRFPCR register by a program.

20.3.11 CANi Transmit FIFO Control Register (CiTFCR) (i = 0 to 5)

Address C0TFCR: H'FF60 054A, C1TFCR: H'FF60 0D4A, C2TFCR: H'FF60 154A, C3TFCR: H'FF60 1D4A, C4TFCR: H'FF60 254A, C5TFCR: H'FF60 2D4A

	b7	b6	b5	b4	b3	b2	b1	b0
	TFEST	TFFST	—	—	TFUST[2:0]			TFE
After Reset	1	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	TFEST	Transmit FIFO Empty Status Bit	0: Unsent message in transmit FIFO 1: No unsent message in transmit FIFO	R
b6	TFFST	Transmit FIFO Full Status Bit	0: Transmit FIFO is not full 1: Transmit FIFO is full (4 unsent messages)	R
b5	—	Reserved	The read data is undefined. The write value should be 0.	R
b4	—	Reserved	This bit is read as 0. The write value should be 0.	R
b3 to b1	TFUST[2:0]	Transmit FIFO Unsent Message Number Status Bit	b3 b2 b1 0 0 0: No unsent message 0 0 1: 1 unsent message 0 1 0: 2 unsent messages 0 1 1: 3 unsent messages 1 0 0: 4 unsent messages 1 0 1: Reserved 1 1 0: Reserved 1 1 1: Reserved	R
b0	TFE	Transmit FIFO Enable Bit	0: Transmit FIFO disabled 1: Transmit FIFO enabled	R/W

Write to the CiTFCR register in CAN operation mode or CAN halt mode.

TFEST Bit

The TFEST bit is set to 1 (no message in transmit FIFO) when the number of unsent messages in the transmit FIFO is 0. The TFEST bit is set to 1 when transmission from the transmit FIFO has been aborted.
The TFEST bit is set to 0 (message in transmit FIFO) when the number of unsent messages in the transmit FIFO is not 0.

TFFST Bit

The TFFST bit is set to 1 (transmit FIFO is full) when the number of unsent messages in the transmit FIFO is 4. The TFFST bit is set to 0 (transmit FIFO is not full) when the number of unsent messages in the transmit FIFO is less than 4. The TFFST bit is set to 0 when transmission from the transmit FIFO has been aborted.

TFUST Bits

The TFUST bits indicate the number of unsent messages in the transmit FIFO.
After the TFE bit is set to 0, the value of the TFUST bit is initialized to 000 when transmission abort or transmission is completed.

TFE Bit

When the TFE bit is set to 1, the transmit FIFO is enabled.
When the TFE bit is set to 0, the transmit FIFO becomes empty (TFEST bit = 1) and then unsent messages from the transmit FIFO are lost as described below:

- If a message from the transmit FIFO is not scheduled for the next transmission or during transmission.
- Following the completion of transmission, a CAN bus error, CAN bus arbitration lost, or entry to CAN halt mode if

a message from the transmit FIFO is scheduled for the next transmission or already during transmission.
Before setting the TFE bit to set to 1 again, ensure that the TFEST bit has been set to 1.
After setting the TFE bit to 1, write transmit data into the CiMB56 register.
Do not set the TFE bit to 1 in normal mailbox mode (MBM bit in the CiCTLR register = 0).

Figure 20.3 shows the transmit FIFO mailbox operation.

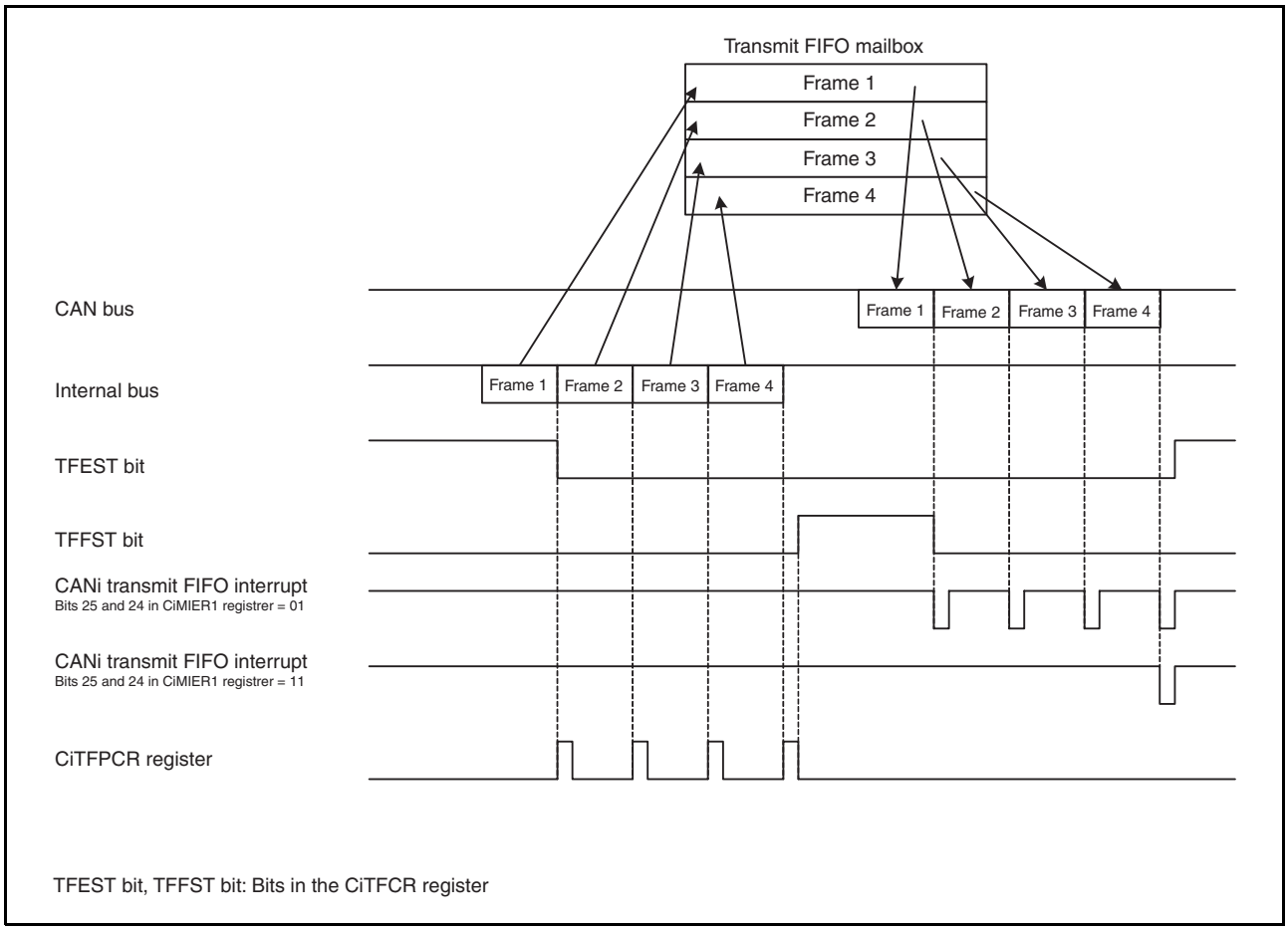


Figure 20.3 Transmit FIFO Mailbox Operation (Bits 25 and 24 in CiMIER1 Register = 01 and 11) (i = 0 to 5)

20.3.12 CANi Transmit FIFO Pointer Control Register (CiTFPCR) (i = 0 to 5)

Address C0TFPCR: H'FF60 054B, C1TFPCR: H'FF60 0D4B, C2TFPCR: H'FF60 154B, C3TFPCR: H'FF60 1D4B, C4TFPCR: H'FF60 254B, C5TFPCR: H'FF60 2D4B

	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	—	—	—	—	—	—	—	—

Bit	Description	R/W
b7 to b0	The CPU-side pointer for the transmit FIFO is incremented by writing H'FF	W

When the transmit FIFO is not full, write H'FF to the CiTFPCR register by a program to increment the CPU-side pointer for the transmit FIFO to the next mailbox location.
Do not write to the CiTFPCR register when the TFE bit in the CiTFCR register is 0 (transmit FIFO disabled).

20.3.13 CANi Status Register (CiSTR) (i = 0 to 5)

Address C0STR: H'FF60 0542, C1STR: H'FF60 0D42, C2STR: H'FF60 1542, C3STR: H'FF60 1D42, C4STR: H'FF60 2542, C5STR: H'FF60 2D42

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	RECST	TRMST	BOST	EPST	SLPST	HLTST	RSTST	EST	TABST	FMLST	NMLST	TFST	RFST	SDST	NDST
After Reset	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15	—	Reserved	This bit is read as 0. The write value should be 0.	R
b14	RECST	Receive Status Flag (receiver)	0: Bus idle or transmission in progress 1: Reception in progress	R
b13	TRMST	Transmit Status Flag (transmitter)	0: Bus idle or reception in progress 1: Transmission in progress or in bus-off state	R
b12	BOST	Bus-Off Status Flag	0: Not in bus-off state 1: In bus-off state	R
b11	EPST	Error-Passive Status Flag	0: Not in error-passive state 1: In error-passive state	R
b10	SLPST	CAN Sleep Status Flag	0: Not in CAN sleep mode 1: In CAN sleep mode	R
b9	HLTST	CAN Halt Status Flag	0: Not in CAN halt mode 1: In CAN halt mode	R
b8	RSTST	CAN Reset Status Flag	0: Not in CAN reset mode 1: In CAN reset mode	R
b7	EST	Error Status Flag	0: No error occurred 1: Error occurred	R
b6	TABST	Transmission Abort Status Flag	0: No mailbox with TRMABT bit = 1 1: Mailbox(es) with TRMABT bit = 1	R
b5	FMLST	FIFO Mailbox Message Lost Status Flag	0: RFMLF bit = 0 1: RFMLF bit = 1	R
b4	NMLST	Normal Mailbox Message Lost Status Flag	0: No mailbox with MSGLOST bit = 1 1: Mailbox(es) with MSGLOST bit = 1	R
b3	TFST	Transmit FIFO Status Flag	0: Transmit FIFO is full 1: Transmit FIFO is not full	R
b2	RFST	Receive FIFO Status Flag	0: No message in receive FIFO 1: Message in receive FIFO	R
b1	SDST	SENTDATA Status Flag	0: No mailbox with SENTDATA bit = 1 1: Mailbox(es) with SENTDATA bit = 1	R
b0	NDST	NEWDATA Status Flag	0: No mailbox with NEWDATA bit = 1 1: Mailbox(es) with NEWDATA bit = 1	R

RECST Bit

The RECST bit is set to 1 when the CAN module performs as a receiver node. The RECST bit is set to 0 when the CAN module performs as a transmitter node or is in bus-idle state.

TRMST Bit

The TRMST bit is set to 1 when the CAN module performs as a transmitter node or is in the bus-off state. The TRMST bit is set to 0 when the CAN module performs as a receiver node or is in bus-idle state.

BOST Bit

The BOST bit is set to 1 when the value of the CiTECR register exceeds 255 and the CAN module is in the bus-off state (TEC ≥ 256). The BOST bit is set to 0 when the CAN module is not in the bus-off state.

EPST Bit

The EPST bit is set to 1 when the value of the CiTECR or CiRECR register exceeds 127 and the CAN module is in error-passive state (128 ≤ TEC < 256 or 128 ≤ REC < 256). The EPST bit is set to 0 when the CAN module is not in the error-passive state.
TEC indicates the value of the transmit error counter (CiTECR register) and REC indicates the value of the receive error counter (CiRECR register).

SLPST Bit

The SLPST bit is set to 1 when the CAN module is in CAN sleep mode. The SLPST bit is set to 0 when the CAN module is not in CAN sleep mode.

HLTST Bit

The HLTST bit is set to 1 when the CAN module is in CAN halt mode. The HLTST bit is set to 0 when the CAN module is not in CAN halt mode. Even when the state is changed from CAN halt mode to CAN sleep mode, the HLTST bit remains 1.

RSTST Bit

The RSTST bit is set to 1 when the CAN module is in CAN reset mode. The RSTST bit is 0 when the CAN module is not in CAN reset mode. Even when the state is changed from CAN reset mode to CAN sleep mode, the RSTST bit remains 1.

EST Bit

The EST bit is 1 when at least one error is detected by the CiEIFR register regardless of the value of the CiEIER register. The EST bit is set to 0 when no error is detected by the CiEIFR register.

TABST Bit

The TABST bit is set to 1 when at least one TRMABT bit in the CiMCTLj register is 1 regardless of the value of the CiMIER register.
The TABST bit is set to 0 when all TRMABT bits are 0.

FMLST Bit

The FMLST bit is set to 1 when the RFMLF bit in the CiRFCR register is 1 regardless of the value of the CiMIER register. The FMLST bit is set to 0 when the RFMLF bit is 0.

NMLST Bit

The NMLST bit is set to 1 when at least one MSGLOST bit in the CiMCTLj register is 1 regardless of the value of the CiMIER register. The NMLST bit is set to 0 when all MSGLOST bits are 0.

TFST Bit

The TFST bit is set to 1 when the transmit FIFO is not full. The TFST bit is set to 0 when the transmit FIFO is full. The TFST bit is set to 0 when normal mailbox mode is selected.

RFST Bit

The RFST bit is set to 1 when the receive FIFO is not empty. The RFST bit is set to 0 when the receive FIFO is empty. The RFST bit is set to 0 when normal mailbox mode is selected.

SDST Bit

The SDST bit is set to 1 when at least one SENTDATA bit in the CiMCTLj register (j = 32 to 63) is 1 regardless of the value of the CiMIER register. The SDST bit is set to 0 when all SENTDATA bits are 0.

NDST Bit

The NDST bit is set to 1 when at least one NEWDATA bit in the CiMCTLj register (j = 0 to 63) is 1 regardless of the value of the CiMIER register. The NDST bit is set to 0 when all NEWDATA bits are 0.

20.3.14 CANi Mailbox Search Mode Register (CiMSMR) (i = 0 to 5)

Address C0MSMR: H'FF60 0553, C1MSMR: H'FF60 0D53, C2MSMR: H'FF60 1553, C3MSMR: H'FF60 1D53, C4MSMR: H'FF60 2553, C5MSMR: H'FF60 2D53

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	MBSM[1:0]	
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b1, b0	MBSM[1:0]	Mailbox Search Mode Select Bits	b1 b0 0 0 : Receive mailbox search mode 0 1 : Transmit mailbox search mode 1 0 : Message lost search mode 1 1 : Channel search mode	R/W

Write to the CiMSMR register in CAN operation mode or CAN halt mode.

MBSM Bits

The MBSM bits select the search mode for the mailbox search function.

When the MBSM bits are 00, receive mailbox search mode is selected. In this mode, the search targets are the NEWDATA bit in the CiMCTLj register (j = 0 to 63) for the normal mailbox and the RFEST bit in the CiRFCR register.

When the MBSM bits are 01, transmit mailbox search mode is selected. In this mode, the search target is the SENTDATA bit in the CiMCTLj register.

When the MBSM bits are 10, message lost search mode is selected. In this mode, the search targets are the MSGLOST bit in the CiMCTLj register for the normal mailbox and the RFMLF bit in the CiRFCR register.

When the MBSM bits are 11, channel search mode is selected. In this mode, the search target is the CiCSSR register. Refer to section 20.3.16, CANi Channel Search Support Register (CiCSSR) (i = 0 to 5).

20.3.15 CANi Mailbox Search Status Register (CiMSSR) (i = 0 to 5)

Address C0MSSR: H'FF60 0552, C1MSSR: H'FF60 0D52, C2MSSR: H'FF60 1552, C3MSSR: H'FF60 1D52, C4MSSR: H'FF60 2552, C5MSSR: H'FF60 2D52

	b7	b6	b5	b4	b3	b2	b1	b0
	SEST	—	MBNST [5:0]					
After Reset	1	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	SEST	Search Result Status Bit	0: Search result found 1: No search result	R
b6	—	Reserved	This bit is read as 0. The write value should be 0.	R
b5 to b0	MBNST [5:0]	Search Result Mailbox Number Status Bit	The search result in the search mode selected by the CiMSMR register is output. Output number: 0 to 63	R

SEST Bit

The SEST bit is set to 1 (no search result) when no corresponding mailbox is found after searching all mailboxes. For example, in transmit mailbox search mode, the SEST bit is set to 1 when no SENTDATA bit for mailboxes is 1. The SEST bit is set to 0 when at least one SENTDATA bit is 1. When the SEST bit is 1, the value of the MBNST bits is undefined.

MBNST Bits

The MBNST bits output the smallest mailbox number that is searched in each mode of the CiMSMR register. In receive mailbox, transmit mailbox, and message lost search modes, the value of the mailbox i.e., the search result to be output, is updated as described below:

- When the NEWDATA, SENTDATA or MSGLOST bit for the output mailbox is set to 0.
- When the NEWDATA, SENTDATA or MSGLOST bit for a higher-priority mailbox is set to 1.

In receive mailbox search and message lost search modes, the receive FIFO (mailbox [60]) is output when the receive FIFO is not empty and there are no unread received messages or no lost messages in any of the normal mailboxes (mailboxes [0] to [55]). In transmit mailbox search mode, the transmit FIFO (mailbox [56]) is not output. Table 20.13 lists the behavior of MBNST bits in FIFO mailbox mode.

In channel search mode, the MBNST bits output the corresponding channel number. After the CiMSSR register is read by a program, the next target channel number is output.

Table 20.13 Behavior of MBNST Bits in FIFO Mailbox Mode

MBSM Bits	Mailbox [56] (Transmit FIFO)	Mailbox [60] (Receive FIFO)
00	Mailbox [56] is not output.	Mailbox [60] is output when no NEWDATA bit for the normal mailbox is set to 1 (a new message is being stored or has been stored to the mailbox) and the receive FIFO is not empty.
01		Mailbox [60] is not output.
10		Mailbox [60] is output when no MSGLOST bit for the normal mailbox is set to 1 (message is overwritten or overrun) and the RFMLF bit is set to 1 (receive FIFO message lost has occurred) in the receive FIFO.
11		Mailbox [60] is not output.

20.3.16 CANi Channel Search Support Register (CiCSSR) (i = 0 to 5)

Address: C0CSSR: H'FF60 0551, C1CSSR: H'FF60 0D51, C2CSSR: H'FF60 1551, C3CSSR: H'FF60 1D51, C4CSSR: H'FF60 2551, C5CSSR: H'FF60 2D51

	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	—	—	—	—	—	—	—	—

Bit	Description	R/W
b7 to b0	When the value for the channel search is input, the channel number is output to the CiMSSR register.	W

The bits in the CiCSSR register, which are set to 1, are encoded by an 8/3 encoder (the lower bit position, the higher priority) and output to the MBNST bits in the CiMSSR register.
The CiMSSR register outputs the updated value whenever the CiMSSR register is read by a program.
Write to the CiCSSR register only when the MBSM bits in the CiMSMR register are 11 (channel search mode). Write to this register in CAN operation mode or CAN halt mode.

Figure 20.4 shows the write and read of registers CiCSSR and CiMSSR.

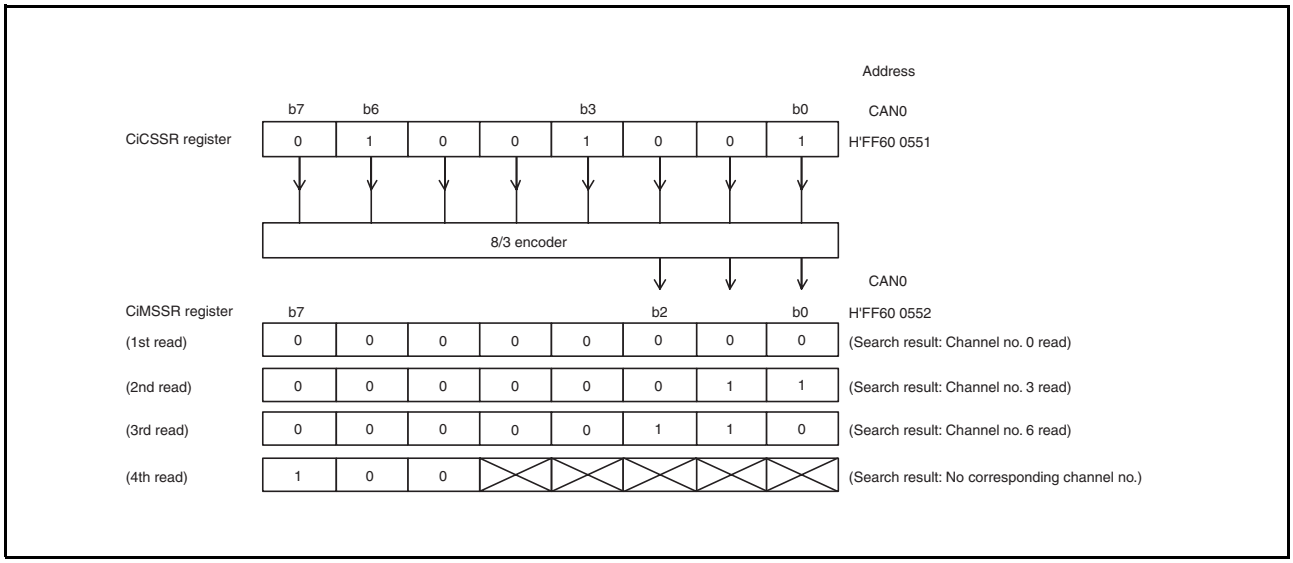


Figure 20.4 Write and Read of Registers CiCSSR and CiMSSR (i = 0 to 5)

The value of the CiCSSR register is also updated whenever the CiMSSR register is read. When the CiCSSR register is read, the value before the 8/3 encoder conversion is read.

20.3.17 CANi Acceptance Filter Support Register (CiAFSR) (i = 0 to 5)

Address C0AFSR: H'FF60 0556, C1AFSR: H'FF60 0D56, C2AFSR: H'FF60 1556, C3AFSR: H'FF60 1D56, C4AFSR: H'FF60 2556, C5AFSR: H'FF60 2D56

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Bit	Description	R/W
b15 to b0	After the standard ID of a received message is written, the value converted for data table search can be read.	R/W

The acceptance filter support unit (ASU) can be used for data table (8 bits × 256) search. In the data table, all standard IDs created by the user are set to be valid/invalid in bit units. When the CiAFSR register is written with the 16-bit unit data including the SID bits in the CiMBj register (j = 0 to 63), in which a received standard ID is stored, a decoded row (byte offset) position and column (bit) position for data table search can be read. The ASU can be used for standard (11-bit) IDs only.

The ASU is enabled in the following cases:

- When the ID to receive cannot be masked by the acceptance filter.

(Example) IDs to receive: H'078, H'087, H'111

- When there are too many IDs to receive and software filtering time is expected to be shortened.

Write to the CiAFSR register in CAN operation mode or CAN halt mode.

Figure 20.5 shows the write and read of the CiAFSR register.

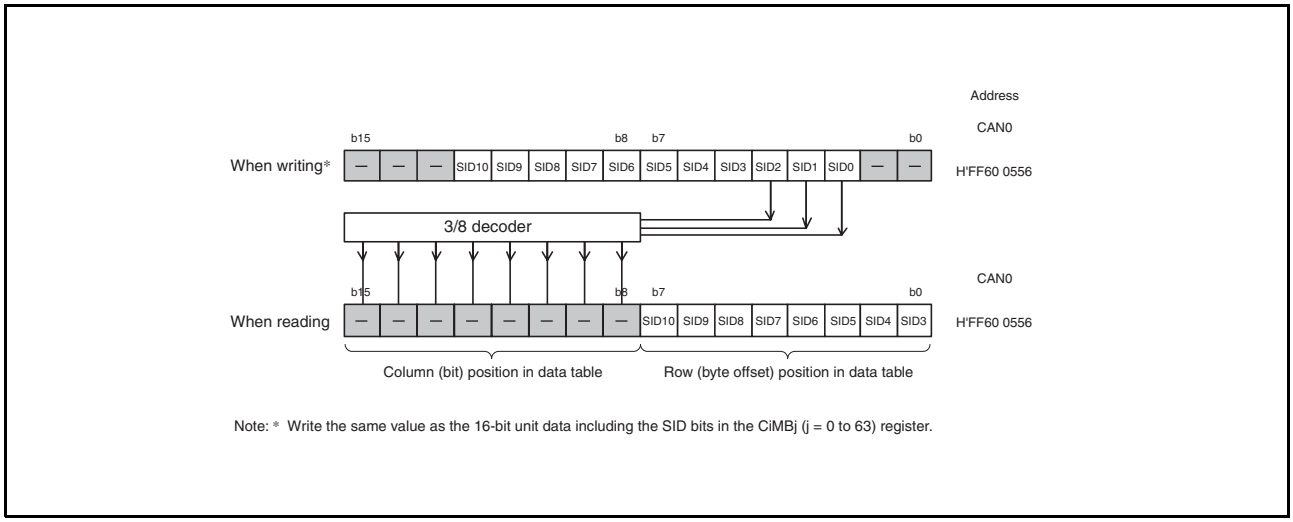


Figure 20.5 Write and Read of CiAFSR Register (i = 0 to 5)

20.3.18 CANi Error Interrupt Enable Register (CiEIER) (i = 0 to 5)

Address C0EIER: H'FF60 054C, C1EIER: H'FF60 0D4C, C2EIER: H'FF60 154C, C3EIER: H'FF60 1D4C, C4EIER: H'FF60 254C, C5EIER: H'FF60 2D4C

	b7	b6	b5	b4	b3	b2	b1	b0
	BLIE	OLIE	ORIE	BORIE	BOEIE	EPIE	EWIE	BEIE
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	BLIE	Bus Lock Interrupt Enable Bit	0: Bus lock interrupt disabled 1: Bus lock interrupt enabled	R/W
b6	OLIE	Overload Frame Transmit Interrupt Enable Bit	0: Overload frame transmit interrupt disabled 1: Overload frame transmit interrupt enabled	R/W
b5	ORIE	Receive Overrun Interrupt Enable Bit	0: Receive overrun interrupt disabled 1: Receive overrun interrupt enabled	R/W
b4	BORIE	Bus-Off Recovery Interrupt Enable Bit	0: Bus-off recovery interrupt disabled 1: Bus-off recovery interrupt enabled	R/W
b3	BOEIE	Bus-Off Entry Interrupt Enable Bit	0: Bus-off entry interrupt disabled 1: Bus-off entry interrupt enabled	R/W
b2	EPIE	Error-Passive Interrupt Enable Bit	0: Error-passive interrupt disabled 1: Error-passive interrupt enabled	R/W
b1	EWIE	Error-Warning Interrupt Enable Bit	0: Error-warning interrupt disabled 1: Error-warning interrupt enabled	R/W
b0	BEIE	Bus Error Interrupt Enable Bit	0: Bus error interrupt disabled 1: Bus error interrupt enabled	R/W

The CiEIER register is used to set the error interrupt enabled/disabled individually for each error interrupt source in the CiEIFR register.
Write to the CiEIER register in CAN reset mode.

BLIE Bit

When the BLIE bit is 0, no error interrupt request is generated even if the BLIF bit in the CiEIFR register is set to 1.
When the BLIE bit is 1, an error interrupt request is generated if the BLIF bit is set to 1.

OLIE Bit

When the OLIE bit is 0, no error interrupt request is generated even if the OLIF bit in the CiEIFR register is set to 1.
When the OLIE bit is 1, an error interrupt request is generated if the OLIF bit is set to 1.

ORIE Bit

When the ORIE bit is 0, an error interrupt request is not generated even if the ORIF bit in the CiEIFR register is set to 1.
When the ORIE bit is 1, an error interrupt request is generated if the ORIF bit is set to 1.

BORIE Bit

When the BORIE bit is 0, an error interrupt request is not generated even if the BORIF bit in the CiEIFR register is set to 1. When the BORIE bit is set to 1, an error interrupt request is generated if the BORIF bit is set to 1.

BOEIE Bit

When the BOEIE bit is 0, no error interrupt request is generated even if the BOEIF bit in the CiEIFR register is set to 1.
When the BOEIE bit is 1, an error interrupt request is generated if the BOEIF bit is set to 1.

EPIE Bit

When the EPIE bit is 0, no error interrupt request is generated even if the EPIF bit in the CiEIFR register is set to 1.
When the EPIE bit is 1, an error interrupt request is generated if the EPIF bit is set to 1.

EWIE Bit

When the EWIE bit is 0, no error interrupt request is generated even if the EWIF bit in the CiEIFR register is set to 1.
When the EWIE bit is 1, an error interrupt request is generated if the EWIF bit is set to 1.

BEIE Bit

When the BEIE bit is 0, no error interrupt request is generated even if the BEIF bit in the CiEIFR register is set to 1.
When the BEIE bit is 1, an error interrupt request is generated if the BEIF bit is set to 1.

20.3.19 CANi Error Interrupt Factor Judge Register (CiEIFR) (i = 0 to 5)

Address C0EIFR: H'FF60 054D, C1EIFR: H'FF60 0D4D, C2EIFR: H'FF60 154D, C3EIFR: H'FF60 1D4D, C4EIFR: H'FF60 254D, C5EIFR: H'FF60 2D4D

	b7	b6	b5	b4	b3	b2	b1	b0
	BLIF	OLIF	ORIF	BORIF	BOEIF	EPIF	EWIF	BEIF
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	BLIF	Bus Lock Detect Flag	0: No bus lock detected 1: Bus lock detected	R/W
b6	OLIF	Overload Frame Transmission Detect Flag	0: No overload frame transmission detected 1: Overload frame transmission detected	R/W
b5	ORIF	Receive Overrun Detect Flag	0: No receive overrun detected 1: Receive overrun detected	R/W
b4	BORIF	Bus-Off Recovery Detect Flag	0: No bus-off recovery detected 1: Bus-off recovery detected	R/W
b3	BOEIF	Bus-Off Entry Detect Flag	0: No bus-off entry detected 1: Bus-off entry detected	R/W
b2	EPIF	Error Passive Detect Flag	0: No error passive detected 1: Error passive detected	R/W
b1	EWIF	Error Warning Detect Flag	0: No error warning detected 1: Error warning detected	R/W
b0	BEIF	Bus Error Detect Flag	0: No bus error detected 1: Bus error detected	R/W

If an event corresponding to each bit occurs, the corresponding bit in the CiEIFR register is set to 1 regardless of the setting of the CiEIER register.

To set each bit to 0, write 0 by a program. If the set timing occurs simultaneously with the clear timing by the program, the bit becomes 1.

When writing 0 to a single bit by a program, use the MOV instruction to ensure that only the specified bit is set to 0 and the other bits are set to 1. Writing 1 has no effect to these bit values.

BLIF Bit

The BLIF bit is set to 1 if 32 consecutive dominant bits are detected on the CAN bus while the CAN module is in CAN operation mode.

After the bit is set to 1, redetection takes place under either of the following conditions:

- After this bit is set to 0 from 1, recessive bits are detected.
- After this bit is set to 0 from 1, the CAN module enters CAN reset mode or CAN halt mode and then enters CAN operation mode again.

OLIF Bit

The OLIF bit is set to 1 if the transmitting condition of an overload frame is detected when the CAN module performs transmission or reception.

ORIF Bit

The ORIF bit is set to 1 when a receive overrun occurs.
This bit is not to set to 1 in overwrite mode. In overwrite mode, a reception complete interrupt request is generated if an overwrite condition occurs and the ORIF bit is not set to 1.
In normal mailbox mode, if an overrun occurs in any of mailboxes [0] to [63] in overrun mode, this bit is set to 1.
In FIFO mailbox mode, if an overrun occurs in any of mailboxes [0] to [55] or the receive FIFO in overrun mode, this bit is set to 1.

BORIF Bit

The BORIF bit is set to 1 when the CAN module recovers from the bus-off state normally by detecting 11 consecutive recessive bits 128 times in the following conditions:

- When the BOM bits in the CiCTLR register are 00.
- When the BOM bits are 10.
- When the BOM bits are 11.

The BORIF bit is not set to 1 if the CAN module recovers from the bus-off state in the following conditions:

- When the CANM bits in the CiCTLR register are set to 01 or 11 (CAN reset mode).
- When the RBOC bit in the CiCTLR register is set to 1 (forcible return from bus-off).
- When the BOM bits are 01.
- When the BOM bits are 11 and the CANM bits are set to 10 (CAN halt mode) before normal recovery occurs.

Table 20.14 lists the behavior of bits BOEIF and BORIF according to BOM bit setting value.

Table 20.14 Behavior of Bits BOEIF and BORIF according to BOM Bit Setting Value

BOM Bits	BOEIF Bit	BORIF Bit
00	Set to 1 on entry to the bus-off state.	Set to 1 on exit from the bus-off state.
01		Do not set to 1.
10		Set to 1 on exit from the bus-off state.
11		Set to 1 if normal bus-off recovery occurs before the CANM bits are set to 10 (CAN halt mode).

BOEIF Bit

The BOEIF bit is set to 1 when the CAN error state becomes bus-off (the TEC (transmit error counter) value exceeds 255).
The BOEIF bit is also set to 1 when the BOM bits in the CiCTLR register are 01 (entry to CAN halt mode automatically at bus-off entry) and the CAN module becomes the bus-off state.

EPIF Bit

The EPIF bit is set to 1 when the CAN error state becomes error-passive (the REC (receive error counter) or TEC value exceeds 127).
The EPIF bit is set to 1 only when the REC or TEC initially exceeds 127. Thus, if 0 is written to the EPIF bit by a program while the REC or TEC remains greater than 127, the EPIF bit is not set to 1 until the REC and TEC go below 127 and then the REC or TEC exceeds 127 again.

EWIF Bit

The EWIF bit is set to 1 when the value of the receive error counter (REC) or transmit error counter (TEC) exceeds 95. The EWIF bit is set to 1 only when the REC or TEC initially exceeds 95. Thus, if 0 is written to the EWIF bit by a program while the REC or TEC remains greater than 95, the EWIF bit is not set to 1 until the REC and TEC go below 95 and then the REC or TEC exceeds 95 again.

BEIF Bit

The BEIF bit is set to 1 when a bus error is detected.

20.3.20 CANi Receive Error Count Register (CiRECR) (i = 0 to 5)

Address C0RECR: H'FF60 054E, C1RECR: H'FF60 0D4E, C2RECR: H'FF60 154E, C3RECR: H'FF60 1D4E, C4RECR: H'FF60 254E, C5RECR: H'FF60 2D4E

	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	0	0	0	0	0	0	0	0

Bit	Description	R/W
b7 to b0	Receive Error Count Function The CiRECR register increments or decrements the counter value according to error status of the CAN module during reception.	R

The CiRECR register indicates the value of the receive error counter.
Refer to the CAN Specifications (ISO11898-1) about the increment/decrement conditions of the receive error counter.
The value in bus-off state is undefined.

20.3.21 CANi Transmit Error Count Register (CiTECR) (i = 0 to 5)

Address C0TECR: H'FF60 054F, C1TECR: H'FF60 0D4F, C2TECR: H'FF60 154F, C3TECR: H'FF60 1D4F, C4TECR: H'FF60 254F, C5TECR: H'FF60 2D4F

	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	0	0	0	0	0	0	0	0

Bit	Description	R/W
b7 to b0	Transmit Error Count Function The CiTECR register increments or decrements the counter value according to error status of the CAN module during transmission.	R

The CiTECR register indicates the value of the transmit error counter.
Refer to the CAN Specifications (ISO11898-1) about the increment/decrement conditions of the transmit error counter.
The value in bus-off state is undefined.

20.3.22 CANi Error Code Store Register (CiECSR) (i = 0 to 5)

Address C0ECSR: H'FF60 0550, C1ECSR: H'FF60 0D50, C2ECSR: H'FF60 1550, C3ECSR: H'FF60 1D50, C4ECSR: H'FF60 2550, C5ECSR: H'FF60 2D50

	b7	b6	b5	b4	b3	b2	b1	b0
	EDPM	ADEF	BE0F	BE1F	CEF	AEF	FEF	SEF
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	EDPM	Error Display Mode Select Bit*1*2	0: Output of first detected error code 1: Output of accumulated error code	R/W
b6	ADEF	ACK Delimiter Error Flag*3*4	0: No ACK delimiter error detected 1: ACK delimiter error detected	R/W
b5	BE0F	Bit Error (dominant) Flag*3*4	0: No bit error (dominant) detected 1: Bit error (dominant) detected	R/W
b4	BE1F	Bit Error (recessive) Flag*3*4	0: No bit error (recessive) detected 1: Bit error (recessive) detected	R/W
b3	CEF	CRC Error Flag*3*4	0: No CRC error detected 1: CRC error detected	R/W
b2	AEF	ACK Error Flag*3*4	0: No ACK error detected 1: ACK error detected	R/W
b1	FEF	Form Error Flag*3*4	0: No form error detected 1: Form error detected	R/W
b0	SEF	Stuff Error Flag*3*4	0: No stuff error detected 1: Stuff error detected	R/W

Notes: 1. Write to the EDPM bit in CAN reset mode or CAN halt mode.
2. If more than one error condition is detected simultaneously, all related bits are set to 1.
3. Writing 1 has no effect to these bit values.
4. When writing 0 to bits SEF, FEF, AEF, CEF, BE1F, BE0F, and ADEF by a program, use the MOV instruction to ensure that only the specified bit is set to 0 and the other bits are set to 1.

The CiECSR register can be used to monitor whether an error has occurred on the CAN bus. Refer to the CAN Specifications (ISO11898-1) to check the generation conditions of each error.

To set each bit except the EDPM bit to 0, write 0 by a program. If the timing at which each bit is set to 1 and the timing at which 0 is written by a program are the same, the relevant bit is set to 1.

EDPM Bit

The EDPM bit selects the output mode of the CiECSR register.

When the EDPM bit is set to 0, the CiECSR register outputs the first error code.

When the EDPM bit is set to 1, the CiECSR register outputs the accumulated error code.

ADEF Bit

The ADEF bit is set to 1 when a form error is detected with the ACK delimiter during transmission.

BE0F Bit

The BE0F bit is set to 1 when a dominant bit error is detected.

BE1F Bit

The BE1F bit is set to 1 when a recessive bit error is detected.

CEF Bit

The CEF bit is set to 1 when a CRC error is detected.

AEF Bit

The AEF bit is set to 1 when an ACK error is detected.

FEF Bit

The FEF bit is set to 1 when a form error is detected.

SEF Bit

The SEF bit is set to 1 when a stuff error is detected.

20.3.23 CANi Time Stamp Register (CiTSR) (i = 0 to 5)

Address C0TSR: H'FF60 0554, C1TSR: H'FF60 0D54, C2TSR: H'FF60 1554, C3TSR: H'FF60 1D54, C4TSR: H'FF60 2554, C5TSR: H'FF60 2D54

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Description	R/W
b15 to b0	Free-running counter value for the time stamp function	R

When the CiTSR register is read, the value of the time stamp counter (16-bit free-running counter) at that moment is read.

The value of the time stamp counter reference clock is a multiple of 1 bit time, as configured by the TSPS bits in the CiCTLR register.

The time stamp counter stops in CAN sleep mode and CAN halt mode, and is initialized in CAN reset mode.

The time stamp counter value is stored to TSL and TSH in the CiMBj register when a received message is stored in a receive mailbox.

The CiTSR register should be read in 16-bit units.

20.3.24 CANi Test Control Register (CiTCR) (i = 0 to 5)

Address C0TCR: H'FF60 0558, C1TCR: H'FF60 0D58, C2TCR: H'FF60 1558, C3TCR: H'FF60 1D58, C4TCR: H'FF60 2558, C5TCR: H'FF60 2D58

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	TSTM[1:0]		TSTE
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R
b2, b1	TSTM[1:0]	CAN Test Mode Select Bits	b2 b1 0 0 : Other than CAN test mode 0 1 : Listen-only mode 1 0 : Self-test mode 0 (external loop back) 1 1 : Self-test mode 1 (internal loop back)	R/W
b0	TSTE	CAN Test Mode Enable Bit	0: CAN test mode disabled 1: CAN test mode enabled	R/W

Write to the CiTCR register in CAN halt mode only.

TSTM Bits

The TSTM bits select the CAN test mode. For details on the CAN test modes, see section 20.3.24 (1), Listen-Only Mode, section 20.3.24 (2), Self-Test Mode 0 (External Loop Back), and section 20.3.24 (3), Self-Test Mode 1 (Internal Loop Back).

TSTE Bit

When the TSTE bit is set to 0, CAN test mode is disabled.
When the TSTE bit is set to 1, CAN test mode is enabled.

(1) Listen-Only Mode

The ISO 11898-1 recommends an optional bus monitoring mode. In listen-only mode, the CAN node is able to receive valid data frames and valid remote frames. It sends only recessive bits on the CAN bus, and the protocol controller is not required to send the ACK bit, overload flag, or active error flag.
Listen-only mode can be used for baud rate detection.
Do not request transmission from any mailboxes in this mode.
Figure 20.6 shows the connection when listen-only mode is selected.

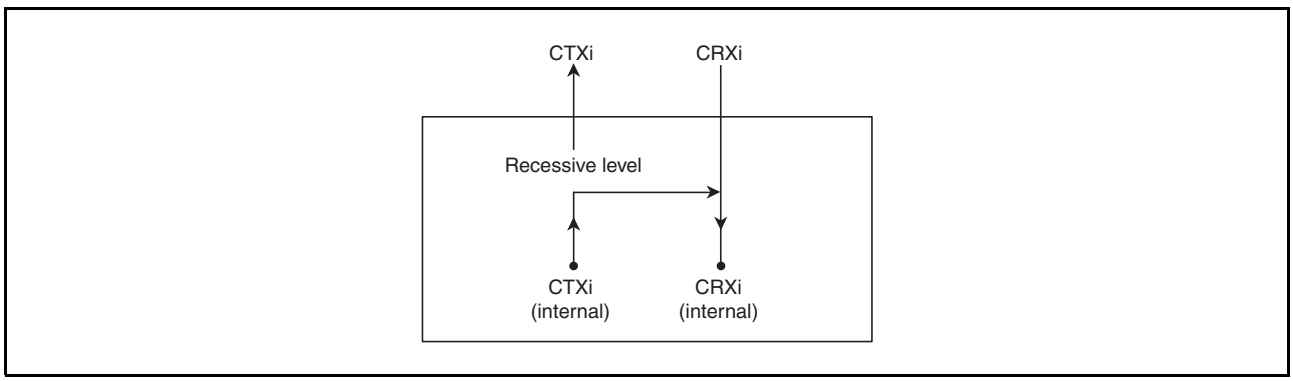


Figure 20.6 Connection when Listen-Only Mode is Selected

(2) Self-Test Mode 0 (External Loop Back)

Self-test mode 0 is provided for CAN transceiver tests.

In this mode, the protocol controller treats its own transmitted messages as messages received via the CAN transceiver and stores them into the receive mailbox. To be independent from external stimulation, the protocol controller generates the ACK bit.

Connect the CTXi/CRXi pins to the CAN transceiver.

Figure 20.7 shows the connection when self-test mode 0 is selected.

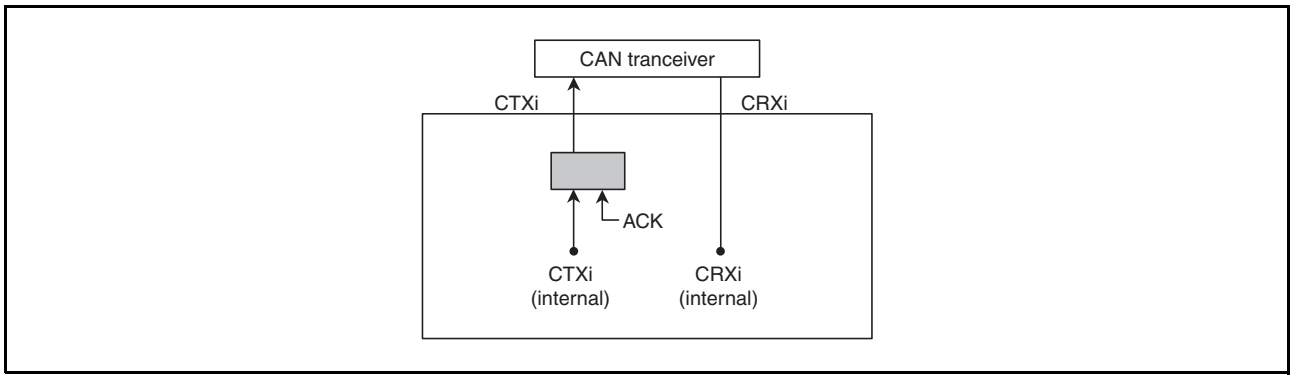


Figure 20.7 Connection when Self-Test Mode 0 is Selected

(3) Self-Test Mode 1 (Internal Loop Back)

Self-test mode 1 is provided for self-test functions.

In this mode, the protocol controller treats its transmitted messages as received messages and stores them into the receive mailbox. To be independent from external stimulation, the protocol controller generates the ACK bit.

In self-test mode 1, the protocol controller performs an internal feedback from the internal CTXi pin to the internal CRXi pin. The input value of the external CRXi pin is ignored. The external CTXi pin outputs only recessive bits. The CTXi/CRXi pins do not need to be connected to the CAN bus or any external device.

Figure 20.8 shows the connection when self-test mode 1 is selected.

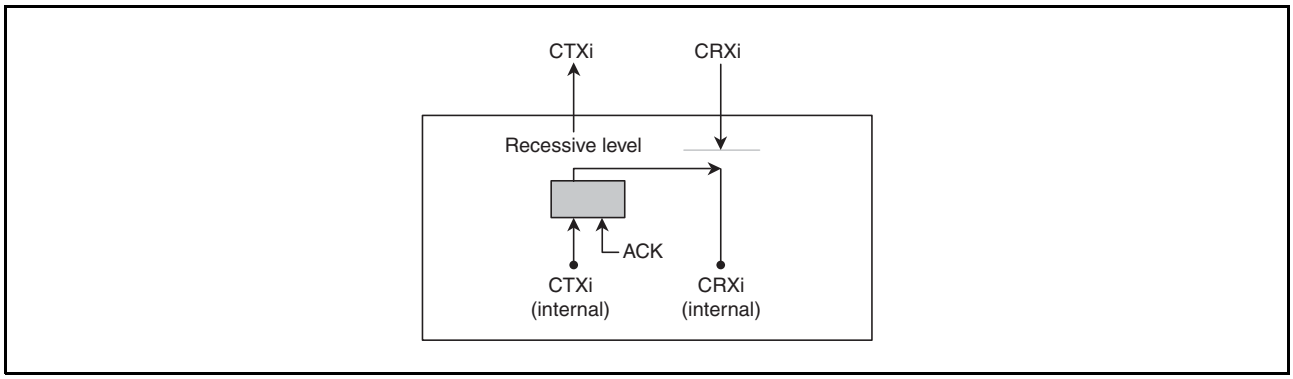


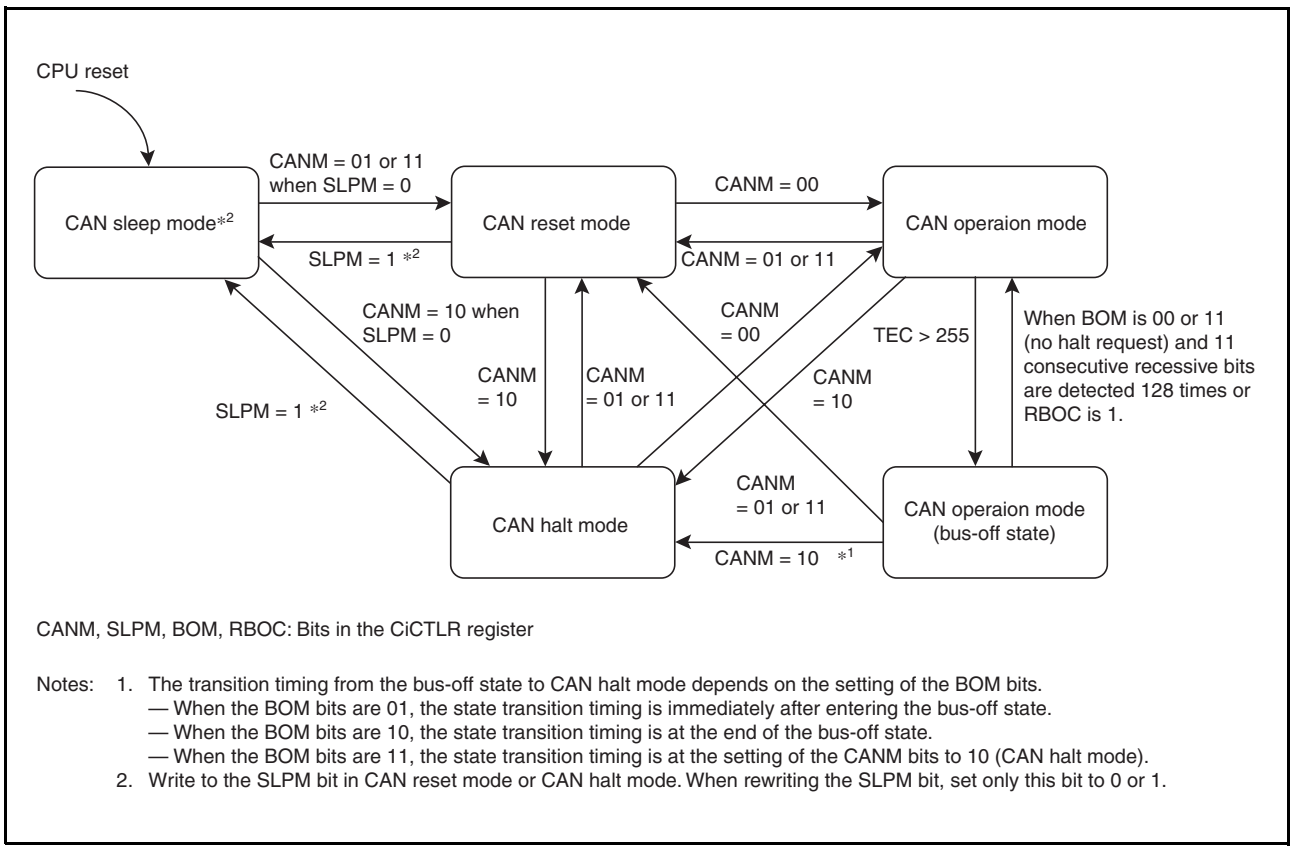
Figure 20.8 Connection when Self-Test Mode 1 is Selected

20.4 Operating Mode

The CAN module has the following four operating modes.

- CAN reset mode
- CAN halt mode
- CAN operation mode
- CAN sleep mode

Figure 20.9 shows the transition between CAN operating modes.



Notes: 1. The transition timing from the bus-off state to CAN halt mode depends on the setting of the BOM bits.
— When the BOM bits are 01, the state transition timing is immediately after entering the bus-off state.
— When the BOM bits are 10, the state transition timing is at the end of the bus-off state.
— When the BOM bits are 11, the state transition timing is at the setting of the CANM bits to 10 (CAN halt mode).
2. Write to the SLPM bit in CAN reset mode or CAN halt mode. When rewriting the SLPM bit, set only this bit to 0 or 1.

Figure 20.9 Transition between CAN Operating Modes (i = 0 to 5)

20.4.1 CAN Reset Mode

CAN reset mode is provided for CAN communication configuration.

When the CANM bits in the CiCTLR register are set to 01 or 11, the CAN module enters CAN reset mode. Then the RSTST bit in the CiSTR register is set to 1. Do not change the CANM bits until the RSTST bit is set to 1. Configure the CiBCR register before exiting CAN reset mode to any other modes.

The following registers are initialized to their reset values after entering CAN reset mode and their initialized values are retained during CAN reset mode:

- CiMCTLj register
- CiSTR register (except bits SLPST and TFST)
- CiEIFR register
- CiRECR register
- CiTECR register
- CiTSR register
- CiMSSR register
- CiMSMR register
- CiRFCR register
- CiTFCR register
- CiTCR register
- CiECSR register (except EDPM bit)

The following registers retain their values after entering CAN reset mode.

- CiCTLR register
- CiSTR register (bits SLPST and TFST)
- Registers CiMIER0 and CiMIER1
- CiEIER register
- CiBCR register
- CiCSSR register
- CiECSR register (EDPM bit only)
- CiMBj register
- Registers CiMKR0 to CiMKR9
- Registers CiFIDCR0 and CiFIDCR1
- Registers CiMKIVLR0 and CiMKIVLR1
- CiAFSR register
- CiRFPCR register
- CiTFPCR register

20.4.2 CAN Halt Mode

CAN halt mode is used for mailbox configuration and test mode setting.

When the CANM bits in the CiCTLR register are set to 10, CAN halt mode is selected. Then the HLTST bit in the CiSTR register is set to 1. Do not change the CANM bits until the HLTST bit is set to 1.

Refer to Table 20.15, Operation in CAN Reset Mode and CAN Halt Mode regarding the state transition conditions when transmitting or receiving.

All registers except bits RSTST, HLTST, and SLPST in the CiSTR register remain unchanged when the CAN module enters CAN halt mode.

Do not change registers CiCTLR (except bits CANM and SLPM) and CiEIER in CAN halt mode. The CiBCR register can be changed in CAN halt mode only when listen-only mode is selected to use for automatic baud rate detection.

Table 20.15 Operation in CAN Reset Mode and CAN Halt Mode

Mode	Receiver	Transmitter	Bus-Off
CAN reset mode (forcible transition) CANM = 11	CAN module enters CAN reset mode without waiting for the end of message reception.	CAN module enters CAN reset mode without waiting for the end of message transmission.	CAN module enters CAN reset mode without waiting for the end of bus-off recovery.
CAN reset mode CANM = 01	CAN module enters CAN reset mode without waiting for the end of message reception.	CAN module enters CAN reset mode after waiting for the end of message transmission*1*4.	CAN module enters CAN reset mode without waiting for the end of bus-off recovery.
CAN halt mode	CAN module enters CAN halt mode after waiting for the end of message reception*2*3.	CAN module enters CAN halt mode after waiting for the end of message transmission*1*4.	[When the BOM bits are 00] A halt request from a program will be acknowledged only after bus-off recovery. [When the BOM bits are 01] CAN module enters automatically CAN halt mode without waiting for the end of bus-off recovery (regardless of a halt request from a program). [When the BOM bits are 10] CAN module enters automatically CAN halt mode after waiting for the end of bus-off recovery (regardless of a halt request from a program). [When the BOM bits are 11] CAN module enters CAN halt mode (without waiting for the end of bus-off recovery) if a halt is requested by a program during bus-off.

Notes: BOM bits: Bits in the CiCTLR register (i = 0 to 5)

1. If several messages are requested to be transmitted, mode transition occurs after the completion of the first transmission. In a case that the CAN reset mode is being requested during suspend transmission, mode transition occurs when the bus is idle, the next transmission ends, or the CAN module becomes a receiver.
2. If the CAN bus is locked at the dominant level, the program can detect this state by monitoring the BLIF bit in the CiEIFR register.
3. If a CAN bus error occurs during reception after CAN halt mode is requested, the CAN mode transits to CAN halt mode.
4. If a CAN bus error or arbitration lost occurs during transmission after CAN reset mode or CAN halt mode is requested, the CAN mode transits to the requested CAN mode.

20.4.3 CAN Sleep Mode

CAN sleep mode is used for reducing current consumption by stopping the clock supply to the CAN module. After MCU hardware reset or software reset, the CAN module starts from CAN sleep mode.

When the SLPM bit in the CiCTLR register is set to 1, the CAN module enters CAN sleep mode. Then the SLPST bit in the CiSTR register is set to 1. Do not change the value of the SLPM bit until the SLPST bit is set to 1. The other registers remain unchanged when the MCU enters CAN sleep mode.

Write to the SLPM bit in CAN reset mode and CAN halt mode. Do not change any registers (except the SLPM bit) during CAN sleep mode. Read operation is still allowed.

When the SLPM bit is set to 0, the CAN module is released from CAN sleep mode. When the CAN module exits CAN sleep mode, the other registers remain unchanged.

20.4.4 CAN Operation Mode (Excluding Bus-Off State)

CAN operation mode is used for CAN communication.

When the CANM bits in the CiCTLR register are set to 00, the CAN module enters CAN operation mode.

Then bits RSTST and HLTST in the CiSTR register are set to 0. Do not change the value of the CANM bits until these bits are set to 0.

If 11 consecutive recessive bits are detected after entering CAN operation mode, the CAN module is in the following states:

- The CAN module becomes an active node on the network that enables transmission and reception of CAN messages.
- Error monitoring of the CAN bus, such as receive and transmit error counters, is performed.

During CAN operation mode, the CAN module may be in one of the following three sub-modes, depending on the status of the CAN bus:

- Idle mode: Transmission or reception is not being performed.
- Receive mode: A CAN message sent by another node is being received.
- Transmit mode: A CAN message is being transmitted. The CAN module may receive its own message simultaneously when self-test mode 0 (TSTM bits in the CiTCR register = 10) or self-test mode 1 (TSTM bits = 11) is selected.

Figure 20.10 shows the sub mode in CAN operation mode.

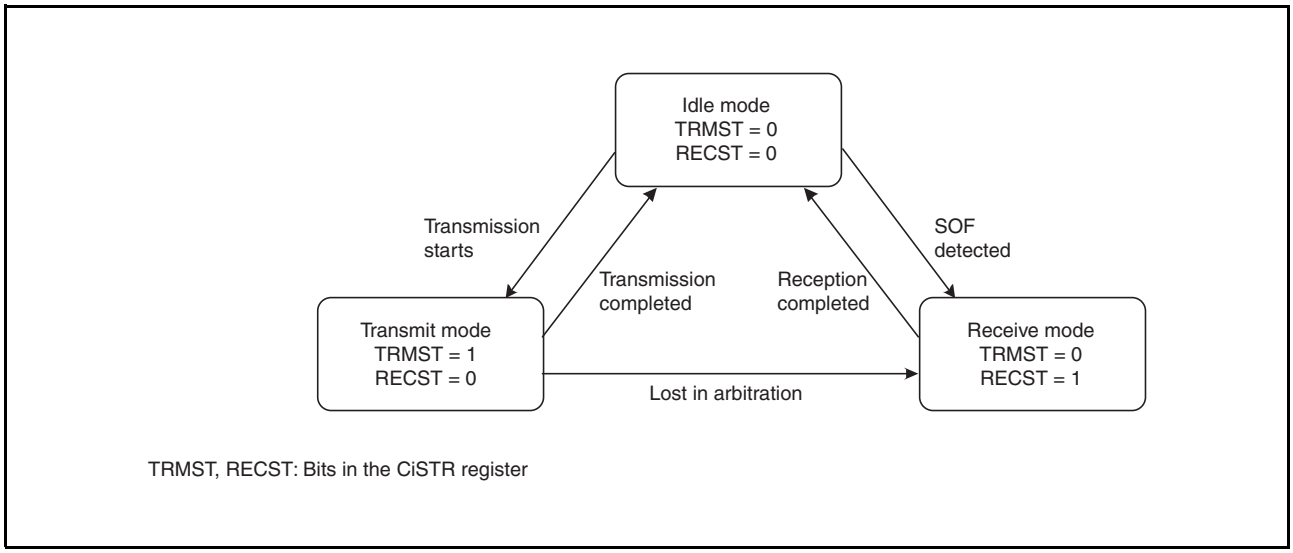


Figure 20.10 Sub Mode in CAN Operation Mode (i = 0 to 5)

20.4.5 CAN Operation Mode (Bus-Off State)

The CAN module enters the bus-off state according to the increment/decrement rules for the transmit/receive error counters in the CAN Specifications.

The following cases apply when recovering from the bus-off state. When the CAN module is in bus-off state, the values of the associated registers, except registers CiSTR, CiEIFR, CiRECR, CiTECR and CiTSR, remain unchanged.

1. When the BOM bits in the CiCTLR register are 00 (normal mode)

   The CAN module enters the error-active state after it has completed the recovery from the bus-off state and CAN communication is enabled instantly. The BORIF bit in the CiEIFR register is set to 1 (bus-off recovery detected) at this time.
2. When the RBOC bit in the CiCTLR register is set to 1 (forcible return from bus-off)

   The CAN module enters the error-active state when it is in bus-off state and the RBOC bit is set to 1. CAN communication is enabled again after 11 consecutive recessive bits are detected. The BORIF bit is not set to 1 at this time.
3. When the BOM bits are 01 (entry to CAN halt mode automatically at bus-off entry)

   The CAN module enters CAN halt mode when it reaches the bus-off state. The BORIF bit is not set to 1 at this time.
4. When the BOM bits are 10 (entry to CAN halt mode automatically at bus-off end)

   The CAN module enters CAN halt mode when it has completed the recovery from bus-off. The BORIF bit is set to 1 at this time.
5. When the BOM bits are 11 (entry to CAN halt mode by a program) and the CANM bits in the CiCTLR register are set to 10 (CAN halt mode) during the bus-off state

   The CAN module enters CAN halt mode when it is in bus-off state and the CANM bits are set to 10 (CAN halt mode). The BORIF bit is not set to 1 at this time.

   If the CANM bits are not set to 10 during bus-off, the same behavior as (1) applies.

20.5 CAN Communication Speed Configuration

The following description explains about the CAN communication speed configuration.

20.5.1 CAN Clock Configuration

This LSI has a CAN clock generator.
The CAN clock can be configured by setting the BRP bits in the CiBCR register.
Figure 20.11 shows the block diagram of CAN clock generator.

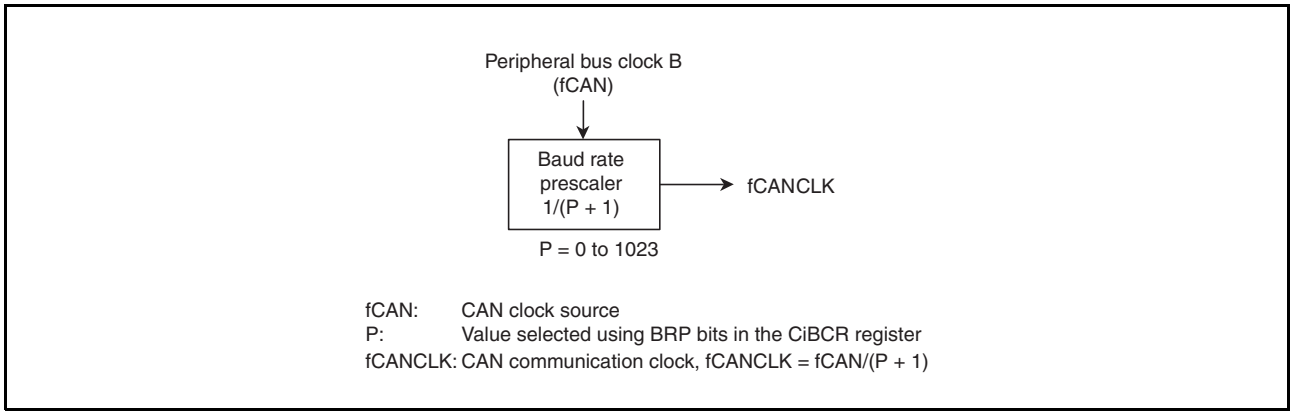


Figure 20.11 Block Diagram of CAN Clock Generator (i = 0 to 5)

20.5.2 Bit Timing Configuration

The bit time consists of the following three segments.
Figure 20.12 shows the bit timing.

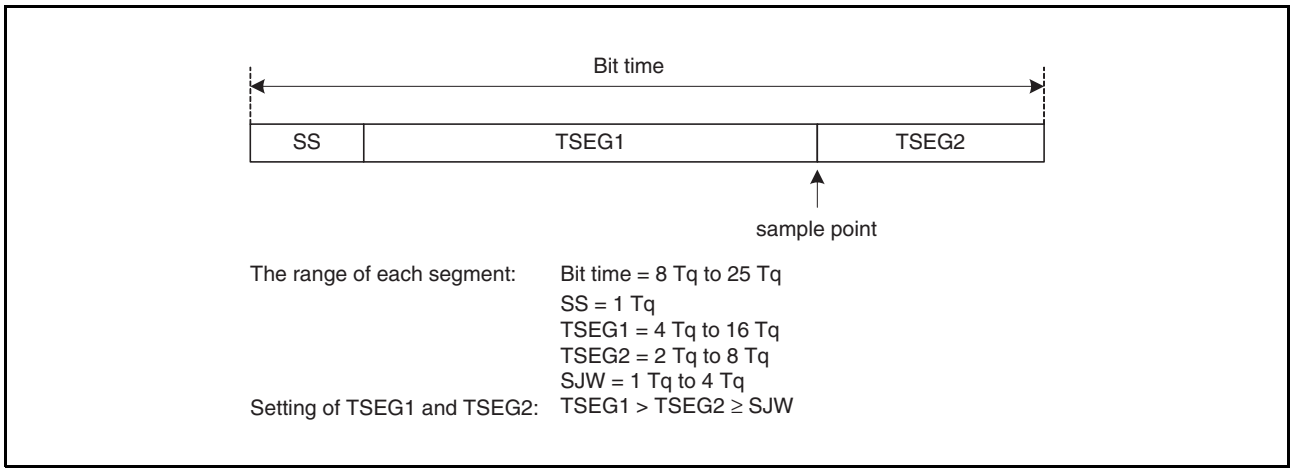


Figure 20.12 Bit Timing

20.5.3 Bit Rate

The bit rate depends on the fCAN (CAN clock source), the division value of the baud rate prescaler, and the number of Tq of one bit time.

$$\text{Bit rate [bps]} = \frac{fCAN}{\text{Baud rate prescaler division value}^* \times \text{number of Tq of one bit time}} = \frac{fCANCLK}{\text{Number of Tq of one bit time}}$$

Note: * Division value of the baud rate prescaler = P + 1 (P: 0 to 1023)
P: Setting value of the BRP bits in the CiBCR register (i = 0 to 5)

Table 20.16 lists bit rate examples.

Table 20.16 Example of Bit Rate

fCAN	40 MHz		32 MHz		24 MHz		20 MHz		16 MHz	
Bit rate	No. of Tq	P + 1	No. of Tq	P + 1	No. of Tq	P + 1	No. of Tq	P + 1	No. of Tq	P + 1
1 Mbps	10 Tq 20 Tq	4 2	8 Tq 16 Tq	4 2	8 Tq	3	10 Tq 20 Tq	2 1	8 Tq 16 Tq	2 1
500 kbps	10 Tq 20 Tq	8 4	8 Tq 16 Tq	8 4	8 Tq 16 Tq	6 3	10 Tq 20 Tq	4 2	8 Tq 16 Tq	4 2
250 kbps	10 Tq 20 Tq	16 8	8 Tq 16 Tq	16 8	8 Tq 16 Tq	12 6	10 Tq 20 Tq	8 4	8 Tq 16 Tq	8 4
83.3 kbps	8 Tq 10 Tq 16 Tq 20 Tq	60 48 30 24	8 Tq 16 Tq	48 24	8 Tq 16 Tq	36 18	8 Tq 10 Tq 16 Tq 20 Tq	30 24 15 12	8 Tq 16 Tq	24 12
33.3 kbps	8 Tq 10 Tq 20 Tq	150 120 60	8 Tq 10 Tq 16 Tq 20 Tq	120 96 60 48	8 Tq 10 Tq 16 Tq 20 Tq	90 72 45 36	8 Tq 10 Tq 20 Tq	75 60 30	8 Tq 10 Tq 16 Tq 20 Tq	60 48 30 24

20.6 Mailbox and Mask Register Structure

Figure 20.13 shows the structure of the CiMBj register.
There are 64 mailboxes with the same structure.

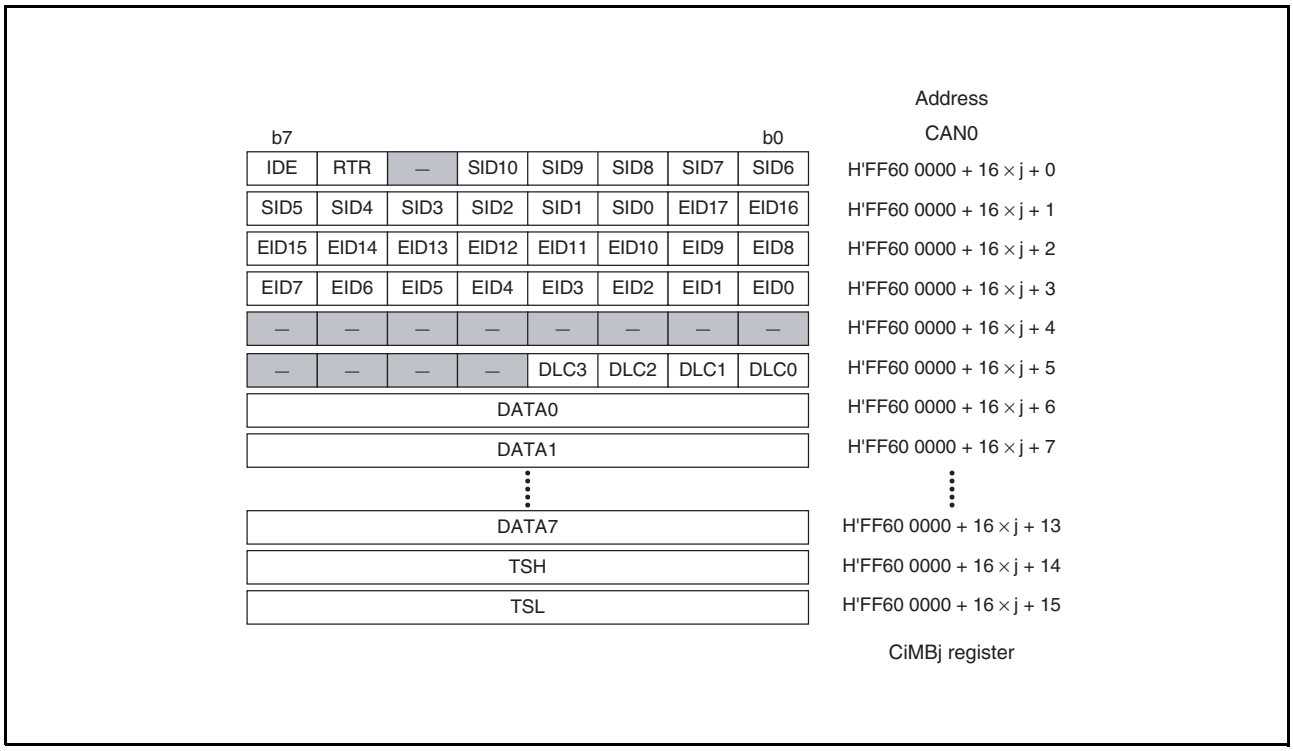


Figure 20.13 Structure of CiMBj Register (i = 0 to 5; j = 0 to 63)

Figure 20.14 shows the structure of registers CiMKR0, CiMKR1, and CiMKR2 to CiMKR9.
There are 10 mask registers with the same structure.

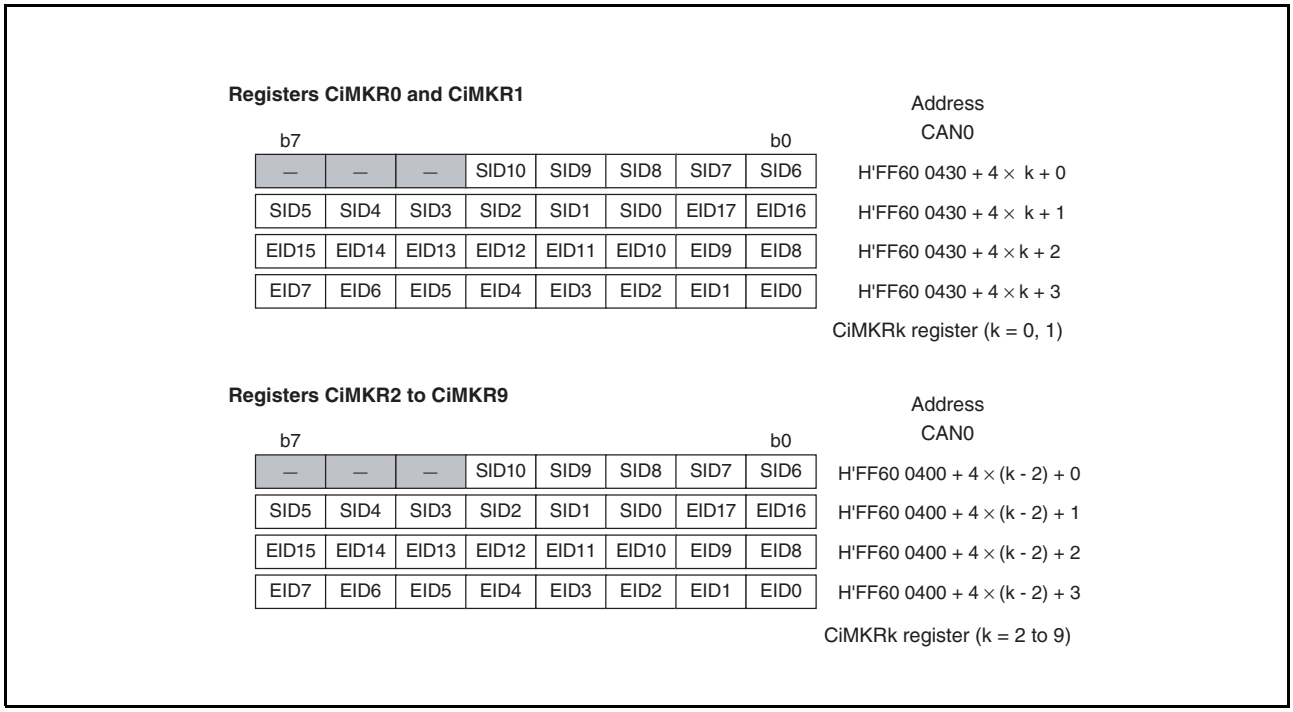


Figure 20.14 Structure of CiMKRk Register (i = 0 to 5, k = 0 to 9)

Figure 20.15 shows the structure of registers CiFIDCR0 and CiFIDCR1.
There are 2 FIFO received ID compare registers with the same structure.

b7							b0	Address CAN0
IDE	RTR	—	SID10	SID9	SID8	SID7	SID6	H'FF60 0420 + 4 × n + 0
SID5	SID4	SID3	SID2	SID1	SID0	EID17	EID16	H'FF60 0420 + 4 × n + 1
EID15	EID14	EID13	EID12	EID11	EID10	EID9	EID8	H'FF60 0420 + 4 × n + 2
EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0	H'FF60 0420 + 4 × n + 3
								CiFIDCRn register

Figure 20.15 Structure of CiFIDCRn register (i = 0 to 5; n = 0, 1)

20.7 Acceptance Filtering and Masking Function

Acceptance filtering and masking function allows the user to receive messages with a specified range of multiple IDs for mailboxes. Registers CiMKR0 to CiMKR9 can perform masking of the standard ID and the extended ID of 29 bits.

- The CiMKR0 register corresponds to mailboxes [0] to [15].
- The CiMKR1 register corresponds to mailboxes [16] to [31].
- The CiMKR2 register corresponds to mailboxes [32] to [35].
- The CiMKR3 register corresponds to mailboxes [36] to [39].
- The CiMKR4 register corresponds to mailboxes [40] to [43].
- The CiMKR5 register corresponds to mailboxes [44] to [47].
- The CiMKR6 register corresponds to mailboxes [48] to [51].
- The CiMKR7 register corresponds to mailboxes [52] to [55].
- The CiMKR8 register corresponds to mailboxes [56] to [59] in normal mailbox mode and the receive FIFO mailboxes [60] to [63] in FIFO mailbox mode.
- The CiMKR9 register corresponds to mailboxes [60] to [63] in normal mailbox mode and the receive FIFO mailboxes [60] to [63] in FIFO mailbox mode.

Registers CiMKIVLR0 and CiMKIVLR1 disable acceptance filtering individually for each mailbox.

The IDE bit in the CiMBj register is enabled when the IDFM bits in the CiCTLR register are 10 (mixed ID mode).

The RTR bit in the CiMBj register selects a data frame or a remote frame.

In FIFO mailbox mode, normal mailboxes (mailboxes [0] to [55]) use the single corresponding register among registers CiMKR0 to CiMKR7 for acceptance filtering. Receive FIFO mailboxes (mailboxes [60] to [63]) use two registers CiMKR8 and CiMKR9 for the acceptance filtering.

Also, the receive FIFO uses two registers CiFIDCR0 and CiFIDCR1 for ID comparison. Bits EID, SID, RTR, and IDE in registers CiMB60 to CiMB63 for the receive FIFO are disabled. As acceptance filtering depends on the result of two ID-mask sets, two ranges of IDs can be received into the receive FIFO.

Registers CiMKIVLR0 and CiMKIVLR1 are disabled for the receive FIFO.

If both setting of standard ID and extended ID are made in the IDE bits in registers CiFIDCR0 and CiFIDCR1 individually, both ID formats are received.

If both setting of data frame and remote frame are made in the RTR bits in registers CiFIDCR0 and CiFIDCR1 individually, both data and remote frames are received.

When combination with two ranges of IDs is not necessary, set the same mask value and the same ID into both of the FIFO ID/mask register sets.

Figure 20.16 shows the correspondence of mask registers to mailboxes. Figure 20.17 shows the acceptance filtering.

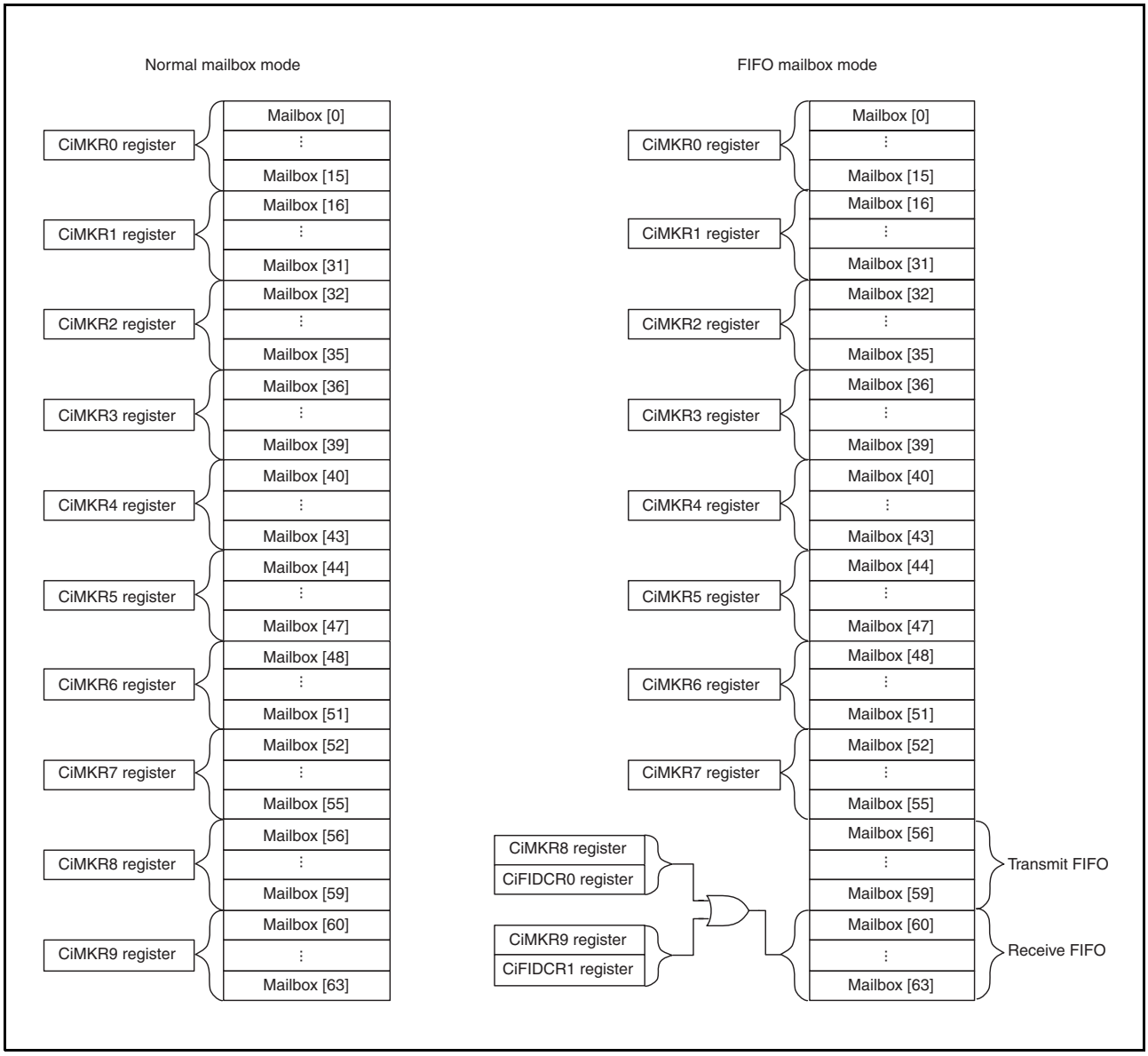
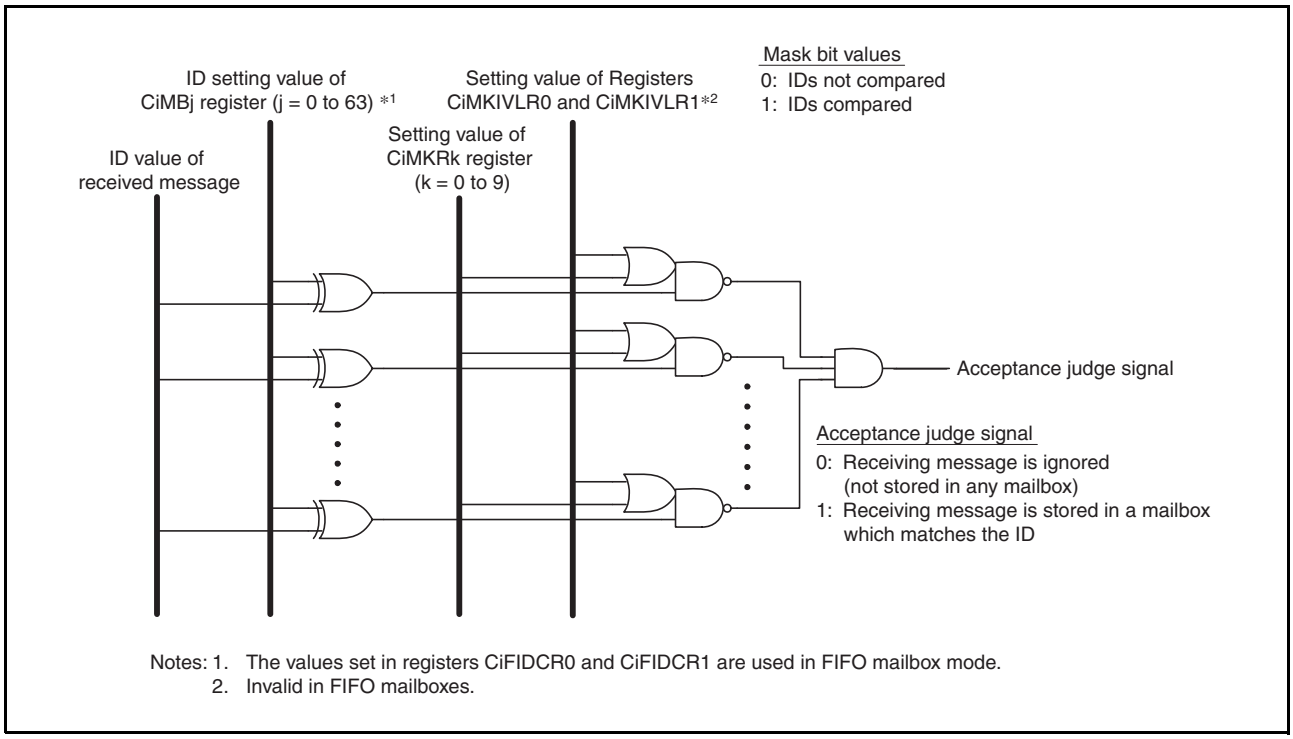


Figure 20.16 Correspondence of Mask Registers to Mailboxes (i = 0 to 5)



Notes: 1. The values set in registers CiFIDCR0 and CiFIDCR1 are used in FIFO mailbox mode.
2. Invalid in FIFO mailboxes.

Figure 20.17 Acceptance Filtering (i = 0 to 5)

20.8 Reception and Transmission

Table 20.17 lists the CAN communication mode configuration.

Table 20.17 Configuration for CAN Receive Mode and Transmit Mode

TRMREQ	RECREQ	ONESHOT	Communication Mode of Mailbox
0	0	0	Mailbox disabled or transmission being aborted
0	0	1	Configurable only when transmission or reception from a mailbox (programmed in one-shot mode) is aborted.
0	1	0	Configured as a receive mailbox for a data frame or a remote frame.
0	1	1	Configured as a one-shot receive mailbox for a data frame or a remote frame.
1	0	0	Configured as a transmit mailbox for a data frame or a remote frame.
1	0	1	Configured as a one-shot transmit mailbox for a data frame or a remote frame.
1	1	0	Do not set.
1	1	1	Do not set.

Note: TRMREQ, RECREQ, ONESHOT: Bits in CiMCTLj register (i = 0 to 5; j = 32 to 63)

When a mailbox is configured as a receive mailbox or a one-shot receive mailbox, note the following:

1. Before a mailbox is configured as a receive mailbox or a one-shot receive mailbox, set the CiMCTLj register to H'00.
2. A received message is stored into the first mailbox that matches the condition according to the result of receive mode configuration and acceptance filtering. Upon deciding a mailbox which stores the received message, the mailbox with the smaller number has higher priority.
3. In CAN operation mode, when a CAN module transmits a message whose ID matches with the ID/mask set of a mailbox configured to receive messages, the CAN module never receives the transmitted data. In self-test mode, however, the CAN module may receive its transmitted data. In this case, the CAN module sends an ACK.

When configuring a mailbox as a transmit mailbox or a one-shot transmit mailbox, note the following:

1. Before a mailbox is configured as a transmit mailbox or one-shot transmit mailbox, ensure that the CiMCTLj register is H'00 and that there is no pending abort process.

20.8.1 Reception

Figure 20.18 shows an operation example of data frame reception in overwrite mode.
This example shows the operation of overwriting the first message when the CAN module receives two consecutive CAN messages that matches the receiving conditions of the CiMCTLj register (j = 0 to 63).

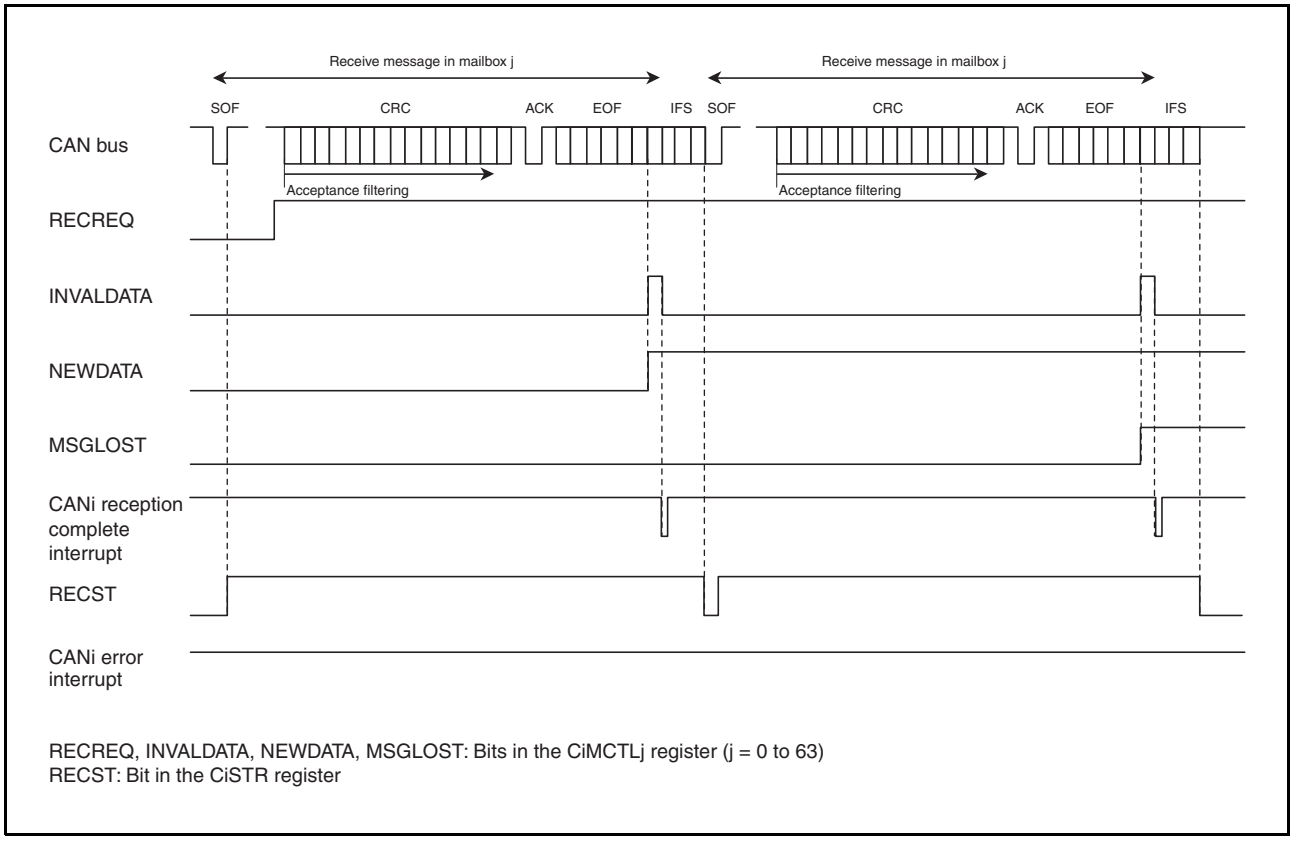


Figure 20.18 Operation Example of Data Frame Reception in Overwrite Mode (i = 0 to 5)

1. When a SOF is detected on the CAN bus, the RECST bit in the CiSTR register is set to 1 (reception in progress) if the CAN module has no message ready to start transmission.
2. The acceptance filter procedure starts at the beginning of the CRC field to select the receive mailbox.
3. After a message has been received, the NEWDATA bit in the CiMCTLj register for the receive mailbox is set to 1 (new data being updated/stored in the mailbox). The INVALDATA bit in the CiMCTLj register is set to 1 (message is being updated) at the same time, and then the INVALDATA bit is set to 0 (message valid) again after the complete message is transferred to the mailbox.
4. When the interrupt enable bit in registers CiMIER0 and CiMIER1 for the receive mailbox is 1 (interrupt enabled), the CANi reception complete interrupt request is generated. This interrupt is generated when the INVALDATA bit is set to 0.
5. After reading the message from the mailbox, the NEWDATA bit needs to be set to 0 by a program.
6. In overwrite mode, if the next CAN message has been received into a mailbox whose NEWDATA bit is still set to 1, the MSGLOST bit in the CiMCTLj register is set to 1 (message has been overwritten). The new received message is transferred to the mailbox. The CANi reception complete interrupt request is generated the same as in 4.

Figure 20.19 shows the operation example of data frame reception in overrun mode. This example shows the operation of overrunning the second message when the CAN module receives two consecutive CAN messages that matches the receiving conditions of the CiMCTLj register (j = 0 to 63).

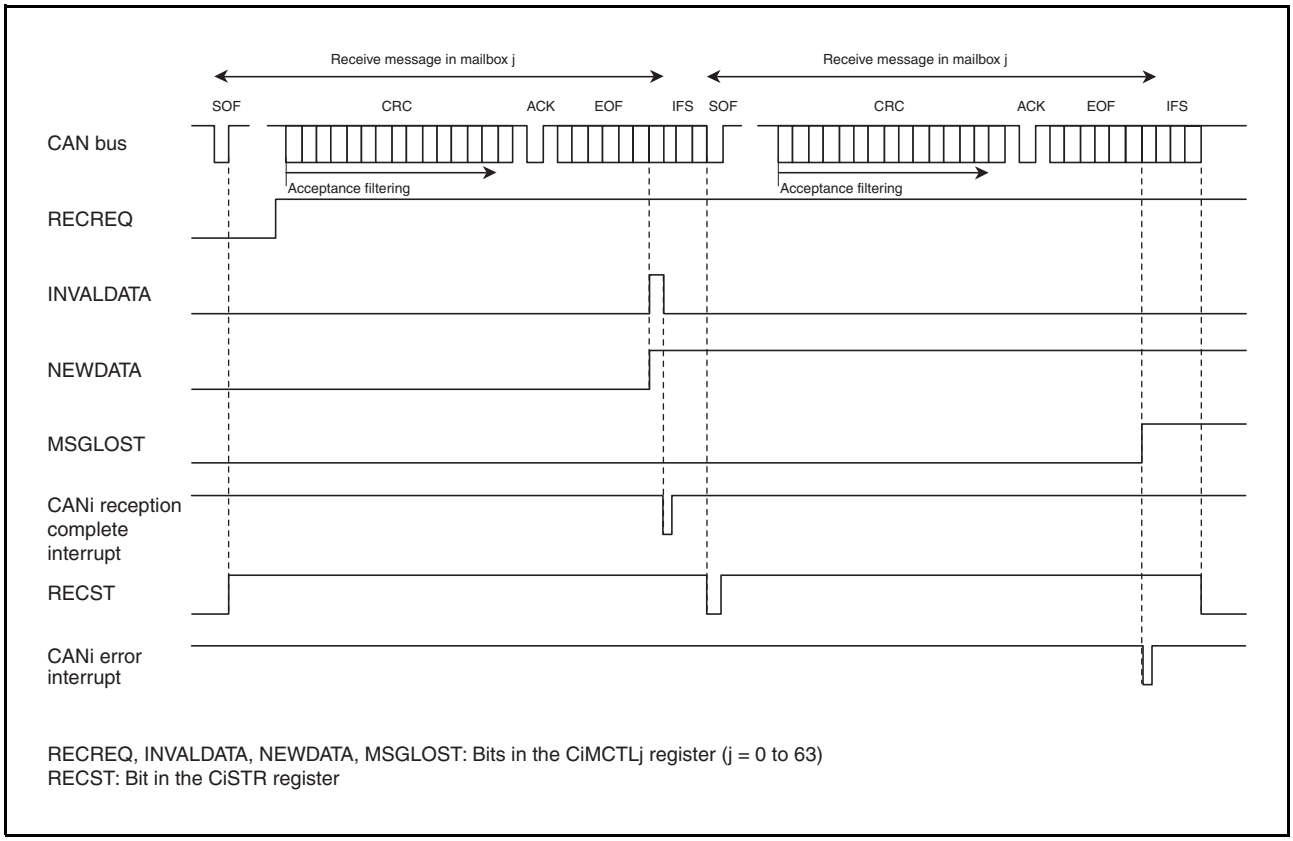


Figure 20.19 Operation Example of Data Frame Reception in Overrun Mode (i = 0 to 5)

1. to 5. are the same as overwrite mode.
6. In overrun mode, if the next message has been received before the NEWDATA bit is set to 0, the MSGLOST bit in the CiMCTLj register is set to 1 (message has been overrun). The new received message is discarded and a CANi error interrupt request is generated if the corresponding interrupt enable bit in the CiEIER register is set to 1 (interrupt enabled).

20.8.2 Transmission

Figure 20.20 shows an operation example of data frame transmission.

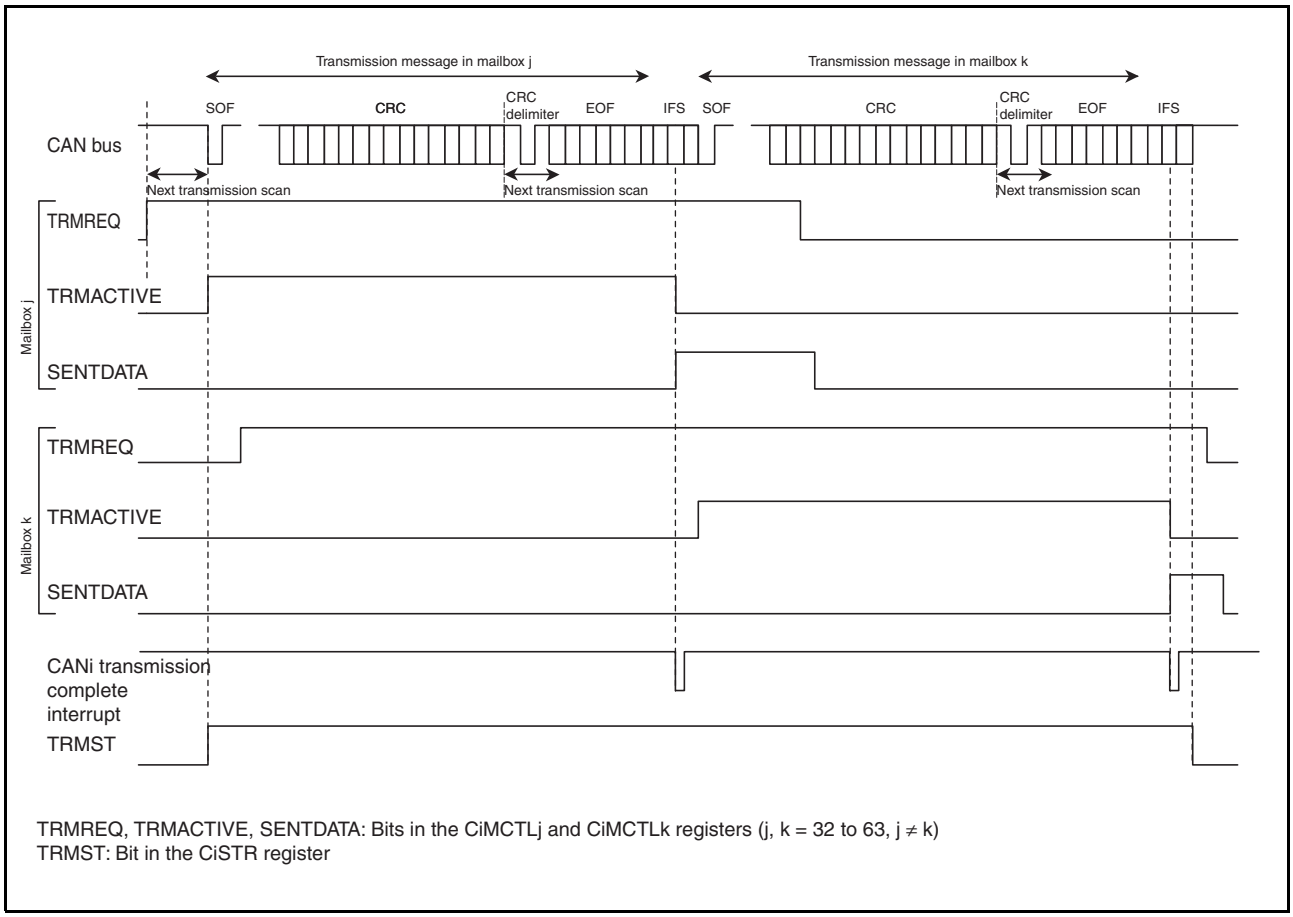


Figure 20.20 Operation Example of Data Frame Transmission (i = 0 to 5)

1. When a TRMREQ bit in the CiMCTLj register (i = 0 to 5, j = 32 to 63) is set to 1 (transmit mailbox) in bus-idle state, the mailbox scan procedure starts to decide the highest-priority mailbox for transmission. Once the transmit mailbox is decided, the TRMACTIVE bit in the CiMCTLj register is set to 1 (from when a transmission request is received until transmission is completed, or an error/arbitration lost has occurred), the TRMST bit in the CiSTR register is set to 1 (transmission in progress), and the CAN module starts transmission. *
2. If other TRMREQ bits are set, the transmission scan procedure starts with the CRC delimiter for the next transmission.
3. If transmission is completed without losing arbitration, the SENTDATA bit in the CiMCTLj register is set to 1 (transmission completed) and the TRMACTIVE bit is set to 0 (transmission is pending, or no transmission request). If the interrupt enable bit in the CiMIER register is 1 (interrupt enabled), the CANi transmission complete interrupt request is generated.
4. When requesting the next transmission from the same mailbox, set bits SENDTDATA and TRMREQ to 0, then set the TRMREQ bit to 1 after checking that bits SENDTDATA and TRMREQ have been set to 0.

Note: * If arbitration is lost after the CAN module starts transmission, the TRMACTIVE bit is set to 0. The transmission scan procedure is performed again to search for the highest-priority transmit mailbox from the beginning of the CRC delimiter. If an error occurs either during transmission or following the arbitration lost, the transmission scan procedure is performed again from the start of the error delimiter to search for the highest-priority transmit mailbox.

20.9 CAN Interrupt

The CAN module provides the following CAN interrupts for each channel.

- CANi wake-up interrupt
- CANi reception complete interrupt
- CANi transmission complete interrupt
- CANi receive FIFO interrupt
- CANi transmit FIFO interrupt
- CANi error interrupt

There are eight types of interrupt sources for the CANi error interrupts. These sources can be determined by checking the CiEIFR register.
- Bus error
- Error-warning
- Error-passive
- Bus-off entry
- Bus-off recovery
- Receive overrun
- Overload frame transmission
- Bus lock

21. 12-Bit A/D Converter (AD0)

This LSI includes a 12-bit successive approximation A/D converter, which consists of one independent unit (AD0). Up to 6 channel analog inputs can be selected by software.

Concerning the 10-bit A/D converter (AD1), refer to section 22, 10-Bit A/D Converter (AD1).

21.1 Introduction

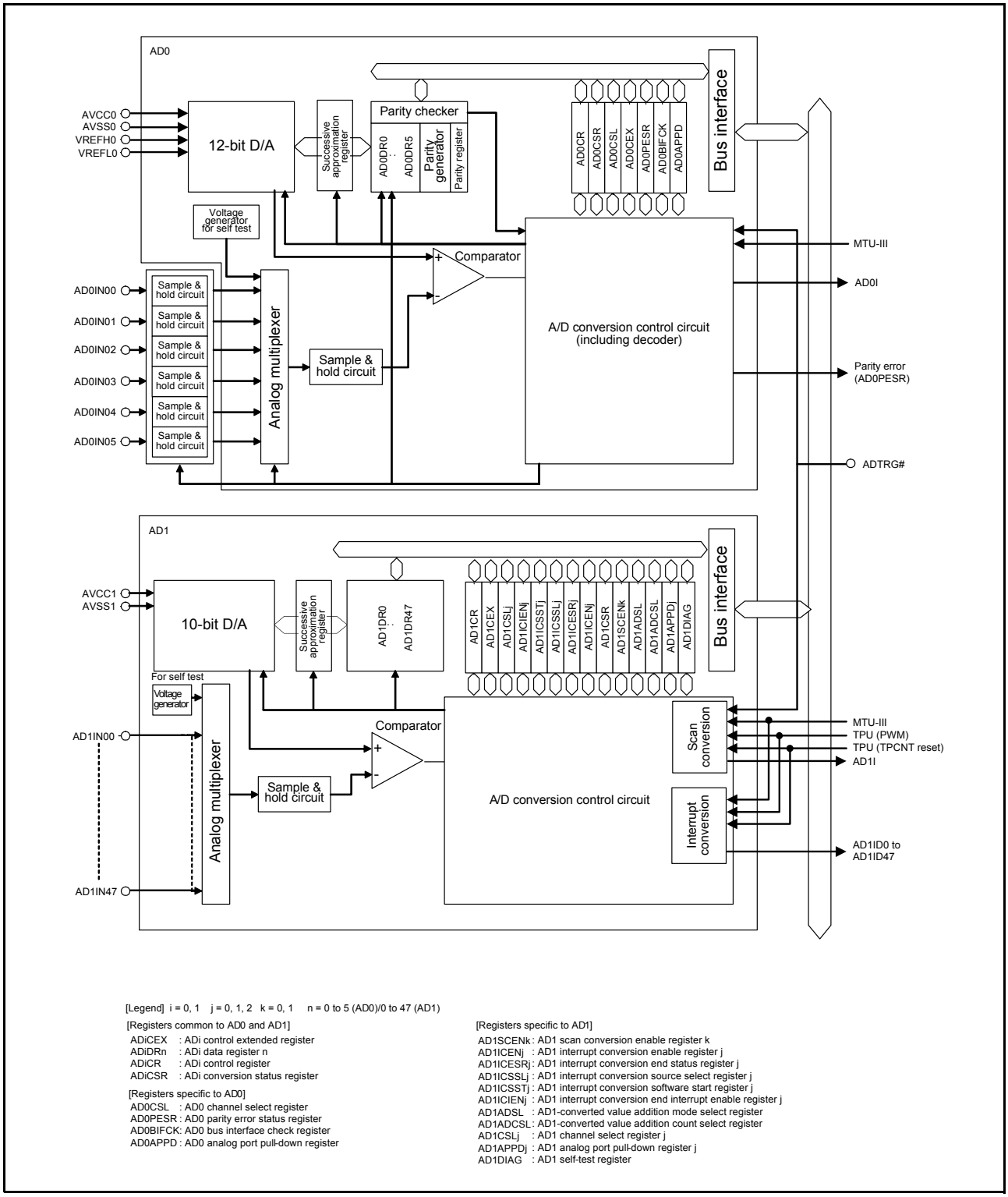
Table 21.1 lists the Specifications of 12-Bit A/D Converter (AD0).

Table 21.1 Specifications of 12-Bit A/D Converter (AD0)

Item	Description
Resolution	• 12 bits
Input channels	• 6 channels (AD0IN00 to AD0IN05)
Minimum conversion time	• Minimum 2.00/1.60 µs/channel (f(PBA) = at 40/50 MHz operation, 80 states (30 states (sample and hold) + 50 states (A/D conversion))
scan conversion modes	• 2 modes Single-cycle scan mode: scanning only once Continuous scan mode: scanning repeatedly without limitation The channels to be scan converted can be selected as desired and A/D conversion is started from the channel with the lowest number (AD0IN00 → AD0IN05).
Identifier addition function	When storing the A/D converted result of each channel in the A/D data register, an identifier corresponding to the channel is stored in the lower four bits.
Register	• Six 12-bit A/D data registers The A/D converted result is held in a 12-bit AD data register corresponding to the channel. (Only the upper 12 bits will be held if the identifier addition function is enabled.)
Sample-and-hold function	Each channel in AD0 (Ch0 to Ch5) includes an independent sample & hold circuit. This enables sampling for more than one channel (up to six channels) to be performed simultaneously.
A/D conversion start request selection	• AD0: Selectable among software trigger (the ADST bit in the AD0 control register (AD0CR)), external trigger (ADTRG# pin)*, and MTU-III (TRGA0N, TRG0N, TRGA1N, TRGA2N, TRGA3N, TRGA4N, TRG4AN, TRG4BN, TRG4ABN, TRGA6N, TRGA7N, TRG7AN, TRG7BN, or TRG7ABN)
Supporting scan conversion end interrupt (ADI) and DMA transfer function	On completion of scanning the channels specified for scan conversion, a scan conversion end interrupt request (AD0I) can be generated or the DMAC can be started up.
Bus interface check register	A register to which only read and write operations irrelevant with A/D converter operation can be performed.
Self-test function of A/D converter	The self-test function of the A/D converter performs the A/D conversion of internally generated voltage values (VREFH × 0, VREFH × 1/2, VREFH × 1), and returns A/D converted values and self-test information on converted signals to the AD0 data register (AD0DRD). After that, the AD0 data register (AD0DRD) are read by software to check whether or not the A/D converted values are in the normal range for the purpose of detecting errors in the A/D converter.
Programmable analog input voltage range	The analog input voltage range is programmable via the voltage applied to the VREFH pin.
Parity function	When storing the A/D converted data in the AD data registers 0 to 5 (AD0DR0 to AD0DR5), parity bits are generated and stored in the AD0 parity register (AD0PR). The parity bits are checked in the A/D module when reading data from the AD data registers 0 to 5 (AD0DR0 to AD0DR5) and the results are reflected in the AD0 parity error status register (ADPESR). When a parity error occurs during the parity check, an A/D parity error interrupt request is issued. (The request can be masked by register setting.)
Clearing of data registers	After reading from the data registers (AD0DR0 to AD0DR5, AD0DRD) by the CPU or DMAC, the registers that were read from can be cleared automatically. (Register setting can enable or disable this function.)
Analog port pull-down function	This function can check for disconnection between an external circuit and an analog port pin.

Note: * The SH72A0 group has no external trigger (ADTRG#).

Figure 21.1 shows Block Diagram of A/D Converter (AD0 to AD1).



[Legend] i = 0, 1 j = 0, 1, 2 k = 0, 1 n = 0 to 5 (AD0)/0 to 47 (AD1)

[Registers common to AD0 and AD1]
ADiCEX : ADi control extended register
ADiDRn : ADi data register n
ADiCR : ADi control register
ADiCSR : ADi conversion status register

[Registers specific to AD0]
AD0CSL : AD0 channel select register
AD0PESR : AD0 parity error status register
AD0BIFCK: AD0 bus interface check register
AD0APPD : AD0 analog port pull-down register

[Registers specific to AD1]
AD1SCENk : AD1 scan conversion enable register k
AD1ICENj : AD1 interrupt conversion enable register j
AD1ICESRj : AD1 interrupt conversion end status register j
AD1ICSSLj : AD1 interrupt conversion source select register j
AD1ICSSTj : AD1 interrupt conversion software start register j
AD1ICIENj : AD1 interrupt conversion end interrupt enable register j
AD1ADSL : AD1-converted value addition mode select register
AD1ADCSL : AD1-converted value addition count select register
AD1CSLj : AD1 channel select register j
AD1APPDj : AD1 analog port pull-down register j
AD1DIAG : AD1 self-test register

Figure 21.1 Block Diagram of A/D Converter (AD0 to AD1)

Table 21.2 to Table 21.4 list the functions assigned to each channel for A/D converter.

Table 21.2 Functions Assigned to Channel for AD0

A/D Converter	A/D Conversion Channel	Scan Conversion									AD0 Data Register (Storage Register for A/D-Converted Value)
		Trigger Source				Self-Test Selection or Channel Selection	Scan Conversion Mode	End Interrupt/ DMA Transfer Request Signal	DMA Transfer Function (DMAC)	End Flag (at Scan Completion)	
		Source 1	Source 2	Source 3	Source 4						
	Self-Test or Pin Name	Software	External Trigger		MTU-III	AD0CEX or AD0CSL					
AD0 6 channels	Self-Test DIAG	ADST (AD0CR)	ADTRG# pin*		—	DIAGM (AD0CEX)	Single cycle scanning or continuous scanning	AD0I	O	ADF (AD0CSR)	AD0DRD
	AD0IN00				TRGA0N, TRG0N, TRGA1N, TRGA2N, TRGA3N, TRGA4N, TRG4AN, TRG4BN, TRG4ABN, TRGA6N, TRGA7N, TRG7AN, TRG7BN, TRG7ABN	AD0CSL0					AD0DR0
	AD0IN01					AD0CSL1					AD0DR1
	AD0IN02					AD0CSL2					AD0DR2
	AD0IN03					AD0CSL3					AD0DR3
	AD0IN04					AD0CSL4					AD0DR4
	AD0IN05					AD0CSL5					AD0DR5

Note: * The SH72A0 group has no external trigger (ADTRG#).

Table 21.3 Functions Assigned to Channel for AD1 (1)

A/D Converter	A/D Conversion Channel *1	Scan Conversion									AD1 Data Register (Storage Register for A/D-Converted Value)	A/D-Converted Value Addition Mode
		Trigger Source				Channel Selection	Scan Conversion Mode	End Interrupt/ DMA Transfer Request Signal	DMA Transfer Function (DMAC)	End Flag (at Scan Completion)		
		Source 1	Source 2	Source 3	Source 4							
	Pin Name	Software	External Trigger	TPU	MTU-III	AD1CSL						AD1ADSL
AD1 24 channels	AD1IN00	ADST (AD1CR)	ADTRG# pin *2	TPO1A TP1CNT reset TPO2A TP2CNT reset TPO3A TP3CNT reset TPO4A TP4CNT reset	TRG0N, TRGA0N, TRGA1N, TRGA2N, TRGA3N, TRGA4N, TRGA6N, TRGA7N	AD1CSL0	Single cycle scanning or continuous scanning	AD1I	O	ADF (AD1CSR)	AD1DR0	AD1ADSL0
	AD1IN01					AD1CSL1					AD1DR1	AD1ADSL1
	AD1IN02					AD1CSL2					AD1DR2	AD1ADSL2
	AD1IN03					AD1CSL3					AD1DR3	AD1ADSL3
	AD1IN04					AD1CSL4					AD1DR4	AD1ADSL4
	AD1IN05					AD1CSL5					AD1DR5	AD1ADSL5
	AD1IN06					AD1CSL6					AD1DR6	AD1ADSL6
	AD1IN07					AD1CSL7					AD1DR7	AD1ADSL7
	AD1IN16					AD1CSL16					AD1DR16	×
	AD1IN17					AD1CSL17					AD1DR17	×
	AD1IN24					AD1CSL24					AD1DR24	×
	AD1IN25					AD1CSL25					AD1DR25	×
	AD1IN26					AD1CSL26					AD1DR26	×
	AD1IN27					AD1CSL27					AD1DR27	×
	AD1IN28					AD1CSL28					AD1DR28	×
	AD1IN29					AD1CSL29					AD1DR29	×
	AD1IN30					AD1CSL30					AD1DR30	×
	AD1IN31					AD1CSL31					AD1DR31	×
	AD1IN42					AD1CSL42					AD1DR42	×
	AD1IN43					AD1CSL43					AD1DR43	×
	AD1IN44					AD1CSL44					AD1DR44	×
	AD1IN45					AD1CSL45					AD1DR45	×
	AD1IN46					AD1CSL46					AD1DR46	×
	AD1IN47					AD1CSL47					AD1DR47	×

Notes: 1. The SH72A0 group has the following eight A/D convertion channels:
AD1IN04, AD1IN05, AD1IN16, AD1IN17, AD1IN24, AD1IN28, AD1IN30, and AD1IN31.
2. The SH72A0 group has no external trigger (ADTRG#).

Table 21.4 Functions Assigned to Channel for AD1 (2)

A/D Converter	A/D Conversion Channel *	Interrupt Conversion										
		Trigger Source				End Interrupt/ DMA Transfer Request Signal	DMA Transfer Function	Enable (Channel Selection)	Source Selection	End Status	End Interrupt Enable	Start Interrupt Conversion Software
		Source 1	Source 2	Source 3	Source 4							
	Pin Name	TPU	TPU	MTU-III	Software (AD1ICSST)		DMAC	AD1ICEN	AD1ICSSL	AD1ICESR	AD1ICIEN	AD1ICSST
AD1 24 channels	AD1IN00			TRGA0N	AD1ICSST0	AD1ID0	O	AD1ICEN0	AD1ICSSL0	AD1ICESR0	AD1ICIEN0	AD1ICSST0
	AD1IN01			TRGA0N	AD1ICSST1	AD1ID1	O	AD1ICEN1	AD1ICSSL1	AD1ICESR1	AD1ICIEN1	AD1ICSST1
	AD1IN02			TRGA1N	AD1ICSST2	AD1ID2	O	AD1ICEN2	AD1ICSSL2	AD1ICESR2	AD1ICIEN2	AD1ICSST2
	AD1IN03			TRGA1N	AD1ICSST3	AD1ID3	O	AD1ICEN3	AD1ICSSL3	AD1ICESR3	AD1ICIEN3	AD1ICSST3
	AD1IN04	TPO1A A/D convertion trigger	TP1CNT counter reset	TRGA2N	AD1ICSST4	AD1ID4	O	AD1ICEN4	AD1ICSSL4	AD1ICESR4	AD1ICIEN4	AD1ICSST4
	AD1IN05	TPO1B A/Dconvertion trigger		TRGA2N	AD1ICSST5	AD1ID5	O	AD1ICEN5	AD1ICSSL5	AD1ICESR5	AD1ICIEN5	AD1ICSST5
	AD1IN06	TPO1C A/Dconvertion trigger		TRGA3N	AD1ICSST6	AD1ID6	O	AD1ICEN6	AD1ICSSL6	AD1ICESR6	AD1ICIEN6	AD1ICSST6
	AD1IN07	TPO1D A/Dconvertion trigger		TRGA3N	AD1ICSST7	AD1ID7	O	AD1ICEN7	AD1ICSSL7	AD1ICESR7	AD1ICIEN7	AD1ICSST7
	AD1IN16	TPO4A A/Dconvertion trigger	TP4CNT counter reset	TRGA0N	AD1ICSST16	AD1ID16	O	AD1ICEN16	AD1ICSSL16	AD1ICESR16	AD1ICIEN16	AD1ICSST16
	AD1IN17	TPO4B A/Dconvertion trigger		TRGA0N	AD1ICSST17	AD1ID17	O	AD1ICEN17	AD1ICSSL17	AD1ICESR17	AD1ICIEN17	AD1ICSST17
	AD1IN24			TRGA4N	AD1ICSST24	AD1ID24	O	AD1ICEN24	AD1ICSSL24	AD1ICESR24	AD1ICIEN24	AD1ICSST24
	AD1IN25			TRGA4N	AD1ICSST25	AD1ID25	O	AD1ICEN25	AD1ICSSL25	AD1ICESR25	AD1ICIEN25	AD1ICSST25
	AD1IN26			TRG0N	AD1ICSST26	AD1ID26	O	AD1ICEN26	AD1ICSSL26	AD1ICESR26	AD1ICIEN26	AD1ICSST26
	AD1IN27			TRG0N	AD1ICSST27	AD1ID27	O	AD1ICEN27	AD1ICSSL27	AD1ICESR27	AD1ICIEN27	AD1ICSST27
	AD1IN28			TRGA6N	AD1ICSST28	AD1ID28	O	AD1ICEN28	AD1ICSSL28	AD1ICESR28	AD1ICIEN28	AD1ICSST28
	AD1IN29			TRGA6N	AD1ICSST29	AD1ID29	O	AD1ICEN29	AD1ICSSL29	AD1ICESR29	AD1ICIEN29	AD1ICSST29
	AD1IN30			TRGA7N	AD1ICSST30	AD1ID30	O	AD1ICEN30	AD1ICSSL30	AD1ICESR30	AD1ICIEN30	AD1ICSST30
	AD1IN31			TRGA7N	AD1ICSST31	AD1ID31	O	AD1ICEN31	AD1ICSSL31	AD1ICESR31	AD1ICIEN31	AD1ICSST31
	AD1IN42			TRG0N	AD1ICSST42	AD1ID42	O	AD1ICEN42	AD1ICSSL42	AD1ICESR42	AD1ICIEN42	AD1ICSST42
	AD1IN43			TRG0N	AD1ICSST43	AD1ID43	O	AD1ICEN43	AD1ICSSL43	AD1ICESR43	AD1ICIEN43	AD1ICSST43
	AD1IN44			TRGA6N	AD1ICSST44	AD1ID44	O	AD1ICEN44	AD1ICSSL44	AD1ICESR44	AD1ICIEN44	AD1ICSST44
	AD1IN45			TRGA6N	AD1ICSST45	AD1ID45	O	AD1ICEN45	AD1ICSSL45	AD1ICESR45	AD1ICIEN45	AD1ICSST45
	AD1IN46			TRGA7N	AD1ICSST46	AD1ID46	O	AD1ICEN46	AD1ICSSL46	AD1ICESR46	AD1ICIEN46	AD1ICSST46
	AD1IN47			TRGA7N	AD1ICSST47	AD1ID47	O	AD1ICEN47	AD1ICSSL47	AD1ICESR47	AD1ICIEN47	AD1ICSST47

Note: * The SH72A0 group has the following eight A/D convertion channels:
AD1IN04, AD1IN05, AD1IN16, AD1IN17, AD1IN24, AD1IN28, AD1IN30, and AD1IN31.

Table 21.5 lists I/O Pins of 12-Bit A/D Converter (AD0).

The six pins of AD0IN00 to AD0IN05 serve as the analog inputs to AD0.

The ADTRG# pin supplies the timing to start scan conversion from outside the LSI. Inputting a low-level signal to the ADTRG# pin requests scan conversion to start in AD0 and AD1. Because the SH72A0 group does not have the ADTRG# pin, no scan conversion can be externally requested.

The AVCC0 and AVSS0 pins are power supply input pins for the analog block in the A/D converter (AD0). The VREFH0 and VREFL0 pins are reference voltage input pins of the A/D converter (AD0).

Table 21.5 I/O Pins of 12-Bit A/D Converter (AD0)

Pin Name	I/O	Function
AVCC0	Input	Analog power supply pin
AVSS0	Input	Analog ground pin
VREFL0	Input	Analog reference voltage pin (VREFL0 < VREFH0)
VREFH0	Input	Analog reference voltage pin (VREFL0 < VREFH0)
AD0IN00	Input	AD0 analog input pin 0
AD0IN01	Input	AD0 analog input pin 1
AD0IN02	Input	AD0 analog input pin 2
AD0IN03	Input	AD0 analog input pin 3
AD0IN04	Input	AD0 analog input pin 4
AD0IN05	Input	AD0 analog input pin 5
ADTRG#	Input	Input pin for scan conversion trigger of A/D*

Note: * The SH72A0 group does not have the ADTRG# pin.

21.2 Registers

Table 21.6 lists Registers of 12-Bit A/D Converter (AD0).

Table 21.6 Registers of 12-Bit A/D Converter (AD0)

Register Name	Symbol	After Reset	Address	Access Size *
AD0 data register 0	AD0DR0	H'0000	H'FFFE 7840	16
AD0 data register 1	AD0DR1	H'0000	H'FFFE 7842	16
AD0 data register 2	AD0DR2	H'0000	H'FFFE 7844	16
AD0 data register 3	AD0DR3	H'0000	H'FFFE 7846	16
AD0 data register 4	AD0DR4	H'0000	H'FFFE 7848	16
AD0 data register 5	AD0DR5	H'0000	H'FFFE 784A	16
AD0 data register DIAG	AD0DRD	H'0000	H'FFFE 783E	16
AD0 control register	AD0CR	H'00	H'FFFE 7800	8
AD0 control extended register	AD0CEX	H'2000	H'FFFE 7830	16
AD0 channel select register	AD0CSL	H'0000	H'FFFE 7820	16
AD0 conversion status register	AD0CSR	H'00	H'FFFE 7802	8
AD0 start trigger select register	AD0STRSL	H'00	H'FFFE 7890	8
AD0 bus interface check register	AD0BIFCK	H'3141 5926	H'FFFE 78C0	8, 16, 32
AD0 parity error status register	AD0PESR	H'0000	H'FFFE 78B0	8, 16
AD0 parity register	AD0PR	H'0000	H'FFFE 78A0	16
AD0 analog port pull-down register	AD0APPD	H'0000	H'FFFE 7834	8

Note: * 16-bit access can be made only on word boundaries, while 32-bit access can be made only on longword boundaries.

21.2.1 AD0 Data Register n (AD0DRn) (n = 0 to 5), AD0 Data Register DIAG (AD0DRD)

The AD0DRn and AD0DRD registers use different formats under the following conditions.

- Setting of the AD data register format select bit (ADRFMT) in the AD0 control extended register (AD0CEX) (left-alignment or right-alignment)
- Setting of identifier bits when the identifier select bit (IDE) in the AD0 control extended register (AD0CEX) is set to 1 (not selected or selected) [The AD0DRD register do not support the identifier addition function]

21.2.1.1 AD0 Data Register n (AD0DRn) (n = 0 to 5)

Address AD0DR0: H'FFFE 7840, AD0DR1: H'FFFE 7842, AD0DR2: H'FFFE 7844, AD0DR3: H'FFFE 7846, AD0DR4: H'FFFE 7848, AD0DR5: H'FFFE 784A

• When left-alignment is selected

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ADD11	ADD10	ADD9	ADD8	ADD7	ADD6	ADD5	ADD4	ADD3	ADD2	ADD1	ADD0	—	—	—	—
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 b4	ADD11 to ADD0	Data Register Bits	12-bit A/D-Converted Value	R
b3 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

• When right-alignment is selected

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	ADD11	ADD10	ADD9	ADD8	ADD7	ADD6	ADD5	ADD4	ADD3	ADD2	ADD1	ADD0
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 b12	—	Reserved	These bits are read as 0. The write value should be 0.	R
b11 to b0	ADD11 to ADD0	Data Register Bits	12-bit A/D-Converted Value	R

• When identifier addition is selected

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ADD11	ADD10	ADD9	ADD8	ADD7	ADD6	ADD5	ADD4	ADD3	ADD2	ADD1	ADD0	ID[3:0]			
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 b4	ADD11 to ADD0	Data Register Bits	12-bit A/D-Converted Value	R
b3 to b0	ID[3:0]	Identifier Bits	Identifier See Table 21.7 for the value to be set.	R

AD0 data register n (AD0DRn) is a 16-bit read-only register which stores the A/D converted results of channels AD0IN00 to AD0IN05. There are six registers accepting analog inputs for channels AD0IN00 to AD0IN05. Registers AD0DR0 to AD0DR5 are initialized to H'0000 by a reset.

Left-shift or right-shift format can be selected by setting the ADRFMT bit in the AD0 control extended register (AD0CEX). At this time, bits ADD11 to ADD0 show the 12-bit A/D-converted value. The other bits are reserved. These bits are read as 0. The write value should be 0.

When the identifier select bit (IDE) in the AD0 control extended register (AD0CEX) is set to identifier addition, the identifier (ID) corresponding to the converted channel is stored in the lower four bits. By reading the ID from the AD0DRn register, whether the correct channel has been converted can be confirmed.

The setting of the A/D data register format select bit (ADRFMT) in the AD0 control extended register (AD0CEX) becomes invalid.

Table 21.7 shows the identifiers to be added.

Table 21.7 List of Identifiers

Additional ID	AD0 Channel
B'0100	AD0IN00
B'0101	AD0IN01
B'0110	AD0IN02
B'0111	AD0IN03
B'1000	AD0IN04
B'1001	AD0IN05
B'1010	—
B'1011	—
B'1100	—

21.2.1.2 AD0 Data Register DIAG (AD0DRD)

Address H'FFFE 783E

• When left-alignment is specified

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ADD 11	ADD 10	ADD9	ADD8	ADD7	ADD6	ADD5	ADD4	ADD3	ADD2	ADD1	ADD0	—	—	DIAGST [1:0]	
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 b4	AD11 to AD0	Data Register Bits	12-bit A/D-Converted Value	R
b3, b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b1, b0	DIAGST [1:0]	Self-Test Status Bits	b1 b0 0 0 : No self test has ever been performed since reset. 0 1 : A self test has been performed at a voltage of VREFH × 0. 1 0 : A self test has been performed at a voltage of VREFH × 1/2. 1 1 : A self test has been performed at a voltage of VREFH × 1.	R

• When right-alignment is specified

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	DIAGST [1:0]		—	—	ADD 11	ADD 10	ADD9	ADD8	ADD7	ADD6	ADD5	ADD4	ADD3	ADD2	ADD1	ADD0
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, b14	DIAGST [1:0]	Self-Test Status Bits	b15b14 0 0 : No self test has ever been performed since reset. 0 1 : A self test has been performed at a voltage of VREFH × 0. 1 0 : A self test has been performed at a voltage of VREFH × 1/2. 1 1 : A self test has been performed at a voltage of VREFH × 1.	R
b13, b12	—	Reserved	These bits are read as 0. The write value should be 0.	R
b11 to b0	ADD11 to ADD0	Data Register Bits	12-bit A/D-Converted Value	R

The AD0 data register DIAG (AD0DRD) is a 16-bit read-only register which stores the A/D converted result of AD0 for self test. The AD0DRD register is initialized to H'0000 by a reset. The AD data register format select bit (ADRFMT) in the AD0 control extended register (AD0CEX) can be used to set either left- or right-alignment. For this operation, the ADD11 to ADD0 bits indicate a 12-bit A/D converted value. AD0DRD also includes the self-test status bits. Other bits are reserved. These bits are read as 0. The write value should be 0.

DIAGST Bit

These bits indicate the conversion voltage at which a self test is performed. For details on self test, refer to section 21.2.3, AD0 Control Extended Register (AD0CEX).

21.2.2 AD0 Control Register (AD0CR)

Address H'FFFE 7800

	b7	b6	b5	b4	b3	b2	b1	b0
	ADST	ADCS	—	ADIE	—	—	TRGE	EXTRG
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	ADST	Scan Conversion Start Bit	0: Stops a scan conversion process. 1: Starts a scan conversion process.	R/W
b6	ADCS	Scan Conversion Mode Select Bit	0: Single-cycle scan mode 1: Continuous scan mode	R/W
b5	—	Reserved	This bit is read as 0. The write value should be 0.	R
b4	ADIE	Interrupt Enable Bit	0: Disables ADI interrupt generation upon scanning completion. 1: Enables ADI interrupt generation upon scanning completion.	R/W
b3, b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b1	TRGE	Trigger Enable Bit	0: Disables scan conversion to be started by an external trigger (ADTRG#) or a MTU-III timer trigger. 1: Enables scan conversion to be started by an external trigger (ADTRG#) or a MTU-III timer trigger.	R/W
b0	EXTRG	Trigger Select Bit	0: Scan conversion is started by a timer source selected by the AD0 start trigger select register (AD0STRSL). 1: Scan conversion is started by an external trigger (ADTRG#).	R/W

Notes: • Starting a scan conversion using an external trigger
If 1 is written to both the TRGE and EXTRG bits when a high-level signal is input to the external trigger pin (ADTRG#), and then if a low level pulse is input to either the ADTRG# pin, either AD0 detects a pulse falling edge and starts the scan conversion process. In this case, the low pulse width must be 1.5 cycles or more of the peripheral bus clock A.
• Independent of the ADST bit, external trigger, and triggers from the timers, starting of a scan conversion process is enabled when the ADSCACT bit in AD0 conversion status register (AD0CSR) is cleared to 0. The startup source for a scan conversion process is not retained.
• Because the SH72A0 group does not have the external trigger pin (ADTRG#), scan conversion cannot be started by external trigger.

The AD0CR register is a 8-bit readable/writable register used to set scan conversion mode for AD0. The AD0CR register is initialized to H'00 by a reset.

ADST Bit

Starts or stops scan conversion. When the ADST bit is set from 0 to 1, the A/D converter detects the rising edge of the ADST bit and then starts scan conversion. When the ADST bit is cleared from 1 to 0, the A/D converter detects the falling edge of the ADST bit and then stops scan conversion. To check whether a scan conversion is being performed, read the ADSCACT bit in the AD0 conversion status register (AD0CSR).

ADCS Bit

Selects scan conversion mode. To prevent incorrect operation, the value of the ADCS bit must be changed while the ADSCACT bit in the AD0 conversion status register (AD0CSR) is cleared to 0. Select single-cycle scan mode or continuous scan mode with this bit. Single-cycle scan mode performs scan conversion once and, upon its completion, stops the scan conversion process. Continuous scan mode continuously repeats the scan conversion process while the scan conversion start bit (ADST) is 1. The scan conversion process can be stopped by writing 0 to the scan conversion start bit (ADST) when the bit is set to 1. The channels selected by the AD0 channel select register (AD0CSL) are A/D-converted with the lowest channel first (AD0IN00 → AD0IN05).
In continuous scan mode, the scan conversion process returns to the first channel when all the selected channels have been converted.

ADIE Bit

Enables or disables generation of the A/D scan conversion end interrupts (AD0I). To prevent incorrect operation, the ADSCACT bit in the AD0 conversion status register (AD0CSR) must be cleared to 0 while the ADIE value is changed.

When the ADF bit in the AD0 conversion status register (AD0CSR) is set to 1 upon completion of scan conversion of the specified channels, an ADI scan conversion end interrupt (AD0I) is generated if this bit is set to 1. The AD0I interrupts can be cleared by clearing the ADF bit to 0 or clearing this bit to 0.

TRGE Bit

Enables or disables scan conversion to be started by an external trigger (ADTRG# pin) or a MTU-III timer trigger.

EXTRG Bit

Selects a trigger source for scan conversion.

The selectable trigger sources are an external trigger (ADTRG# pin) and the startup sources of the MTU-III, which are selected by the AD0 start trigger select register (AD0STRSL).

21.2.3 AD0 Control Extended Register (AD0CEX)

Address H'FFFE 7830

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ADRFMT	—	—	—	DIAGM	DIAGLD	DIAGVAL [1:0]		—	PAIE	ACE	—	IDE	—	—	—
After Reset	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15	ADRFMT	AD Data Register Format Select Bit	0: Left-alignment is selected for the AD data register format. 1: Right-alignment is selected for the AD data register format.	R/W
b14	—	Reserved	This bit is read as 0. The write value should be 0.	R
b13	—	Reserved	This bit is read as 1. The write value should be 1.	R
b12	—	Reserved	This bit is read as 0. The write value should be 0.	R
b11	DIAGM	Self-Test Enable Bit	0: Self test of the A/D converter is not performed. 1: Self test of the A/D converter is performed.	R/W
b10	DIAGLD	Self-Test Mode Select Bit	0: Self-test voltages are automatically rotated. 1: Self-test voltage is fixed by bits DIAGVAL.	R/W
b9, b8	DIAGVAL [1:0]	Self-Test Voltage Select Bits	b9 b8 0 0 : Reserved 0 1 : Perform a self test at a voltage of VREFH × 0 1 0 : Perform a self test at a voltage of VREFH × 1/2 1 1 : Perform a self test at a voltage of VREFH × 1	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R
b6	PAIE	Parity Interrupt Enable Bit	0: Disables output of a parity error interrupt request 1: Enables output of a parity error interrupt request	R/W
b5	ACE	Automatic Clearing Enable Bit	0: Disables automatic clearing of AD0DRn and AD0DRD after they have been read 1: Enables automatic clearing of AD0DRn and AD0DRD after they have been read	R/W
b4	—	Reserved	This bit is read as 0. The write value should be 0.	R
b3	IDE	Identifier Enable Bit	0: Does not add identifier to the AD0DRn register 1: Adds identifier to the AD0DRn register	R/W
b2 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

ADRFMT Bit

Specifies left-alignment or right-alignment for the data to be stored in the AD0 data register.

The format of the AD0 data register associated with a channel on which identifier addition mode is selected is fixed to left-alignment, irrespective of the ADRFMT bit value.

For details on the format of the AD0 data register, refer to section 21.2.1, AD0 Data Register n (AD0DRn) (n = 0 to 5), AD0 Data Register DIAG (AD0DRD).

DIAGM Bit

The self-test function is used to detect errors in the A/D converter (AD0). The AD0 convert the internally generated voltage values VREFH × 0, VREFH × 1/2, and VREFH × 1. Upon completion of the conversion, A/D converted values and information on converted voltages are stored in the AD0 data register DIAG (AD0DRD). After that, the AD0DRD register is read by software to determine whether the A/D converted values are in the normal range or not. Self test is performed before the lowest-numbered channel is converted in the scan conversion process.

In each execution of a self test, one of the three voltage values is converted, and the three voltage values are automatically rotated each time a self test is executed. The execution time for self test is equal to the A/D conversion time for a channel. To prevent incorrect operation, the ADSCACT bit in the AD0 conversion status register (AD0CSR) must be cleared to 0 while the DIAGM value is changed.

DIAGLD Bit

Selects whether the three voltage values that are converted in the self-test process are to be rotated or to be fixed. Clearing the DIAGLD bit to 0 causes a conversion to be performed with rotating voltage values in the following sequence: VREFH × 0 VREFH × 1/2 VREFH × 1. If a self test is performed beginning with VREFH × 0 upon reset, the voltage does not return to VREFH × 0 even after the completion of the scan conversion. If a scan conversion is executed again, rotation is resumed from where it ended in the previous conversion operation.

Setting the DIAGLD bit to 1 causes a conversion at a fixed voltage value that is selected by the DIAGVAL bit in the AD0 control extended register. (Automatic rotation is not performed.) Clearing the DIAGLD bit to 0 again causes the start of rotation from a fixed voltage value (in a loading function).

DIAGVAL Bit

These bits are set when a self-test converted value is fixed. For details, refer to the description of the DIAGLD bit. If the value of these bits is B'00 (reserved) which is the initial value, executing self test with the DIAGLD bit set to 1 is prohibited.

PAIE Bit

Selects whether or not to output the parity error that occurred when reading from the AD0 data register as an interrupt request.

ACE Bit

Enables or disables automatic clearing of the AD0DRn and AD0DRD registers after they have been read by the CPU or DMAC. When this bit is set to 1, the AD0DRn and AD0DRD registers are automatically cleared to H'0000 after the CPU or DMAC reads from the AD0DRn and AD0DRD registers. This function enables update failures of the AD0DRn and AD0DRD registers to be detected. When the AD0DRn and AD0DRD registers are cleared, the parity and identifier bits are also cleared.

IDE Bit

Selects whether or not to add an identifier to the A/D converted value to be stored in the AD0 data register (AD0DRn). If identifier addition is selected, left-alignment is used for storing the A/D data in the AD0DRn register, regardless of the ADRFMT setting.

21.2.4 AD0 Channel Select Register (AD0CSL)

Address H'FFFE 7820

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	AD0 CSL5	AD0 CSL4	AD0 CSL3	AD0 CSL2	AD0 CSL1	AD0 CSL0
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 b6	—	Reserved	These bits are read as 0. The write value should be 0.	R
b5 to b0	AD0CSL5 to AD0CSL0	AD0 Conversion Channel Select Bits	Setting the ANS bit to 1 selects AD0INm as a channel to perform A/D conversion. 0: AD0INm is not subject to scan conversion. 1: AD0INm is subject to scan conversion.	R/W

Note: m = 00 to 05

The AD0CSL register is used to select channels that are subject to scan conversion.

To prevent incorrect operation, the ADSCACT bit in the AD0CSR register must be cleared to 0 while the AD0CSL register values are changed.

21.2.5 AD0 Conversion Status Register (AD0CSR)

Address H'FFFE 7802

	b7	b6	b5	b4	b3	b2	b1	b0
	ADSCACT	—	—	—	—	—	—	ADF
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	ADSCACT	Scan Conversion Status Bit	0: Scan conversion process is in idle state. 1: Scan conversion process is being executed.	R
b6 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R
b0	ADF	Single Scan End Flag	0: Indicates that the scan conversion process is in idle state. 1: Indicates that a single scan has been completed and the A/D-converted values on all selected AD0INm channels have been transferred to the AD0DRn register.	R/W

Note: n = 0 to 5, m = 00 to 05

ADSCACT Bit

This is a status bit indicating whether the scan conversion process is in the idle state or it is being executed.
Since this is a read-only bit, writing a value to this bit has no meaning. However, if a value is to be written to this bit, it must always be 0.

ADF Bit

This bit is set to 1 each time scanning ends in the scan conversion process (when all selected channels are converted). 1 cannot be written to this bit. When the ADF bit is set to 1, either a scan conversion end interrupt or a DMA transfer request to the DMAC can be generated. In this manner, processing such as storing the contents of the AD0 data register to the RAM can be implemented by means of either software or the DMAC.

[Clearing conditions]

- 0 is written to this bit after reading 1.
- The DMAC is started up by AD0I.

[Setting condition]

- All analog conversion has been completed during each scanning in scan conversion process.

21.2.6 AD0 Start Trigger Select Register (AD0STRSL)

Address H'FFFE 7890

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	ADSTRS [4:0]				
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R
b4 to b0	ADSTRS [4:0]	A/D Start Trigger Select Bit	The relationship between the startup sources and the settings of these bits is shown in Table 21.8.	R/W

The AD0STRSL register selects the A/D conversion start trigger of the MTU-III to be used as the A/D conversion startup source when the TRGE bit in the AD0CR register is set to 1 while the EXTRG bit in the AD0CR register is cleared to 0.

ADSTRS Bit

The A/D conversion startup source from the on-chip peripheral modules is selected by the combination of these bits.

Table 21.8 List of A/D Conversion Startup Sources (AD0STRSL) (1)

Module	Source	Ch	Remarks	ADSTRS [4]	ADSTRS [3]	ADSTRS [2]	ADSTRS [1]	ADSTRS [0]
ADC	ADTRG	—	A/D conversion startup trigger pin*	0	0	0	0	0
MTU-III	TRGA0N	0	TGRA0 of IC/OC	0	0	0	0	1
	TRGA1N	1	TGRA1 of IC/OC	0	0	0	1	0
	TRGA2N	2	TGRA2 of IC/OC	0	0	0	1	1
	TRGA3N	3	TGRA3 of IC/OC	0	0	1	0	0
	TRGA4N	4	IC/OC of TGRA4, low level of MT4CNT in complementary PWM mode	0	0	1	0	1
	TRGA6N	6	TGRA6 of IC/OC	0	0	1	1	0
	TRGA7N	7	IC/OC of TGRA7, low level of MT7CNT in complementary PWM mode	0	0	1	1	1
	TRG0N	0	TGRE0 compare	0	1	0	0	0
	TRG4AN	4	Compare match between MT4ADSRCSEA and MT4CNT (interrupt skipping function 1)	0	1	0	0	1
	TRG4BN	4	Compare match between MT4ADSRCSEB and MT4CNT (interrupt skipping function 1)	0	1	0	1	0
	TRG4AN or TRG4BN	4	Compare match between MT4ADSRCSEA and MT4CNT or MT4ADSRCSEB and MT4CNT (interrupt skipping function 1)	0	1	0	1	1
	TRG4ABN	4	Compare match between MT4ADSRCSEA and MT4CNT or MT4ADSRCSEB and MT4CNT (interrupt skipping function 2)	0	1	1	0	0
	TRG7AN	7	Compare match between MT7ADSRCSEA and MT7CNT (interrupt skippingfunction 1)	0	1	1	0	1
	TRG7BN	7	Compare match between MT7ADSRCSEB and MT7CNT (interrupt skipping function 1)	0	1	1	1	0
	TRG7AN or TRG7BN	7	Compare match between MT7ADSRCSEA and MT7CNT or MT7ADSRCSEB and MT7CNT (interrupt skipping function 1)	0	1	1	1	1
	TRG7ABN	7	Compare match between MT7ADSRCSEA and MT7CNT or MT7ADSRCSEB and MT7CNT (interrupt skipping function 2)	1	0	0	0	0

Note: * Because the SH72A0 group does not have the external trigger pin (ADTRG#), the A/D converter cannot be activated by external trigger.

21.2.7 AD0 Bus Interface Check Register (AD0BIFCK)

Address H'FFFE 78C0

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	RWR[31:16]															
After Reset	0	0	1	1	0	0	0	1	0	1	0	0	0	0	0	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RWR[15:0]															
After Reset	0	1	0	1	1	0	0	1	0	0	1	0	0	1	1	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	RWR[31:0]	General Read/Write Register	32-bit register to which a desired value can be written to or read from*.	R/W

Note: * Read/write in longwords is executed by dividing a longword into upper and lower words and performing operations twice on it. Accordingly, read/write to the upper word and lower word cannot be performed in the same clock cycle.

The AD0BIFCK register is initialized to H'3141 5926 by a reset.

21.2.8 AD0 Parity Register (AD0PR)

Address H'FFFE 78A0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	PR[5:0]					
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 b6	—	Reserved	These bits are read as 0. The write value should be 0.	R
b5 to b0	PR[5:0]	Parity Bits Generated Using AD0DRn Data Bits	The PRn bit indicates the parity operation result of the conversion result.	R

The AD0PR register performs parity operation when the A/D converter stores the converted value in the AD0DRn registers and store the parity operation result in the corresponding bit of the A/D parity register.
When identifier addition mode is selected, the information related to these is included in the result for which parity generation is performed.
As a result, if the identifier addition mode setting is changed after the A/D converter has written data to the AD0DRn register, the data format value subject to the parity check may be different at writing and reading when reading the AD0DRn register by the CPU, and a parity check error may occur.
The parity generated in this module is an even parity. If the automatic clearing function is enabled, the corresponding A/D parity bit is also cleared when the AD0DRn register is read from. Therefore, in a case requiring the A/D parity information, read this register before reading the AD0DRn register.

Note: n = 0 to 5

21.2.9 AD0 Analog Port Pull-Down Register (AD0APPD)

Address (H'FFFE7834)

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	AD0APPD5	AD0APPD4	AD0APPD3	AD0APPD2	AD0APPD1	AD0APPD0
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R
b5 to b0	AD0APPDn	Bits Controlling Pull-Down MOS Elements for Analog Port Pins	0: The pull-down MOS elements are switched off. 1: The pull-down MOS elements are switched on.	R/W

Note: n = 0 to 5

The AD0APPD register switches the pull-down MOS elements for the analog port pins off or on. After return from the power off state, this register is reset to 0.

Analog input and digital input pin functions are multiplexed with the I/O port pins for use with the 12-bit A/D converter.
If a pin is not in use as an analog input pin due to the port function selection register for the I/O port, the pull-down setting for the pin becomes invalid. However, this is not reflected in the port pull-down register.
For the details on port-function selection, see section 13.2.4, Pull-Up Control Register 0 (PUR0).

AD0APPDn Bit (n = 0 to 5)

[Setting Condition]
Setting an AD0APPDn bit to 1 turns on the pull-down MOS element attached to the analog port corresponding to the bit.
For an outline of the pull-down function of the analog ports, see Figure 21.8.

21.2.10 AD0 Parity Error Status Register (AD0PESR)

Address H'FFFE 78B0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	PES[5:0]					
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 b6	—	Reserved	These bits are read as 0. The write value should be 0.	R
b5 to b0	PES[5:0]	AD0DRn Register Parity Status Flag	0: No parity error has occurred. 1: Parity error has occurred.	R

Note: n = 0 to 5

The AD0PESR register performs a parity check for the data read from AD0DRn and store the check result.
Since a parity check is executed when the data is output to the CPU or DMAC, even if the data stored in the AD0DRn registers are erroneous, the error is not reflected in these registers until the data is read from the AD0DRn register. Therefore, to check if there are parity errors with these registers, first read the AD0DRn register to be parity checked and then read these registers.
The AD0PESR register is initialized to H'0000 by a reset.

PES Flag

[Setting condition]

- AD0DRn data output to the CPU or DMAC contains a parity error (value is updated when AD0DRn is read)

[Clearing conditions]

- AD0DRn data output to the CPU or DMAC has no parity error (value is updated when AD0DRn is read)
- 0 is written to the PESn bit after reading 1.

21.2.11 Interface with CPU

The AD0 data register is a 16-bit register. The peripheral bus connected to the CPU is 16 bits. The AD0 data register must be read in units of words. If the A/D data register is read in byte units by dividing a word into upper and lower bytes and performing read operations twice on it, the A/D converted value read in the first read operation and that read in the second read operation may change. To avoid this error, the A/D data register should not be read in byte units.

21.3 Operations

21.3.1 Scan Conversion

A scan conversion is performed in two operating modes: single-cycle scan mode and continuous scan mode. In single-cycle scan mode, one or more specified channels are scanned once. In continuous scan mode, one or more specified channels are scanned until the ADST bit in the AD0 control register (AD0CR) is cleared to 0 (changed from 1 to 0) by software.

Single-cycle scan mode is selected by clearing the ADCS bit in the AD0 control register (AD0CR) to 0, while continuous scan mode is selected by setting the ADCS bit to 1. When scan conversion is started in either mode, A/D conversion is performed for AD0INm channels selected by the AD0 channel select register (AD0CSL), starting from the channel with the lowest number m (m = 00 to 05). AD0IN00, AD0IN01, ... AD0IN05 in AD0.

In single-cycle scan mode, after A/D converting (scanning) all selected channels once, the A/D converter sets the ADF bit in the AD0 conversion status register (AD0CSR) to 1 and then clears the ADSCACT bit in the AD0CSR register to 0 to complete the scan conversion. In continuous scan mode, after scanning all selected channels once, the A/D converter sets the ADF bit to 1 and then continues scanning. The ADF bit is set to 1 each time scanning on a specified channel is completed.

To stop the scanning, write 0 to the ADST bit when it is 1. Writing 0 to the ADST bit when it is 0 does not affect the A/D converter. Similarly, writing 1 to the ADST bit when it is 1 does not affect the A/D converter. Therefore, to stop a scan conversion started by a request other than the ADST bit, first write 1 to the ADST bit and then write 0 to it.

When the ADF bit is set to 1 while the ADIE bit in the AD0CR register is set to 1, an ADI interrupt request is generated. To clear the ADF bit to 0, write 0 to the ADF bit after reading it as 1. When the DMAC is started by an ADI interrupt, the ADF bit is automatically cleared to 0 and the ADI interrupt is also cleared.

21.3.2 Single-Cycle Scan Conversion Mode

The following is an example operation of single-scan conversion where three channels AD0IN00, AD0IN03, and AD0IN05 are selected and an AD0I interrupt is enabled.

1. Clear the ADCS bit in the AD0 control register (AD0CR) to 0 and set the ADIE bit in the AD0CR register to 1.
2. Set the AD0CSL0, AD0CSL3, and AD0CSL5 bits in the AD0 channel select register (AD0CSL) to 1.
3. Set the ADST bit in the AD0 control register (AD0CR) to 1 to start scan conversion. If the ADST bit is already set to 1, write 1 to it after clearing it to 0.
4. Starting the scan conversion sets the ADSCACT bit to 1, and the following operations are performed in order.

(1) Sampling & analog value hold process

(2) Self-test conversion process

On completion of self test, the A/D conversion result with the self-test status added is stored in the AD0DRD register according to the DIAGLD and DIAGVAL bit settings in the AD0CEX register.

(3) A/D conversion

On completion of A/D conversion for AD0IN00, the A/D converted value is transferred to the AD0DR0 register.

(4) Operation (3) is performed for AD0IN03 and AD0IN05 in order.

5. When the A/D converted values of all the selected channels (AD0IN00, AD0IN03, and AD0IN05) have been transferred to the AD0DRn registers, the ending process is executed. After the ending process, the ADF bit is set to 1. At this time, an AD0I interrupt is generated since the ADIE bit is set to 1. The ADSCACT bit is cleared to 0 and the scan conversion is completed.
6. Next, the AD0I interrupt handler is started. In the interrupt handler, clear the AD0I interrupt by writing 0 to the ADF bit after reading it as 1. After that, read the contents of the AD0DR0, AD0DR3, and AD0DR5 registers.
7. Terminate the AD0I interrupt handler.

Figure 21.2 shows Example Operation in Single-Cycle Scan Mode (AD0).

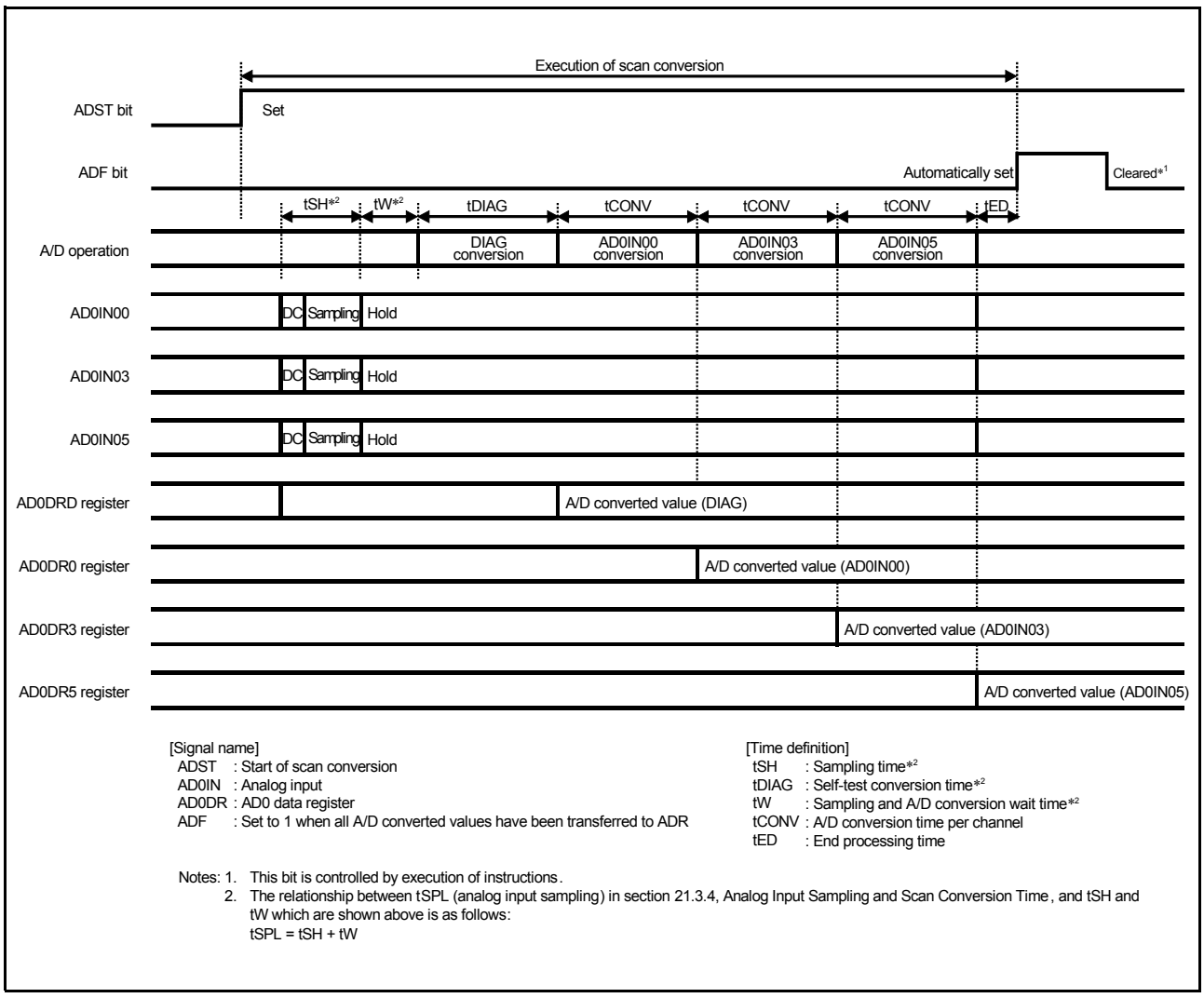


Figure 21.2 Example Operation in Single-Cycle Scan Mode (AD0)

21.3.3 Continuous Scan Conversion Mode

The following is an operation example of continuous scan conversion in which three channels AD0IN00, AD0IN03, and AD0IN05 are selected and an AD0I interrupt is enabled.

1. Set the ADCS and ADIE bits in the AD0 control register (AD0CR) to 1.
2. Set the AD0CSL0, AD0CSL3 and AD0CSL5 bits in the AD0 channel select register (AD0CSL) to 1.
3. Set the ADST bit in the AD0 control register (AD0CR) to 1 to start scan conversion. If the ADST bit is already 1, write 1 to it after clearing it to 0.
4. Starting scan conversion sets the ADSCACT bit to 1, and the following operations are performed in order.
   (1) Sampling & analog value hold process.
   (2) Self-test conversion process
   On completion of self test, the A/D conversion result with the self-test status added is stored in the AD0DRD register according to the DIAGLD and DIAGVAL bit settings in the AD0CEX register.
   (3) A/D conversion
   On completion of A/D conversion for AD0IN00, the A/D converted value is transferred to the AD0DR0 register.
   (4) Operation (3) is performed for AD0IN03 and AD0IN05 in order.
5. When the A/D converted values of all the selected channels (AD0IN00, AD0IN03, and AD0IN05) have been transferred to the AD0DRn registers, the ending process is executed. After the ending process, the ADF bit is set to 1. At this time, an AD0I interrupt is generated since the ADIE bit is set to 1.
   Then, step 4 is repeatedly executed to perform scan conversion.
6. The AD0I interrupt handler is started simultaneously. In the interrupt handler, clear the AD0I interrupt by writing 0 to the ADF bit after reading it as 1. After that, read the contents of the AD0DR0, AD0DR3, and AD0DR5 registers.
7. Terminate the AD0I interrupt handler.
8. Steps 4 to 7 are repeated as long as the ADST bit is 1. Clearing the ADST bit to 0 clears the ADSCACT bit to 0, and completes scan conversion. To start scan conversion again, set the ADST bit to 1.

Figure 21.3 shows Example Operation in Continuous Scan Mode (AD0).

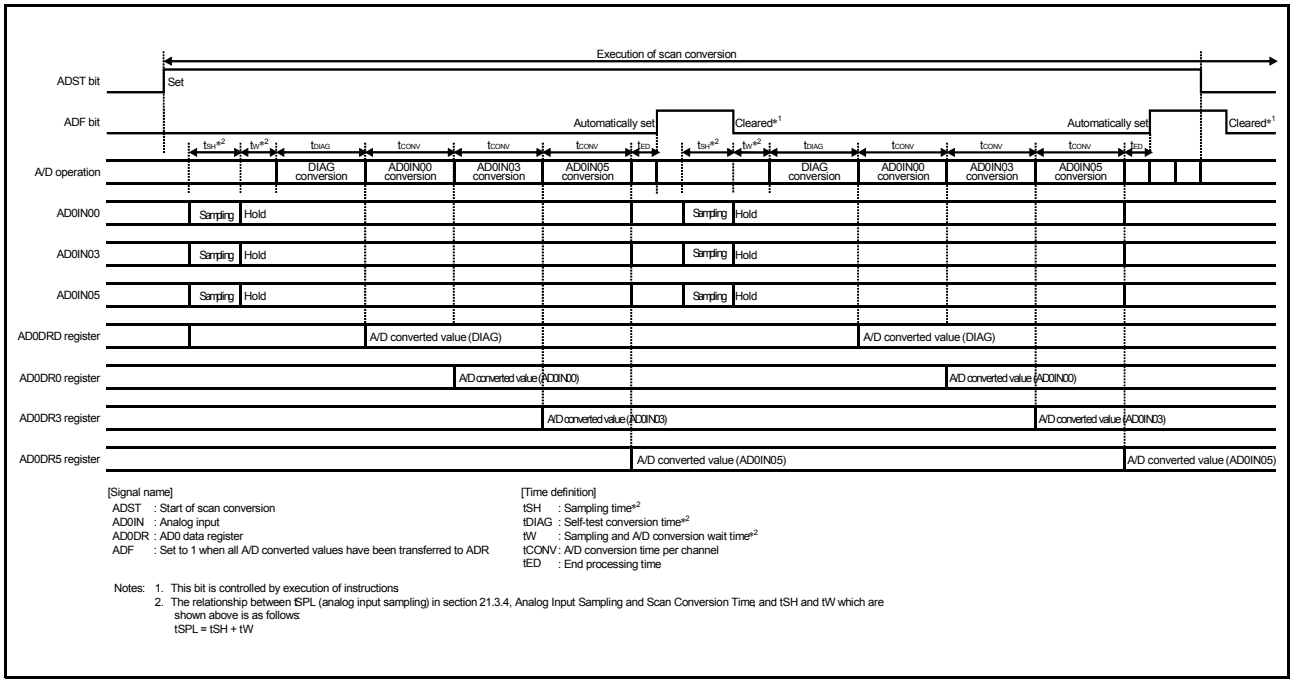


Figure 21.3 Example Operation in Continuous Scan Mode (AD0)

21.3.4 Analog Input Sampling and Scan Conversion Time

AD0 includes sample-and-hold circuits. Scan conversion can be activated by software, MTU-III trigger, or external trigger (the ADTRG# pin)*1. In scan conversion, when start-of-scan-conversion delay time (tD) has passed after the ADST bit is set to 1, AD0 samples the analog input, performs the Diag conversion process, and then begins the A/D conversion process.

Figure 21.4 shows a timing chart for a scan conversion by software activation in single-cycle scan mode. Figure 21.5 shows a timing chart for a scan conversion by MTU-III trigger source and external trigger (the ADTRG# pin) source in single-cycle scan mode.

Scan conversion time (tSCAN) includes start-of-scan-conversion delay time (tD), analog input sampling time (tSPL), self-test conversion time (tDIAG)*2, A/D conversion processing time (tCONV), and end-of-scan-conversion delay time (tED). Table 21.9 lists Scan Conversion Time (AD0).

The scan conversion time (tSCAN) in single-cycle scan mode for which the number of selected channels is n can be determined as follows:

$$tSCAN = tD + tSPL + tDIAG + (tCONV \times n) + tED$$

The scan conversion time for the first cycle in continuous scan mode is tSCAN for single-cycle scan minus tED.

The scan conversion time for the second and subsequent cycles in continuous scan mode is a fixed period equal to tSPL + tDIAG + (tCONV × n).

Notes: 1.Because the SH72A0 group does not have the external trigger pin (ADTRG#), scan conversion cannot be activated by external trigger.
2.When self test is not to be performed, tDIAG becomes 0.

Table 21.9 Scan Conversion Time (AD0)

Item		Symbol		f (PBA) (cyc) ADTRG# (External Trigger)	f (PBA) (cyc) MTU-III, Trigger Source	f (PBA) (cyc) Software Activation	Unit
Start-of-scan-conversion delay time *2		tD		7 *1	6	7	State
Analog input sampling time	Sampling-of-pin time	tSPL	tSH	20	20	20	
	Sampling-of-pin and A/D conversion wait time		tW	10	10	10	
Self-test conversion time *2		tDIAG		50	50	50	
A/D conversion processing time *2		tCONV		50	50	50	
End-of-scan-conversion delay time *2		tED		12	12	12	
Scan conversion time *3		tSCAN		399 (Max.)	398 (Max.)	399 (Max.)	

Notes: 1. For the external trigger input delay time, refer to “A/D Conversion Timing” in section 30, Electrical Characteristics.
2. For each timing of tD, tDIAG, tCONV, and tED, see Figure 21.4 and Figure 21.5.
3. The maximum value of the scan conversion time is generated by selecting a self-test conversion and using a 6 channel scan conversion.

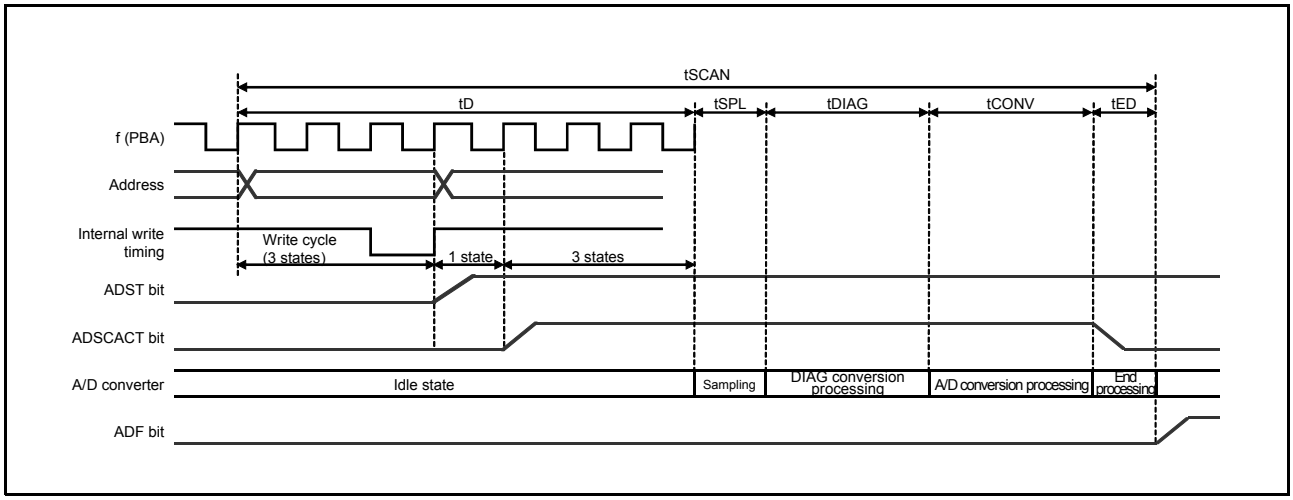
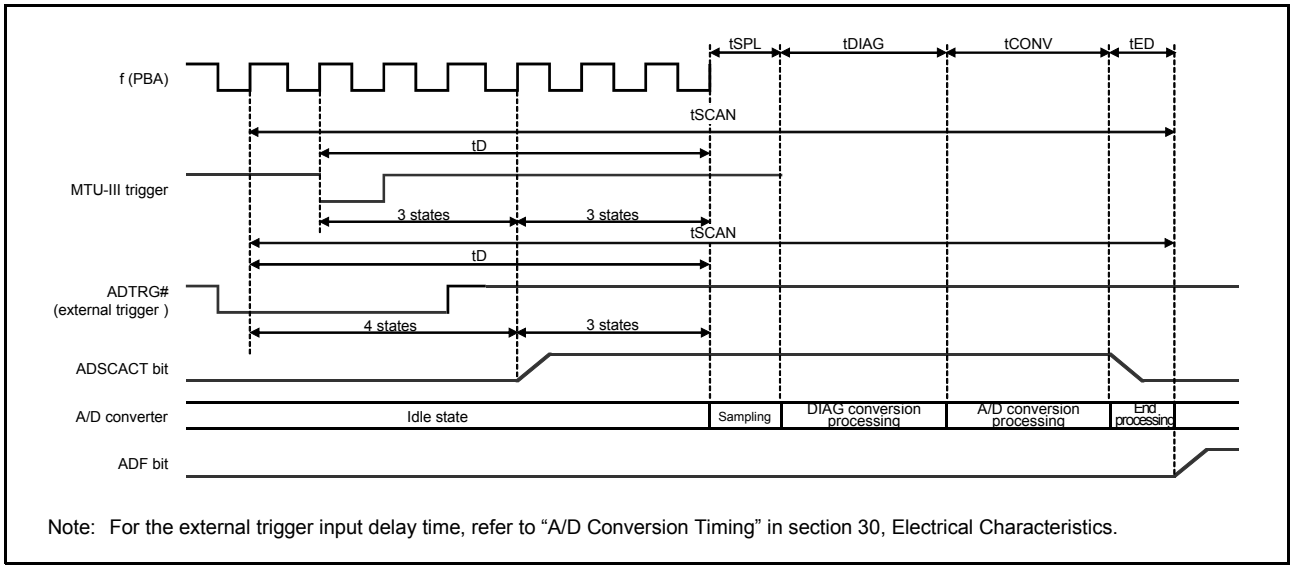


Figure 21.4 AD0 Scan Conversion Timing (Generated by Software Activation)



Note: For the external trigger input delay time, refer to “A/D Conversion Timing” in section 30, Electrical Characteristics.

Figure 21.5 AD0 Scan Conversion Timing (Generated by MTU-III, Trigger Source, or ADTRG# Source)

21.3.5 Access to Data Register (AD0DRn and AD0DRD)

The A/D data registers (AD0DRn and AD0DRD) are 16-bit registers used to store the A/D conversion results. When storing the A/D conversion result in the AD0 data register, parity operation (identifier included) is performed and the parity result is held. When the CPU or DMAC reads from the AD0 data register, the saved parity result and the AD0 data register value are parity checked. If the parity check results do not match, the respective bit in the AD0PR register is set to 1. If interrupts are enabled by the PAIE bit in the AD0CEX register at this time, a parity error interrupt request is sent to the interrupt controller. Note that even when the parity check result is a mismatch, the access to the data register is valid and data will be read from the AD0 data register.

21.3.6 Usage Example of AD0 Data Register Automatic Clearing Function

Setting the ACE bit in the AD0CEX register to 1 automatically clears the AD0 data registers (AD0DRn and AD0DRD) to H'0000 when the AD0DRn and AD0DRD registers are read by the CPU or DMAC.

This function is used to detect update failures of the AD0DRn and AD0DRD registers.

Examples in which the function to automatically clear the AD0 data register is enabled and disabled are shown below.

In a case where the ACE bit is 0 (initial value), if the A/D conversion result (H'0222) is not written to the AD0 data register for some reason, the old data (H'0111) will become the AD0 data register value. Furthermore, if this AD0 data register value is written to a general register using an A/D scan conversion end interrupt, the old data (H'0111) can be saved in the general register. When checking data that has not been updated, it is necessary to frequently save the old data in the RAM or a general register.

In a case where the ACE bit is 1, when the AD0 data register value = H'0111 is read by the CPU or DMAC, the AD0 data register value is automatically cleared to H'0000. After that, if the A/D conversion result of H'0222 cannot be transferred to the AD0 data register for some reason, the cleared data (H'0000) remains as the AD0 data register value. If this AD0 data register value is written to a general register using an A/D scan conversion end interrupt at this point, H'0000 will be saved in the general register. Occurrence of an AD0 data register update failure can be determined by simply checking whether the read data value is H'0000.

If automatic clearing of the AD0 data register is selected, when data is read from the AD0 data register, the converted data and identifier bits as well as the respective bits in the A/D parity registers will be cleared. Even when reading and writing conflict in the microcomputer, a parity error will not be accidentally generated.

21.3.7 Identifier Addition Function

When the IDE bit in the AD0CEX register is set to 1, the identifier corresponding to the converted channel will be stored in the lower four bits at the same time the A/D conversion result is stored in the AD0 data register (AD0DRn).

When the identifier addition function is enabled, the A/D data register format is fixed to left-alignment.

In A/D-converted value addition mode, only the upper 12 bits of the addition result will be valid.

21.3.8 Starting Scan Conversion with External Trigger

AD0 can be activated by the input of an external trigger. (Only the SH72A2 group. The SH72A0 group cannot activate AD0 by external trigger.) After setting AD0 start trigger select register (AD0STRSL) to H'00 and applying a high-level signal to the ADTRG# pin, both the TRGE and EXTRG bits in AD0 control register (AD0CR) should be set to 1. If a low-level signal is then input to the ADTRG# pin, AD0 will detect a falling edge of the pulse and set the ADSCACT bit to 1. Figure 21.6 shows External Trigger Input Timing.

The timing at which a scan conversion is started after the ADSCACT bit is set to 1 is the same as the case where the ADST bit is set to 1 from 0 by software. For details on pin function settings, refer to section 13, I/O Ports.

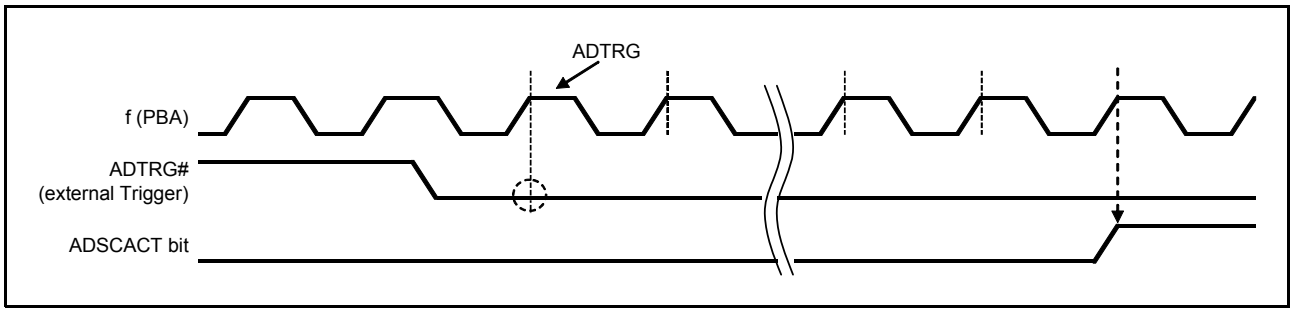


Figure 21.6 External Trigger Input Timing

21.3.9 Starting Scan Conversion with Trigger from Peripheral Modules

A scan conversion can be activated by a timer trigger of the MTU-III. To start up a scan conversion by a timer trigger, the TRGE bit should be set to 1 and the EXTRG bit should be cleared to 0 in AD0 control register (AD0CR) and the relevant source bit should be selected by the ADSTRS bit in the AD0 start trigger select register (AD0STRSL). If a startup source is entered in this situation, the ADSDACT bit is set to 1. The timing at which a scan conversion is started after the ADSCACT bit is set to 1 is the same as the case where the ADST bit is set to 1 from 0 by software.

21.4 Interrupt Sources and DMA Transfer Request

21.4.1 Interrupt Request on Completion of Each Scan Conversion

AD0 can send scan conversion end interrupt requests AD0 to the CPU.

By setting the ADIE bit in the AD0 control register (AD0CR) to 1, AD0I interrupt is enabled; by clearing the ADIE bit to 0, AD0I interrupt is disabled.

In addition, the DMAC can be started up when AD0I interrupt is generated. In this case, interrupt is not sent to the CPU. If the DMAC is started upon AD0I interrupt, the ADF bit in the AD0 conversion status register (AD0CSR) is automatically cleared to 0 when data transfer is performed by the DMAC.

For details on DMAC settings, refer to section 12, DMAC.

Note: The ADF bit is not cleared by an interrupt request to the CPU.

21.5 Definition of A/D Conversion Accuracy

The definition of A/D conversion accuracy is described below.

- Resolution

This indicates the number of digital output codes in the A/D converter.

- Offset error

This error, which is exclusive of quantization error, is a deviation of the analog input voltage value from the ideal A/D conversion characteristics when the digital output changes from a minimum voltage value B'0000 0000 0000 to B'0000 0000 0001 (Figure 21.7).

- Full scale error

This error, which is exclusive of quantization error, is a deviation of the analog input voltage value from the ideal A/D conversion characteristics when the digital output changes from B'1111 1111 1110 to B'1111 1111 1111 (Figure 21.7).

- Quantization error

This error, which is inherent to the A/D converter, is given as 1/2LSB (Figure 21.7).

- Nonlinearity error

This error, which is exclusive of offset error, full scale error and quantization error, is a deviation from the ideal A/D conversion characteristics through the zero-scale and full-scale transitions.

- Absolute accuracy

This is a deviation of the digital value from the analog input value. This includes offset error, full scale error, quantization error, and nonlinearity error.

Figure 21.7 shows Definition of A/D Conversion Accuracy.

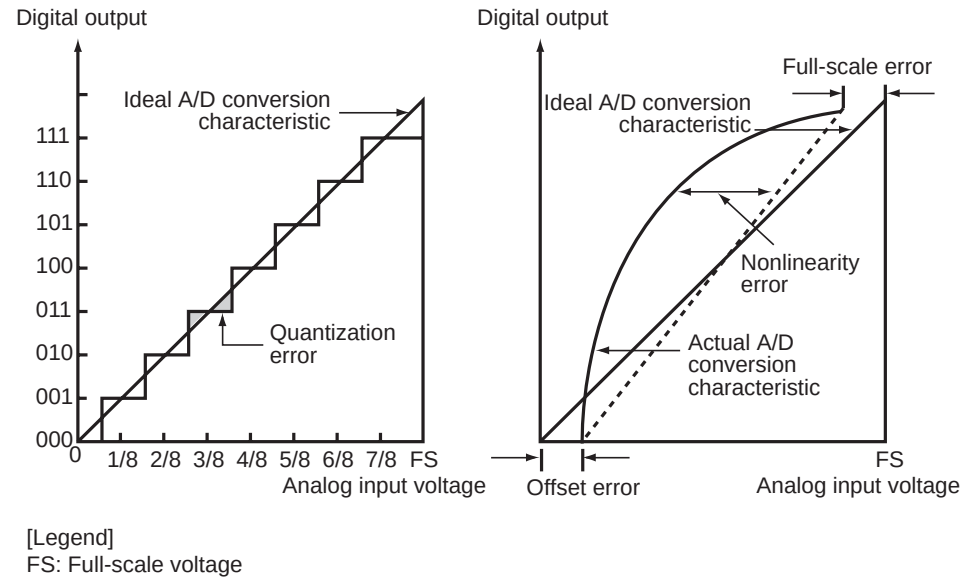


Figure 21.7 Definition of A/D Conversion Accuracy

21.6 Pull-Down Function of Analog Ports

Each analog port has a pull-down MOS element. Setting the AD0APPD5 to AD0APPD0 bits to 1 (the default is 0) turns on the pull-down MOS elements for the corresponding analog channels.

When voltage from an external circuit is being applied to an analog port pin, the result of A/D conversion can be used to check for disconnection between them.

- Examples

Normal operation: The result of conversion is near the voltage applied from the external circuit.

Abnormal operation (disconnection): The result of conversion is near AVSS.

Note: Analog input and digital input pin functions are multiplexed with the I/O port pins for use with the 12-bit A/D converter. If a pin is not in use as an analog input pin due to the port function selection register for the I/O port, the pull-down setting for the pin becomes invalid. However, this is not reflected in the port pull-down register. For the details on port-function selection, see section 13.2.24, Port Ni Function Select Register (PNiS) (i = 00 to 05).

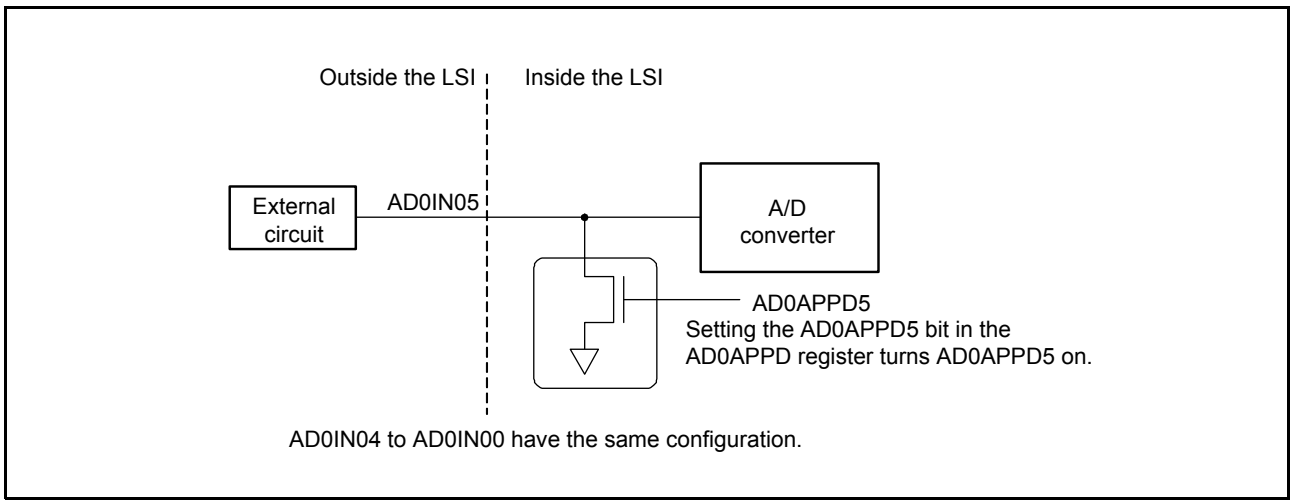


Figure 21.8 Schematic of Analog Port Pull-Down Function

21.7 Notes on 12-Bit A/D Converter (AD0)

21.7.1 Analog Input Voltage Range

The voltage applied to an analog input pin during A/D conversion should be within the following range:
VREFL0 ≤ AD0INm (m = 00 to 05) ≤ VREFH0

21.7.2 Relationship among AVCC0, AVSS0, VCC, and VSS

When using the A/D converter, make sure that the following relationships are held among AVCC0, AVSS0, VCC and VSS:
AVCC0 = 5.0V ± 0.5V, AVCC0 ≤ VCC + 1.0V, AVSS0 = VSS
When the A/D converter is not used, AVCC pin must not be open. In this case, the following relationship should be held between AVSS and VSS:
AVSS0 = VSS, AVCC0 = VCC

21.7.3 Allowable Settings for Pins VREFH0 and VREFL0

The allowable settings for the VREFH0 pin are as follows:
4.5 V ≤ VREFH0 ≤ AVCC0 (When the A/D converter is used)
AVSS ≤ VREFH0 ≤ AVCC0 (When the A/D converter is not used)
If the above relationships are not held, it can adversely affect the reliability of the LSI.
For the VREFL0 pin, the following relationships should be observed.
VREFL0 = AVSS0 = VSS

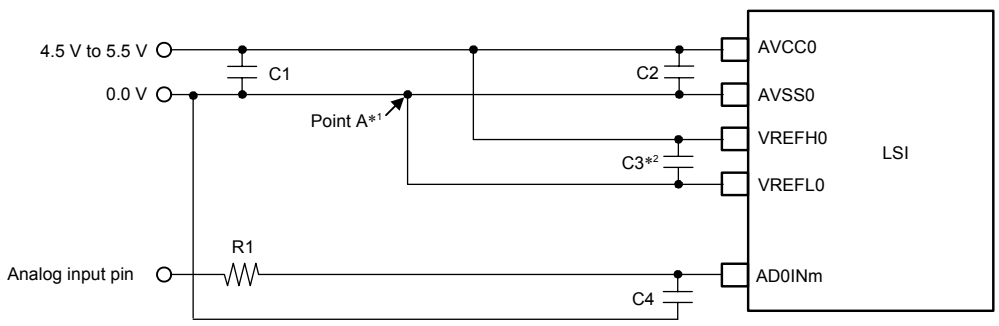
21.7.4 Precautions on Board Design

For designing a board, to the maximum extent possible the digital circuits should be laid out separately from the analog circuits. Layouts involving the crossing of signal lines for digital circuits and signal lines for analog circuits, or placing them in proximity to each other, should be avoided. If the dissimilar signal lines are placed in close proximity to each other, the resulting induction can lead to a malfunction of the analog circuits or produce an adverse impact on A/D conversion values.
It should be noted that the analog input pins (AD0IN00 to AD0IN05), the analog reference voltages (VREFH0, VREFL0), and the analog power supply (AVCC0) should be isolated from the digital circuits by means of analog grounding (AVSS0). In addition, the analog ground (AVSS0) should be connected in one point to a stable digital ground (VSS) on the board. To prevent the influence of the noise generated by the 10-bit A/D converter, separate AVCC0 from AVCC1 as much as possible and design the board so that the common impedance is small.

21.7.5 Precautions on Noise Measures

A protection circuit to prevent the analog input pins (AD0INm: m = 00 to 05) from damages, by such abnormal voltages as surges, should be connected between AVCC0 and AVSS0 and between VREFH0 and VREFL0, as shown in figures 21.9 and 21.10. Connect a capacitor as a filter between the analog input pin (AD0INm) and AVSS0.
The values for the bypass capacitors between AVCC0 and AVSS0 and between VREFH0 and VREFL0 in figures 21.9 and 21.10 are for reference. Consider the actual values when designing the board. Also bypass capacitors connected between VREFH0 and VREFL0, or a filter capacitor connected to an analog input pin (AD0INm), should be connected to the AVSS0. Since connecting a filter capacitor as shown in figures 21.9 and 21.10 can cause an error by averaging the input currents to analog input pins (AD0INm), care must be taken to choose appropriate circuit constants. Figure 21.9 is a basic connection example of power supplies. Figure 21.10 is a connection example of power supplies, which is not subject to the board. This connection can lead to a potential difference between AVCC0

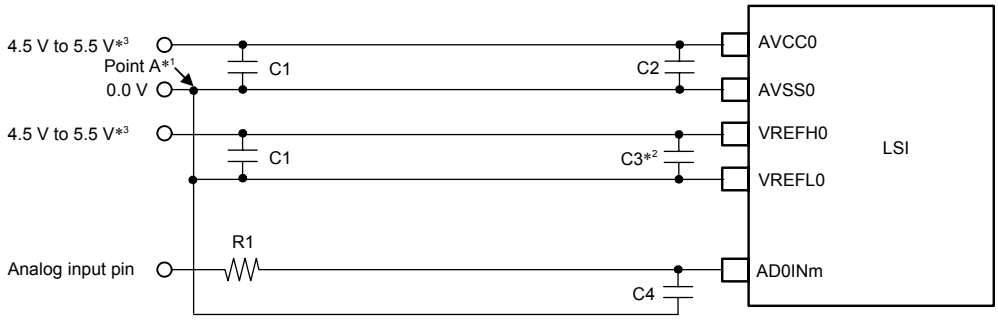
and VREFH0.



[Reference value] C1 10.0 µF to 100.0 µF*3
C2, C3 0.1 µF to 1.0 µF*3
C4 0.1 µF
R1 3 kΩ

Notes: 1. A potential difference between AVSS0 and VREFL0 will lead to an offset error. Accordingly, design the board so that the line impedances at AVSS (point A) and that at VREFL0 are small and equal.
2. If the full-scale error worsens, VREFH0 may be unstable. To improve the stability, decrease the line impedance at VREFH0 or increase the capacitance of C3.
3. Determine the ideal value by evaluating the capacitor on the actual board where it is to be used.

Figure 21.9 Basic Connection Example of Analog Power Supplies and Analog Input Pin



[Reference value] C1 10.0 µF to 100.0 µF*4
C2, C3 0.1 µF to 1.0 µF*4
C4 0.1 µF
R1 3 kΩ

Notes: 1. A potential difference between AVSS0 and VREFL0 will lead to an offset error. Accordingly, design the board so that the line impedances at AVSS (point A) and that at VREFL0 are small and equal.
2. If the full-scale error worsens, VREFH0 may be unstable. To improve the stability, decrease the line impedance at VREFH0 or increase the capacitance of C3.
3. Providing separate power supplies for AVCC0 and VREFH0 can lead to a potential difference between the two. Make sure that VREFH0 is from 4.5 V to AVCC0 and the ground is common. Consider note 1 when designing the board.
4. Determine the ideal value by evaluating the capacitor on the actual board where it is to be used.

Figure 21.10 Connection Example of Analog Power Supplies and Analog Input Pin (Not Subject to Board and Leading to Potential Difference between AVCC0 and VREFH)

21.7.6 Precautions on Transitions to Low Power Mode

Stop A/D conversion before changing the mode to standby mode.

After setting the ADST bit in AD0CR, wait until the analog circuits in the A/D converter have stopped. To secure this waiting time, set the bits according to the following procedure.

(1) Set the TRGE bit in AD0CR to 0.
(2) Set the ADST bit in AD0CR to 0.
(3) Check that A/D conversion stops.
(Stopping A/D conversion requires at least (f(PBA) × six cycles).)
(4) Change the mode to standby mode.

21.7.7 Precautions on Using Port Pins

Use the following pins with care when using the 12-bit A/D converter in the SH72A2 Group.

(1) The PL10, PL11, and PL12 pins should not be used as the port output pins.
(2) The PL12 pin should not be used as the output compare output pin or the PWM output pin with the TP04A function.

22. 10-Bit A/D Converter (AD1)

This LSI includes a 10-bit successive approximation A/D converter, which consists of one independent unit (AD1). Up to 24 channel analog inputs can be selected in the SH72A2 group. Up to 8 channel analog inputs can be selected in the SH72A0 group by software.
In this section, explanations for the SH72A2 group are described.
Concerning the 12-bit A/D converter, refer to section 21, 12-Bit A/D Converter (AD0).

Note: Some of the AD1 analog pin numbers are missing. The pin numbers are inconsecutive. Analog pin numbers AD1IN00 to AD1IN47 indicate that the following pins are present.

(1) SH72A2 Group:
AD1IN00 to AD1IN07, AD1IN16, AD1IN17, AD1IN24 to AD1IN31, AD1IN42 to AD1IN47

(2) SH72A0 Group:
AD1IN04 to AD1IN05, AD1IN16, AD1IN17, AD1IN24, AD1IN28, AD1IN30, AD1IN31

22.1 Introduction

Table 22.1 and 22.2 list the Specifications of 10-Bit A/D Converter (AD1).

Table 22.1 Specifications of 10-Bit A/D Converter (AD1) (1)

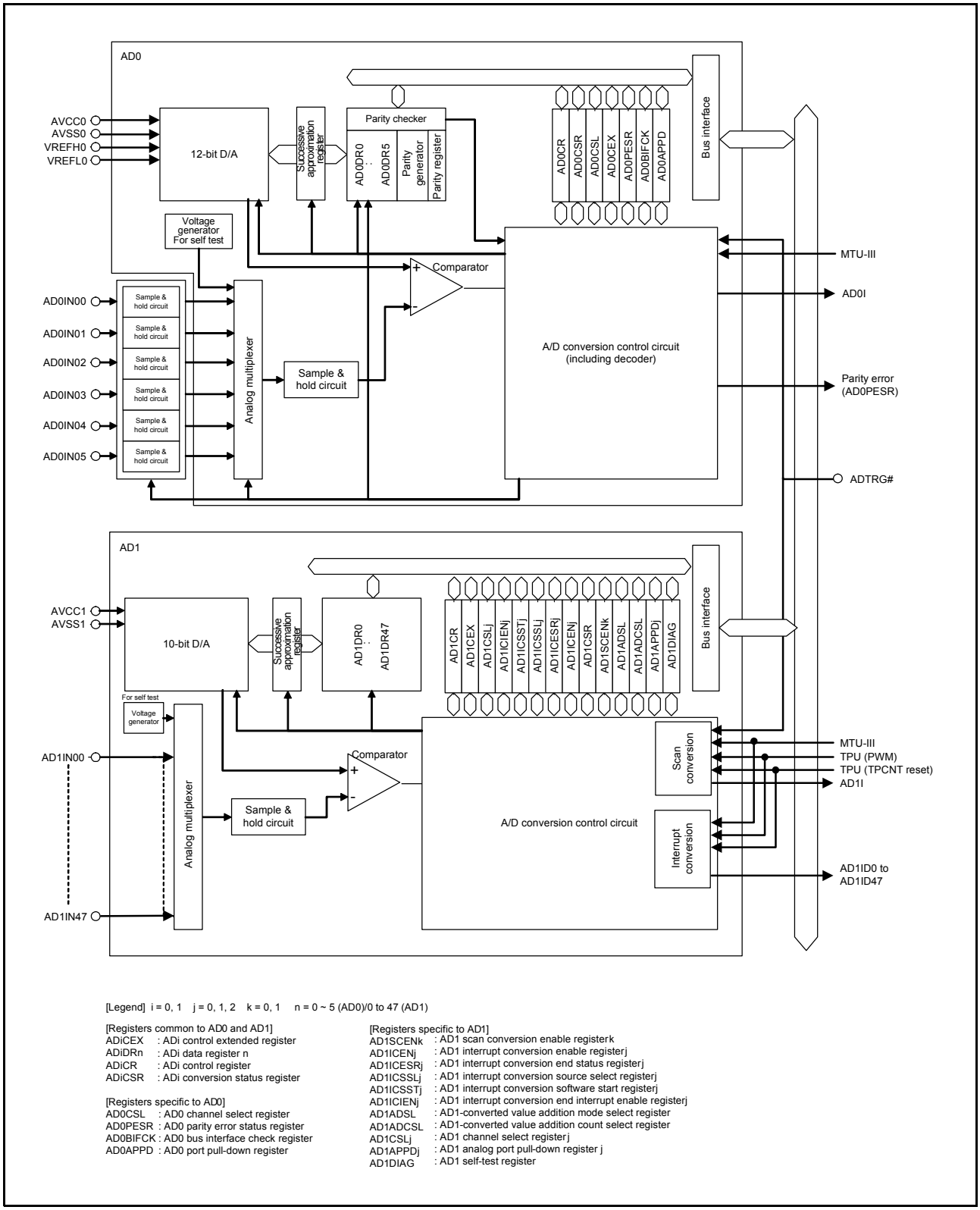
Item	Description
Resolution	• 10 bits
Input channels	• 24 channels (AD1: 24 channels (AD1IN00 to AD1IN47)*1
Minimum conversion time	• AVCC = 5 V: 1.0 µs/channel (f (PBA) = 50 MHz, 50 conversion states)
Scan conversion modes	• 2 modes Single cycle scan mode: Scanning performed only once Continuous scan mode: Scanning performed repeatedly The channels subject to scanning are selectable. A/D conversion proceeds in order from lower-numbered to higher-numbered channels (AD1IN00 to AD1IN47 for AD1).
A/D-converted value addition mode	The same channel is A/D converted two to four times in succession, and the sum of the converted values is stored in the AD1 data register. A/D-converted value addition mode supports channels AD1IN00 to AD1IN07. The use of the average of the results can improve the precision of A/D conversion, depending on the types of noise components that are present. This function, however, cannot guarantee an improvement in A/D conversion accuracy.*2
Registers	• Twenty-four 10-bit AD data registers*3
Sample and hold function	• A/D converter module (AD1) includes a sample and hold circuit.
Two ways of starting scan conversion	• AD1: Selectable among software trigger (the ADST bit in the AD1CR register), external trigger (ADTRG#)*4, TPU (PWM) timer trigger (TPO1A to TPO4A), resets TP1CNT to TP4CNT, and MTU-III timer trigger (TRG0N, TRGA0N to TRGA4N, TRGA6N, or TRGA7N)

Notes: 1. The SH72A0 group has eight channel.
2. The SH72A0 group supports AD1IN04 to AD1IN05.
3. The SH72A0 group has eight registers.
4. The SH72A0 group has no external trigger (the ADTRG# pin).

Table 22.2 Specifications of 10-Bit A/D Converter (AD1) (2)

Item	Description
Interrupt-triggered conversion	• AD1: Independently from scan conversion, it is possible to preferentially process channels requested by TPU (PWM) timer trigger (TPU1 to TPU4), resets TP1CNT to TP4CNT, MTU-III timer trigger (TRG0N, TRGA0N to TRGA4N, TRGA6N, or TRGA7N), or software trigger for A/D conversion. This function supports channels AD1IN00 to AD1IN47. A/D conversion is executed on only AD1IN04 to AD1IN07, AD1IN16, and AD1IN17 for which the interrupt conversion is requested by the TPU timer trigger or the resets. When an interrupt conversion and a scan conversion conflict, the scan conversion is suspended and A/D conversion is executed preferentially on the channel for which the interrupt conversion was requested. On completion of the interrupt conversion, the scan conversion is resumed on the channel there A/D conversion was interrupted.
Support for scan conversion end interrupt (ADI), interrupt conversion end interrupt (ADID), and DMA transfer function	• On completion of scanning for scan conversion, a scan conversion end interrupt request (ADI) can be generated or the DMAC can be started. On completion of interrupt conversion on channels AD1IN00 to AD1IN47, an interrupt conversion end interrupt request (AD1ID0 to AD1ID47) can be generated, or the DMAC (AD1ID0 to AD1ID47) can be started.
Pull-down function of analog ports	This function can check for disconnection between an external circuit and an analog port pin.
Self-test function of A/D converter	The self-test function of the A/D converter performs the A/D conversion of internally generated voltage values (AVCC1 × 0, AVCC1 × 1/2, AVCC1 × 1). Have software read ADDR and ADDIARG and check whether or not the A/D-converted values are in the normal ranges on completion of A/D conversion.

Figure 22.1 shows Block Diagram of A/D Converter (AD0 and AD1).



[Legend] i = 0, 1 j = 0, 1, 2 k = 0, 1 n = 0 ~ 5 (AD0)/0 to 47 (AD1)

[Registers common to AD0 and AD1]
ADiCEX : ADi control extended register
ADiDRn : ADi data register n
ADiCR : ADi control register
ADiCSR : ADi conversion status register

[Registers specific to AD0]
AD0CSL : AD0 channel select register
AD0PESR : AD0 parity error status register
AD0BIFCK : AD0 bus interface check register
AD0APPD : AD0 port pull-down register

[Registers specific to AD1]
AD1SCENk : AD1 scan conversion enable registerk
AD1ICENj : AD1 interrupt conversion enable registerj
AD1ICESRj : AD1 interrupt conversion end status registerj
AD1ICSSLj : AD1 interrupt conversion source select registerj
AD1ICSSTj : AD1 interrupt conversion software start registerj
AD1ICIENj : AD1 interrupt conversion end interrupt enable registerj
AD1ADSL : AD1-converted value addition mode select register
AD1ADCSL : AD1-converted value addition count select register
AD1CSLj : AD1 channel select registerj
AD1APPDj : AD1 analog port pull-down register j
AD1DIAG : AD1 self-test register

Figure 22.1 Block Diagram of A/D Converter (AD0 and AD1)

Table 22.3 to Table 22.5 list the functions assigned to each channel for A/D converter.

Table 22.3 Functions Assigned to Each Channel for AD0

A/D Converter	A/D Conversion Channel	Scan Conversion									AD0 Data Register (Storage Register for A/D-Converted Value)
		Trigger Source				Self-Test Selection or Channel Selection	Scan Conversion Mode	End Interrupt/ DMA Transfer Request Signal	DMA Transfer Function (DMAC)	End Flag (at Scan Completion)	
		Source 1	Source 2	Source 3	Source 4						
	Self Test or Pin Name	Software	External Trigger		MTU-III	AD0CEX or AD0CSL					
AD0 6 channels	Self-Test DIAG	ADST (AD0CR)	ADTRG# pin*		—	DIAGM (AD0CEX)	Single cycle scanning or continuous scanning	AD0I	O	ADF (AD0CSR)	AD0DRD
	AD0IN00				TRGA0N, TRG0N, TRGA1N, TRGA2N, TRGA3N, TRGA4N, TRG4AN, TRG4BN, TRG4ABN, TRGA6N, TRGA7N, TRG7AN, TRG7BN, TRG7ABN	AD0CSL0					AD0DR0
	AD0IN01					AD0CSL1					AD0DR1
	AD0IN02					AD0CSL2					AD0DR2
	AD0IN03					AD0CSL3					AD0DR3
	AD0IN04					AD0CSL4					AD0DR4
	AD0IN05					AD0CSL5					AD0DR5

Note: * The SH72A0 group has no external trigger (ADTRG#).

Table 22.4 Functions Assigned to Each Channel for AD1 (1)

A/D Converter	A/D Conversion Channel*1	Scan Conversion									AD1 Data Register (Storage Register for A/D-Converted Value)	A/D-Converted Value Addition Mode
		Trigger Source				Channel Selection	Scan Conversion Mode	End Interrupt/ DMA Transfer Request Signal	DMA Transfer Function (DMAC)	End Flag (at Scan Completion)		
		Source 1	Source 2	Source 3	Source 4							
	Pin Name	Software	External Trigger	TPU	MTU-III	AD1CSL						AD1ADSL
AD1 24 channels	AD1IN00	ADST (AD1CR)	ADTRG# pin *2	TPO1A TP1CNT reset TPO2A TP2CNT reset TPO3A TP3CNT reset TPO4A TP4CNT reset	TRG0N, TRGA0N, TRGA1N, TRGA2N, TRGA3N, TRGA4N, TRGA6N, TRGA7N	AD1CSL0	Single cycle scanning or continuous scanning	AD1I	○	ADF (AD1CSR)	AD1DR0	AD1ADSL0
	AD1IN01					AD1CSL1					AD1DR1	AD1ADSL1
	AD1IN02					AD1CSL2					AD1DR2	AD1ADSL2
	AD1IN03					AD1CSL3					AD1DR3	AD1ADSL3
	AD1IN04					AD1CSL4					AD1DR4	AD1ADSL4
	AD1IN05					AD1CSL5					AD1DR5	AD1ADSL5
	AD1IN06					AD1CSL6					AD1DR6	AD1ADSL6
	AD1IN07					AD1CSL7					AD1DR7	AD1ADSL7
	AD1IN16					AD1CSL16					AD1DR16	×
	AD1IN17					AD1CSL17					AD1DR17	×
	AD1IN24					AD1CSL24					AD1DR24	×
	AD1IN25					AD1CSL25					AD1DR25	×
	AD1IN26					AD1CSL26					AD1DR26	×
	AD1IN27					AD1CSL27					AD1DR27	×
	AD1IN28					AD1CSL28					AD1DR28	×
	AD1IN29					AD1CSL29					AD1DR29	×
	AD1IN30					AD1CSL30					AD1DR30	×
	AD1IN31					AD1CSL31					AD1DR31	×
	AD1IN42					AD1CSL42					AD1DR42	×
	AD1IN43					AD1CSL43					AD1DR43	×
	AD1IN44					AD1CSL44					AD1DR44	×
	AD1IN45					AD1CSL45					AD1DR45	×
	AD1IN46					AD1CSL46					AD1DR46	×
	AD1IN47					AD1CSL47					AD1DR47	×

Notes: 1. The SH72A0 group has the following eight A/D conversion channels:
AD1IN04, AD1IN05, AD1IN16, AD1IN17, AD1IN24, AD1IN28, AD1IN30, and AD1IN31.
2. The SH72A0 group has no external trigger (ADTRG#).

Table 22.5 Functions Assigned to Each Channel for AD1 (2)

A/D Converter	A/D Conversion Channel*	Interrupt Conversion										
		Trigger Source				End Interrupt/ DMA Transfer Request Signal	DMA Transfer Function	Enable (Channel Selection)	Source Selection	End Status	End Interrupt Enable	Start Interrupt Conversion Software
		Source 1	Source 2	Source 3	Source 4							
	Pin Name	TPU	TPU	MTU-III	Software (AD1ICSST)		DMAC	AD1ICEN	AD1ICSSL	AD1ICESR	AD1ICIEN	AD1ICSST
AD1 24 channels	AD1IN00			TRGA0N	AD1ICSST0	AD1ID0	○	AD1ICEN0	AD1ICSSL0	AD1ICESR0	AD1ICIEN0	AD1ICSST0
	AD1IN01			TRGA0N	AD1ICSST1	AD1ID1	○	AD1ICEN1	AD1ICSSL1	AD1ICESR1	AD1ICIEN1	AD1ICSST1
	AD1IN02			TRGA1N	AD1ICSST2	AD1ID2	○	AD1ICEN2	AD1ICSSL2	AD1ICESR2	AD1ICIEN2	AD1ICSST2
	AD1IN03			TRGA1N	AD1ICSST3	AD1ID3	○	AD1ICEN3	AD1ICSSL3	AD1ICESR3	AD1ICIEN3	AD1ICSST3
	AD1IN04	TPO1A A/D conversion trigger	TP1CNT counter reset	TRGA2N	AD1ICSST4	AD1ID4	○	AD1ICEN4	AD1ICSSL4	AD1ICESR4	AD1ICIEN4	AD1ICSST4
	AD1IN05	TPO1B A/D conversion trigger		TRGA2N	AD1ICSST5	AD1ID5	○	AD1ICEN5	AD1ICSSL5	AD1ICESR5	AD1ICIEN5	AD1ICSST5
	AD1IN06	TPO1C A/D conversion trigger		TRGA3N	AD1ICSST6	AD1ID6	○	AD1ICEN6	AD1ICSSL6	AD1ICESR6	AD1ICIEN6	AD1ICSST6
	AD1IN07	TPO1D A/D conversion trigger		TRGA3N	AD1ICSST7	AD1ID7	○	AD1ICEN7	AD1ICSSL7	AD1ICESR7	AD1ICIEN7	AD1ICSST7
	AD1IN16	TPO4A A/D conversion trigger	TP4CNT counter reset	TRGA0N	AD1ICSST16	AD1ID16	○	AD1ICEN16	AD1ICSSL16	AD1ICESR16	AD1ICIEN16	AD1ICSST16
	AD1IN17	TPO4B A/D conversion trigger		TRGA0N	AD1ICSST17	AD1ID17	○	AD1ICEN17	AD1ICSSL17	AD1ICESR17	AD1ICIEN17	AD1ICSST17
	AD1IN24			TRGA4N	AD1ICSST24	AD1ID24	○	AD1ICEN24	AD1ICSSL24	AD1ICESR24	AD1ICIEN24	AD1ICSST24
	AD1IN25			TRGA4N	AD1ICSST25	AD1ID25	○	AD1ICEN25	AD1ICSSL25	AD1ICESR25	AD1ICIEN25	AD1ICSST25
	AD1IN26			TRG0N	AD1ICSST26	AD1ID26	○	AD1ICEN26	AD1ICSSL26	AD1ICESR26	AD1ICIEN26	AD1ICSST26
	AD1IN27			TRG0N	AD1ICSST27	AD1ID27	○	AD1ICEN27	AD1ICSSL27	AD1ICESR27	AD1ICIEN27	AD1ICSST27
	AD1IN28			TRGA6N	AD1ICSST28	AD1ID28	○	AD1ICEN28	AD1ICSSL28	AD1ICESR28	AD1ICIEN28	AD1ICSST28
	AD1IN29			TRGA6N	AD1ICSST29	AD1ID29	○	AD1ICEN29	AD1ICSSL29	AD1ICESR29	AD1ICIEN29	AD1ICSST29
	AD1IN30			TRGA7N	AD1ICSST30	AD1ID30	○	AD1ICEN30	AD1ICSSL30	AD1ICESR30	AD1ICIEN30	AD1ICSST30
	AD1IN31			TRGA7N	AD1ICSST31	AD1ID31	○	AD1ICEN31	AD1ICSSL31	AD1ICESR31	AD1ICIEN31	AD1ICSST31
	AD1IN42			TRG0N	AD1ICSST42	AD1ID42	○	AD1ICEN42	AD1ICSSL42	AD1ICESR42	AD1ICIEN42	AD1ICSST42
	AD1IN43			TRG0N	AD1ICSST43	AD1ID43	○	AD1ICEN43	AD1ICSSL43	AD1ICESR43	AD1ICIEN43	AD1ICSST43
	AD1IN44			TRGA6N	AD1ICSST44	AD1ID44	○	AD1ICEN44	AD1ICSSL44	AD1ICESR44	AD1ICIEN44	AD1ICSST44
	AD1IN45			TRGA6N	AD1ICSST45	AD1ID45	○	AD1ICEN45	AD1ICSSL45	AD1ICESR45	AD1ICIEN45	AD1ICSST45
	AD1IN46			TRGA7N	AD1ICSST46	AD1ID46	○	AD1ICEN46	AD1ICSSL46	AD1ICESR46	AD1ICIEN46	AD1ICSST46
	AD1IN47			TRGA7N	AD1ICSST47	AD1ID47	○	AD1ICEN47	AD1ICSSL47	AD1ICESR47	AD1ICIEN47	AD1ICSST47

Note: * The SH72A0 group has the following eight A/D conversion channels:
AD1IN04, AD1IN05, AD1IN16, AD1IN17, AD1IN24, AD1IN28, AD1IN30, and AD1IN31.

Table 22.6 lists I/O pins of 10-bit A/D converter (AD1).
The 24 pins of AD1IN00 to AD1IN47 serve as the analog inputs to AD1. The ADTRG# pin supplies the timing to start scan conversion from outside the LSI. Inputting a low-level signal to the ADTRG# pin requests scan conversion to start in AD1. The SH72A0 group has eight analog input pins. It does not have the ADTRG# pin.
Pin switching is required for pins which are multiplexed with other functions. For details, refer to section 13, I/O Ports.

Table 22.6 I/O Pins of 10-Bit A/D Converter (AD1)

Module Name	Pin Name*	I/O	Function
AD1	AVCC1	Input	Analog power supply pin (Pins VREF1 (input pin for analog reference voltage) and AVCC1 serve as shared pins.)
	AVSS1	Input	Analog ground pin
	AD1IN00	Input	AD1 analog input pin 0
	AD1IN01	Input	AD1 analog input pin 1
	AD1IN02	Input	AD1 analog input pin 2
	AD1IN03	Input	AD1 analog input pin 3
	AD1IN04	Input	AD1 analog input pin 4
	AD1IN05	Input	AD1 analog input pin 5
	AD1IN06	Input	AD1 analog input pin 6
	AD1IN07	Input	AD1 analog input pin 7
	AD1IN16	Input	AD1 analog input pin 16
	AD1IN17	Input	AD1 analog input pin 17
	AD1IN24	Input	AD1 analog input pin 24
	AD1IN25	Input	AD1 analog input pin 25
	AD1IN26	Input	AD1 analog input pin 26
	AD1IN27	Input	AD1 analog input pin 27
	AD1IN28	Input	AD1 analog input pin 28
	AD1IN29	Input	AD1 analog input pin 29
	AD1IN30	Input	AD1 analog input pin 30
	AD1IN31	Input	AD1 analog input pin 31
	AD1IN42	Input	AD1 analog input pin 42
	AD1IN43	Input	AD1 analog input pin 43
	AD1IN44	Input	AD1 analog input pin 44
	AD1IN45	Input	AD1 analog input pin 45
	AD1IN46	Input	AD1 analog input pin 46
	AD1IN47	Input	AD1 analog input pin 47
	ADTRG#	Input	Input pin for scan conversion trigger of AD0 and AD1

Note: * The SH72A0 group has the following pins:
AVCC1, AVSS1, AD1IN04, AD1IN05, AD1IN16, AD1IN17, AD1IN24, AD1IN28, AD1IN30, and AD1IN31.

22.2 Registers

Table 22.7 and Table 22.8 list Registers of 10-Bit A/D Converter (AD1).

Table 22.7 Registers of 10-Bit A/D Converter (AD1) (1)

Register Name	Symbol	After Reset	Address	Access Size*
AD1 control register	AD1CR	H'00	H'FFFE 8000	8
AD1 conversion status register	AD1CSR	H'00	H'FFFE 8002	8
AD1 interrupt conversion enable register 0	AD1ICEN0	H'0000	H'FFFE 8004	8, 16
AD1 interrupt conversion enable register 1	AD1ICEN1	H'0000	H'FFFE 8006	8, 16
AD1 interrupt conversion enable register 2	AD1ICEN2	H'0000	H'FFFE 8008	8, 16
AD1 interrupt conversion end status register 0	AD1ICESR0	H'0000	H'FFFE 800A	8, 16
AD1 interrupt conversion end status register 1	AD1ICESR1	H'0000	H'FFFE 800C	8, 16
AD1 interrupt conversion end status register 2	AD1ICESR2	H'0000	H'FFFE 800E	8, 16
AD1 interrupt conversion source select register 0	AD1ICSSL0	H'0000	H'FFFE 8010	8, 16
AD1 interrupt conversion source select register 1	AD1ICSSL1	H'0000	H'FFFE 8012	8, 16
AD1 interrupt conversion source select register 2	AD1ICSSL2	H'0000	H'FFFE 8014	8, 16
AD1 interrupt conversion software start register 0	AD1ICSST0	H'0000	H'FFFE 8016	8, 16
AD1 interrupt conversion software start register 1	AD1ICSST1	H'0000	H'FFFE 8018	8, 16
AD1 interrupt conversion software start register 2	AD1ICSST2	H'0000	H'FFFE 801A	8, 16
AD1 interrupt conversion end interrupt enable register 0	AD1ICIEN0	H'0000	H'FFFE 801C	8, 16
AD1 interrupt conversion end interrupt enable register 1	AD1ICIEN1	H'0000	H'FFFE 801E	8, 16
AD1 interrupt conversion end interrupt enable register 2	AD1ICIEN2	H'0000	H'FFFE 8020	8, 16
AD1-converted value addition mode select register	AD1ADSL	H'00	H'FFFE 8022	8
AD1-converted value addition count select register	AD1ADCSL	H'00	H'FFFE 8024	8
AD1 channel select register 0	AD1CSL0	H'0000	H'FFFE 8026	8, 16
AD1 channel select register 1	AD1CSL1	H'0000	H'FFFE 8028	8, 16
AD1 channel select register 2	AD1CSL2	H'0000	H'FFFE 802A	8, 16
AD1 scan conversion enable register 0	AD1SCEN0	H'0000	H'FFFE 802C	8, 16
AD1 scan conversion enable register 1	AD1SCEN1	H'0000	H'FFFE 802E	8, 16
AD1 control extended register	AD1CEX	H'0000	H'FFFE 8030	8, 16
AD1 data register 0	AD1DR0	H'0000	H'FFFE 8040	16
AD1 data register 1	AD1DR1	H'0000	H'FFFE 8042	16
AD1 data register 2	AD1DR2	H'0000	H'FFFE 8044	16
AD1 data register 3	AD1DR3	H'0000	H'FFFE 8046	16
AD1 data register 4	AD1DR4	H'0000	H'FFFE 8048	16
AD1 data register 5	AD1DR5	H'0000	H'FFFE 804A	16
AD1 data register 6	AD1DR6	H'0000	H'FFFE 804C	16
AD1 data register 7	AD1DR7	H'0000	H'FFFE 804E	16
AD1 data register 16	AD1DR16	H'0000	H'FFFE 8060	16
AD1 data register 17	AD1DR17	H'0000	H'FFFE 8062	16
AD1 data register 24	AD1DR24	H'0000	H'FFFE 8070	16
AD1 data register 25	AD1DR25	H'0000	H'FFFE 8072	16
AD1 data register 26	AD1DR26	H'0000	H'FFFE 8074	16
AD1 data register 27	AD1DR27	H'0000	H'FFFE 8076	16

Table 22.8 Registers of 10-Bit A/D Converter (AD1) (2)

Register Name	Symbol	After Reset	Address	Access Size*
AD1 data register 28	AD1DR28	H'0000	H'FFFE 8078	16
AD1 data register 29	AD1DR29	H'0000	H'FFFE 807A	16
AD1 data register 30	AD1DR30	H'0000	H'FFFE 807C	16
AD1 data register 31	AD1DR31	H'0000	H'FFFE 807E	16
AD1 data register 42	AD1DR42	H'0000	H'FFFE 8094	16
AD1 data register 43	AD1DR43	H'0000	H'FFFE 8096	16
AD1 data register 44	AD1DR44	H'0000	H'FFFE 8098	16
AD1 data register 45	AD1DR45	H'0000	H'FFFE 809A	16
AD1 data register 46	AD1DR46	H'0000	H'FFFE 809C	16
AD1 data register 47	AD1DR47	H'0000	H'FFFE 809E	16
AD1 analog port pill-down register 0	AD1APPD0	H'0000	H'FFFE 8034	8, 16
AD1 analog port pill-down register 1	AD1APPD1	H'0000	H'FFFE 8036	8, 16
AD1 analog port pill-down register 2	AD1APPD2	H'0000	H'FFFE 8038	8, 16
AD1 self-test register	AD1DIAG	H'0000	H'FFFE 8032	8

Note: * 16-bit access can be made only at word boundaries.

22.2.1 AD1 Data Register n (AD1DRn) (n = 0 to 47)

Address: AD1DR0: H'FFFE 8040, AD1DR1: H'FFFE 8042, AD1DR2: H'FFFE 8044, AD1DR3: H'FFFE 8046, AD1DR4: H'FFFE 8048, AD1DR5: H'FFFE 804A, AD1DR6: H'FFFE 804C, AD1DR7: H'FFFE 804E, AD1DR16: H'FFFE 8060, AD1DR17: H'FFFE 8062, AD1DR24: H'FFFE 8070, AD1DR25: H'FFFE 8072, AD1DR26: H'FFFE 8074, AD1DR27: H'FFFE 8076, AD1DR28: H'FFFE 8078, AD1DR29: H'FFFE 807A, AD1DR30: H'FFFE 807C, AD1DR31: H'FFFE 807E, AD1DR42: H'FFFE 8094, AD1DR43: H'FFFE 8096, AD1DR44: H'FFFE 8098, AD1DR45: H'FFFE 809A, AD1DR46: H'FFFE 809C, AD1DR47: H'FFFE 809E

Note: n = 0 to 7, 16, 17, 24 to 31, or 42 to 47.

- When A/D-converted value addition mode is not selected

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0	—	—	—	—	—	—
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 b6	AD9 to AD0	Data Register Bits	10-Bit A/D-Converted Value	R
b5 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

- When right-alignment is selected

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0
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 b10	—	Reserved	These bits are read as 0. The write value should be 0.	R
b9 to b0	AD9 to AD0	Data Register Bits	10-Bit A/D-Converted Value	R

- When A/D-converted value addition mode is selected

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	AD11	AD10	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0	—	—	—	—
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 b4	AD11 to AD0	Data Register Bits	Sum of All Values Added in A/D-Converted Value Addition Mode	R
b3 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

AD1 data registers n (AD1DRn (n = 0 to 47))* are read-only registers which store the A/D converted results of channels AD1IN00 to AD1IN47.

Registers AD1DR0 to AD1DR47 use different formats under the following conditions.

- Setting of the AD data register format select bit (ADRFMT) in the AD1 control extended register (AD1CEX) (left-alignment or right-alignment)
- Setting of the A/D-converted value addition channel select bit (AD1ADSLn) in the AD1-converted value addition mode select register (AD1ADSL) (not selected or selected)

When A/D-converted value addition mode is not selected, left-shift or right-shift format can be selected by setting the ADRFMT bit in the AD1 control extended register (AD1CEX). At this time, bits AD9 to AD0 show the 10-bit A/D-converted value. The other bits are reserved. These bits are read as 0. The write value should be 0. A/D-converted value addition mode can only be set in registers AD1DR0 to AD1DR7. Registers AD1DR8 to AD1DR47 cannot select the setting of A/D-converted value addition mode.

When A/D-converted value addition mode is selected, the setting of the ADRFMT bit is invalid. At this time, bits AD11 to AD0 show the sum of all the values added in A/D converted value addition mode. The other bits are reserved. These bits are read as 0. The write value should be 0.

The following minimum and maximum values apply to channels for which A/D-converted value addition mode is selected:

1st conversion: H'0000 ≤ AD1DRn (n = 0 to 7) ≤ H'3FF0

2nd conversion: H'0000 ≤ AD1DRn (n = 0 to 7) ≤ H'7FE0

3rd conversion: H'0000 ≤ AD1DRn (n = 0 to 7) ≤ H'BFD0

4th conversion: H'0000 ≤ AD1DRn (n = 0 to 7) ≤ H'FFC0

Note: * n = 0 to 7, 16, 17, 24 to 31, or 42 to 47

22.2.2 AD1 Control Register (AD1CR)

Address AD1CR: H'FFFE 8000

	b7	b6	b5	b4	b3	b2	b1	b0
	ADST	ADCS	—	ADIE	—	—	TRGE	EXTRG
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	ADST	Scan Conversion Start Bit	0: Stops a scan conversion process. 1: Starts a scan conversion process.	R/W
b6	ADCS	Scan Conversion Mode Select Bit	0: Single-cycle scan mode 1: Continuous scan mode	R/W
b5	—	Reserved	This bit is read as 0. The write value should be 0.	R
b4	ADIE	Interrupt Enable Bit	0: Disables ADI interrupt generation upon scanning completion. 1: Enables ADI interrupt generation upon scanning completion.	R/W
b3, b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b1	TRGE	Trigger Enable Bit	0: Disables scan conversion to be started by an external trigger (ADTRG#) or TPU (PWM) timer trigger/TPCNT reset/MTU-III timer trigger. 1: Enables scan conversion to be started by an external trigger (ADTRG#) or TPU (PWM) timer trigger/TPCNT reset/MTU-III timer trigger.	R/W
b0	EXTRG	Trigger Select Bit	0: Scan conversion is started by a TPU (PWM) timer trigger/ TPCNT reset/MTU-III timer trigger. 1: Scan conversion is started by an external trigger (ADTRG#).	R/W

Notes: • Staring an interrupt conversion and a scan conversion simultaneously
If 1 is written to the TRGE bit and 0 is written to the EXTRG bit in AD1, and 1 is written to the AD1ICEN4 bit and the AD1ICEN5 bit in the AD1 interrupt conversion enable register 0 (AD1ICEN0), and then the timer TPO1A trigger and the timer TPO1B trigger are simultaneously input from the TPU (PWM), the AD1 executes the following operations in the indicated sequence: AD1IN04 interrupt conversion → AD1IN05 interrupt conversion → scan conversion. To execute a scan conversion only, clear both the AD1ICEN4 and AD1ICEN5 bits to 0. Either AD1IN04 or AD1IN05 can also execute a single-channel interrupt conversion. Similar operations can also be accomplished through combinations of an AD1 scan conversion and an AD1IN04 interrupt conversion by using the timer TP0CNT reset trigger and the timer TPO1A trigger from the TPU (PWM).

- Starting a scan conversion using an external trigger
If 1 is written to both the TRGE and EXTRG bits when high level signals are input to the external trigger pins (ADTRG#), and then if a low level pulse is input to either the ADTRG# pin, either AD1 detects a pulse falling edge and starts the scan conversion process. In this case, the low pulse width must be 1.5 cycles or more of the peripheral bus clock A.
The required high pulse width depends on the settings of the CKS bit.
At the time when the CKS is 0: 2 cycles or more of the peripheral bus clock A
At the time when the CKS is 1: 4 cycles or more of the peripheral bus clock A
Because the SH72A0 group has no external trigger (ADTRG#), scan conversion cannot be started by external trigger.
- Independent of the ADST bit, external triggers and TPU (PWM) timer trigger/TPCNT reset/MTU-III timer trigger, startup of a scan conversion is enabled when the ADSCACT bit in the AD1 conversion status register (AD1CSR) is cleared to 0. The startup source for a scan conversion is not retained.
- Regarding the startup cycle time for scan conversion and interrupt conversion initiated by the TPU (PWM) timer trigger/TPCNT reset/MTU-III timer trigger, specify a TPU (PWM) timer trigger/TPCNT reset/MTU-III timer trigger cycle time that exceeds the scan conversion time (for example, 56 states when the CKS bit is cleared to 0 and 112 states when the CKS is set to 1, respectively, for conversion on a single channel) and the interrupt conversion time (for example, 50 states when the CKS bit is set to 1 and 100 states, for conversion on a single channel using one trigger source). For details on the timer trigger cycle setting, refer to section 15, Timer Pulse Unit (TPU).

ADST Bit

Starts or stops scan conversion.

When the ADST bit is set from 0 to 1, the A/D converter detects the rising edge of the ADST bit and then starts scan conversion. When the ADST bit is cleared from 1 to 0, the A/D converter detects the falling edge of the ADST bit and then stops scan conversion. The ADST bit does not affect interrupt conversion. To check whether a scan conversion is being performed, read the ADSCACT bit in the AD1CSR register.

ADCS Bit

Selects scan conversion mode. To prevent incorrect operation, the value of the ADCS bit must be changed while the ADSCACT bit in the AD1CSR register is cleared to 0. In single-cycle scan mode, scanning is performed once and, upon completion, scan conversion ends. In continuous scan mode, scanning is repeated indefinitely. Scan conversion can be stopped by writing 0 to the ADST bit when the bit is set to 1. In scan conversion, A/D conversion proceeds in order from lower-numbered to higher-numbered channels (AD1IN00 to AD1IN47 for AD1). In continuous scan mode, the conversion process returns to the first channel when all the selected channels have been converted.

ADIE Bit

Enables or disables generation of the A/D scan conversion end interrupt (ADI). To prevent incorrect operation, the value of the ADIE bit must be changed while the ADSCACT bit in the AD1CSR register is cleared to 0.
When the ADF bit in the AD1CSR register is set to 1 upon completion of each scan in the scan conversion process, an ADI interrupt is generated when the ADIE bit is set to 1. The ADI interrupt can be cleared by clearing the ADF bit to 0 or clearing the ADIE bit to 0.

TRGE Bit

- AD1: Enables or disables scan conversion to be started by an external trigger (ADTRG#) or TPU (PWM) timer trigger (TPU1 to TPU4)/TP1CNT to TP4CNT reset/MTU-III timer trigger (TRG0N, TRGA0N to TRGA4N, TRGA6N or TRGA7N)).

EXTRG Bit

Selects a trigger source for scan conversion. Either the external trigger (ADTRG#) or the TPU (PWM) timer trigger/ TPCNT reset/MTU-III timer trigger.

22.2.3 AD1 Control Extended Register (AD1CEX)

Address AD1CEX: H'FFFE 8030

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ADRFMT	—	—	—	—	—	—	—	CKS	—	—	—	—	—	ITTRGS [1:0]	
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	ADRFMT	AD Data Register Format Select Bit	0: AD data register format is left-shift 1: AD data register format is right-shift	R/W
b14 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R
b7	CKS	Clock Select Bit	0: A/D conversion time = 50 states (f (PBA) conversion) 1: A/D conversion time = 100 states (f (PBA) conversion)	R/W
b6 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b1, b0	ITTRGS [1:0]	Expanded Interrupt Conversion Trigger Source Select Bit	b1 b0 0 0 : AD1IN04 to AD1IN07, AD1IN16 and AD1IN17 interrupt conversions are triggered by the source 1 TPU (PWM) timer. 0 1 : AD1IN04 to AD1IN07, AD1IN16 and AD1IN17 interrupt conversions are triggered by the source 2 TPCNT reset. 1 0 : Setting prohibited 1 1 : AD1IN00 to AD1IN47 interrupt conversions are triggered by the source 3 MTU-III timer.	R/W

Note: n = 00 to 07, 16, 17, 24 to 31, or 42 to 47

ADRFMT Bit

For details on the AD data register formats, refer to section 22.2.1, AD1 Data Register n (AD1DRn) (n = 0 to 47).

CKS Bit

Selects A/D conversion time. To prevent incorrect operation, the values of the ADSCACT and ADITACT bits in the AD1CSR register must be 0 when the value of the CKS bit is changed.

ITTRGS Bit

Selects the source 1 timer, the source 2 timer, or the source 3 timer as the interrupt conversion trigger source for AD1IN00 to AD1IN47. The setting of the ITTRGS bit has effect only when the AD1ICENn bit in the AD1 interrupt conversion enable register (AD1ICENj (j = 0 to 2)) is set to 1 and the AD1ICSSLn bit in the AD1 interrupt conversion source select register (AD1ICSSLj (j = 0 to 2)) is cleared to 0. See Table 22.4 and Table 22.5.

Note: The source 1 timer, the source 2 timer, or the source 3 timer can be selected as the interrupt conversion trigger source for AD1IN04 to AD1IN07, AD1IN16, and AD1IN17.
The source 1 timer and the source 2 timer cannot be selected as the ohter interrupt conversion trigger source.

22.2.4 AD1 Channel Select Register j (AD1CSLj) (j = 0 to 2)

22.2.4.1 AD1 Channel Select Register 0 (AD1CSL0)

Address AD1CSL0: H'FFFE 8026

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	AD1CSL7	AD1CSL6	AD1CSL5	AD1CSL4	AD1CSL3	AD1CSL2	AD1CSL1	AD1CSL0
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 b8	—	Reserved	These bits are read as 0. The write value should be 0.	R
b7 to b0	AD1CSLn	AD1 Channel Select Bits	0: AD1INm is not selected. 1: AD1INm is selected.	R/W

Note: n = 0 to 7, m = 00 to 07

The AD1CSL0 register is used to select channels that are subject to scan conversion.
To prevent incorrect operation, the ADSCACT bit in the AD1CSR register must be cleared to 0 while the AD1CSL0 register value is changed.

Note: The AD1CSL0 register selects scan conversion channels; it is not used to select interrupt conversion channels. An interrupt conversion channel is selected by the AD1 interrupt conversion enable register (AD1ICEN).
If a channel is selected by both the AD1CSL0 and AD1ICEN registers, it is subject to conversion in both scan conversion and interrupt conversion. A channel that is selected only by the AD1CEN register is excluded from the list of channels eligible for scan conversion, and only receives an interrupt conversion.

AD1CSLn Bit (n = 0 to 7)

Setting the AD1CSLn bit to 1 selects AD1INm. The correspondence between AD1INm and the AD1CSLn bit is shown in Table 22.4 and Table 22.5.

22.2.4.2 AD1 Channel Select Register 1 (AD1CSL1)

Address AD1CSL1: H'FFFE 8028

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	AD1CSL31	AD1CSL30	AD1CSL29	AD1CSL28	AD1CSL27	AD1CSL26	AD1CSL25	AD1CSL24	—	—	—	—	—	—	AD1CSL17	AD1CSL16
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 b8	AD1CSLn	AD1 Channel Select Bits	0: AD1INm is not selected. 1: AD1INm is selected.	R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b1, b0	AD1CSLn	AD1 Channel Select Bits	0: AD1INm is not selected. 1: AD1INm is selected.	

Note: n = 16, 17, 24 to 31, m = 16, 17, 24 to 31

The AD1CSL1 register is used to select channels that are subject to scan conversion. To prevent incorrect operation, the ADSCACT bit in the AD1CSR register must be cleared to 0 while the AD1CSL1 register values are changed.

Note: The AD1CSL1 register selects scan conversion channels; it is not used to select interrupt conversion channels. An interrupt conversion channel is selected by the AD1 interrupt conversion enable register (AD1ICEN).
If a channel is selected by both the AD1CSL1 and AD1ICEN registers, it is subject to conversion in both scan conversion and interrupt conversion. A channel that is selected only by the AD1ICEN register is excluded from the list of channels eligible for scan conversion, and only receives an interrupt conversion.

AD1CSLn Bit (n = 16, 17, 24 to 31)

Setting the AD1CSLn bit to 1 selects AD1INm. The correspondence between AD1INm and the AD1CSLn bit is shown in Table 22.4 and Table 22.5.

22.2.4.3 AD1 Channel Select Register 2 (AD1CSL2)

Address AD1CSL2 : H'FFFE 802A

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	AD1CSL47	AD1CSL46	AD1CSL45	AD1CSL44	AD1CSL43	AD1CSL42	—	—	—	—	—	—	—	—	—	—
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 b10	AD1CSLn	AD1 Channel Select Bits	0: AD1INm is not selected. 1: AD1INm is selected.	R/W
b9 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

Note: n = 42 to 47, m = 42 to 47

The AD1CSL2 register is used to select channels that are subject to scan conversion. To prevent incorrect operation, the ADSCACT bit in the AD1CSR register must be cleared to 0 while the AD1CSL2 register values are changed.

Note: The AD1CSL2 register selects scan conversion channels; it is not used to select interrupt conversion channels. An interrupt conversion channel is selected by the AD1 interrupt conversion enable register (AD1ICEN).
If a channel is selected by both the AD1CSL2 and AD1ICEN registers, it is subject to conversion in both scan conversion and interrupt conversion. A channel that is selected only by the AD1ICEN register is excluded from the list of channels eligible for scan conversion, and only receives an interrupt conversion.

AD1CSLn Bit (n = 42 to 47)

Setting the AD1CSLn bit to 1 selects AD1INm. The correspondence between AD1INm and the AD1CSLn bit is shown in Table 22.4 and Table 22.5.

22.2.5 AD1 Conversion Status Register (AD1CSR)

Address AD1CSR: H'FFFE 8002

	b7	b6	b5	b4	b3	b2	b1	b0
	ADSCACT	ADITACT	—	—	—	—	—	ADF
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	ADSCACT	Scan Conversion Status Bit	0: Indicates that the Scan conversion process is in idle state. 1: Indicates that the Scan conversion process is being executed.	R
b6	ADITACT	Interrupt Conversion Status Bit	0: Indicates that the interrupt conversion process is in idle state. 1: Indicates that the Interrupt conversion process is being executed.	R
b5 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R
b0	ADF	Single Scan End Flag	0: Indicates that the scan conversion process is in idle state. 1: Indicates that a single scan has been completed and the A/D-converted values on all selected AD1INn has been transferred to the AD1 data register.	R/W

Note: n = 0 to 7, m = 00 to 07

ADSCACT Bit

Indicates whether the scan conversion process is in the idle state or it is being executed. This is a read-only bit and cannot be written. If an interrupt conversion is started during a scan conversion, the A/D converter stops the scan conversion process and preferentially executes the interrupt conversion. However, until such time that all scan conversion is completed, the ADSCACT bit maintains to be set to 1 and is not cleared to 0.

ADITACT Bit

Indicates whether the interrupt conversion process is in the idle state or it is being executed. This is a read-only bit and cannot be written. The ADSCACT and ADITACT bits can indicate the status of the AD1. For details, see Table 22.9.

Table 22.9 Relationship between AD1 Status, ADSCACT, and ADITACT

ADSCACT Bit	ADITACT Bit	AD1 Status	Source of Scan Conversion	Source of Interrupt Conversion
0	0	Idle state	No	No
	1	Interrupt conversion	No	Yes
1	0	Scan conversion	Yes	No
	1	Interrupt conversion	Yes	Yes

ADF Bit

This bit is set to 1 each time scanning ends in the scan conversion process (when all selected channels are converted). 1 cannot be written to this bit.

When the ADF bit is set to 1, either a scan conversion end interrupt or a DMA transfer request to the DMAC can be generated (the interrupt or the request can be selected by the interrupt controller circuit). In this manner, processing such as storing the contents of the AD data register to the RAM can be implemented by means of either software or the DMAC.

[Conditions for clearing to 0]

- 0 is written to this bit after reading 1.

[Condition for setting to 1]

- All analog conversion has been completed during each scanning in scan conversion process.

22.2.6 AD1-Converted Value Addition Mode Select Register (AD1ADSL)

Address AD1ADSL: H'FFFE 8022

	b7	b6	b5	b4	b3	b2	b1	b0
	AD1ADSL7	AD1ADSL6	AD1ADSL5	AD1ADSL4	AD1ADSL3	AD1ADSL2	AD1ADSL1	AD1ADSL0
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	AD1ADSL7 to AD1ADSL0	AD-Converted Value Addition Channel Select Bits	0: A/D-converted value addition mode is not selected. 1: A/D-converted value addition mode (successively 2 to 4 times of addition) is selected.	R/W

Note: n = 0 to 7, m = 00 to 07

The AD1ADSL register selects channel AD1INm (m = 00 to 07) on which A/D conversion is performed successively two to four times, after which the converted values are added (integrated).

AD1ADSLn Bit

When the AD1ADSLn bit is set to 1, the A/D converter performs conversion on AD1INm successively 2 to 4 times and returns the added (integrated) conversion results to the AD1 data register n. If the AD1ADSLn bit is cleared to 0, the A/D converter performs a normal 1-time conversion of AD1INm and returns the conversion result to the AD1 data register n. Irrespective of whether a scan conversion or an interrupt conversion is to be performed, the A/D converter determines whether or not to perform an addition according to the AD1ADSLn value. To prevent incorrect operation, both the ADSCACT and ADITACT bits in the AD1CSR register must be cleared to 0 while the AD1ADSLn bit value is changed.
The correspondence between AD1INm and the AD1ADSLn bit is shown in Table 22.4 and Table 22.5. How to select the addition count is described in section 22.2.7, AD1-Converted Value Addition Count Select Register (AD1ADCSL).

Figure 22.2 shows Scan Conversion Sequence with Bits AD1ADSL2 and AD1ADSL6 Set to 1, based on the assumption that the addition count is set to 4, and that channels AD1IN00 to AD1IN07 are selected. The conversion process begins with AD1IN00. The AD1IN02 conversion is performed successively 4 times, and the addition (integration) value is returned to the data register, after which the AD1IN03 conversion process is started. If an interrupt conversion is requested during of a scan conversion, the scan conversion process is stopped and an A/D conversion is preferentially executed on the channel in which an interrupt conversion was requested. Upon completion of the interrupt conversion, the scan conversion process is resumed from the A/D conversion on the interrupted channel. However, if the AD1ADSLn bit of the interrupted channel (AD1INm) is set to 1, even if addition has been performed at least once (two to four times), the conversion is restarted from the 1st conversion.

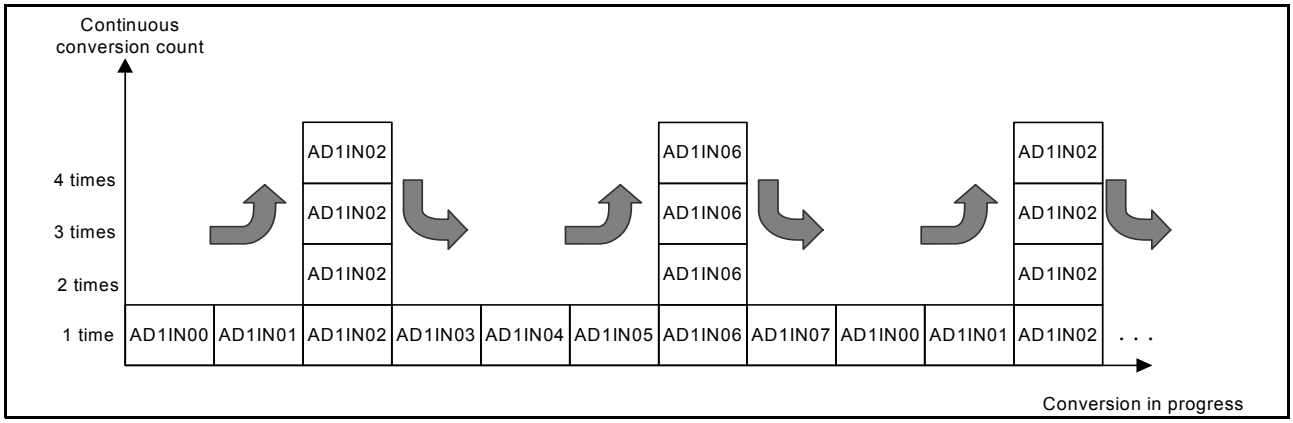


Figure 22.2 Scan Conversion Sequence with Bits AD1ADSL2 and AD1ADSL6 Set to 1

22.2.7 AD1-Converted Value Addition Count Select Register (AD1ADCSL)

Address AD1ADCSL: H'FFFE 8024

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	ADC[1:0]	
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b1, b0	ADC[1:0]	Addition Count Select Bits	b1 b0 0 0 : 1-time conversion (normal conversion) 0 1 : 2-time conversion 1 0 : 3-time conversion 1 1 : 4-time conversion	R/W

The AD1ADCSL register sets the addition count for channels for which A/D-converted value addition mode is selected.

ADC Bit

These bits select the number of additions to be performed in A/D-converted value addition mode. These bits have no effect on the A/D conversion of channels for which A/D-converted value addition mode is not selected.
To prevent incorrect operation, both the ADSCACT and ADITACT bits in the AD1CSR register must be cleared to 0 while the ADC1 and ADC0 bit values are changed.

22.2.8 AD1 Scan Conversion Enable Register k (AD1SCENk) (k = 0, 1)

22.2.8.1 AD1 Scan Conversion Enable Register 0 (AD1SCEN0)

Address H'FFFE 802C

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	AD1SCEN15	AD1SCEN14	AD1SCEN13	—	—	—	—	—	—	—	—	AD1SCEN4	AD1SCEN3	AD1SCEN2	AD1SCEN1	—
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 b13	AD1SCEN15 to AD1SCEN13	Scan Conversion Request Enable Bits	0: Disable a scan conversion request from the TPU (PWM) timer trigger/the TPCNT reset. 1: Enable a scan conversion request from the TPU (PWM) timer trigger/the TPCNT reset.	R/W
b12 tob5	—	Reserved	These bits are read as 0. The write value should be 0.	R
b4 tob1	AD1SCEN4 to AD1SCEN1	Scan Conversion Request Enable Bits	0: Disable a scan conversion request from the TPU (PWM) timer trigger/the TPCNT reset. 1: Enable a scan conversion request from the TPU (PWM) timer trigger/the TPCNT reset.	R/W
b0	—	Reserved	This bit is read as 0. The write value should be 0.	R

Note: n = 1 to 4, 13 to 15

The AD1SCEN0 register enables or disables an interrupt request for AD1IN00 to AD1IN47.

AD1SCENn Bit (n = 1 to 4, 13 to 15)

Setting the AD1SCENn bit to 1 enables the scan conversion request to the corresponding TPU (PWM). The correspondence between the AD1SCENn bit and the scan request trigger source is shown in Table 22.10.

22.2.8.2 AD1 Scan Conversion Enable Register 1 (AD1SCEN1)

Address H'FFFE 802E

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	AD1SCEN31	AD1SCEN30	AD1SCEN29	AD1SCEN28	AD1SCEN27	AD1SCEN26	AD1SCEN25	AD1SCEN24	—	—	—	—	—	—	—	AD1SCEN16
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 b8	AD1SCEN31 to AD1SCEN24	Scan Conversion Request Enable Bits	0: Disable a scan conversion request from the MTU-III timer. 1: Enable a scan conversion request from the MTU-III timer.	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R
b0	AD1SCEN16	Scan Conversion Request Enable Bit	0: Disable a scan conversion request from the TPU (PWM) timer trigger/the TPCNT reset. 1: Enable a scan conversion request from the TPU (PWM) timer trigger/the TPCNT reset.	R/W

Note: n = 16, 24 to 31

The AD1SCEN1 register enables or disables an interrupt request for AD1IN00 to AD1IN47.

AD1SCENn Bit (n = 24 to 31)

Setting the AD1SCENn bit to 1 enables the scan conversion request to the corresponding MTU-III timer. The correspondence between the AD1SCENn bit and the scan request trigger source is shown in Table 22.10.

AD1SCENn Bit (n = 16)

Setting the AD1SCENn bit to 1 enables the scan conversion request to the corresponding PWM timer. The correspondence between the AD1SCENn bit and the scan request trigger source is shown in Table 22.10.

Table 22.10 Correspondence between AD1SCENn Bit and Scan Request Trigger Source

Register Name	Bit Name	Scan Conversion Trigger Corresponding to AD1SCEN Bit
AD1SCEN1	AD1SCEN31	MTU-III timer trigger (TRG0N)
	AD1SCEN30	MTU-III timer trigger (TRGA7N)
	AD1SCEN29	MTU-III timer trigger (TRGA6N)
	AD1SCEN28	MTU-III timer trigger (TRGA4N)
	AD1SCEN27	MTU-III timer trigger (TRGA3N)
	AD1SCEN26	MTU-III timer trigger (TRGA2N)
	AD1SCEN25	MTU-III timer trigger (TRGA1N)
	AD1SCEN24	MTU-III timer trigger (TRGA0N)
	AD1SCEN16	TPU (PWM) timer trigger (TP4CNT reset)
AD1SCEN0	AD1SCEN15	TPU (PWM) timer trigger (TP3CNT reset)
	AD1SCEN14	TPU (PWM) timer trigger (TP2CNT reset)
	AD1SCEN13	TPU (PWM) timer trigger (TP1CNT reset)
	AD1SCEN4	TPU (PWM) timer trigger (TPO4A)
	AD1SCEN3	TPU (PWM) timer trigger (TPO3A)
	AD1SCEN2	TPU (PWM) timer trigger (TPO2A)
	AD1SCEN1	TPU (PWM) timer trigger (TPO1A)

22.2.9 AD1 Interrupt Conversion Enable Register j (AD1ICENj) (j = 0 to 2)

22.2.9.1 AD1 Interrupt Conversion Enable Register 0 (AD1ICEN0)

Address AD1ICEN0: H'FFFE 8004

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	AD1ICEN7	AD1ICEN6	AD1ICEN5	AD1ICEN4	AD1ICEN3	AD1ICEN2	AD1ICEN1	AD1ICEN0
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 b8	—	Reserved	These bits are read as 0. The write value should be 0.	R
b7 to b0	AD1ICENn	Interrupt Conversion Request Enable Bits	0: Disables an interrupt conversion request to AD1INm by TPU (PWM) timer trigger/TPCNT reset/MTU-III timer trigger or software trigger (AD1ICSSTn). 1: Enables an interrupt conversion request to AD1INm by TPU (PWM) timer trigger/TPCNT reset/MTU-III timer trigger or software trigger (AD1ICSSTn).	R/W

Note: n = 0 to 7, m = 00 to 07

The AD1ICEN0 register enables or disables an interrupt conversion request for AD1IN00 to AD1IN07.

AD1ICENn Bit (n = 0 to 7)

Setting the AD1ICENn bit to 1 enables the interrupt conversion request to the corresponding AD1INm channel. The correspondence among the AD1ICENn bit, AD1INm, and the interrupt request trigger source is shown in Table 22.4 and Table 22.5.

22.2.9.2 AD1 Interrupt Conversion Enable Register 1 (AD1ICEN1)

Address AD1ICEN1 : H'FFFE 8006

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	AD1ICEN31	AD1ICEN30	AD1ICEN29	AD1ICEN28	AD1ICEN27	AD1ICEN26	AD1ICEN25	AD1ICEN24	—	—	—	—	—	—	AD1ICEN17	AD1ICEN16
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 b8	AD1ICENn	Interrupt Conversion Request Enable Bits	0: Disables an interrupt conversion request to AD1INm by TPU (PWM) timer trigger/TPCNT reset/MTU-III timer trigger or software trigger (AD1ICSSTn). 1: Enables an interrupt conversion request to AD1INm by TPU (PWM) timer trigger/TPCNT reset/MTU-III timer trigger or software trigger (AD1ICSSTn).	R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b1, b0	AD1ICENn	Interrupt Conversion Request Enable Bits	0: Disables an interrupt conversion request to AD1INm by TPU (PWM) timer trigger/TPCNT reset/MTU-III timer trigger or software trigger (AD1ICSSTn). 1: Enables an interrupt conversion request to AD1INm by TPU (PWM) timer trigger/TPCNT reset/MTU-III timer trigger or software trigger (AD1ICSSTn).	R/W

Note: n = 16, 17, 24 to 31, m = 16, 17, 24 to 31

The AD1ICEN1 register enables or disables an interrupt conversion request for AD1IN16, AD1IN17, AD1IN24 to AD1IN31.

AD1ICENn Bit (n = 16, 17, 24 to 31)

Setting the AD1ICENn bit to 1 enables the interrupt conversion request to the corresponding AD1INm channel. The correspondence among the AD1ICENn bit, AD1INm, and the interrupt request trigger source is shown in Table 22.4 and Table 22.5.

22.2.9.3 AD1 Interrupt Conversion Enable Register 2 (AD1ICEN2)

Address AD1ICEN2 : H'FFFE 8008

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	AD1ICEN47	AD1ICEN46	AD1ICEN45	AD1ICEN44	AD1ICEN43	AD1ICEN42	—	—	—	—	—	—	—	—	—	—
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 b10	AD1ICENn	Interrupt Conversion Request Enable Bits	0: Disables an interrupt conversion request to AD1INm by TPU (PWM) timer trigger/TPCNT reset/MTU-III timer trigger or software trigger (AD1ICSSTn). 1: Enables an interrupt conversion request to AD1INm by TPU (PWM) timer trigger/TPCNT reset/MTU-III timer trigger or software trigger (AD1ICSSTn).	R/W
b9 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

Note: n = 42 to 47, m = 42 to 47

The AD1ICEN2 register enables or disables an interrupt conversion request for AD1IN42 to AD1IN47.

AD1ICENn Bit (n = 42 to 47)

Setting the AD1ICENn bit to 1 enables the interrupt conversion request to the corresponding AD1INm channel. The correspondence among the AD1ICENn bit, AD1INm, and the interrupt request trigger source is shown in Table 22.4 and Table 22.5.

22.2.10 AD1 Interrupt Conversion Source Select Register j (AD1ICSSLj) (j = 0 to 2)

22.2.10.1 AD1 Interrupt Conversion Source Select Register 0 (AD1ICSSL0)

Address AD1ICSSL0: H'FFFE 8010

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	AD1ICSSL7	AD1ICSSL6	AD1ICSSL5	AD1ICSSL4	AD1ICSSL3	AD1ICSSL2	AD1ICSSL1	AD1ICSSL0
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 b8	—	Reserved	These bits are read as 0. The write value should be 0.	R
b7 to b0	AD1ICSSLn	Interrupt Conversion Source Select Bits	0: Trigger source 1 TPU(PWM) timer trigger, trigger source 2 TPCNT reset, or trigger source 3 MTU-III timer trigger used as AD1INm interrupt conversion request source 1: Software trigger (AD1ICSSTn) used as AD1INm interrupt conversion request source	R/W

Note: n = 0 to 7, m = 00 to 07

The AD1ICSSL0 register selects the trigger source for interrupt conversion. Either a TPU (PWM) timer trigger/ TPCNT reset/MTU-III timer trigger or a software trigger caused by writing to the AD1ICSST0 to AD1ICSST2 registers can be selected.

AD1ICSSLn Bit (n = 0 to 7)

If the AD1ICSSLn bit is cleared to 0 and the AD1ICENn bit in the AD1 interrupt conversion enable register is set to 1, the A/D converter performs edge detection and begins interrupt conversion on AD1INm when a trigger source 1 TPU (PWM) timer trigger, a trigger source 2 TPCNT reset, or a trigger source 3 MTU-III timer trigger is input. If the AD1ICSSLn bit is set to 1, the A/D converter performs edge detection and begins interrupt conversion on AD1INm when 1 is written to the AD1ICSSTn bit in the AD1 interrupt conversion software start register. Selection of a trigger source 1 TPU (PWM) timer trigger, a trigger source 2 TPCNT reset, or a trigger source 3 MTU-III timer trigger is accomplished by setting the ITTRGS bit in the AD1CEX register. The correspondence among the AD1ICSSLn bit, AD1INm, and the interrupt request trigger source is shown in Table 22.4 and Table 22.5.

22.2.10.2 AD1 Interrupt Conversion Source Select Register 1 (AD1ICSSL1)

Address AD1ICSSL1 : H'FFFE 8012

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	AD1ICSSL31	AD1ICSSL30	AD1ICSSL29	AD1ICSSL28	AD1ICSSL27	AD1ICSSL26	AD1ICSSL25	AD1ICSSL24	—	—	—	—	—	—	AD1ICSSL17	AD1ICSSL16
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 b8	AD1ICSSLn	Interrupt Conversion Source Select Bits	0: Trigger source 1 TPU (PWM) timer trigger, trigger source 2 TPCNT reset, or trigger source 3 MTU-III timer trigger used as AD1INm interrupt conversion request source 1: Software trigger (AD1ICSSTn) used as AD1INm interrupt conversion request source	R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b1, b0	AD1ICSSLn	Interrupt Conversion Source Select Bits	0: Trigger source 1 TPU (PWM) timer trigger, trigger source 2 TPCNT reset, or trigger source 3 MTU-III timer trigger used as AD1INm interrupt conversion request source 1: Software trigger (AD1ICSSTn) used as AD1INm interrupt conversion request source	R/W

Note: n = 16, 17, 24 to 31, m = 16, 17, 24 to 31

The AD1ICSSL1 register selects the trigger source for interrupt conversion. Either a TPU (PWM) timer trigger/ TPCNT reset/MTU-III timer trigger or a software trigger caused by writing to the AD1ICSST0 to AD1ICSST2 registers can be selected.

AD1ICSSLn Bit (n = 16, 17, 24 to 31)

If the AD1ICSSLn bit is cleared to 0 and the AD1ICENn bit in the AD1 scan conversion enable register is set to 1, the A/D converter performs edge detection and begins interrupt conversion on AD1INm when a trigger source 1 TPU (PWM) timer trigger, a trigger source 2 TPCNT reset, or a trigger source 3 MTU-III timer trigger is input. If the AD1ICSSLn bit is set to 1, the A/D converter performs edge detection and begins interrupt conversion on AD1INm when 1 is written to the AD1ICSSTn bit in the AD1 interrupt conversion software start register. Selection of trigger source 1 TPU (PWM) timer trigger, trigger source 2 TPCNT reset, or trigger source 3 MTU-III timer trigger is accomplished by setting the ITTRGS bit in the AD1CEX register. The correspondence among the AD1ICSSLn bit, AD1INm, and the interrupt request trigger source is shown in Table 22.4 and Table 22.5.

22.2.10.3 AD1 Interrupt Conversion Source Select Register 2 (AD1ICSSL2)

Address AD1ICSSL2 : H'FFFE 8014

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	AD1ICSSL47	AD1ICSSL46	AD1ICSSL45	AD1ICSSL44	AD1ICSSL43	AD1ICSSL42	—	—	—	—	—	—	—	—	—	—
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 b10	AD1ICSSLn	Interrupt Conversion Source Select Bits	0: Trigger source 1 TPU (PWM) timer trigger, trigger source 2 TPCNT reset, or trigger source 3 MTU-III timer trigger used as AD1INm interrupt conversion request source 1: Software trigger (AD1ICSSTn) used as AD1INm interrupt conversion request source	R/W
b9 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

Note: n = 42 to 47, m = 42 to 47

The AD1ICSSL2 register selects the trigger source for interrupt conversion. Either a TPU (PWM) timer trigger/ TPCNT reset/MTU-III timer trigger or a software trigger caused by writing to the AD1ICSST0 to AD1ICSST2 registers can be selected.

AD1ICSSLn Bit (n = 42 to 47)

If the AD1ICSSLn bit is cleared to 0 and the AD1ICENn bit in the AD1 scan conversion enable register is set to 1, the A/D converter performs edge detection and begins interrupt conversion on AD1INm when a trigger source 1 TPU (PWM) timer trigger, a trigger source 2 TPCNT reset, or a trigger source 3 MTU-III timer trigger is input. If the AD1ICSSLn bit is set to 1, the A/D converter performs edge detection and begins interrupt conversion on AD1INm when 1 is written to the AD1ICSSTn bit in the AD1 interrupt conversion software start register. Selection of trigger source 1 TPU (PWM) timer trigger, trigger source 2 TPCNT reset, or trigger source 3 MTU-III timer trigger is accomplished by setting the ITTRGS bit in the AD1CEX register. The correspondence among the AD1ICSSLn bit, AD1INm, and the interrupt request trigger source is shown in Table 22.4 and Table 22.5.

22.2.11 AD1 Interrupt Conversion Software Start Register j (AD1ICSSTj) (j = 0 to 2)

22.2.11.1 AD1 Interrupt Conversion Software Start Register 0 (AD1ICSST0)

Address AD1ICSST0: H'FFFE 8016

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	AD1ICSST7	AD1ICSST6	AD1ICSST5	AD1ICSST4	AD1ICSST3	AD1ICSST2	AD1ICSST1	AD1ICSST0
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 b8	—	Reserved	These bits are read as 0. The write value should be 0.	W
b7 to b0	AD1ICSSTn	Interrupt Conversion Software Trigger Bits	0: No interrupt conversion request (software trigger) on AD1INm 1: Interrupt conversion request (software trigger) on AD1INm	W

Note: n = 0 to 7, m = 00 to 07

The AD1ICSST0 register starts an interrupt conversion by software. The AD1ICSST0 register is a write-only register and it is always read as 0s.

AD1ICSSTn Bit (n = 0 to 7)

If the AD1ICSSTn bit in the AD1 interrupt conversion source select register corresponding to AD1INm is set to 1 and the AD1ICENn bit in the AD1 interrupt conversion enable register is set to 1, the A/D converter performs edge detection and begins interrupt conversion on AD1INm when 1 is written to the AD1ICSSTn bit. Write 0 to the AD1ICSSTn bit corresponding to each channel (AD1INm) not subject to interrupt conversion requests. Any AD1INm channel for which 0 is written to the corresponding AD1ICSSTn bit is not affected by any of these operations. When interrupt conversion requests are issued, the interrupt sources are stored in internal circuits in AD1INm units. When an interrupt conversion on a channel (AD1INm) with a stored source is performed and completed, the source associated with the AD1INm is cleared. Consequently, writing 1 to the AD1ICSSTn bit and subsequently writing 0 to it does not clear the source associated with AD1INm, so interrupt conversion is executed. However, writing 1 to the AD1ICSSTn bit when a source is already pending does not cause interrupt conversion to be performed on AD1INm twice.
There is one source per channel. This also applies to the execution of an interrupt conversion in response to a request from a PMW timer trigger. See Table 22.4 and Table 22.5 for the correspondence between each AD1ICSSTn bit and AD1INm channel.

22.2.11.2 AD1 Interrupt Conversion Software Start Register 1 (AD1ICSST1)

Address AD1ICSST1 : H'FFFE 8018

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	AD1ICSST31	AD1ICSST30	AD1ICSST29	AD1ICSST28	AD1ICSST27	AD1ICSST26	AD1ICSST25	AD1ICSST24	—	—	—	—	—	—	AD1ICSST17	AD1ICSST16
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 b8	AD1ICSSTn	Interrupt Conversion Software Trigger Bits	0: No interrupt conversion request (software trigger) on AD1INm 1: Interrupt conversion request (software trigger) on AD1INm	W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	W
b1, b0	AD1ICSSTn	Interrupt Conversion Software Trigger Bits	0: No interrupt conversion request (software trigger) on AD1INm 1: Interrupt conversion request (software trigger) on AD1INm	W

Note: n = 16, 17, 24 to 31, m = 16, 17, 24 to 31

The AD1ICSST1 register starts an interrupt conversion by software. The AD1ICSST1 register is write-only register and it is always read as 0s.

AD1ICSSTn Bit (n = 16, 17, 24 to 31)

If the AD1ICSSTn bit in the AD1 interrupt conversion source select register corresponding to AD1INm is set to 1 and the AD1ICENn bit in the AD1 interrupt conversion enable register is set to 1, the A/D converter performs edge detection and begins interrupt conversion on AD1INm when 1 is written to the AD1ICSSTn bit. Write 0 to the AD1ICSSTn bit corresponding to each channel (AD1INm) not subject to interrupt conversion requests. Any AD1INm channel for which 0 is written to the corresponding AD1ICSSTn bit is not affected by any of these operations. When interrupt conversion requests are issued, the interrupt sources are stored in internal circuits in AD1INm units. When an interrupt conversion on a channel (AD1INm) with a stored source is performed and completed, the source associated with the AD1INm is cleared. Consequently, writing 1 to the AD1ICSSTn bit and subsequently writing 0 to it does not clear the source associated with AD1INm, so interrupt conversion is executed. However, writing 1 to the AD1ICSSTn bit when a source is already pending does not cause interrupt conversion to be performed on AD1INm twice.

There is one source per channel. This also applies to the execution of an interrupt conversion in response to a request from a PWM timer trigger. See Table 22.4 and Table 22.5 for the correspondence between each AD1ICSSTn bit and AD1INm channel.

22.2.11.3 AD1 Interrupt Conversion Software Start Register 2 (AD1ICSST2)

Address AD1ICSST2 : H'FFFE 801A

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	AD1ICSST47	AD1ICSST46	AD1ICSST45	AD1ICSST44	AD1ICSST43	AD1ICSST42	—	—	—	—	—	—	—	—	—	—
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 b10	AD1ICSSTn	Interrupt Conversion Software Trigger Bits	0: No interrupt conversion request (software trigger) on AD1INm 1: Interrupt conversion request (software trigger) on AD1INm	W
b9 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	W

Note: n = 42 to 47, m = 42 to 47

The AD1ICSST2 register starts an interrupt conversion by software. The AD1ICSST2 register is write-only register and it is always read as 0s.

AD1ICSSTn Bit (n = 42 to 47)

If the AD1ICSSTn bit in the AD1 interrupt conversion source select register corresponding to AD1INm is set to 1 and the AD1ICENn bit in the AD1 interrupt conversion enable register is set to 1, the A/D converter performs edge detection and begins interrupt conversion on AD1INm when 1 is written to the AD1ICSSTn bit. Write 0 to the AD1ICSSTn bit corresponding to each channel (AD1INm) not subject to interrupt conversion requests. Any AD1INm channel for which 0 is written to the corresponding AD1ICSSTn bit is not affected by any of these operations. When interrupt conversion requests are issued, the interrupt sources are stored in internal circuits in AD1INm units. When an interrupt conversion on a channel (AD1INm) with a stored source is performed and completed, the source associated with the AD1INm is cleared. Consequently, writing 1 to the AD1ICSSTn bit and subsequently writing 0 to it does not clear the source associated with AD1INm, so interrupt conversion is executed. However, writing 1 to the AD1ICSSTn bit when a source is already pending does not cause interrupt conversion to be performed on AD1INm twice.
There is one source per channel. This also applies to the execution of an interrupt conversion in response to a request from a PWM timer trigger. See Table 22.4 and Table 22.5 for the correspondence between each AD1ICSSTn bit and AD1INm channel.

22.2.12 AD1 Interrupt Conversion End Status Register j (AD1ICESRj) (j = 0 to 2)

22.2.12.1 AD1 Interrupt Conversion End Status Register 0 (AD1ICESR0)

Address AD1ICESR0: H'FFFE 800A

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	AD1ICESR7	AD1ICESR6	AD1ICESR5	AD1ICESR4	AD1ICESR3	AD1ICESR2	AD1ICESR1	AD1ICESR0
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 b8	—	Reserved	These bits are read as 0. The write value should be 0.	R
b7 to b0	AD1ICESRn	Interrupt Conversion End Flag	0: Indicates that an interrupt conversion process on AD1INm is in idle state. 1: Indicates that an interrupt conversion process on AD1INm has been completed and the conversion result has been transferred to the AD1DRn register.	R/W

Notes: • Even when the AD1ICESRn bit is not cleared to 0, an interrupt conversion request for AD1INm can be accepted. Exercise care regarding the timing of storing values in AD1 data register n.
• n = 0 to 7, m = 00 to 07

The AD1ICESR0 register indicates that an interrupt conversion has been completed. When an interrupt conversion has been completed, the AD1ICESRn bit corresponding to the channel (AD1INm) is set to 1.

AD1ICESRn Bit (n = 0 to 7)

AD1ICESRn is a status flag bit which indicates that an interrupt conversion has been completed. 1 must not be written to the AD1ICESRn bit. When the AD1ICESRn bit is set to 1, an AD1INm interrupt conversion end interrupt (AD1IDn) can be generated. See Table 22.4 and Table 22.5 for correspondence between AD1ICESRn and AD1INm.

[Condition for clearing to 0]

- Writing 0 to the AD1ICESRn bit after reading it as 1.

[Condition for setting to 1]

- An interrupt conversion process on AD1INm has been completed.

22.2.12.2 AD1 Interrupt Conversion End Status Register 1 (AD1ICESR1)

Address AD1ICESR1 : H'FFFE 800C

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	AD1IC ESR31	AD1IC ESR30	AD1IC ESR29	AD1IC ESR28	AD1IC ESR27	AD1IC ESR26	AD1IC ESR25	AD1IC ESR24	—	—	—	—	—	—	AD1IC ESR17	AD1IC ESR16
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 b8	AD1ICESRn	Interrupt Conversion End Flag	0: Indicates that an interrupt conversion process on AD1INm is in idle state. 1: Indicates that an interrupt conversion process on AD1INm has been completed and the conversion result has been transferred to the AD1DRn bit.	R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b1, b0	AD1ICESRn	Interrupt Conversion End Flag	0: Indicates that an interrupt conversion process on AD1INm is in idle state. 1: Indicates that an interrupt conversion process on AD1INm has been completed and the conversion result has been transferred to the AD1DRn bit.	R/W

Notes: • Even when the AD1ICESRn bit is not cleared to 0, an interrupt conversion request for AD1INm can be accepted. Exercise care regarding the timing of storing values in AD1 data register n.
• n = 16, 17, 24 to 31, m = 16, 17, 24 to 31

The AD1ICESR1 register indicates that an interrupt conversion has been completed. When an interrupt conversion has been completed, the AD1ICESRn bit corresponding to the channel (AD1INm) is set to 1.

AD1ICESRn Bit (n = 16, 17, 24 to 31)

AD1ICESRn is a status flag bit which indicates that an interrupt conversion has been completed. 1 must not be written to the AD1ICESRn bit. When the AD1ICESRn bit is set to 1, an AD1INm interrupt conversion end interrupt (AD1IDn) can be generated. See Table 22.4 and Table 22.5 for correspondence between AD1ICESRn and AD1INm.

[Condition for clearing to 0]

- Writing 0 to the AD1ICESRn bit after reading it as 1.

[Condition for setting to 1]

- An interrupt conversion process on AD1INm has been completed.

22.2.12.3 AD1 Interrupt Conversion End Status Register 2 (AD1ICESR2)

Address AD1ICESR2 : H'FFFE 800E

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	AD1ICESR47	AD1ICESR46	AD1ICESR45	AD1ICESR44	AD1ICESR43	AD1ICESR42	—	—	—	—	—	—	—	—	—	—
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 b10	AD1ICESRn	Interrupt Conversion End Flag	0: Indicates that an interrupt conversion process on AD1INm is in idle state. 1: Indicates that an interrupt conversion process on AD1INm has been completed and the conversion result has been transferred to the AD1DRn bit.	R/W
b9 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

Notes: • Even when the AD1ICESRn bit is not cleared to 0, an interrupt conversion request for AD1INm can be accepted. Exercise care regarding the timing of storing values in AD1 data register n.
• n = 42 to 47, m = 42 to 47

The AD1ICESR2 register indicates that an interrupt conversion has been completed. When an interrupt conversion has been completed, the AD1ICESRn bit corresponding to the channel (AD1INm) is set to 1.

AD1ICESRn Bit (n = 42 to 47)

AD1ICESRn is a status flag bit which indicates that an interrupt conversion has been completed. 1 must not be written to the AD1ICESRn bit. When the AD1ICESRn bit is set to 1, an AD1INm interrupt conversion end interrupt (AD1IDn) can be generated. See Table 22.4 and Table 22.5 for correspondence between AD1ICESRn and AD1INm.

[Condition for clearing to 0]

- Writing 0 to the AD1ICESRn bit after reading it as 1.

[Condition for setting to 1]

- An interrupt conversion process on AD1INm has been completed.

22.2.13 AD1 Interrupt Conversion End Interrupt Enable Register j (AD1ICIENj) (j = 0 to 2)

22.2.13.1 AD1 Interrupt Conversion End Interrupt Enable Register 0 (AD1ICIEN0)

Address AD1ICIEN0 : H'FFFE 801C

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	AD1ICIEN7	AD1ICIEN6	AD1ICIEN5	AD1ICIEN4	AD1ICIEN3	AD1ICIEN2	AD1ICIEN1	AD1ICIEN0
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 b8	—	Reserved	These bits are read as 0. The write value should be 0.	R
b7 to b0	AD1ICIENn	Interrupt Conversion End Interrupt Enable Bits	0: Disables an interrupt request upon completion of AD1INm interrupt conversion (AD1IDn) or a DMA request. 1: Enables an interrupt request upon completion of AD1INm interrupt conversion (AD1IDn) or a DMA request.	R/W

Note: n = 0 to 7, m = 00 to 07

The AD1ICIEN0 register enables or disables an A/D interrupt conversion end interrupt generation when the AD1ICESRn bit in the AD1 interrupt conversion end status register j (AD1ICESR0 to AD1ICESR2) is set to 1.

AD1ICIENn Bit (n = 0 to 7)

The AD1ICIENn bit enables or disables an AD1INm interrupt conversion end interrupt (AD1IDn) to be generated. To prevent incorrect operation, the ADITACT bit in the AD1 conversion status register (AD1CSR) must be cleared to 0 while the AD1ICIENn bit value is changed.

If the AD1ICIENn bit is 1 when the AD1ICESRn bit in the interrupt conversion end status register is set to 1 upon completion of AD1INm interrupt conversion, the AD1IDn signal is generated.

The AD1IDn signal can be cleared by clearing the AD1ICESRn bit or the AD1ICIENn bit to 0.

The correspondence among the AD1ICIENn bit, AD1INm, and AD1IDn is shown in Table 22.4 and Table 22.5.

22.2.13.2 AD1 Interrupt Conversion End Interrupt Enable Register 1 (AD1ICIEN1)

Address AD1ICIEN1 : H'FFFE 801E

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	AD1ICIEN31	AD1ICIEN30	AD1ICIEN29	AD1ICIEN28	AD1ICIEN27	AD1ICIEN26	AD1ICIEN25	AD1ICIEN24	—	—	—	—	—	—	AD1ICIEN17	AD1ICIEN16
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 b8	AD1ICIENn	Interrupt Conversion End Interrupt Enable Bits	0: Disables an interrupt request upon completion of AD1INm interrupt conversion (AD1IDn) or a DMA request. 1: Enables an interrupt request upon completion of AD1INm interrupt conversion (AD1IDn) or a DMA request.	R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b1, b0	AD1ICIENn	Interrupt Conversion End Interrupt Enable Bits	0: Disables an interrupt request upon completion of AD1INm interrupt conversion (AD1IDn) or a DMA request. 1: Enables an interrupt request upon completion of AD1INm interrupt conversion (AD1IDn) or a DMA request.	R/W

Note: n = 16, 17, 24 to 31, m = 16, 17, 24 to 31

The AD1ICIEN1 register enables or disables an A/D interrupt conversion end interrupt generation when the AD1ICESRn bit in the AD1 interrupt conversion end status register j (AD1ICESR0 to AD1ICESR2) is set to 1.

AD1ICIENn Bit (n = 16, 17, 24 to 31)

The AD1ICIENn bit enables or disables an AD1INm interrupt conversion end interrupt (AD1IDn) to be generated. To prevent incorrect operation, the ADITACT bit in the AD1 conversion status register (AD1CSR) must be 0 while the AD1ICIENn bit is changed.

If the AD1ICIENn bit is 1 when the AD1ICESRn bit in the interrupt conversion end status register is set to 1 upon completion of AD1INm interrupt conversion, the AD1IDn signal is generated.

The AD1IDn signal can be cleared by clearing the AD1ICESRn bit or the AD1ICIENn bit to 0.

See Table 22.4 and Table 22.5 for correspondence among the AD1ICIENn bit, AD1INm, and AD1IDn.

22.2.13.3 AD1 Interrupt Conversion End Interrupt Enable Register 2 (AD1ICIEN2)

Address AD1ICIEN2 : H'FFFE 8020

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	AD1ICIEN47	AD1ICIEN46	AD1ICIEN45	AD1ICIEN44	AD1ICIEN43	AD1ICIEN42	—	—	—	—	—	—	—	—	—	—
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 b10	AD1ICIENn	Interrupt Conversion End Interrupt Enable Bits	0: Disables an interrupt request upon completion of AD1INm interrupt conversion (AD1IDn) or a DMA request. 1: Enables an interrupt request upon completion of AD1INm interrupt conversion (AD1IDn) or a DMA request.	R/W
b9 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

Note: n = 42 to 47, m = 42 to 47

The AD1ICIEN2 register enables or disables an A/D interrupt conversion end interrupt generation when the AD1ICESRn bit in the AD1 interrupt conversion end status register j (AD1ICESR0 to AD1ICESR2) is set to 1.

AD1ICIENn Bit (n = 42 to 47)

The AD1ICIENn bit enables or disables an AD1INm interrupt conversion end interrupt (AD1IDn) to be generated. To prevent incorrect operation, the ADITACT bit in the AD1 conversion status register (AD1CSR) must be 0 while the AD1ICIENn bit is changed.

If the AD1ICIENn bit is 1 when the AD1ICESRn bit in the interrupt conversion end status register is set to 1 upon completion of AD1INm interrupt conversion, the AD1IDn signal is generated.

The AD1IDn signal can be cleared by clearing the AD1ICESRn bit or the AD1ICIENn bit to 0.

See Table 22.4 and Table 22.5 for correspondence among the AD1ICIENn bit, AD1INm, and AD1IDn.

22.2.14 AD1 Analog Port Pull-Down Register j (AD1APPDj) (j=0 to 2)

22.2.14.1 AD1 Analog Port Pull-Down Register 0 (AD1APPD0)

Address H'FFFE8034

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	AD1APPD7	AD1APPD6	AD1APPD5	AD1APPD4	AD1APPD3	AD1APPD2	AD1APPD1	AD1APPD0
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 b8	—	Reserved	These bits are read as 0. The write value should be 0.	R
b7 to b0	AD1APPDn	Bits Controlling Pull-Down MOS Elements for Analog Port Pins	0: The pull-down MOS elements are switched off. 1: The pull-down MOS elements are switched on.	R/W

Note: n = 0 to 7

The AD1APPD0 register performs the on-off control of the analog port pull-down MOS.

AD1APPDn Bit (n = 0 to 7)

Setting an AD1APPDn bit to 1 turns on the pull-down MOS element attached to the analog port corresponding to the bit.

For an outline of the pull-down function of the analog ports, see Figure 22.11.

Note: After return from the power off state, this register is reset to 0.
Analog input and digital input pin functions are multiplexed with the I/O port pins for use with the 10-bit A/D converter.
If a pin is not in use as an analog input pin due to the port function selection register for the I/O port, the pull-down setting for the pin becomes invalid. However, this is not reflected in the port pull-down register.
For the details on port-function selection, see section 13.2.21, Port Ji Function Select Register (PJiS) (i = 00 to 11).

22.2.14.2 AD1 Analog Port Pull-Down Register 1 (AD1APPD1)

Address H'FFFE8036

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	AD1APPD31	AD1APPD30	AD1APPD29	AD1APPD28	AD1APPD27	AD1APPD26	AD1APPD25	AD1APPD24	—	—	—	—	—	—	AD1APPD17	AD1APPD16
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 b8	AD1APPDn	Bits Controlling Pull-Down MOS Elements for Analog Port Pins	0: The pull-down MOS elements are switched off. 1: The pull-down MOS elements are switched on.	R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b1, b0	AD1APPDn	Bits Controlling Pull-Down MOS Elements for Analog Port Pins	0: The pull-down MOS elements are switched off. 1: The pull-down MOS elements are switched on.	R/W

Note: n=16, 17, 24 to 31

The AD1APPD1 register performs the on-off control of the analog port pull-down MOS.

AD1APPDn Bit (n=16, 17, 24 to 31)

Setting an AD1APPDn bit to 1 turns on the pull-down MOS element attached to the analog port corresponding to the bit.

For an outline of the pull-down function of the analog ports, see Figure 22.11.

Note: After return from the power off state, this register is reset to 0.
Analog input and digital input pin functions are multiplexed with the I/O port pins for use with the 10-bit A/D converter.
If a pin is not in use as an analog input pin due to the port function selection register for the I/O port, the pull-down setting for the pin becomes invalid. However, this is not reflected in the port pull-down register.
For the details on port-function selection, see section 13.2.21, Port Ji Function Select Register (PJiS) (i = 00 to 11).

22.2.14.3 AD1 Analog Port Pull-Down Register 2 (AD1APPD2)

Address H'FFFE8038

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	AD1APPD47	AD1APPD46	AD1APPD45	AD1APPD44	AD1APPD43	AD1APPD42	—	—	—	—	—	—	—	—	—	—
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 b10	AD1APPDn	Bits Controlling Pull-Down MOS Elements for Analog Port Pins	0: The pull-down MOS elements are switched off. 1: The pull-down MOS elements are switched on.	R/W
b9 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

Note: n = 42 to 47

The AD1APPD2 register performs the on-off control of the analog port pull-down MOS.

AD1APPDn Bit (n = 42 to 47)

Setting an AD1APPDn bit to 1 turns on the pull-down MOS element attached to the analog port corresponding to the bit.

For an outline of the pull-down function of the analog ports, see Figure 22.11.

Note: After return from the power off state, this register is reset to 0.
Analog input and digital input pin functions are multiplexed with the I/O port pins for use with the 10-bit A/D converter.
If a pin is not in use as an analog input pin due to the port function selection register for the I/O port, the pull-down setting for the pin becomes invalid. However, this is not reflected in the port pull-down register.
For the details on port-function selection, see section 13.2.21, Port Ji Function Select Register (PJiS) (i = 00 to 11).

22.2.15 AD1 Self-Test Register (AD1DIAG)

Address H'FFFE8032

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	DIAG[1:0]	
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b1, b0	DIAG[1:0]	A/D Self-Diagnosis Bits	b1 b0 0 0 : The self-test function is off. 0 1 : A/D conversion is enabled at a voltage of AVCC1 × 0. 1 0 : A/D conversion is enabled at a voltage of AVCC1 1/2. 1 1 : A/D conversion is enabled at a voltage of AVCC1 × 1.	R/W

The AD1DIAG register controls generation of the internal voltages for use in detecting faults in the A/D converter.

DIAG Bit

Setting the DIAG bits to a value other than 00 generates an internal voltage in accord with the setting, and this voltage is A/D converted.

At this time, the analog inputs are ignored. To prevent incorrect operation, the ADST bit in the AD1CR register must be cleared to 0 while the DIAG bits are set.

22.2.16 Interface with CPU

The AD1 data register is a 16-bit register that is connected to the CPU via the 16-bit peripheral bus. The AD1 data register must be read in units of words (16 bits). If the AD1 data register is read in byte units by dividing a word into upper and lower bytes and performing read operations twice on it, the A/D converted value read in the first read operation and that read in the second read operation may change. To avoid this error, the AD1 data register should not be read in byte units.

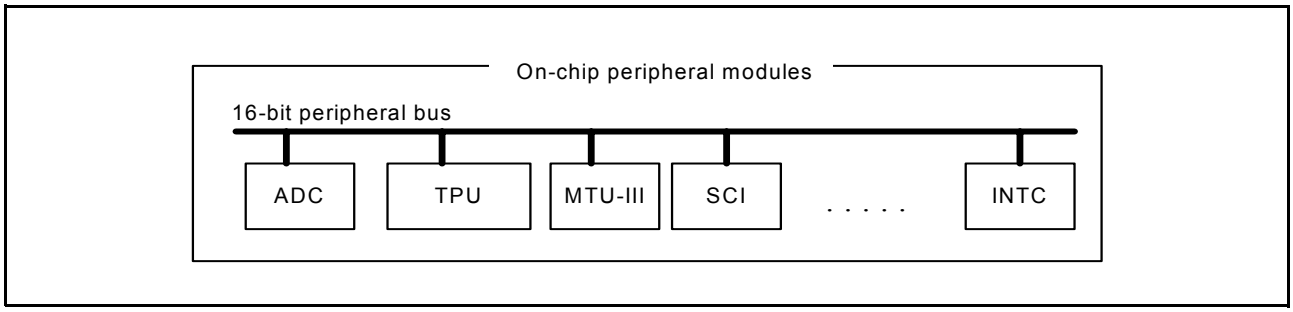


Figure 22.3 Interface between CPU and A/D Converter (ADC)

22.3 Operations

22.3.1 Scan Conversion

A scan conversion is performed in two operating modes: single-cycle scan mode and continuous scan mode. In single-cycle scan mode, one or more specified channels are scanned once. In continuous scan mode, one or more specified channels are scanned until the ADST bit in the AD1 control register (AD1CR) is cleared to 0 (changed from 1 to 0) by software.

Single-cycle scan mode is selected by clearing the ADCS bit in the AD1 control register (AD1CR) to 0, while continuous scan mode is selected by setting the ADCS bit to 1. Scan conversion proceeds in order from lower-numbered to higher-numbered channels: AD1IN00 to AD1IN47 for AD1.

In single-cycle scan mode, after scanning all selected channels once, the A/D converter sets the ADF bit in the AD1 conversion status register (AD1CSR) to 1 and then clears the ADSCACT bit in the AD1CSR register to 0 to completes the scan conversion. In continuous scan mode, after scanning all selected channels once, the A/D converter sets the ADF bit to 1 and then continues to scanning. The ADF bit is set to 1 each time scanning on specified channels is completed.

To stop the scanning, write 0 to the ADST bit when it is 1. Writing 0 to the ADST bit when it is 0 does not affect the A/D converter. Similarly, writing 1 to the ADST bit when it is 1 does not affect the A/D converter. Therefore, to stop a scan conversion started by a request other than the ADST bit, first write 1 to the ADST bit and then write 0 to it.

When the ADF bit is set to 1 while the ADIE bit in the AD1 control register (AD1CR) is set to 1, an ADI interrupt request is generated. To clear the ADF bit to 0, write 0 to the ADF bit after reading it as 1.

22.3.2 Single-Cycle Scan Conversion Mode

The following is an example operation of single-scan conversion where three channels AD1IN00, AD1IN03 and AD1IN07 are selected and an AD1I interrupt is enabled.

1. Clear the ADCS bit in the AD1 control register (AD1CR) to 0 and set the ADIE bit in the AD1CR register to 1.
2. Set bits AD1CSL0, AD1CSL3 and AD1CSL7 in the AD1 channel select register (AD1CSL) to 1.
3. Set the ADST bit in the AD1 control register (AD1CR) to 1 to start scan conversion. If the ADST bit is already set to 1, write 1 to it after clearing it to 0.
   In this case, write 1 to the ADST bit after the interval of a specified period * or more.

   Note: At the time when the CKS is 0: 2 cycles of the peripheral bus clock A
   At the time when the CKS is 1: 4 cycles of the peripheral bus clock A

4. Starting the scan conversion sets the ADSCACT bit to 1. Then, the A/D conversion on channel AD1IN00 is started. On completion of the A/D conversion, the A/D converted value is transferred to the AD1DR0 register. After that, channels AD1IN03 and AD1IN07 are scanned in the order in the same way as in AD1IN00.
5. When the A/D converted values of all the selected channels (AD1IN00, AD1IN03 and AD1IN07) have been transferred to the AD1DRn register, the ADF bit is set to 1. At this time, an AD1I interrupt is generated since the ADIE bit is set to 1. The ADSCACT bit is cleared to 0 and the scan conversion is completed.
6. Next, the AD1I interrupt handler is started. In the interrupt handler, clear the AD1I by writing 0 to the ADF bit after reading it as 1. After that, read the contents of AD1DR0, AD1DR3 and AD1DR7.
7. Complete the AD1I interrupt handler.

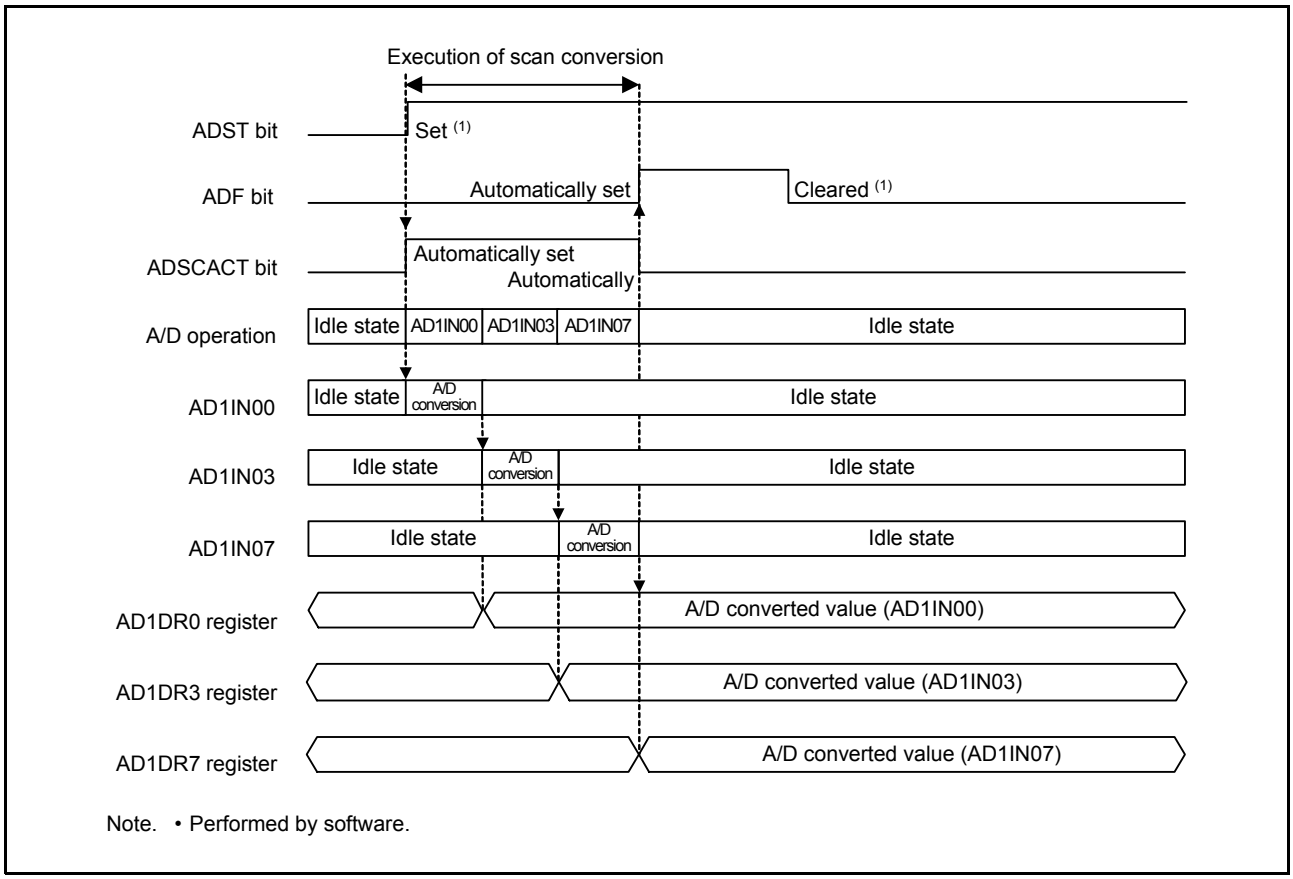


Figure 22.4 Example Operation in Single-Cycle Scan Mode

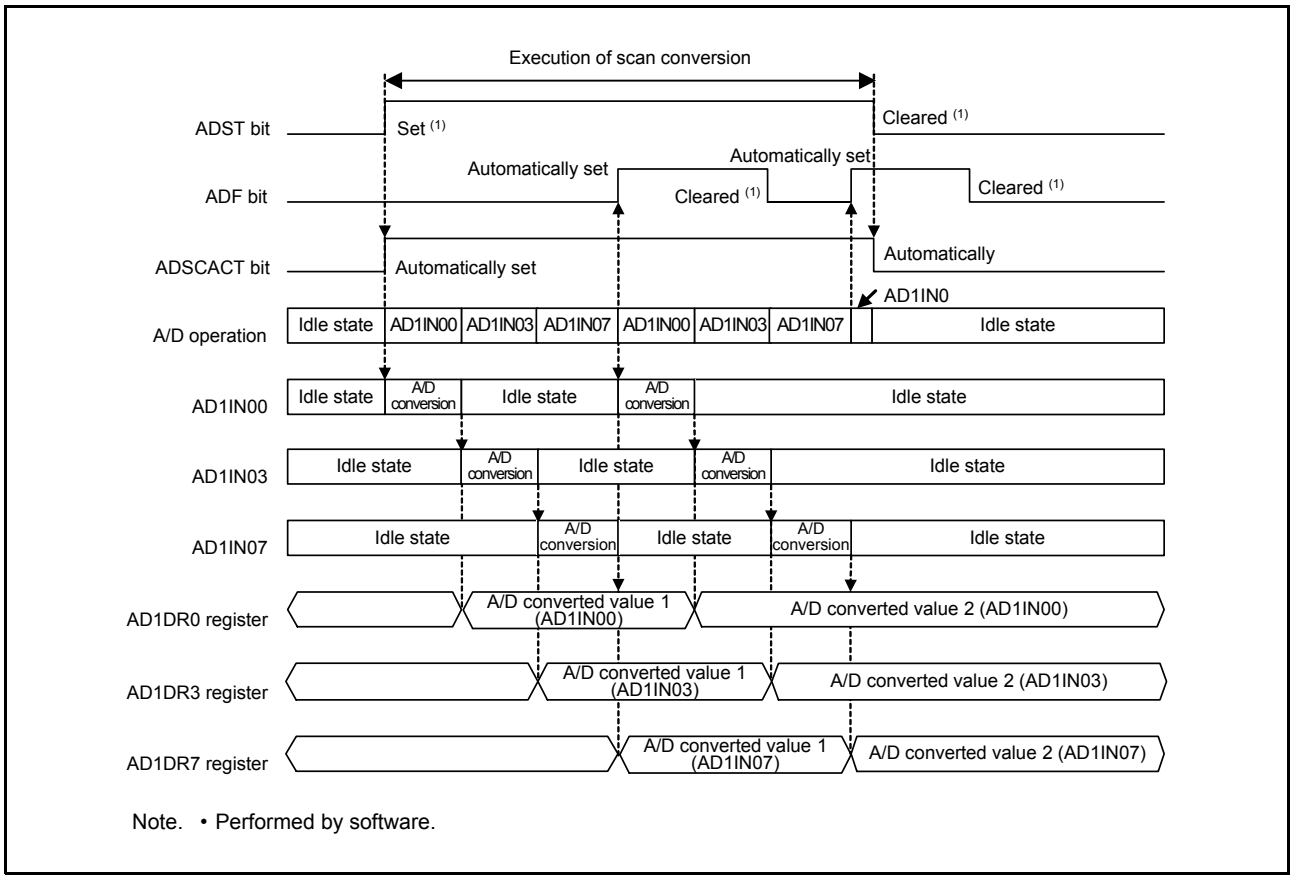
22.3.3 Continuous Scan Conversion Mode

The following is an example operation of continuous scan conversion where three channels AD1IN00, AD1IN03 and AD1IN07 are selected and an AD1I interrupt is enabled. The same operations can also apply to AD2.

1. Set the ADCS bit and ADIE bit in the AD1 control register (AD1CR) to 1.
2. Set bits AD1CSL0, AD1CSL3, and AD1CSL7 in the AD1 channel select register (AD1CSL) to 1.
3. Set the ADST bit in the AD1 control register (AD1CR) to 1 to start scan conversion. If the ADST bit is already 1, write 1 to it after clearing it to 0.
   In this case, write 1 to the ADST bit after the interval of a specified period *1 or more.

Note 1. At the time when the CKS is 0: 2 cycles of the peripheral bus clock A
At the time when the CKS is 1: 4 cycles of the peripheral bus clock A

4. Starting the scan conversion sets the ADSCACT bit to 1. Then, the A/D conversion on channel AN0 is started. On completion of the A/D conversion, the A/D converted value is transferred to the AD1DR00 register. After that, channels AD1IN03 and AD1IN07 are scanned in the order in the same way as in AD1IN00.
5. When the A/D converted values of all the selected channels (AD1IN00, AD1IN03, and AD1IN07) have been transferred to the AD1DRn register, the ADF bit is set to 1. At this time, an AD1I interrupt is generated since the ADIE bit is set to 1. Also, the scan conversion returns to the start.
6. The AD1I interrupt handler is started simultaneously. In the interrupt handler, clear the AD1I interrupt by writing 0 to the ADF bit after reading it as 1. After that, read the contents of the AD1DR0, AD1DR3, and AD1DR7 registers.
7. Complete the AD1I interrupt handler.
8. Steps 4 to 7 are repeated as long as the ADST bit is 1. Clearing the ADST bit to 0 clears the ADSCACT bit to 0, and completes the scan conversion. Setting the ADST bit to 1 initiates scan conversion.



Note. • Performed by software.

Figure 22.5 Example Operation in Continuous Scan Mode

22.3.4 Interrupt Conversion

When a TPU (PWM) timer trigger/TPCNT reset/MTU-III timer trigger or software trigger is requested on channels AD1IN00 to AD1IN47, A/D conversion is performed on the requested channels. In scan conversion all selected channels are converted when a request is received, but in interrupt conversion channels are converted one at a time in response to individual requests.

To perform interrupt conversion, set the AD1ICENn bit in the AD1ICEN0 to AD1ICEN2 registers to 1. Then select the trigger source by bits AD1ICSSLn bit in the AD1ICSSL0 to AD1ICSSL2 registers. When interrupt conversion is requested by the selected trigger source, A/D conversion is performed on the corresponding AD1INm channel. On completion of the interrupt conversion on the AD1INm channel, the AD1ICESRn bit in the AD1ICESR0 to AD1ICESR2 registers is set to 1. The AD1ICESRn bit is set to 1 each time an interrupt conversion is performed on an AD1INm channel. Furthermore, if any interrupt conversion is performed, the ADITACT bit in the AD1CSR0 to AD1CSR2 registers is set to 1. When A/D conversion has been completed on all AD1INm channels to which interrupt conversion is requested, the ADITACT bit is cleared to 0.

When interrupt conversion requests conflict, A/D conversion is performed according to the priority. AD1 is prioritized as AD1IN00 > AD1IN01 > ... AD1IN46 > AD1IN47, that is, the lower channel number corresponds to the higher priority. When interrupt conversion is requested on another channel (AD1INj, AD1INk) during the interrupt conversion on channel AD1INi, the A/D conversion is not interrupted during the conversion regardless of the priority. In this case, on completion of the A/D conversion on channel AD1INi, A/D conversion is performed according to the priority on remaining channels (in this case, AD1INj and AD1INk) in which interrupt conversion requests are pending. Therefore, the priority on interrupt conversion determines which channel is to be converted for the next operation. When a single trigger source generates interrupt conversion requests on two channels or multiple trigger sources simultaneously generate interrupt conversion requests, A/D conversion is performed according to this priority.

When interrupt conversion is requested during scan conversion, the scan conversion on channel AD1INi is suspended, and A/D conversion on the other channel (AD1ANj) in which the interrupt conversion was requested is performed. On completion of the interrupt conversion on channel AD1ANj, the scan conversion is resumed from the interrupted channel (AD1INi). This scheme ensures that the length of time required from the initiation of an interrupt conversion request to the completion of it is always constant. This makes it possible, for example, to perform A/D conversion in pin-point accuracy by synchronizing them with the operation of A/D conversion sources that are external to the LSI.

When the AD1ICESRn bit is set to 1 while the AD1IDEn bit in the AD1ICIEN register is set to 1, an AD1IDn interrupt is requested. To clear the AD1ICESRn bit to 0, write 0 to the AD1ICESRn bit after reading it as 1. The DMA transfer of the DMAC is supported on channels AD1IN00 (AD1ID0) to AD1IN47 (AD1ID47).

22.3.5 Example Operation of Interrupt Conversion

The following is an example of interrupt conversion operation where TPU timer TPO1D is selected as the trigger source for channel AD1IN07 and TPU timers TPO4A and TPO4B are selected as the trigger source for channels AD1IN16 and AD1IN17.

1. Set bits AD1ICEN7, AD1ICEN16, and AD1ICEN17 in the AD1 interrupt conversion enable register (AD1ICEN) to 1.
2. Clear bits AD1ICSSL7, AD1ICSSL16, and AD1ICSSL17 in the AD1 interrupt conversion source select register (AD1ICSSL) to 0.
3. Subsequently, interrupt conversion requests are generated by TPO1D, TPO4A and TPO4B at intervals specified by the PWM registers. For details on the PWM registers, refer to section 15, Timer Pulse Unit (TPU).
4. When interrupt conversion is requested by TPO1D, the ADITACT bit in the AD1CSR register is set to 1 and interrupt conversion on channel AD1IN07 is performed. On completion of A/D conversion on AD1IN07, the A/D converted value of AD1IN07 is transferred to the AD1DR7 and the AD1ICESR7 bit in the AD1ICESR register is set to 1. The ADITACT bit is cleared to 0 and the interrupt conversion is completed. Furthermore, if the AD1ICIEN7 bit in the AD1ICIEN0 register is 1, an AD1ID07 interrupt is requested to the CPU.
5. When interrupt conversion is requested by TPO4A and TPO4B, the ADITACT bit in the AD1CSR register is set to 1, and interrupt conversion on channels AD1IN16 is performed. Then A/D conversion on channel AD1IN16 is performed. On completion of the conversion, the A/D converted value of AD1IN16 is transferred to the AD1DR16 register, and the AD1ICESR16 bit in the AD1ICESR register is set to 1. An A/D conversion on channel AD1IN17 is then performed. On completion of the conversion, the A/D converted value of AD1IN17 is transferred to the AD1DR17 register, and the AD1ICESR17 bit in the AD1ICESR register is set to 1. The ADITACT bit is cleared to 0 and the interrupt conversion is completed. Further, if bits AD1ICIEN16 and AD1ICIEN17 in the AD1ICIEN register are set to 1 when either the AD1ICESR16 or AD1ICESR17 bit is set to 1, the A/D converter requests an AD1ID16 or an AD1ID17 interrupt to the CPU.
6. Subsequently, steps 4 to 5 are repeated. The following is an example operation when requests by TPO1D, TPO4A and TPO4B conflict.

- Example Operation 1

When a TPO1D and TPO4B interrupt conversion request is input during the A/D conversion on channel AD1IN16 due to a TPO4A interrupt conversion request, the request is processed as follows.

The TPO1D and TPO4B interrupt source is retained in the A/D converter until conversion on channel AD1IN16 completes. When A/D conversion on channel AD1IN16 finishes, A/D conversion is performed on channels AD1IN07 and AD1IN17, in that order, according to their priority.

- Example Operation 2

When interrupt conversion requests by TPO1D, TPO4A and TPO4B are input simultaneously, the requests are processed as follows.

The TPO1D, TPO4A and TPO4B interrupt sources are retained in the A/D converter. Then A/D conversion is performed on channels AD1IN07, AD1IN16, and AD1IN17, in that order, according to their priority.

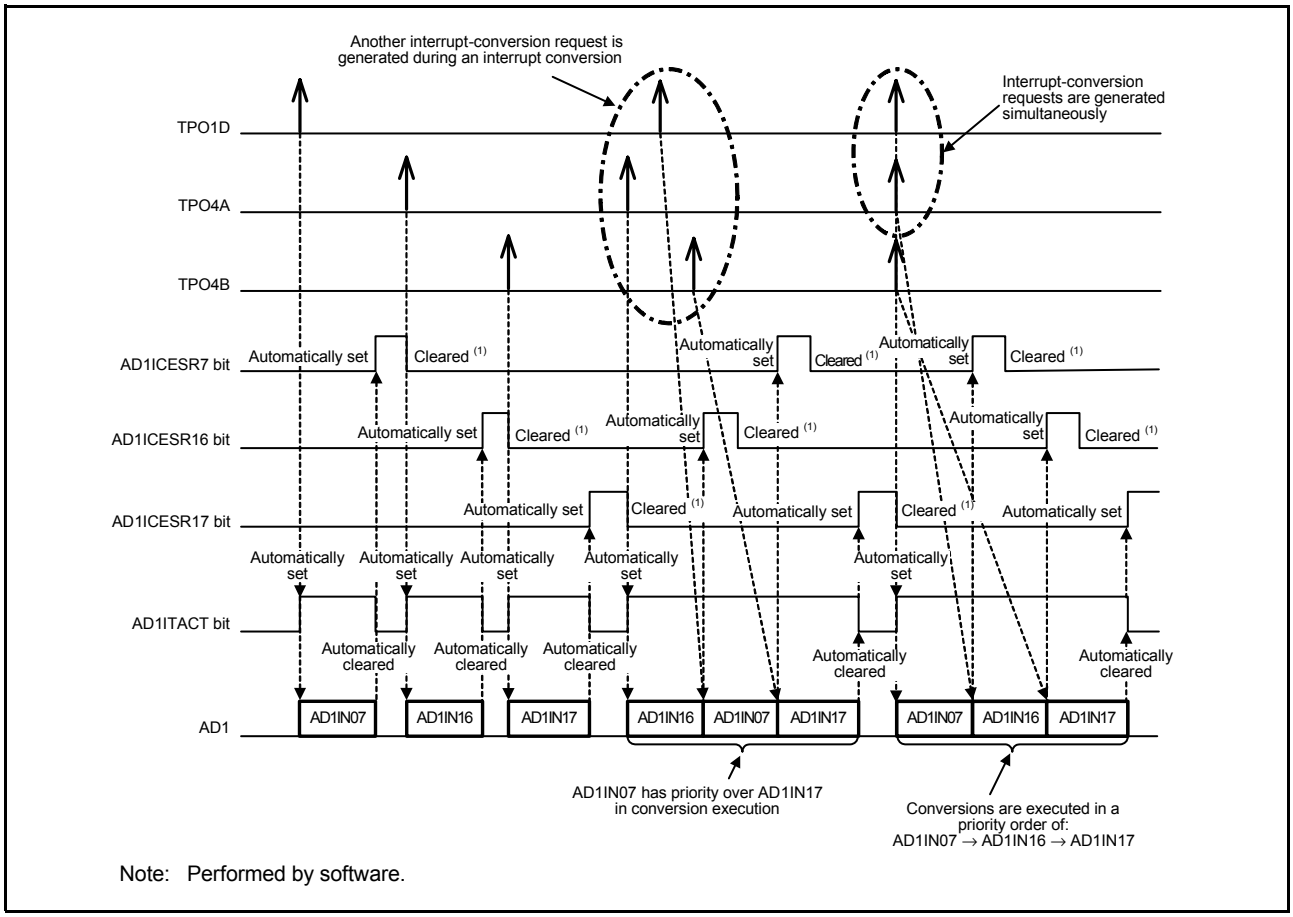


Figure 22.6 Example Operation of Interrupt Conversion

22.3.6 Interrupt Conversion during Scan Conversion

The following is an example operation where single scan conversion on three channels AD1IN04, AD1IN07, and AD1IN17 are started by a scan conversion request from TPU (PWM) timer TPO1A and then an interrupt conversion on channel AD1IN06 is started by a interrupt conversion request from TPU (PWM) timer TPO4A.

1. Clear the ADCS and EXTRG bits in the AD1 control register (AD1CR) to 0, and set the TRGE bit in the AD1CR register to 1.
2. Set the AD1CSL4, AD1CSL7, and AD1CSL17 bits in the AD1 channel select register (AD1CSL) to 1.
3. Set the AD1ICEN16 bit in the AD1 interrupt conversion enable register (AD1ICEN0) to 1.
4. Clear the AD1ICSSL16 bit in the AD1 interrupt conversion source select register (AD1ICSSL0) to 0.
5. Subsequently, a scan conversion request is generated by TPU (PWM) timer TPO1A and an interrupt conversion request is generated by TPU (PWM) timer TPO4A at intervals specified by the PWM registers. For details on the PWM registers, refer to section 15, Timer Pulse Unit (TPU).
6. When scan conversion is requested by TPU (PWM) timer TPO1A, the ADSCACT bit is set to 1. Then, A/D conversion on channels AD1IN04, AD1IN07, and AD1IN017 is performed in the order. On completion of the conversion, the ADF bit is set to 1 and the ADSCACT bit is cleared to 0, indicating that the scan conversion is completed.
7. When interrupt conversion is requested by TPU (PWM) timer TPO4A, the ADITACT bit is set to 1 and interrupt conversion on channel AD1IN16 is performed. On completion of the A/D conversion on channel AD1IN16, the AD1ICESR16 bit in the AD1ICESR1 register is set to 1 and the ADITACT bit is cleared to 0, indicating that the interrupt conversion is completed.
8. Subsequently, steps 6 to 7 are repeated. The following is an example operation where a scan conversion and an interrupt conversion conflict.

- Example Operation

When a TPU (PWM) timer TPO4A interrupt conversion request is input during the A/D conversion on channel AD1IN07 in the scan conversion due to a TPU (PWM) timer TPO1A scan conversion request, the request is processed as follows.

The TPU (PWM) timer TPO4A interrupt source is retained in the A/D converter, and the scan conversion on channel AD1IN07 is suspended. The priority is applied to channels AD1IN07 and AD1IN17 on which scan conversion is pending, and is applied to AD1IN16 which is the current request. In this case, the A/D conversion on channels AD1IN16, AD1IN07, and AD1IN17 in the order.

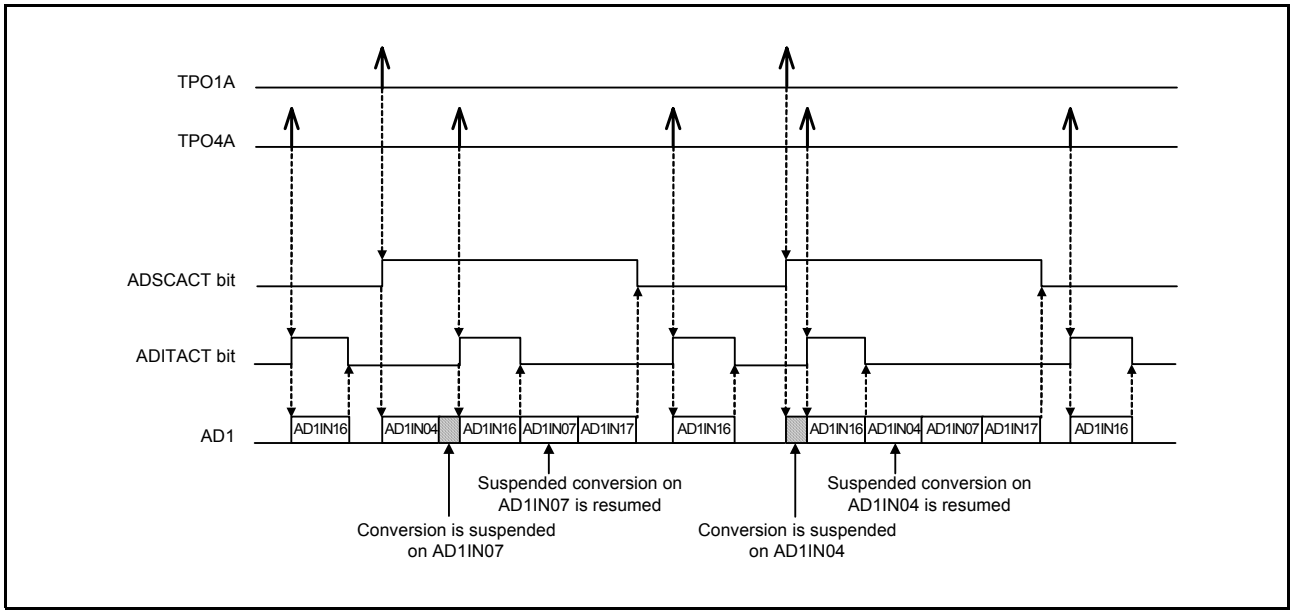


Figure 22.7 Operation Example of Interrupt Conversion during Scan Conversion

22.3.7 Analog Input Sampling and Scan Conversion Time

The A/D converter includes the sample and hold circuit. When start-of-scan-conversion delay time (tD) have passed after the ADST bit in the AD1CR register is set to 1, the A/D converter samples the analog input, and then begins the conversion process.

Figure 22.8 shows a timing chart for a scan conversion on one channel in single-cycle scan mode. Scan conversion time (tSCAN) includes start-of-scan-conversion delay time (tD), analog input sampling time (tSPL), A/D conversion processing time (tCONV) and end-of-scan-conversion delay time (tED). The scan conversion time is shown in Table 22.11.

The scan conversion time (tSCAN) in single-cycle scan mode for which the number of selected channels is n can be determined according to the following equation:

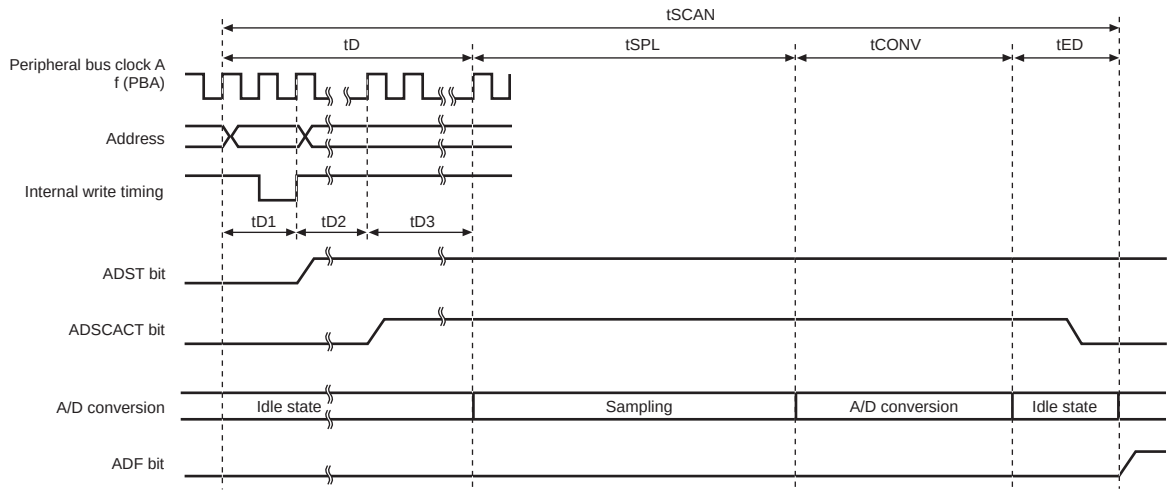
$tSCAN = tD + \{(tSPL + tCONV) \times n\} + tED$

The scan conversion time for the first cycle in continuous scan mode is tSCAN for single-cycle scan minus tED.

The scan conversion time for the second and subsequent cycles in continuous scan mode is a fixed time, which is equal to $\{(tSPL + tCONV) \times n\}$.

Table 22.11 Scan Conversion Time

Item	Symbol	f (PBA) = 50MHz (f (PBA) conversion)		Unit
		CKS = “0”	CKS = “1”	
Start-of-scan-conversion delay time	tD	7	11 to 12	State
Write cycle	tD1	2	2	
Synchronization time	tD2	2	3 to 4	
Time until sampling is started after the rising of the ADSCACT bit	tD3	3	6	
Analog input sampling time	tSPL	20	40	
A/D conversion processing time	tCONV	30	60	
End-of-scan-conversion delay time	tED	4	7	
Scan conversion time	tSCAN	61	118 to 119	



[Legend]
tD : Start-of-scan-conversion delay time
tSPL : Analog input sampling time
tCONV : A/D conversion time
tED : End-of-scan-conversion delay time
tSCAN : Scan conversion time
tD1 : Write cycle
tD2 : Synchronization time
tD3 : Time until sampling is started after the rising of the ADSCACT bit

Figure 22.8 Timing Diagram for Scan Conversion (Single Channel, Single Cycle)

22.3.8 Starting Scan Conversion with External Trigger

The A/D converter can be activated by the input of an external trigger (only the SH72A2 group. The SH72A0 group cannot activate the A/D converter by external trigger). To start up the A/D converter by an external trigger, the pin function should be set up. Concerning pin function settings, refer to section 13, I/O Ports. After applying a high level signal to the ADTRG# pin, both the TRGE and EXTRG bits in the AD1 control register (AD1CSR) should be set to 1. If a low level signal is then input to the ADTRG# pin, the A/D converter detects a pulse fall edge and sets the ADSCACT bit to 1.

Figure 22.9 shows an external of trigger input timing. Table 22.12 lists scan conversion time for external trigger input. The timing at which a scan conversion is started after the ADSCACT bit is set to 1 is the same as the case where the ADST bit is set to 1 from 0 by software. For details on pin function setting, refer to section 13, I/O Ports.

To stop the scan conversion process while it is in progress, write 1 to the ADST bit and then write 0 to it.

Table 22.12 Scan Conversion Time for External Trigger Input

Item	Symbol	f (PBA) = 50MHz (f (PBA) conversion)		Unit
		CKS = “0”	CKS = “1”	
Start-of-scan-conversion delay time	tD	8	13 to 14	State
Time until the rising of the ADSCACT bit after the falling of the ADTRG# pin is sampled	tD4	5	7 to 8	
Time until sampling is started after the falling of the ADSCACT bit	tD3	3	6	
Analog input sampling time	tSPL	20	40	
A/D conversion processing time	tCONV	30	60	
End-of-scan-conversion delay time	tED	4	7	
Scan conversion time	tSCAN	62	120 to 121	

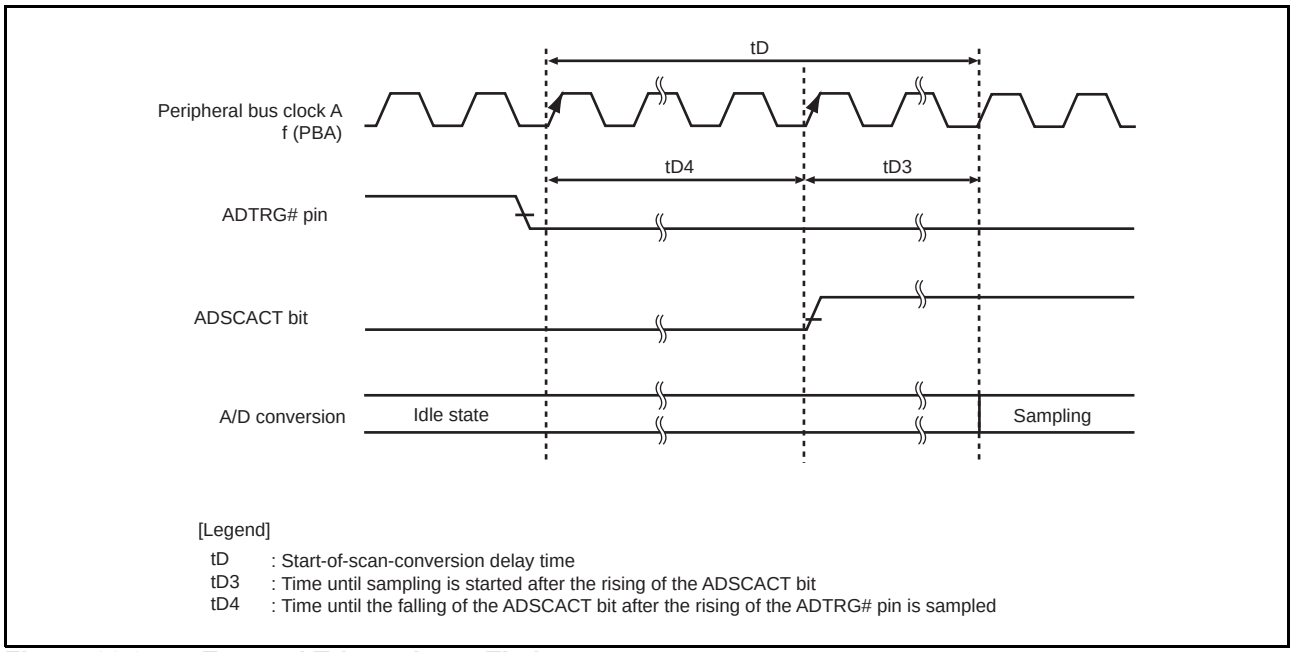


Figure 22.9 External Trigger Input Timing

22.3.9 Starting Scan Conversion with TPU (PWM) Timer Trigger, TPCNT Reset, or MTU-III Timer Trigger

A scan conversion can be activated by a TPU (PWM) timer trigger, TPCNT reset, or MTU-III timer trigger. To start up a scan conversion by a TPU (PWM) timer trigger, TPCNT reset, or MTU-III timer trigger, the TRGE bit in the AD1 control register (AD1CR) is set to 1, and the EXTRG bit is set to 0. If a TPU (PWM) timer trigger, TPCNT reset, or MTU-III timer trigger is entered in this situation, the ADSCACT bit is set to 1. The timing at which a scan conversion is started after the ADSCACT bit is set to 1 is the same as the case where the ADST bit is set to 1 from 0 by software. To stop the scan conversion process while it is in progress, write 1 to the ADST bit and then write 0 to it.

22.4 Interrupt Sources and DMA Transfer Request

22.4.1 Interrupt Requests on Completion of Scan Conversion

The A/D converter can generate a scan conversion end interrupt request (ADI) to the CPU. By setting the ADIE bit in the AD1 control register (AD1CR) to 1, an ADI interrupt is enabled; by clearing the bit to 0, an ADI interrupt is disabled. In addition, the DMAC can be started up when an ADI interrupt is generated. In this case, interrupts are not generated to the CPU.
For details on DMAC settings, refer to section 12, DMAC and section 8, Interrupt Controller (INTC).

Note: * The ADF bit is not cleared by an interrupt request to the CPU.

22.4.2 Interrupt Requests on Completion of Interrupt Conversion

The A/D converter can generate interrupt conversion end interrupt requests (AD1ID0 to AD1ID47) to the CPU upon completion of an interrupt conversion. By setting the AD1IDE0 to AD1IDE47 and bits in the AD1 interrupt conversion end interrupt enable registers (AD1ICIEN0 to AD1ICIEN2) to 1, the AD1ID0 to AD1ID47 interrupts are enabled, by clearing the bits to 0, the AD1ID0 to AD1ID47 interrupts are disabled. In addition, the DMAC can be started up by setting registers in the interrupt control circuit when an ADI interrupt is generated. In this case, interrupts are not generated to the CPU.
For details on DMAC settings, refer to section 12, DMAC and section 8, Interrupt Controller (INTC).

Note: * The AD1ICESR bit is not cleared by an interrupt request to the CPU.

22.5 Definition of A/D Conversion Accuracy

The definition of A/D conversion accuracy is described below.

- Resolution

This indicates the number of digital output codes in the A/D converter

- Quantization error

This error, which is inherent to the A/D converter, is given as 1/2LSB (Figure 22.10).

- Offset error

This error, which is exclusive of quantization error, is a deviation of the analog input voltage value from the ideal A/D conversion characteristics when the digital output changes from a minimum voltage value B'0000 0000 0000 to B'0000 0000 0001 (Figure 22.10).

- Full scale error

This error, which is exclusive of quantization error, is a deviation of the analog input voltage value from the ideal A/D conversion characteristics when the digital output changes from B'1111 1111 1110 to B'1111 1111 1111 (Figure 22.10).

- Nonlinearity error

This error, which is exclusive of offset error, full scale error and quantization error, is a deviation from the ideal A/D conversion characteristics through the zero-scale and full-scale transitions (Figure 22.10).

- Absolute accuracy

This is a deviation of the digital value from the analog input value. This includes offset error, full scale error, quantization error, and nonlinearity error.

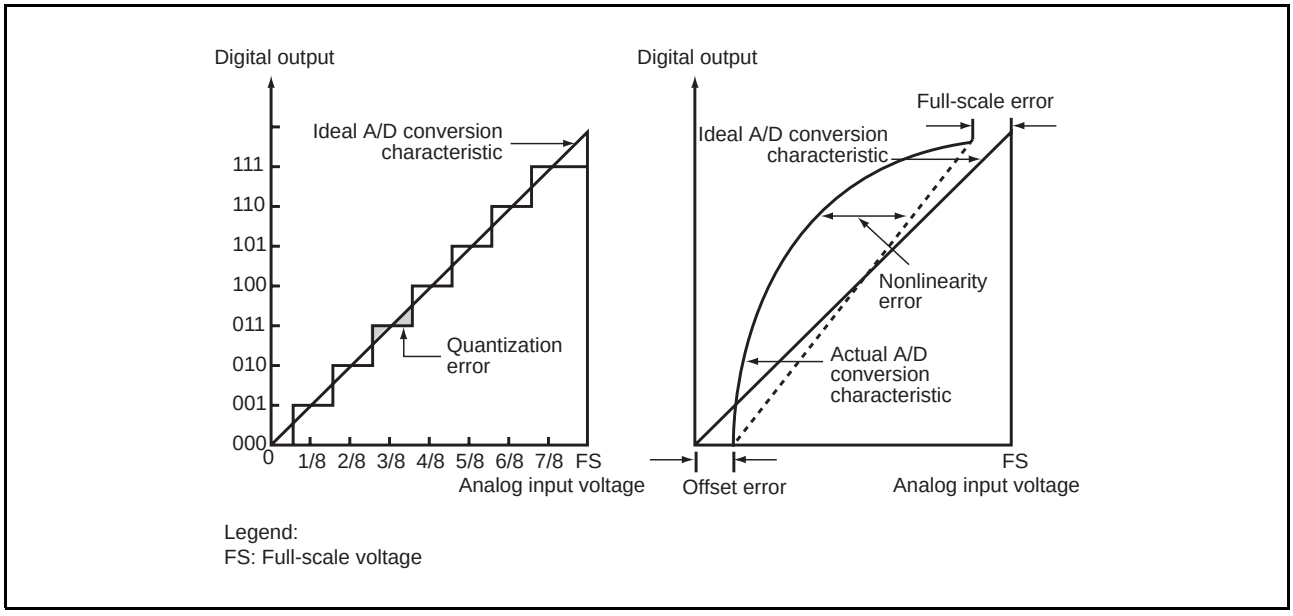


Figure 22.10 Definition of A/D Conversion Accuracy

22.6 Analog Port Pull-Down Function

Each analog port has a pull-down MOS element. Setting the AD1APPD0 to AD1APPD47 bits to 1 (the default is 0) turns on the pull-down MOS elements for the corresponding analog channels.

When voltage from an external circuit is being applied to an analog port pin, the result of A/D conversion can be used to check for disconnection between them.

- Examples

Normal operation: The result of conversion is near the voltage applied from the external circuit.

Abnormal operation (disconnection): The result of conversion is near AVSS.

Note: Analog input and digital input pin functions are multiplexed with the I/O port pins for use with the 10-bit A/D converter. If a pin is not in use as an analog input pin due to the port function selection register for the I/O port, the pull-down setting for the pin becomes invalid. However, this is not reflected in the port pull-down register. For the details on port-function selection, see section 13.2.21, Port Ji Function Select Register (PJiS) (i = 00 to 11), section 13.2.22, Port Ki Function Select Register (PKiS) (i = 00, 01, and 08 to 15), and section 13.2.23, Port Li Function Select Register (PLiS) (i = 10 to 15).

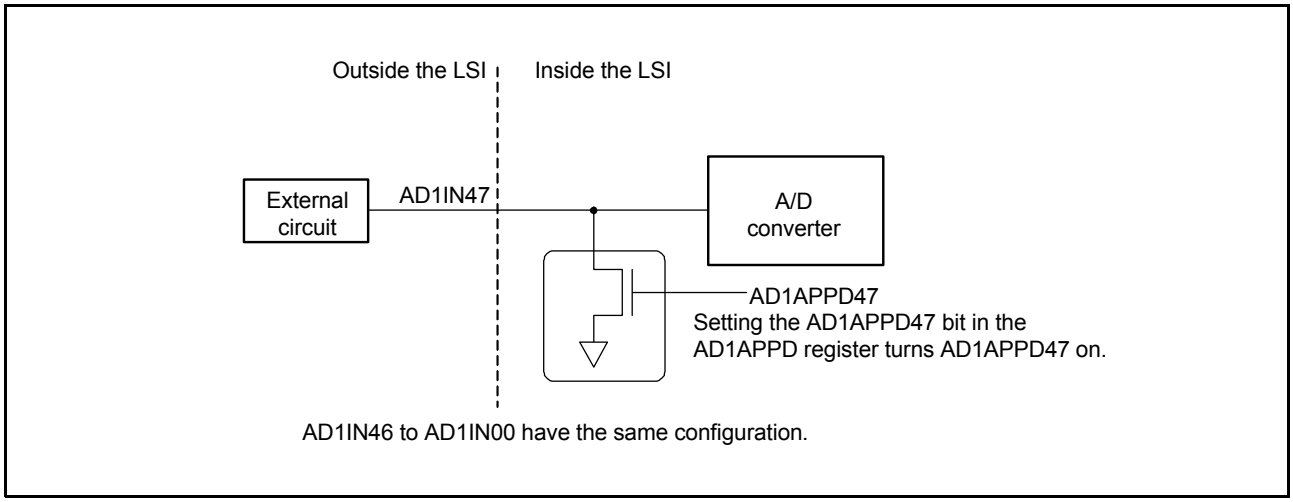


Figure 22.11 Schematic of Analog Port Pull-Down Function

22.7 Self-Test of A/D Converter

Software can handle self-diagnosis of the A/D converter.

AD1DIAG selects the internal voltage to be subject to A/D conversion, and this voltage is A/D converted. After that, software reads AD1DR and AD1DIAG and checks whether or not the A/D-converted value is in the normal range.

When a voltage to be subject to A/D conversion is selected, the analog inputs are ignored. To prevent incorrect operation, the ADST pin in AD1CR must be 0 while AD1DIAG is set.

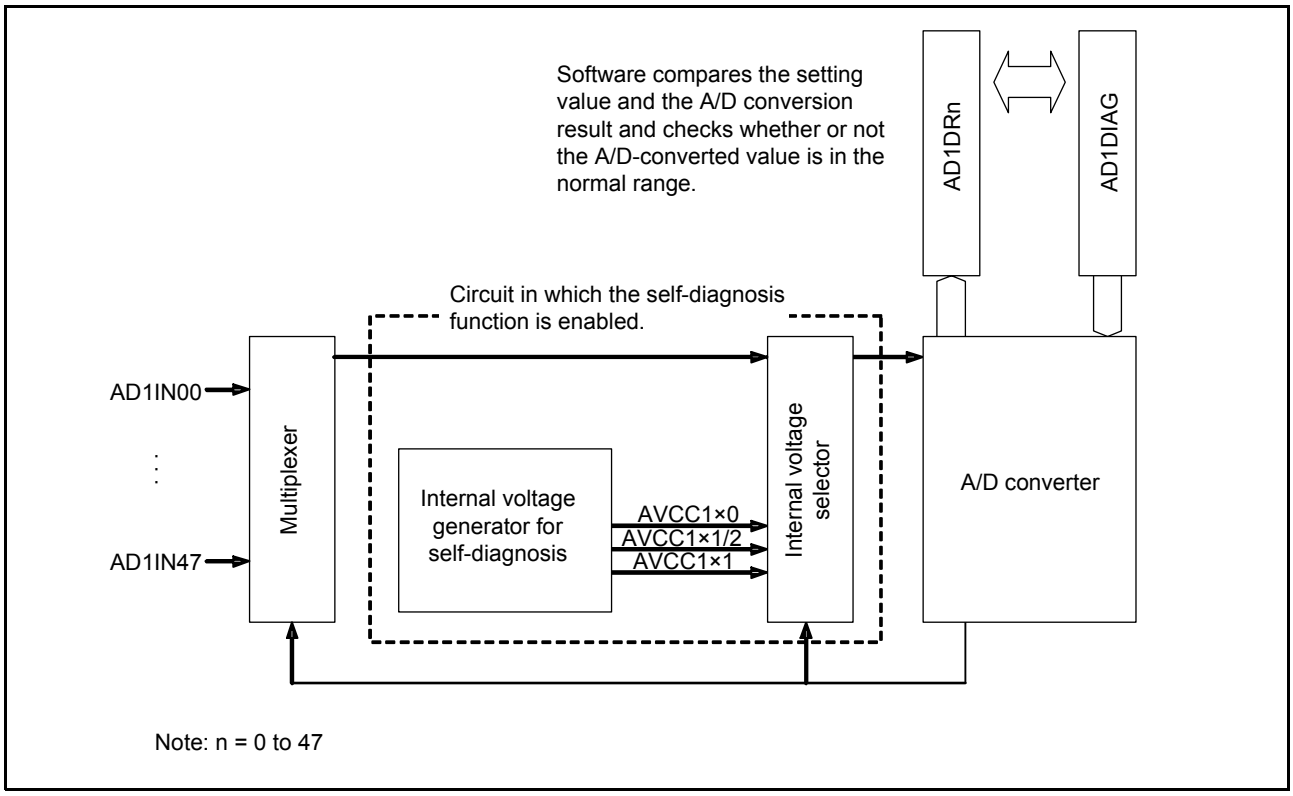


Figure 22.12 Schematic of Self-Diagnosis of A/D Converter

Table 22.13 Ideal A/D-Converted Values (in Normal Range)

Selected voltage	AD1DR (in normal range)
AVCC1×0	H'0000 (H'0000 to H'0400*)
AVCC1×1/2	H'7FC0 (H'7BC0 to H'H83C0*)
AVCC1×1	H'FFC0 (H'FBC0 to H'FFC0*)

Note: * The range of values includes those corresponding to errors in the voltage from the internal voltage generator.

22.8 Notes on 10-Bit A/D Converter (AD1)

22.8.1 Analog Input Voltage Range

The voltage applied to the analog input pin (AD1INm) during A/D conversion should be within a range of AVSS1 ≤ AD1INm (m = 00 to 47) ≤ AVCC1.

22.8.2 Relationship among AVCC1, AVSS1, VCC, and VSS

When using the A/D converter ADC1, make sure that the following relationships are held among AVCC1, AVSS1, VCC and VSS:
AVCC1 = VCC±0.3V, AVSS1 = VSS
When the A/D converter is not used, make sure that the following relationships are held among AVCC1, AVSS1, VCC, and VSS: AVSS1 = VSS, AVCC1 = VCC

22.8.3 Precautions on Board Design

For designing a board, to the maximum extent possible the digital circuits should be laid out separately from the analog circuits. Layouts involving the crossing of signal lines for digital circuits and signal lines for analog circuits, or placing them in proximity to each other, should be avoided. If the dissimilar signal lines are placed in close proximity to each other, the resulting induction can lead to a malfunction of the analog circuits or produce an adverse impact on A/D conversion values.
It should be noted that the analog input pins (AD1IN00 to AD1IN47), the analog power supply (AVCC1) should be isolated from the digital circuits by means of analog grounding (AVSS1). In addition, the analog ground (AVSS1) should be connected in one point to a stable digital ground (VSS) on the board. To prevent the influence of the noise generated by the 12-bit A/D converter, separate AVCC0 from AVCC1 as much as possible and design the board so that the common impedance is small.

22.8.4 Precautions on Noise Measures

A protection circuit to prevent the analog input pins (AD1INm) from damages, by such abnormal voltages as excessive surges, should be connected between AVCC1 and AVSS1, as illustrated in Figure 22.13. Also bypass capacitors connected to AVCC1, or a filter capacitor connected to an analog input pin (AD1INm), should be connected to the AVSS1. Since connecting a filter capacitor can cause an error by averaging the input currents to the analog input pin (AD1INm), care must be taken to choose appropriate circuit constants, as shown in Figure 22.13.

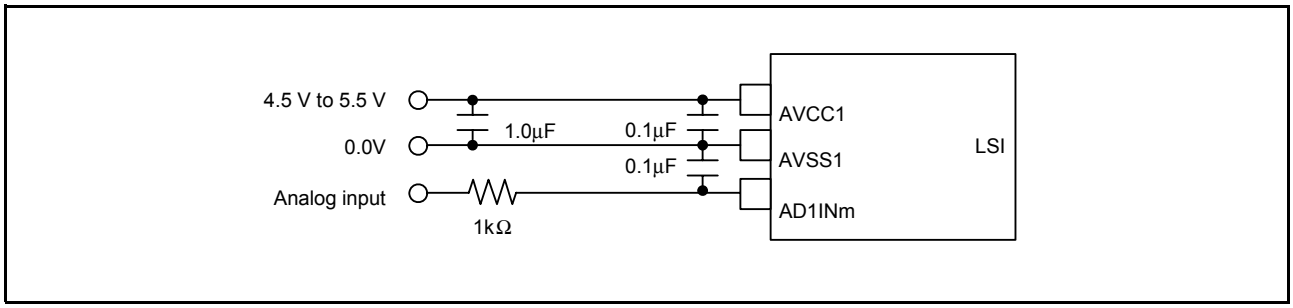


Figure 22.13 Example of Protection Circuit for Analog Input Pin

22.8.5 Precaution on Using Analog Input Pins as Digital Inputs/Output Pins

When using analog input pins as digital inputs/output pins, keep the following in mind. Concerning pin-function selection, refer to section 13, I/O Ports.

The analog input pins for the 10-bit A/D converter are divided into group A and group B.

(1) Precautions on differences in usage between the analog pin groups

If multiple analog input pins are to be used as a mixture of analog input pins and digital input/output pins, the analog pins in group B must be used as analog input pins and the analog pins in group A must be used as digital input/output pins. If analog pins in group B are in use as digital input/output pins, do not use analog pins in group A as analog inputs.

(2) Precaution on pins within the same group

When multiple analog input pins are used as a mixture of analog input pins and digital input/output pins, analog input and digital input/output pins must not be mixed within the same group (for example, using AD1IN00 as an analog input/output and AD1IN01 as a digital input).

Analog pins in group A: AD1IN00 to AD1IN07

Analog pins in group B: AD1IN16, AD1IN17, AD1IN24 to AD1IN31, AD1IN42 to AD1IN47

22.8.6 Precautions on Transitions to Low Power Mode

Stop A/D conversion before changing the mode to standby mode.

After setting the ADST bit in AD1CR, wait until the analog circuits in the A/D converter have stopped. To secure this waiting time, set the bits according to the following procedure.

(1) Set the TRGE bit in AD1CR to 0.
(2) Set the ADST bit in AD1CR to 0.
(3) Set the CKS bit in AD1CEX to 0 (f (PBA)).
(4) Check that A/D conversion stops.
(Stopping A/D conversion requires at least (f(PBA) × six cycles).)
(5) Change the mode to standby mode.

23. CRC Calculator

23.1 Introduction

The CRC (Cyclic Redundancy Check) calculator is used for error detection in data blocks. One of two generator polynomials (CRC-CCITT ($X^{16} + X^{12} + X^5 + 1$) or CRC-32 ($X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X^1 + 1$)) can be used for generating the CRC code.

The CRC code generated is set in the CRCDi register every time 1 byte of data is written to the CRCIN register after the initial value is written to the CRCDi register.

In this section, the variable i in the CRCDi register represents a value of 0 or 1.

Figure 23.1 shows the CRC Calculator Block Diagram.

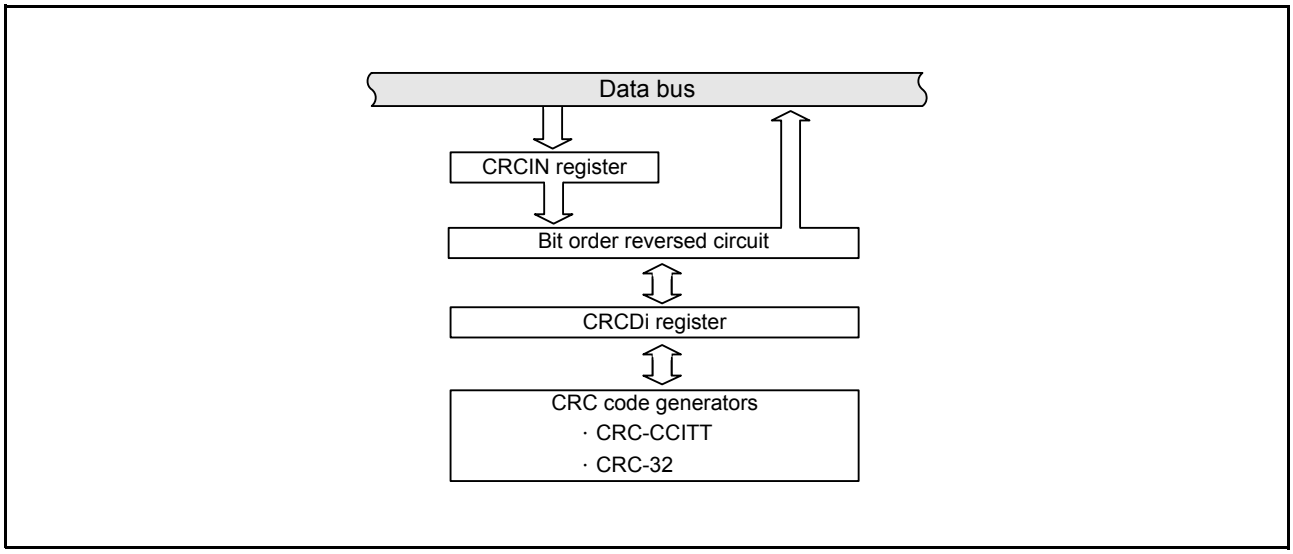


Figure 23.1 CRC Calculator Block Diagram

23.2 Registers

Table 23.1 lists the CRC Calculator Registers.

Table 23.1 CRC Calculator Registers

Register Name	Symbol	After Reset	Address	Access Size
CRC control register	CRCCR	H'00	H'FF46 5000	8
CRC data input register	CRCIN	Undefined	H'FF46 5001	8
CRC-CCITT data register	CRCD0	Undefined	H'FF46 5002	16
CRC-32 data register	CRCD1	Undefined	H'FF46 5004	32

23.2.1 CRC Control Register (CRCCR)

Address H'FF46 5000

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	CRCMD	COB
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b1	CRCMD	CRC Mode Set Bit	0: CRC-CCITT Mode 1: CRC-32 Mode	R/W
b0	COB	CRC Bit Order Reverse Control Bit	0: Bit order reversed circuit invalid 1: Bit order reversed circuit valid	R/W

This register controls read and write operations in registers CRCDi and CRCIN.

CRCMD Bit

When the CRCMD bit is set to 0, the calculation is performed in CRC-CCITT mode and then the result is stored in the CRCD0 register.

When the CRCMD bit is set to 1, the calculation is performed in CRC-32 mode and then the result is stored in the CRCD1 register.

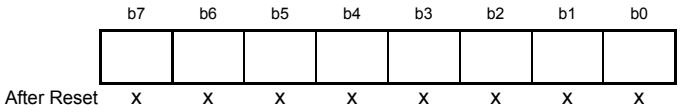
COB Bit

When the COB bit is set to 0, write data to registers CRCDi and CRCIN with the bit order reversed. When the CRC code is read from the CRCDi register, reverse the bit order by a program as the CRC code in reversed bit order can be read.

When the COB bit is set to 1, write data to registers CRCDi and CRCIN without reversing the bit order. When the CRC code is read from the CRCDi register, the CRC code in non-reversed bit order can be read.

23.2.2 CRC Data Input Register (CRCIN)

Address H'FF46 5001



Bit	Description	R/W
b7 to b0	The data to be used for the CRC calculation is input.	R/W

This register inputs the data to be used for the CRC calculation.
When the COB bit in the CRCCR register is 0, write data to this register with the bit order reversed.
When the COB bit in the CRCCR register is 1, write data to this register without reversing the bit order.

23.2.3 CRC-CCITT Data Register (CRCD0)

Address H'FF46 5002

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Bit	Description	R/W
b15 to b0	The calculation result in CRC-CCITT mode is stored.	R/W

This register stores the calculation result when the CRCMD bit in the CRCCR register is 0 (in CRC-CCITT mode).
When the COB bit is 0, write the initial value with the bit order reversed. Then if data is written to the CRCIN register, the CRC code in reversed bit order can be read from this register.
When the COB bit is 1, write data to this register without the bit order reversed. If data is written to the CRCIN register, the CRC code in non-reversed bit order can be read from this register.
When the CRCMD bit in the CRCCR register is 1, this register is not used and retains the last calculation result.

23.2.4 CRC-32 Data Register (CRCD1)

Address H'FF46 5004

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
After Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Bit	Description	R/W
b31 to b0	The calculation result in CRC-32 mode is stored.	R/W

This register stores the calculation result when the CRCMD bit in the CRCCR register is 1 (in CRC-32 mode).
When the COB bit is 0, write the initial value with the bit order reversed. Then if data is written to the CRCIN register, the CRC code in reversed bit order can be read from this register.
When the COB bit is 1, write data to the register without the bit order reversed. If data is written to the CRCIN register, the CRC code in non-reversed bit order can be read from this register.
When the CRCMD bit in the CRCCR register is 0, this register is not used and retains the last calculation result.

23.3 Operations

23.3.1 Calculation Example in CRC-CCITT Mode

The following is an example of the CRC calculations, in which the CRC code of H'80C4 in CRC-CCITT mode is generated.

23.3.1.1 When COB Bit is 0:

(1) Reverse the bit order of H'80C4 in byte units by a program.
H'80 → H'01, H'C4 →H'23
(2) Write B'00 to the CRCCR register.
(3) Write the initial value H'0000 to the CRCD0 register.
(4) Write the reversed-bit-order value of H'80 (H'01) to the CRCIN register. Then H'1189, the reversed-bit-order value of H'9188 which is the CRC code of H'80, is stored in the CRCD0 register.
(5) Write the reversed-bit-order value of H'C4 (H'23) to the CRCIN register. Then H'0A41, the reversed-bit-order value of H'8250 which is the CRC code of H'80C4, is stored in the CRCD0 register.

23.3.1.2 When COB Bit is 1:

(1) Write B'01 to the CRCCR register.
(2) Write the initial value H'0000 to the CRCD0 register.
(3) Write H'80 to the CRCIN register. Then the CRC code of H'80 (H'9188) is stored in the CRCD0 register.
(4) Write H'C4 to the CRCIN register. Then the CRC code of H'80C4 (H'8250) is stored in the CRCD0 register.

23.3.2 Calculation Example in CRC-32 Mode

The following is an example of the CRC calculations, in which the CRC code of H'80C4 in CRC-32 mode is generated.

23.3.2.1 When COB Bit is 0:

(1) Reverse the bit order of H'80C4 in byte units by a program.
H'80 → H'01, H'C4 → H'23
(2) Write B'10 to the CRCCR register.
(3) Write the initial value (H'0000 0000) to the CRCD1 register.
(4) Write the reversed-bit-order value of H'80 (H'01) to the CRCIN register. Then H'7707 3096, the reversed-bit-order value of H'690C E0EE which is the CRC code of H'80, is stored in the CRCD1 register.
(5) Write the reversed-bit-order value of H'C4 (H'23) to the CRCIN register. Then H'BB7C 4033, the reversed-bit-order value of H'CC02 3EDD which is the CRC code of H'80C4, is stored in the CRCD1 register.

23.3.2.2 When COB Bit is 1:

(1) Write B'11 to the CRCCR register.
(2) Write the initial value (H'0000 0000) to the CRCD1 register.
(3) Write H'80 to the CRCIN register. Then the CRC code of H'80 (H'690C E0EE) is stored in the CRCD1 register.
(4) Write H'C4 to the CRCIN register. Then the CRC code of H'80C4 (H'CC02 3EDD) is stored in the CRCD1 register.

24. ROM

24.1 Introduction

This LSI incorporate 512 Kbytes of flash memory (ROM) for the storage of instruction code. The flash memory has the following features.

Table 24.1 list the ROM Specifications.

Table 24.1 ROM Specifications

Item	Description
Flash-memory MAT	User MAT: 512 Kbytes User boot MAT: 32 Kbytes
Read	High-speed reading through ROM cache
Programing and erasing methods	The ROM can be programmed and erased by commands issued through the peripheral bus (P bus) to the ROM/EEPROM-dedicated sequencer (FCU)
Programing and erasing unit	Programming units: 256-byte. Erasing unit: User boot MAT: All User MAT: Block
On-board programming modes	Three types (boot mode, user program mode, user boot mode)
Protection mode	Software protection
Programming and erasing time and count	Refer to section 30, Electrical Characteristics.

- Two types of flash-memory MATs

The ROM has two types of memory areas (hereafter referred to as memory MATs) in the same address space. These two MATs can be switched by the start-up mode or bank switching through the control register.

For addresses H'0000 8000 to H'0007 FFFF, undefined data is read and programming and erasing are ignored when the user boot MAT is selected.

User MAT: 512 Kbytes

User boot MAT: 32 Kbytes

Figure 24.1 shows the Memory MAT Configuration in ROM.

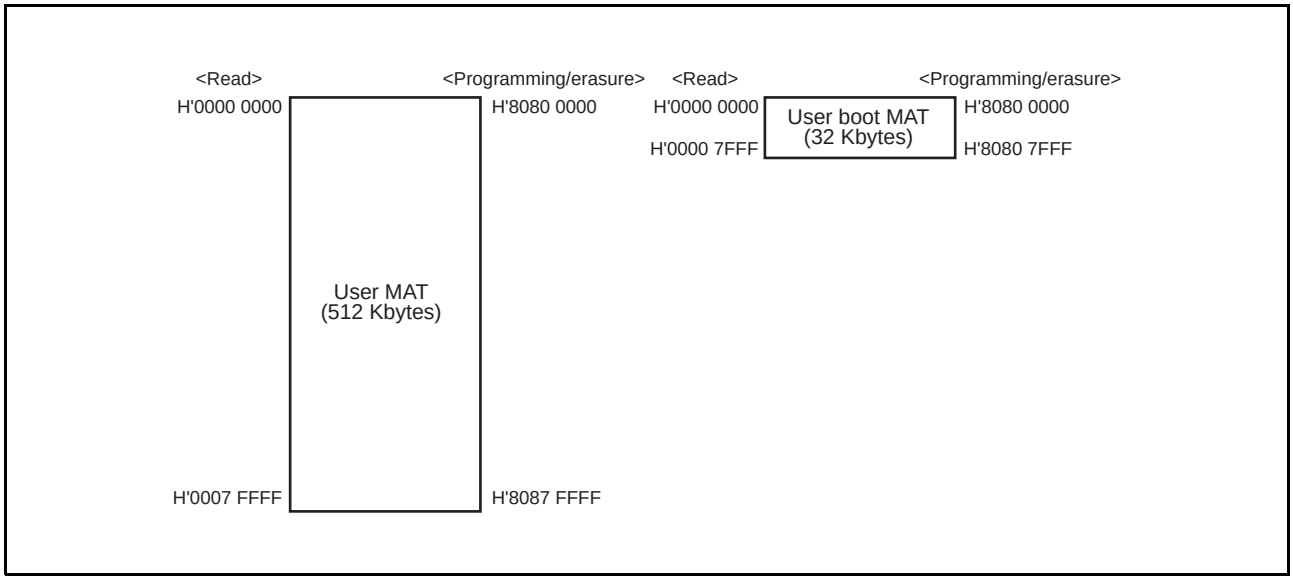


Figure 24.1 Memory MAT Configuration in ROM

- High-speed reading through ROM cache

Both the user MAT and user boot MAT can be read at high speed through the ROM cache.

- Programming and erasing methods

The ROM can be programmed and erased by commands issued through the peripheral bus (P bus) to the ROM/EEPROM-dedicated sequencer (FCU).

While the flash control unit (FCU) is programming or erasing the ROM, the CPU can execute a program located outside the ROM. While the FCU is programming or erasing the EEPROM, the CPU can execute a program in the ROM. When the FCU suspends programming or erasure, the CPU can execute a program in the ROM, and then the FCU can resume programming or erasure. While the FCU suspends erasure, areas other than the erasure-suspended area can be programmed.

- Programming/erasing unit

The user MAT and user boot MAT are programmed in 256-byte units. The entire area of the user boot MAT is always erased at one time. The user MAT can be erased in block units.
Figure 24.2 shows the block configuration of the user MAT. The user MAT is divided into eight 8-Kbyte blocks and seven 64-Kbyte blocks in this LSI.

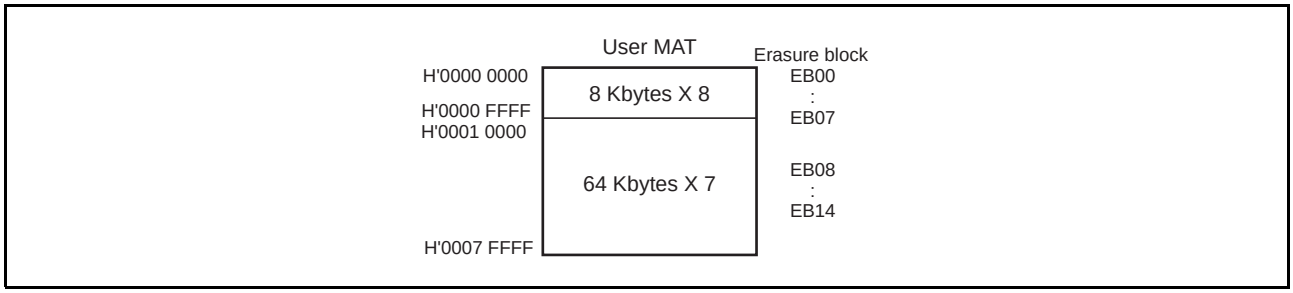


Figure 24.2 Block Configuration of User MAT

- Three types of on-board programming modes

[Boot mode]
The user MAT and user boot MAT can be programmed using the SCI. The bit rate for SCI communications between the host and this LSI can be automatically adjusted.
[User program mode]
The user MAT can be programmed with a desired interface. A transition from single-chip mode is enabled.
[User boot mode]
The user MAT can be programmed with a desired interface. To make a transition to this mode, a reset is needed.

- Protection modes

This LSI supports a mode to protect memory against programming or erasure: software protection by the FENTRY0 bit in the FENTRYR register or lock bit settings. The FENTRY0 bit in FENTRYR register enables or disables ROM programming or erasure by the FCU. A lock bit is included in each erasure block of the user MAT to protect memory against programming or erasure.
The LSI also provides a function to suspend programming or erasure when abnormal operation is detected during programming or erasure.

- Programming and erasing time and count

Refer to section 30, Electrical Characteristics.

Figure 24.3 shows the Block Diagram of ROM.

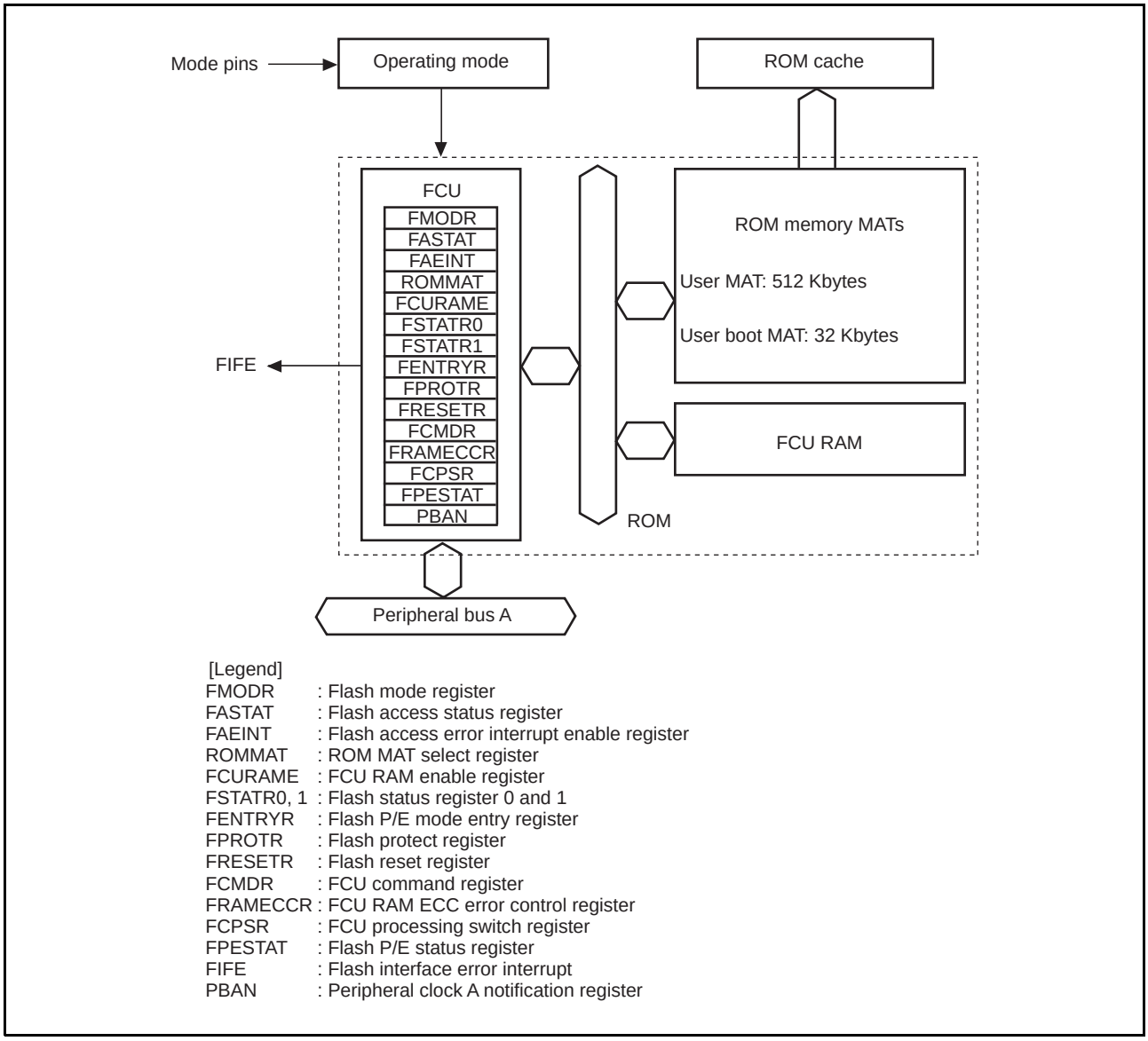


Figure 24.3 Block Diagram of ROM

24.2 Input/Output Pins

Table 24.2 shows the Pin Configuration used for the ROM. The combination of MD0 to MD1 pin levels and the ASEMD pin level determines the ROM programming mode (refer to section 24.4, Overview of ROM-Related Modes). In boot mode, the ROM can be programmed or erased by the host connected via the RXD1 and TXD1 pins (refer to section 24.5, Boot Mode).

Table 24.2 Pin Configuration

Pin Name	I/O	Function
RESET#	Input	This LSI enters a hardware reset state when this signal goes low.
MD0 to MD1, ASEMD	Input	These pins specify the operating mode.
RXD1	Input	Receives data through SCI1 (communications with host)
TXD1	Output	Transmits data through SCI1 (communications with host)

24.3 Registers

Table 24.3 list the ROM Registers. Some of these registers have EEPROM-related bits. The ROM-related registers are initialized by a reset.

Table 24.3 ROM Registers

Register Name	Symbol	After Reset	Address	Access Size
Flash mode register	FMODR	H'00	H'FFFF A802	8
Flash access status register	FASTAT	H'00	H'FFFF A810	8
Flash access error interrupt enable register	FAEINT	H'9F	H'FFFF A811	8
ROM MAT select register	ROMMAT	H'0000 H'0001	H'FFFF A820	8*2, 16
FCU RAM enable register	FCURAME	H'0000	H'FFFF A854	8*2, 16
Flash status register 0	FSTATR0	H'80 *1	H'FFFF A900	8, 16
Flash status register 1	FSTATR1	H'00 *1	H'FFFF A901	8, 16
Flash P/E mode entry register	FENTRYR	H'0000 *1	H'FFFF A902	8*2, 16
Flash protect register	FPROTR	H'0000 *1	H'FFFF A904	8*2, 16
Flash reset register	FRESETR	H'0000	H'FFFF A906	8*2, 16
FCU command register	FCMDR	H'FFFF *1	H'FFFF A90A	8, 16
FCU RAM ECC error control register	FRAMECCR	H'02 *1	H'FFFF A90C	8
FCU processing switch register	FCPSR	H'0000 *1	H'FFFF A918	8, 16
Flash P/E status register	FPESTAT	H'0000 *1	H'FFFF A91C	8, 16
Peripheral clock A notification register	PBAN	H'00 *1	H'FFFF A938	8, 16

Notes: 1. This register can be initialized by a reset, or by setting the FRESET bit in the FRESETR register to 1.
2. The 8-bit unit of access is only for reading this register.

24.3.1 Flash Mode Register (FMODR)

Address H'FFFF A802

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	FRDMD	—	—	—	—
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R
b4	FRDMD	FCU Read Mode Select Bit	Selects the read mode to read the ROM or EEPROM using FCU. This bit specifies the check method for the lock bits in the ROM (refer to section 24.6.1, FCU Command List, and section 24.6.3 12, Reading Lock Bit), whereas this bit must be set to make the blank check command available for use in the EEPROM (refer to section 25, EEPROM). 0: Selects the memory area read mode. The mode to read the lock bits in the ROM in ROM lock bit read mode. 1: Selects the register read mode. The mode to read the lock bits in the ROM using the lock bit read 2 command.	R/W
b3 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

The FMODR register specifies the FCU operation mode.
The FMODR register is initialized by a reset.

24.3.2 Flash Access Status Register (FASTAT)

Address H'FFFF A810

	b7	b6	b5	b4	b3	b2	b1	b0
	ROMAE	—	—	CMDLK	EEPAE	EEPIFE	EEPRPE	EEPWPE
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	ROMAE	ROM Access Error Bit	Indicates whether or not a ROM access error has been generated. If this bit becomes 1, the ILGLERR bit in the FSTATR0 register is set to 1 and the FCU enters a command-locked state. 0: No ROM access error has occurred. 1: A ROM access error has occurred. [Conditions to become 1] • An access command is issued to ROM program/erase addresses H'8080 0000 to H'8087 FFFF while the FENTRY0 bit in the FENTRYR register is 1 in ROM P/E normal mode. • An access command is issued to ROM program/erase addresses H'8080 0000 to H'8087 FFFF while the FENTRY0 bit in the FENTRYR register is 0. • A read access command is issued to ROM read addresses H'0000 0000 to H'0007 FFFF while the FENTRYR register value is not H'0000. • A block erase, program, or lock bit program command is issued while the user boot MAT is selected. • An access command is issued to an address other than ROM program/erase addresses H'8080 0000 to H'8080 7FFF while the user boot MAT is selected. [Condition to become 0] • 0 is written to this bit after reading 1 from the ROMAE bit.	R/(W) *
b6, b5	—	Reserved	These bits are read as 0. The write value should be 0.	R
b4	CMDLK	FCU Command Lock Bit	Indicates whether the FCU is in command-locked state (refer to section 24.8.2, Error Protection). 0: The FCU is not in a command-locked state 1: The FCU is in a command-locked state [Condition to become 1] • The FCU detects an error and enters command-locked state. [Condition to become 0] • The FCU completes the status-clear command processing while the FASTAT register is H'10.	R
b3	EEPAE	EEPROM Access Error Bit	Refer to section 25, EEPROM.	R/(W) *
b2	EEPIFE	EEPROM Instruction Fetch Error Bit	Refer to section 25, EEPROM.	R/(W) *
b1	EEPRPE	EEPROM Read Protect Error Bit	Refer to section 25, EEPROM.	R/(W) *
b0	EEPWPE	EEPROM Program/Erase Protect Error Bit	Refer to section 25, EEPROM.	R/(W) *

Note: * Writing 0 after reading 1 is only allowed in order to clear the flag.

The FASTAT register indicates the access error status for the ROM and EEPROM. If any bit in the FASTAT register is set to 1, the FCU enters command-locked state (refer to section 24.8.2, Error Protection). To cancel a command-locked state, set the FASTAT register to H'10, and then issue a status-clear command to the FCU.
The FASTAT register is initialized by a reset.

24.3.3 Flash Access Error Interrupt Enable Register (FAEINT)

Address H'FFFF A811

	b7	b6	b5	b4	b3	b2	b1	b0
	ROMAEIE	—	—	CMDLKIE	EEPAEIE	EEPIFEIE	EEPRPEIE	EEPWPEIE
After Reset	1	0	0	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b7	ROMAEIE	ROM Access Error Interrupt Enable Bit	Enables or disables an FIFE interrupt request when a ROM access error occurs and the ROMAE bit in the FASTAT register becomes 1. 0: Does not generate an FIFE interrupt request when ROMAE = 1. 1: Generates an FIFE interrupt request when ROMAE = 1.	R/W
b6, b5	—	Reserved	These bits are read as 0. The write value should be 0.	R
b4	CMDLKIE	FCU Command Lock Interrupt Enable Bit	Enables or disables an FIFE interrupt request when FCU command-locked state is entered and the CMDLK bit in the FASTAT register becomes 1. 0: Does not generate an FIFE interrupt request when CMDLK = 1 1: Generates an FIFE interrupt request when CMDLK = 1	R/W
b3	EEPAEIE	EEPROM Access Error Interrupt Enable Bit	Refer to section 25, EEPROM.	R/W
b2	EEPIFEIE	EEPROM Instruction Fetch Error Interrupt Enable Bit	Refer to section 25, EEPROM.	R/W
b1	EEPRPEIE	EEPROM Read Protect Error Interrupt Enable Bit	Refer to section 25, EEPROM.	R/W
b0	EEPWPEIE	EEPROM Program/Erase Protect Error Interrupt Enable Bit	Refer to section 25, EEPROM.	R/W

Note: After writing to the FAEINT register, execute a FAEINT register read instruction, and five or more NOP instructions.

The FAEINT register enables or disables output of flash interface error (FIFE) interrupts.

The FAEINT register is initialized by a reset.

24.3.4 ROM Mat Select Register (ROMMAT)

Address H'FFFF A820

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	KEY[7:0]								—	—	—	—	—	—	—	ROMSEL
After Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0/1

Bit	Symbol	Bit Name	Description	R/W
b15 to b8	KEY[7:0]	Key Code Bits	These bits enable or disable ROMSEL bit modification. The data written to these bits are not stored.	R/(W) *
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R
b0	ROMSEL	ROM MAT Select Bit	Selects a memory MAT in the ROM. The initial value is 1 when the LSI is started in user boot mode; otherwise, the initial value is 0. Writing to this bit is enabled only when this register is accessed in word size and H'3B is written to the KEY bits. 0: Selects the user MAT 1: Selects the user boot MAT	R/W

Notes: After writing to the ROMMAT register, execute a ROMMAT register read instruction, and five or more NOP instructions.
* Write data is not retained.

The ROMMAT register switches memory MATs in the ROM.
The ROMMAT is initialized by a reset.
Writing to the upper byte is enabled only when this register is accessed in word size and specific value is written. The data written to the upper byte is not retained in the register.

24.3.5 FCU RAM Enable Register (FCURAME)

Address H'FFFF A854

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	KEY[7:0]								—	—	—	—	—	—	—	FCRME
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 b8	KEY[7:0]	Key Code Bits	These bits enable or disable FCRME bit modification. The data written to these bits are not stored.	R/(W) *
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R
b0	FCRME	FCU RAM Enable Bit	Enables or disables access to the FCU RAM. Writing to this bit is enabled only when this register is accessed in word size and H'C4 is written to the KEY bits. Before writing to the FCU RAM, clear FENTRYR to H'0000 to stop the FCU. 0: Disables access to FCU RAM 1: Enables access to FCU RAM	R/W

Note: * Write data is not retained.

The FCURAME register enables or disables access to the FCU RAM area.
The FCURAME register is initialized by a reset.
Writing to the upper byte is enabled only when this register is accessed in word size and specific value is written. The data written to the upper byte is not retained in the register.

24.3.6 Flash Status Register 0 (FSTATR0)

Address H'FFFF A900

	b7	b6	b5	b4	b3	b2	b1	b0
	FRDY	ILGLERR	ERSERR	PRGERR	SUSRDY	—	ERSSPD	PRGSPD
After Reset	1	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	FRDY	Flash Ready Bit	Indicates the processing state in the FCU. 0: Programming or erasure processing, programming or erasure suspension processing, lock bit read 2 command processing, or EEPROM blank check is in progress (refer to section 25, EEPROM). 1: None of the above is in progress.	R
b6	ILGLERR	Illegal Command Error Bit	Indicates that the FCU has detected an illegal command or illegal ROM or EEPROM access. When this bit is 1, the FCU is in command-locked state (refer to section 24.8.2, Error Protection). 0: The FCU has not detected any illegal command or illegal ROM/EEPROM access 1: The FCU has detected an illegal command or illegal ROM/EEPROM access [Conditions to become 1] • The FCU has detected an illegal command. • The FCU has detected an illegal ROM/EEPROM access (the ROMAE, EEPAE, EEPIFE, EEPRPE, or EEPWPE bit in FASTAT register is 1). • The FENTRYR setting is illegal. [Condition to become 0] • The FCU completes the status-clear command processing while FASTAT register is H'10.	R
b5	ERSERR	Erasure Error Bit	Indicates the result of ROM or EEPROM erasure by the FCU. When this bit is 1, the FCU is in command-locked state (refer to section 24.8.2, Error Protection). 0: Erasure processing has been completed successfully 1: An error has occurred during erasure [Conditions to become 1] • An error has occurred during erasure. • A block erase command has been issued for the area protected by a lock bit. [Condition to become 0] • The FCU completes the status-clear command processing.	R
b4	PRGERR	Programming Error Bit	Indicates the result of ROM or EEPROM programming by the FCU. When this bit is 1, the FCU is in command-locked state (refer to section 24.8.2, Error Protection). 0: Programming has been completed successfully 1: An error has occurred during programming [Conditions to become 1] • An error has occurred during programming. • A programming command has been issued for the area protected by a lock bit. [Condition to become 0] • The FCU completes the status-clear command processing.	R
b3	SUSRDY	Suspend Ready Bit	Indicates whether the FCU is ready to accept a P/E suspend command. 0: The FCU cannot accept a P/E suspend command 1: The FCU can accept a P/E suspend command [Condition to become 1] • After initiating programming/erasure, the FCU has entered a state where it is ready to accept a P/E suspend command. [Conditions to become 0] • The FCU has accepted a P/E suspend command. • The FCU has entered a command-locked state during programming or erasure.	R
b2	—	Reserved	This bit is read as 0. The write value should be 0.	R

Bit	Symbol	Bit Name	Description	R/W
b1	ERSSPD	Erasure-Suspended Status Bit	Indicates that the FCU has entered an erasure suspension process or an erasure-suspended status (refer to section 24.6.4, Suspending Operation). 0: The FCU is in a status other than the below- mentioned. 1: The FCU is in an erasure suspension process or an erasure-suspended status. [Condition to become 1] • The FCU has initiated an erasure suspend command. [Condition to become 0] • The FCU has accepted a resume command.	R
b0	PRGSPD	Programming-Suspended Status Bit	Indicates that the FCU has entered a write suspension process or a write suspend status (refer to section 24.6.4, Suspending Operation). 0: The FCU is in a status other than the below- mentioned. 1: The FCU is in a write suspension process or a write-suspended status. [Condition to become 1] • The FCU has initiated a write suspend command. [Condition to become 0] • The FCU has accepted a resume command.	R

The FSTATR0 register indicates the FCU status.

The FRTATR0 register is initialized by a reset, or by setting the FRESET bit in the FRESETR register to 1.

24.3.7 Flash Status Register 1 (FSTATR1)

Address H'FFFF A901

	b7	b6	b5	b4	b3	b2	b1	b0
	FCUERR	—	—	FLOCKST	—	—	FRDTCT	FRCRCT
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	FCUERR	FCU Error Bit	Indicates an error has occurred during the CPU processing in the FCU. 0: No error has occurred during the CPU processing in the FCU 1: An error has occurred during the CPU processing in the FCU [Condition to become 0] The FRESET bit in the FRESETR register is set to 1. When FCUERR bit is 1, set the FRESET bit to 1 to initialize the FCU, and then copy the FCU firmware again from the FCU firmware area to the FCU RAM area.	R
b6, b5	—	Reserved	These bits are read as 0. The write value should be 0.	R
b4	FLOCKST	Lock Bit Status Bit	Reflects the lock bit data read through lock bit read 2 command execution. When the FRDY bit becomes 1 after the lock bit read 2 command is issued, valid data is stored in this bit. This bit value is retained until the next lock bit read 2 command is completed. 0: Protected state 1: Non-protected state	R
b3, b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b1	FRDTCT	FCU RAM ECC 2-Bit Error Detection Monitoring Bit	Indicates that a 2-bit error has been detected when the FCU is reading RAM. 0: No 2-bit error has been detected. 1: A 2-bit error has been detected. When FRDTCT bit is 1, set the FRESET bit to 1 to initialize the FCU, and then copy the FCU firmware again from the FCU firmware area to the FCU RAM area.	R
b0	FRCRCT	FCU RAM ECC 1-Bit Error Correction Monitoring Bit	Indicates that a 1-bit error has been corrected when the FCU is reading RAM. 0: No 1-bit error has been corrected. 1: A 1-bit error has been corrected. When FRCRCT bit is 1, set the FRESET bit to 1 to initialize the FCU, and then copy the FCU firmware again from the FCU firmware area to the FCU RAM area.	R

The FSTATR1 register indicates the FCU status.

The FSTATR1 register is initialized by a reset, or by setting the FRESET bit in the FRESETR register to 1.

24.3.8 Flash P/E Mode Entry Register (FENTRYR)

Address H'FFFF A902

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	FEKEY [7:0]								FENTRYD	—	—	—	—	—	—	FENTRY0
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 b8	FEKEY [7:0]	Key Code Bits	These bits enable or disable rewriting of the FENTRYD and FENTRY0 bits. Write data to these bits are not retained.	R/(W) *
b7	FENTRYD	EEPROM P/E Mode Entry	Refer to section 25, EEPROM.	R/W
b6 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R
b0	FENTRY0	ROM P/E Mode Entry 0 Bit	Specifies the P/E mode for the 512-Kbyte ROM (read addresses: H'0000 0000 to H'0007 FFFF; program/erase addresses: H'8080 0000 to H'8087 FFFF). 0: The 512-Kbyte ROM is in read mode 1: The 512-Kbyte ROM is in P/E mode Programming is enabled when the following conditions are all satisfied: • The FRDY bit in the FSTATR0 register is 1. • H'AA is written to FEKEY in word access. [Condition to become 1] • 1 is written to the FENTRY0 register while the write enabling conditions are satisfied and FENTRYR register is H'0000. [Conditions to become 0] • The FRDY bit in the FSTATR0 register becomes 1. • This register is written to in byte access. • A value other than H'AA is written to the FEKEY bits in word access. • 0 is written to the FENTRY0 register while the write enabling conditions are satisfied. • FENTRYR register is written to while FENTRYR register is not H'0000 and the write enabling conditions are satisfied.	R/W

Notes: After writing to the FENTRYR register, execute a FENTRYR register read instruction, and five or more NOP instructions.
* Write data is not retained.

The FENTRYR register specifies the P/E mode for the ROM or EEPROM. To specify the P/E mode for the ROM or EEPROM so that the FCU can accept commands, set either of the FENTRYD and FENTRY0 bits to 1.
The FENTRYR register can be initialized by a reset, or by setting the FRESET bit in the FRESETR register to 1.

Note: • This register can be written to only when a specified value is written to the upper byte in word access; the register is initialized when a value not allowed for the register is written to the upper byte. The data written to the upper byte is not stored in the register.

24.3.9 Flash Protect Register (FPROTR)

Address H'FFFF A904

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	FPKEY [7:0]								—	—	—	—	—	—	—	FPROCTCN
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 b8	FPKEY [7:0]	Key Code Bits	These bits enable or disable FPROTCN bit modification. The data written to these bits are not stored.	R/(W) *
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R
b0	FPROCTCN	Lock Bit Protect Cancel Bit	Enables or disables protection through the lock bits against programming and erasure. 0: Enables protection through the lock bits 1: Disables protection through the lock bits [Condition to become 1] • H'55 is written to the FPKEY bits and 1 is written to FPROTCN bit in word access while the FENTRYR register value is not H'0000. [Conditions to become 0] • This register is written to in byte access. • A value other than H'55 is written to FPKEY bits in word access. • H'55 is written to FPKEY bits and 0 is written to FPROTCN bit in word access. • The FENTRYR register value is H'0000.	R/W

Note: * Write data is not retained.

The FPROTR register enables or disables the protection function through the lock bits against programming and erasure. The FPROTR register is initialized by a reset, or by setting the FRESET bit in the FRESETR register to 1.

Note: • This register can be written to only when a specified value is written to the upper byte in word access; the register is initialized when a value not allowed for the register is written to the upper byte. The data written to the upper byte is not stored in the register.

24.3.10 Flash Reset Register (FRESETR)

Address H'FFFF A906

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	FRKEY [7:0]								—	—	—	—	—	—	—	FRESET
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 b8	FRKEY [7:0]	Key Code Bits	These bits enable or disable FRESET bit modification. The data written to these bits are not stored.	R/(W) *
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R
b0	FRESET	Flash Reset Bit	Setting this bit to 1 forcibly terminates programming/erasure of ROM or EEPROM and initializes the FCU. A high voltage is applied to the ROM/EEPROM memory units during programming and erasure. To ensure sufficient time for the voltage applied to the memory unit to drop, keep the value of the FRESET bit at 1 for a period of t_{RESW2} (refer to section 30, Electrical Characteristics) when the FCU is initialized. Do not read from the ROM/EEPROM units while the value of the FRESET bit is kept at 1. The FCU commands are unavailable for use while the FRESET bit is set to 1, since this initializes the FENTRYR register. This bit can be written only when H'CC is written to the FRKEY bits in word access. 0: Issue no reset to the FCU. 1: Issues a reset to the FCU.	R/W

Note: * Write data is not retained.

The FRESETR register is used for the initialization of the FCU.

The FRESETR register is initialized by a reset.

Writing to the upper byte is enabled only when this register is accessed in word size and specific value is written. The data written to the upper byte is not retained in the register.

24.3.11 FCU Command Register (FCMDR)

Address H'FFFF A90A

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	CMDR [7:0]								PCMDR [7:0]							
After Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b15 to b8	CMDR [7:0]	Command Register Bits	These bits store the latest command accepted by the FCU.	R
b7 to b0	PCMDR [7:0]	Precommand Register Bits	These bits store the previous command accepted by the FCU.	R

The FCMDR register stores the commands that the FCU has accepted.
The FCMDR register is initialized by a reset, or by setting the FRESET bit in the FRESETR register to 1.

Table 24.4 shows the states of the FCMDR register after acceptance of the various commands. For details on the blank check, refer to section 25.6, User Mode, User Program Mode, and User Boot Mode.

Table 24.4 FCMDR Status after a Command is Accepted

Command	CMDR	PCMDR
Normal mode transition	H'FF	Previous command
Status read mode transition	H'70	Previous command
Lock bit read mode transition (lock bit read 1)	H'71	Previous command
Program	H'E8	Previous command
Block erase	H'D0	H'20
P/E suspend	H'B0	Previous command
P/E resume	H'D0	Previous command
Status register clear	H'50	Previous command
Lock bit read 2 blank check	H'D0	H'71
Lock bit program	H'D0	H'77
Peripheral function clock A notification register	H'E9	Previous command

24.3.12 FCU RAM ECC Error Control Register (FRAMECCR)

Address H'FFFF A90C

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	FRDCLE	FRCCLE
After Reset	0	0	0	0	0	0	1	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b1	FRDCLE	FCU Command Lock Enabling Bit upon FCU RAM 2-Bit Error Detection	Enables or disables an FCU command-lock request upon detection of an ECC 2-bit error when the FCU is reading RAM. If an ECC 2-bit error is detected while this bit is set to 1, the CMDLK bit in FASTAT register is set to 1. 0: Issues no FCU command-lock request upon detection of a 2-bit error. 1: Issues an FCU command-lock request upon detection of a 2-bit error.	R/W
b0	FRCCLE	FCU Command Lock Enabling Bit upon FCU RAM 1-Bit Error Correction	Enables or disables an FCU command-lock request upon correction of an ECC 1-bit error when the FCU is reading RAM. If an ECC 1-bit error is detected while this bit is set to 1, the CMDLK bit in FASTAT register is set to 1. Indicates that a 1-bit error has been corrected when the FCU is reading RAM. 0: No 1-bit error has been corrected. 1: A 1-bit error has been corrected.	R/W

The FRAMECCR register enables or disables an FCU command lock request upon correction of an ECC 1-bit error or detection of and ECC 2-bit error when the FCU is reading the RAM. FRAMECCR register enables or disables an FCU command lock request, but it does not control the FRDTCT and FRCRCT bits in the Flash status register 1 (FSTATR1).

The FRAMECCR register is initialized by a reset.

24.3.13 FCU Processing Switch Register (FCPSR)

Address H'FFFF A918

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ESUSPMD
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 b1	—	Reserved	These bits are read as 0. The write value should be 0.	R
b0	ESUSPMD	Erasure-Suspended Mode Bit	Erasure-suspended mode This LSI does not use this mode. The write value should always be 0.	R/W

The FCPSR register selects a function to make the FCU suspend erasure.

The FCPSR register is initialized by a reset, or by setting the FRESET bit in the FRESETR register to 1.

24.3.14 Flash P/E Status Register (FPESTAT)

Address H'FFFF A91C

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	PEERRST [7:0]							
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 b8	—	Reserved	These bits are read as 0. The write value should be 0.	R
b7 to b0	PEERRST [7:0]	P/E Error Status Bits	Indicates the source of an error that occurs during programming/erasure. This bit value is only valid if the PRGERR or ERSERR bit value in the FSTATR0 register is 1; otherwise the bit retains the value to indicate the source of an error that previously occurred. H'01: A write attempt made to an area protected by the lock bits H'02: A write error caused by other source than the above H'11: An erase attempt made to an area protected by the lock bits H'12: An erase error caused by other source than the above Other than above: Reserved	R

The FPESTAT register indicates the result of programming/erasure of the ROM/EEPROM.
The FPESTAT register is initialized by a reset, or by setting the FRESET bit in the FRESETR register to 1.

24.3.15 Peripheral Clock A Notification Register (PBAN)

Address H'FFFF A938

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	PBAN [7:0]							
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 b8	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b7 to b0	PBAN [7:0]	Peripheral clock A notification bit	Notifies the ROM/EEPROM-dedicated sequencer (FCU) of the frequency of peripheral bus clock A.	R/W

PBAN is initialized by a reset or by setting the FRESET bit of FRESETR to 1.

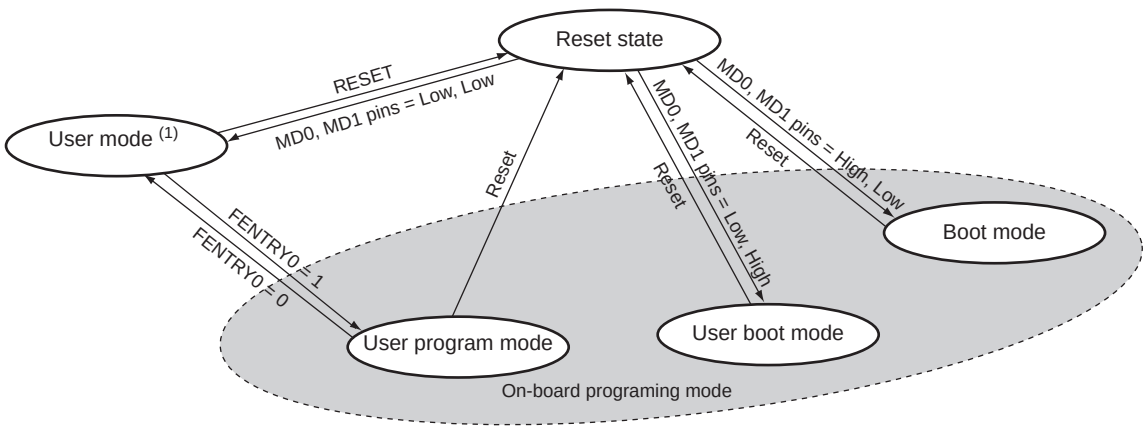
PBAN Bits

Before programming or erasing the ROM, set the PBAN bits to match the frequency of peripheral bus clock A. When the peripheral bus clock A notification command is issued, the FCU is notified of the specified value. Do not change the frequency while programming or erasing the ROM/EEPROM.
The value to be specified is the result of converting the operating frequency in MHz units to binary.
Example: When the operating frequency of peripheral clock A is 50 MHz:
Convert 50 to binary, so set the PBAN bits to H'32 (B'0011 0010).

Note: Set the PBAN bits to the frequency of peripheral clock A in MHz. If the specified frequency is different from the actual frequency, data in the ROM/EEPROM may be corrupted.

24.4 Overview of ROM-Related Modes

Figure 24.4 shows the ROM-Related Mode Transition in this LSI.



Note 1. MCU single-chip mode of this LSI is a user mode.

Figure 24.4 ROM-Related Mode Transition

- The ROM can be read but cannot be programmed or erased in user mode.
- The ROM can be read, programmed, and erased on the board in user program mode, user boot mode, and boot mode.

Table 24.5 compares programing- and erasure-related items for the boot mode, user program mode, and user boot mode.

Table 24.5 Comparison of Programming Modes

Item	Boot Mode	User Program Mode	User Boot Mode
Mode transition	MD0, MD1 pins = High, Low	MD0, MD1 pins = Low, Low Write 1 to the FENTRY0 bit	MD0, MD1 pins = Low, High
Programming/erasure environment	On-board programming		
Programming/erasure enabled MAT	User MAT and user boot MAT	User MAT	User MAT
Programming/erasure control	Host	FCU	FCU
Entire area erasure	Available (automatic)	Available	Available
Block erasure	Available *1	Available	Available
Programming data transfer	From host via SCI	From any device via RAM	From any device via RAM
Reset-start MAT	Embedded program stored MAT	User MAT	User boot MAT *2
Transition to MCU operating mode	MD0, MD1 pins = Low, Low and reset	Write 0 to the FENTRY0 bit	MD0, MD1 pins = Low, Low and reset

Notes: 1. The entire area is erased when the LSI is started. After that, a specified block can be erased.
2. After the LSI is started in the embedded program stored MAT and the boot program provided by Renesas Corp. is executed, execution starts from the location indicated by the reset vector of the user boot MAT.

- The user boot MAT can be programmed or erased only in boot mode.
- In boot mode, the user MAT, user boot MAT, and EEPROM data MAT are all erased immediately after the LSI is started. The user MAT, user boot MAT, and data MAT can then be programmed from the host via the SCI. The ROM can also be read after this entire area erasure.
- In user boot mode, a boot operation with a desired interface can be implemented through mode pin settings different from those in user program mode.
- In boot mode or user boot mode, the boot program uses the on-chip RAM. Therefore, once the RAM is disabled via the RAM Enable Control Register (RAMEN) and a reset is issued, the data prior to the reset is no longer retained in the RAM after booting is initiated in boot mode or user boot mode (refer to section 27, RAM Control).

24.5 Boot Mode

24.5.1 System Configuration

To program or erase the user MAT and user boot MAT in boot mode, send control commands and programming data from the host. The on-chip SCI of this LSI is used in asynchronous mode for communications between the host and this LSI. The tool for sending control commands and programming data must be prepared in the host. When this LSI is started in boot mode, the program in the embedded program stored MAT is executed. This program automatically adjusts the SCI bit rate and performs communications between the host and this LSI by means of the control command method.

Figure 24.5 shows the System Configuration in Boot Mode. The NMI and INT0 to INT13 interrupts are ignored in this mode, but these pins must be fixed to non-active state. Note that the AUD cannot be used in this mode.

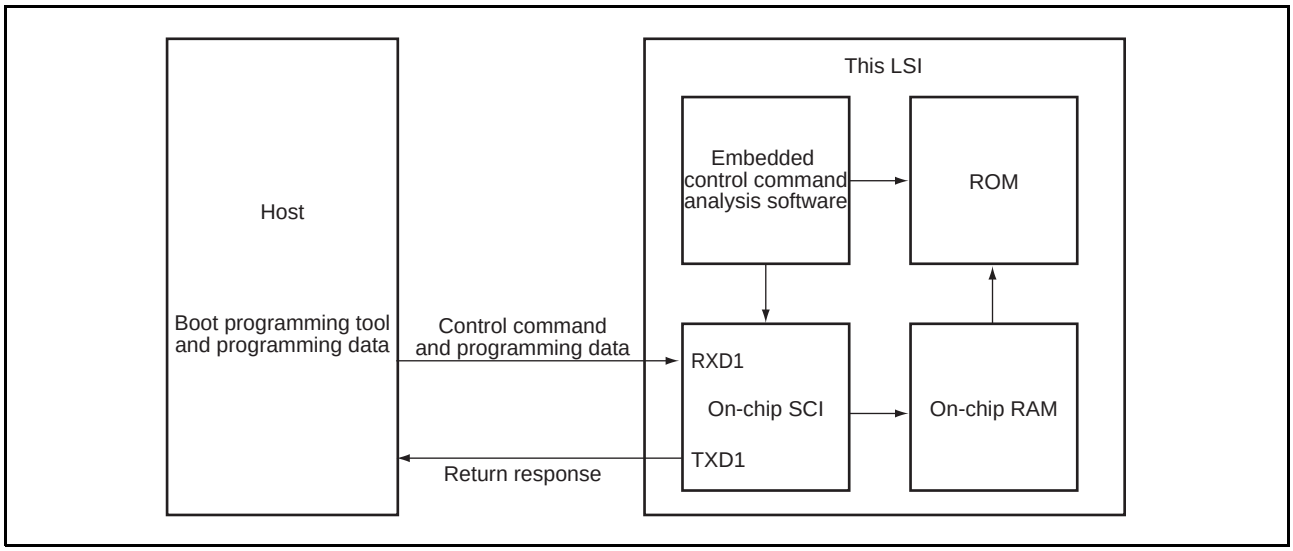


Figure 24.5 System Configuration in Boot Mode

24.5.2 State Transition in Boot Mode

Figure 24.6 shows the State Transition in Boot Mode.

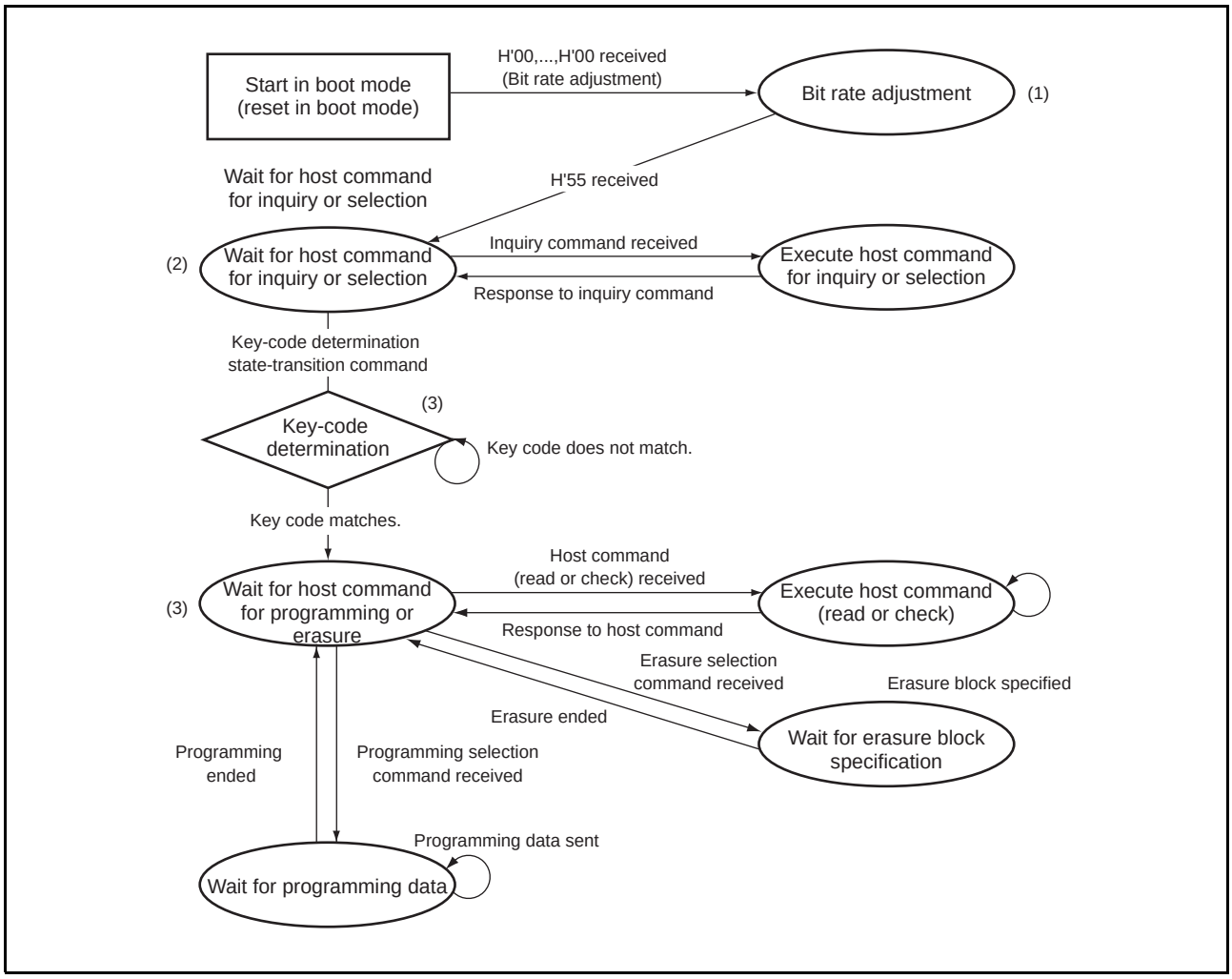


Figure 24.6 State Transition in Boot Mode

(1) Bit Rate Adjustment

After this LSI is started in boot mode, it automatically adjusts the bit rate for communications between the host and SCI. After automatic adjustment of the bit rate, the LSI sends H'00 to the host. After the LSI has successfully received H'55 sent from the host, the LSI waits for a host command for inquiry or selection. For details on bit rate adjustment, refer to section 24.5.3, Automatic Adjustment of Bit Rate.

(2) Waiting for Host Command for Inquiry or Selection

In this state, the host inquires regarding MAT information (such as the size, configuration, and start address) and the supported functions, and selects the device, clock mode, and bit rate. Upon reception of a programming/erasure state transition command sent from the host, this LSI erases the entire area of each of the user MAT, user boot MAT, and EEPROM data MAT and waits for a host command for programming or erasure. For details of inquiry/selection host commands, refer to section 24.5.4, Inquiry/Selection Host Command Wait State.

(3) Key-code determination

This state is for judging whether the key code sent from the host matches that with which the chip is programmed. If they do match, the chip is placed in the state of waiting for programming or erasure. Furthermore, if the key code in the chip is in the initial state (has not been programmed), any key code will be authenticated.

(4) Waiting for Host Command for Programming or Erasure

In this state, this LSI performs programming or erasure according to the command sent from the host. The LSI enters programming data wait state, erasure block specification wait state, or command (read or check) processing state depending on the received command.

Upon reception of a programming selection command, the LSI waits for programming data. After the programming selection command, send the programming start address and programming data from the host.

Specifying H'FFFF FFFF as the programming start address terminates programming processing and the LSI makes a transition from the programming data wait state to programming/erasure command wait state.

Upon reception of an erasure selection command, the LSI waits for erasure block specification. After the erasure selection command, send the erasure block number from the host. Specifying H'FF as the erasure block number terminates erasure processing and the LSI makes a transition from the erasure block specification wait state to programming/erasure command wait state. As the entire area of each of the user MAT, user boot MAT, and EEPROM data MAT is erased before the LSI enters programming/erasure command wait state after it is started in boot mode, erasure processing is not needed except for the case when the data programmed in boot mode should be erased without resetting the LSI.

In addition to programming and erasing commands, many other host commands are provided for use in programming/erasure command wait state; these include commands for checksum, blank check (erasure check), memory read, and status inquiry. For details on these host commands, refer to section 24.5.5, Programming/Erasing Host Command Wait State.

24.5.3 Automatic Adjustment of Bit Rate

When this LSI is started in boot mode, it measures the low-level (H'00) period of the data that is continuously sent from the host in asynchronous SCI communications. During this measurement, set the SCI transmit/receive format to 8-bit data, 1 stop bit, and no parity, and set the bit rate to 9,600 bps or 19,200 bps. This LSI calculates the bit rate of the host SCI by means of the measured low-level period, and then sends H'00 to the host after completing the bit rate adjustment. When the host has received H'00 successfully, it must send H'55 to this LSI. If the host has failed to receive H'00, restart this LSI in boot mode to calculate and adjust the bit rate again. When this LSI has received H'55, it returns H'E6 to the host, or when it has failed to receive H'55, it returns H'FF.

Figure 24.7 shows the SCI Transmit/Receive Format for Automatic Adjustment of Bit Rate.
Figure 24.8 shows the Communication Sequence between Host and this LSI.

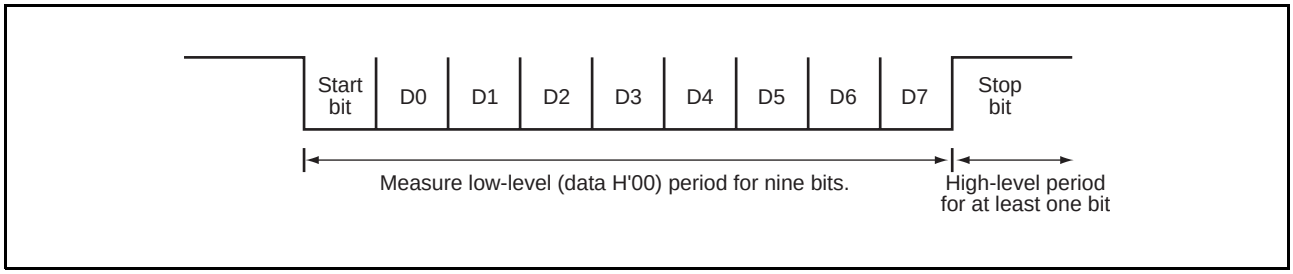


Figure 24.7 SCI Transmit/Receive Format for Automatic Adjustment of Bit Rate

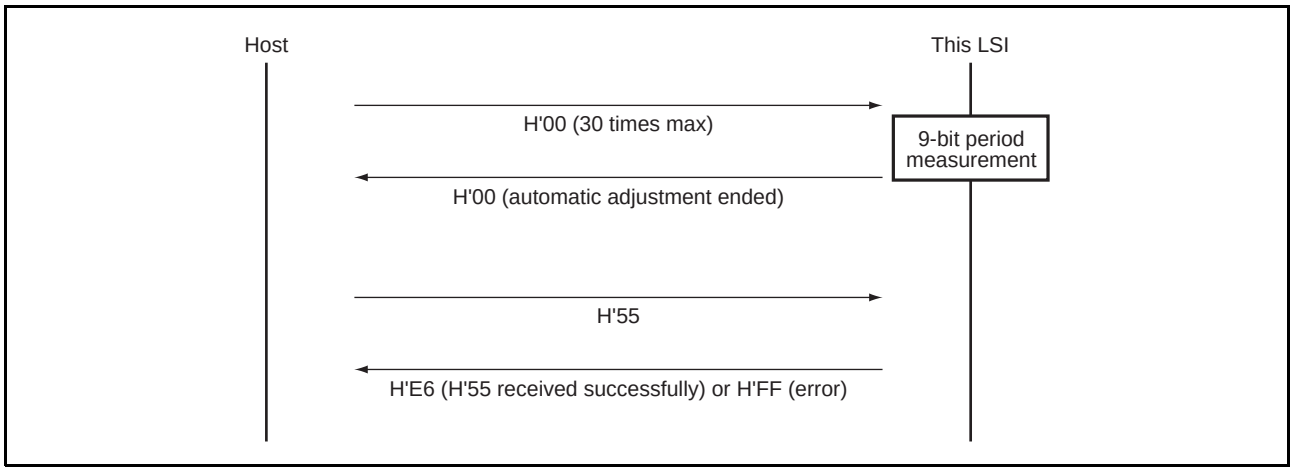


Figure 24.8 Communication Sequence between Host and this LSI

The bit rate may not be adjusted correctly depending on the bit rate of the host SCI or the peripheral clock frequency of this LSI. Satisfy the SCI communications condition as shown in Table 24.6.

Table 24.6 Condition for Automatic Adjustment of Bit Rate

Host SCI Bit Rate	Peripheral Clock Frequency of this LSI	
	SH72A2 Group	SH72A0 Group
9,600 bps	10 to 12.5 MHz	8 to 10 MHz
19,200 bps	10 to 12.5 MHz	8 to 10 MHz

24.5.4 Inquiry/Selection Host Command Wait State

Table 24.7 shows the host command available in inquiry/selection host command wait state. The boot program status inquiry command can also be used in programming/erasure host command wait state. The other commands can only be used in inquiry/selection host command wait state.

Table 24.7 Inquiry/Selection Host Commands

Host Command Name	Function
Supported device inquiry	Inquires regarding the device codes and the product codes for the embedded programs
Device selection	Selects a device code
Clock mode inquiry	Inquires regarding the clock mode
Clock mode selection	Selects a clock mode
Multiplication ratio inquiry	Inquires regarding the number of clock types, the number of multiplication/division ratios, and the multiplication /division ratios
Operating frequency inquiry	Inquires regarding the number of clock types and the maximum and minimum operating frequencies
User boot MAT information inquiry	Inquires regarding the number of user boot MATs and the start and end addresses
User MAT information inquiry	Inquires regarding the number of user MATs and the start and end addresses
Erasure block information inquiry	Inquires regarding the number of blocks and the start and end addresses
Programming size inquiry	Inquires regarding the size of programming data
Simultaneous two-MAT programming information inquiry	Inquires regarding the availability of simultaneous two-MAT programming function
New bit rate selection	Modifies the bit rate of SCI communications between the host and this LSI
Transition to the key-code determination state	Places the chip in the key-code determination state
Key-code checking command	Sends the key code
Boot program status inquiry	Inquires regarding the state of this LSI

If the host has sent an undefined command, this LSI returns a response indicating a command error in the format shown below. The command field holds the first byte of the undefined command sent from the host.

Error response	H'80	Command

In inquiry/selection host command wait state, send selection commands from the host in the order of device selection, clock mode selection, and new bit rate selection to set up this LSI according to the responses to inquiry commands. Note that the supported device inquiry and clock mode inquiry commands are the only inquiry commands that can be sent before the clock mode selection command; other inquiry commands must not be issued before the clock mode selection command. If commands are issued in an incorrect order, this LSI returns a response indicating a command error. Figure 24.9 shows the Example of Procedure to Use Inquiry/Selection Host Commands.

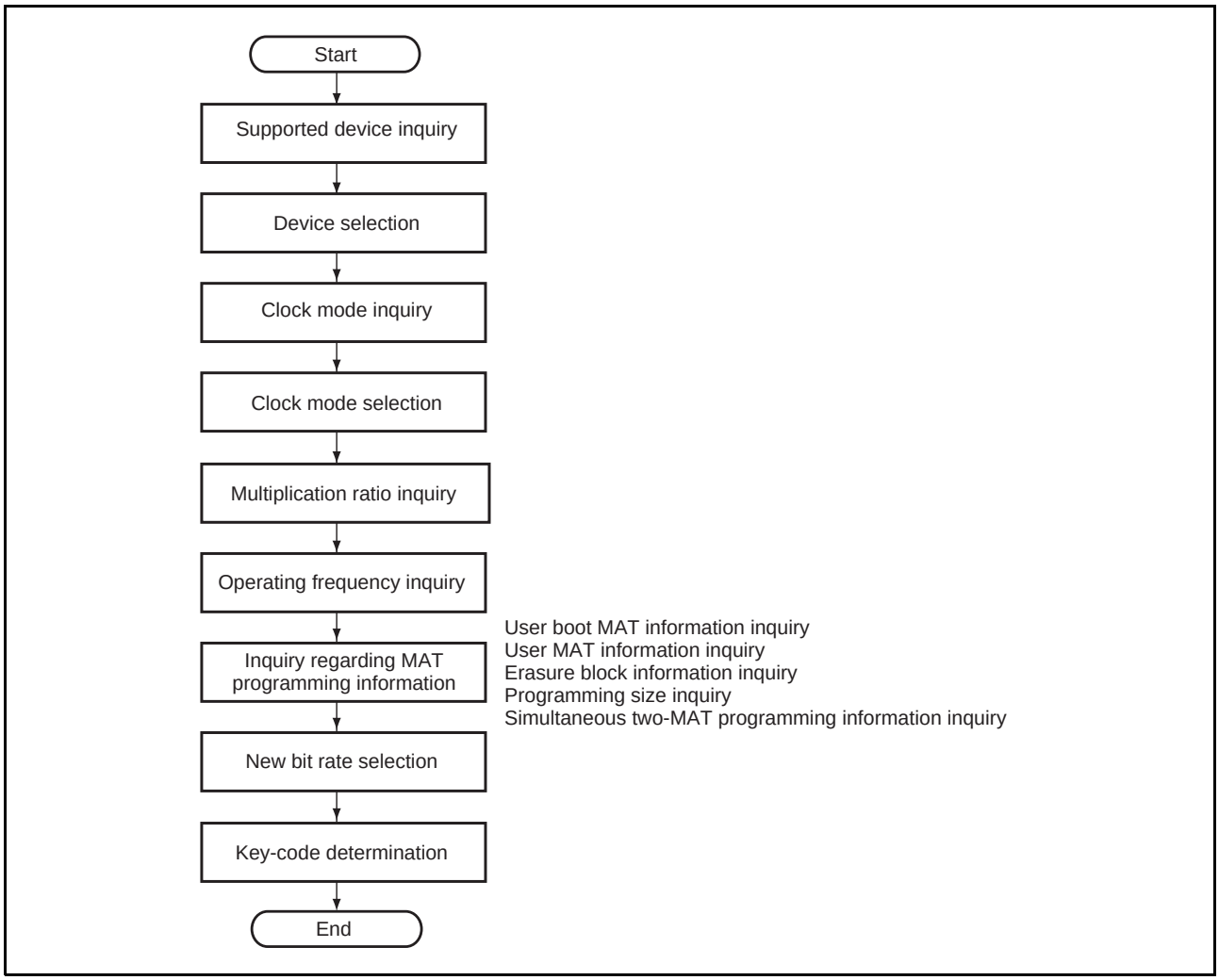


Figure 24.9 Example of Procedure to Use Inquiry/Selection Host Commands

Each host command is described in detail below. The “command” in the description indicates a command sent from the host to this LSI and the “response” indicates a response sent from this LSI to the host. The “checksum” is byte-size data calculated so that the sum of all bytes to be sent by this LSI becomes H'00.

(1) Supported Device Inquiry

In response to a supported device inquiry command sent from the host, this LSI returns the information concerning the devices supported by the embedded program for boot mode. If the supported device inquiry command comes after the host has selected a device, this LSI only returns the information concerning the selected device.

Command	H'20			
Response	H'30	Size	Device count	
	Character count	Device code		Product code
	Character count	Device code		Product code
	:	:		:
	Character count	Device code		Product code
	SUM			

[Legend]
Size (1 byte): Total number of bytes in the device count, character count, device code, and product code fields
Device count (1 byte): Number of device types supported by the embedded program for boot mode
Character count (1 byte): Number of characters included in the device code and product code fields
Device code (4 bytes): ASCII code for the product name of the chip
Product code (n bytes): ASCII code for the supported device name
SUM (1 byte): Checksum (in response)

(2) Device Selection

In response to a device selection command sent from the command, this LSI checks if the selected device is supported. When the selected device is supported, this LSI specifies this device as the device for use and returns a response (H'06). If the selected device is not supported or the sent command is illegal, this LSI returns an error response (H'90). Even when H'01 has been returned as the number of supported devices in response to a supported device inquiry command, issue a device selection command to specify the device code that has been returned as the result of the inquiry.

Command	H'10	Size	Device code	SUM
Response	H'06			
Error response	H'90	Error		

[Legend]
Size (1 byte): Number of characters in the device code field (fixed at 2)
Device code (4 bytes): ASCII code for the product name of the chip
(one of the device codes returned in response to the supported device inquiry command)
SUM (1 byte): Checksum
Error (1 byte): Error code
H'11: Checksum error (illegal command)
H'21: Incorrect device code error

(3) Clock Mode Inquiry

In response to a clock mode inquiry command sent from the host, this LSI returns the supported clock modes. If the clock mode inquiry command comes after the host has selected a clock mode, this LSI only returns the information concerning the selected clock mode.

Command	H'21			
Response	H'31	Size		
	Mode	Mode	:	Mode
	SUM			

[Legend]
Size (1 byte): Total number of bytes in the mode count and mode fields
Mode (1 byte): Supported clock mode (for example, H'01 indicates clock mode 1)
SUM (1 byte): Checksum

(4) Clock Mode Selection

In response to a clock mode selection command sent from the host, this LSI checks if the selected clock mode is supported. When the selected mode is supported, this LSI specifies this clock mode for use and returns a response (H'06). If the selected mode is not supported or the sent command is illegal, this LSI returns an error response (H'91). Be sure to issue a clock mode selection command only after issuing a device selection command. Even when H'00 or H'01 has been returned as the number of supported clock modes in response to a clock mode inquiry command, issue a clock mode selection command to specify the clock mode that has been returned as the result of the inquiry.

Command	H'11	Size	Mode	SUM
Response	H'06			
Error response	H'91	Error		

[Legend]
Size (1 byte): Number of characters in the mode field (fixed at 1)
Mode (1 byte): Clock mode (one of the clock modes returned in response to the clock mode inquiry command)
SUM (1 byte): Checksum
Error (1 byte): Error code
H'11: Checksum error (illegal command)
H'22: Incorrect clock mode error

(5) Multiplication Ratio Inquiry

In response to a multiplication ratio inquiry command sent from the host, this LSI returns the clock types, the number of multiplication/division ratios, and the multiplication division ratios supported.

Command	H'22				
Response	H'32	Size	Clock type count		
	Multiplication ratio count	Multiplication ratio	Multiplication ratio	:	Multiplication ratio
	Multiplication ratio count	Multiplication ratio	Multiplication ratio	:	Multiplication ratio
	:	:	:	:	:
	Multiplication ratio count	Multiplication ratio	Multiplication ratio	:	Multiplication ratio
	SUM				

[Legend]
Size (1 byte): Total number of bytes in the clock type count, multiplication ratio count, and multiplication ratio fields
Clock type count (1 byte): Number of clock types
(for example, H'02 indicates two clock types; that is, an internal clock and a peripheral clock)
Multiplication ratio count (1 byte): Number of supported multiplication/division ratios (for example, H'03 indicates that three multiplication ratios are supported for the internal clock (x4, x6, and x8))
Multiplication ratio (1 byte): A positive value indicates a multiplication ratio (for example, H'04 = 4 = multiplication by 4)
A negative value indicates a division ratio (for example, H'FE = -2 = division by 2)
SUM (1 byte): Checksum

(6) Operating Clock Frequency Inquiry

In response to an operating clock frequency inquiry command sent from the host, this LSI returns the minimum and maximum frequencies for each clock.

Command	H'23			
Response	H'33	Size	Clock type count	
	Minimum frequency		Maximum frequency	
	Minimum frequency		Maximum frequency	
	:		:	
	Minimum frequency		Maximum frequency	
	Minimum frequency		Maximum frequency	
	SUM			

[Legend]
Size (1 byte): Total number of bytes in the clock type count, minimum frequency, and maximum frequency fields
Clock type count (1 byte): Number of clock types
(for example, H'02 indicates two clock types; that is, an internal clock and a peripheral clock)
Minimum frequency (2 bytes): Minimum value of the operating frequency (for example, H'07D0 indicates 20.00 MHz).
This value should be calculated by multiplying the frequency value (MHz) to two decimal places by 100.
Maximum frequency (2 bytes): Maximum value of the operating frequency represented in the same format as the minimum frequency
SUM (1 byte): Checksum

(7) User Boot MAT Information Inquiry

In response to a user boot MAT information inquiry command sent from the host, this LSI returns the number of user boot MATs and their addresses.

Command	H'24		
Response	H'34	Size	MAT count
	MAT start address		
	MAT end address		
	MAT start address		
	MAT end address		
	:		
	MAT start address		
	MAT end address		
	SUM		

[Legend]
Size (1 byte): Total number of bytes in the MAT count, MAT start address, and MAT end address fields
MAT count (1 byte): Number of user boot MATs (consecutive areas are counted as one MAT)
MAT start address (4 bytes): Start address of a user boot MAT
MAT end address (4 bytes): End address of a user boot MAT
SUM (1 byte): Checksum

(8) User MAT Information Inquiry

In response to a user MAT information inquiry command sent from the host, this LSI returns the number of user MATs and their addresses.

Command	H'25		
Response	H'35	Size	MAT count
	MAT start address		
	MAT end address		
	MAT start address		
	MAT end address		
	:		
	MAT start address		
	MAT end address		
	SUM		

[Legend]
Size (1 byte): Total number of bytes in the MAT count, MAT start address, and MAT end address fields
MAT count (1 byte): Number of user MATs (consecutive areas are counted as one MAT)
MAT start address (4 bytes): Start address of a user MAT
MAT end address (4 bytes): End address of a user MAT
SUM (1 byte): Checksum

(9) Erasure Block Information Inquiry

In response to an erasure block information inquiry command sent from the host, this LSI returns the number of erasure blocks in the user MAT and their addresses.

Command	H'26		
Response	H'36	Size	Block count
	MAT start address		
	MAT end address		
	MAT start address		
	MAT end address		
	:		
	MAT start address		
	MAT end address		
	SUM		

[Legend]
Size (2 bytes): Total number of bytes in the block count, block start address, and block end address fields
Block count (1 byte): Number of erasure blocks in the user MAT
Block start address (4 bytes): Start address of an erasure block
Block end address (4 bytes): End address of an erasure block
SUM (1 byte): Checksum

(10) Programming Size Inquiry

In response to a programming size inquiry command sent from the host, this LSI returns the programming size.

Command	H'27			
Response	H'37	Size	Programming size	SUM

[Legend]
Size (1 byte): Number of characters included in the programming size field (fixed at two)
Programming size (2 bytes): Programming unit (bytes)
SUM (1 byte): Checksum

(11) Simultaneous Two-MAT Programming Information Inquiry

In response to an inquiry from the host concerning simultaneous two-MAT programming information, this LSI returns whether two MATs can be programmed simultaneously and the start address of the available MATs. This LSI does not support simultaneous programming of two MATs.

Command	H'28		
Response	H'38	Size	Mode
	First MAT start address		
	Second MAT start address		
	SUM		

[Legend]
Size (1 byte): Total number of bytes in the mode, first MAT start address, and second MAT start address fields (fixed at 5 in this LSI)
Mode (1 byte): Programming mode (H'01 in this LSI)
H'01: One-MAT programming,
H'10: Simultaneous two-MAT programming
First MAT start address (1 byte): Start address of the first MAT (H'0000 0000 in this LSI)
Second MAT start address (1 byte): Start address of the second MAT (no data in this LSI)
No data is sent in one-MAT programming mode.
SUM (1 byte): Checksum

(12) New Bit Rate Selection

In response to a new bit rate selection command sent from the host, this LSI checks if the on-chip SCI can be set to the

selected new bit rate. When the SCI can be set to the new bit rate, this LSI returns a response (H'06) and sets the SCI to the new bit rate. If the SCI cannot be set to the new bit rate or the sent command is illegal, this LSI returns an error response (H'BF). Upon reception of response H'06, the host waits for a one-bit period in the previous bit rate with which the new bit rate selection command has been sent, and then sets the host bit rate to the new one. After that, the host sends confirmation data (H'06) in the new bit rate, and this LSI returns a response (H'06) to the confirmation data.

Be sure to issue a new bit rate selection command only after a clock mode selection command.
Figure 24.10 shows the New Bit Rate Selection Sequence.

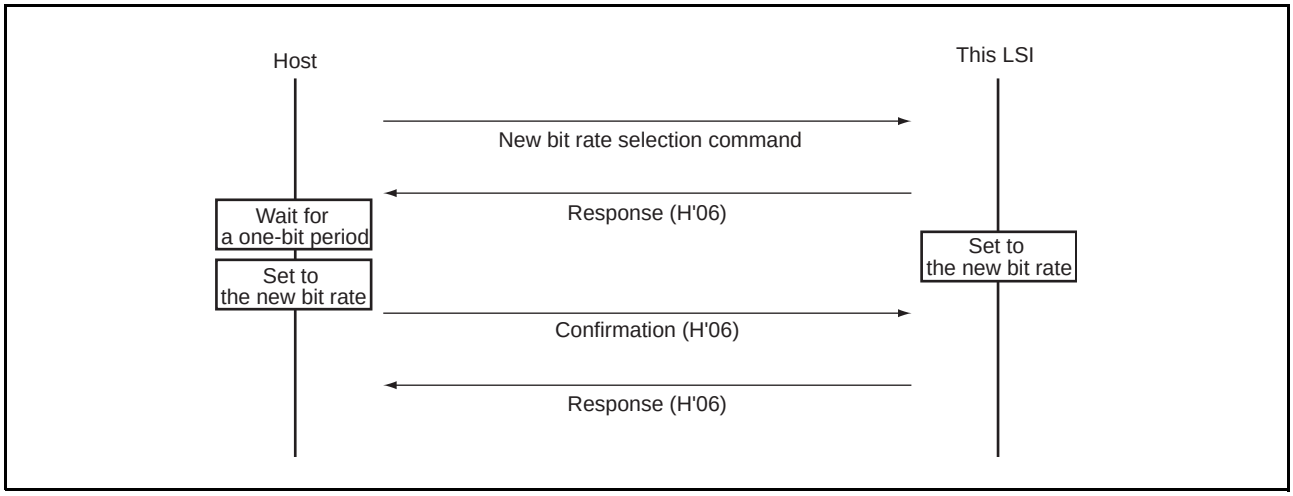


Figure 24.10 New Bit Rate Selection Sequence

Command	H'3F	Size	Bit rate	Input frequency
	Clock type count	Multiplication ratio 1	Multiplication ratio 2	
	SUM			
Response	H'06			
Error Response	H'BF	Error		
Confirmation	H'06			
Response	H'06			

[Legend]
Size (1 byte): Total number of bytes in the bit rate, input frequency, clock type count, and multiplication ratio fields
Bit rate (2 bytes): New bit rate (for example, H'00C0 indicates 19200 bps)
1/100 of the new bit rate value should be specified.
Input frequency (2 bytes): Clock frequency input to this LSI (for example, H'07D0 indicates 20.00 MHz)
This value should be calculated by multiplying the input frequency value to two decimal places by 100.
Clock type count (1 byte): Number of clock types
(for example, H'02 indicates two clock types; that is an internal clock and a peripheral clock)
Multiplication ratio 1 (1 byte): Multiplication/division ratio of the input frequency to obtain the internal clock
A positive value indicates a multiplication ratio (for example, H'04 = 4 = multiplication by 4)
A negative value indicates a division ratio (for example, HFE = -2 = division by 2)
Multiplication 2 (1 byte): Multiplication/division ratio of the input frequency to obtain the peripheral clock
This value is represented in the same format as multiplication ratio 1
SUM (1 byte): Checksum
Error (1 byte): Error code (refer to Table 24.8)

Table 24.8 Error Code

Code	Description
H'11	• Checksum error
H'24	• Bit rate selection error A bit rate selection error occurs when the bit rate selected through a new bit rate selection command cannot be set for the SCI of this LSI within an error of 4%. The bit rate error can be obtained by the following equation from the bit rate (B) selected through a new bit rate selection command, the input frequency (f_{EX}), multiplication ratio 2 (MPϕ), the SCiBR register setting (N) in SCI, and the CKS bits value (N) in SCiCR register. Bit rate error (%) $= \{\frac{f_{EX} \times M_{P\Phi} \times 10^6}{(N+1) \times B \times 32 \times 2^{n-1}} - 1\} \times 100$
H'25	• Input frequency error An input frequency error occurs when the input frequency specified through a new bit rate selection command is outside the range from the minimum to maximum input frequencies for the clock mode selected through a clock mode selection command.
H'26	• Multiplication ratio error A multiplication ratio error occurs when the multiplication ratio specified through a new bit rate selection command does not match the clock mode selected through a clock mode selection command. To check the selectable multiplication ratios, issue a multiplication ratio inquiry command.
H'27	• Operating frequency error An operating frequency error occurs when this LSI cannot operate at the operating frequencies selected through a new bit rate selection command. This LSI calculates the operating frequencies from the input frequency and multiplication ratios specified through a new bit rate selection command and checks if each calculated frequency is within the range from the minimum to maximum frequencies for the respective clock. To check the minimum and maximum operating frequencies for each clock, issue an operating clock frequency inquiry command.

(13) Transition to the key-code determination state

The chip enters the key-code determination state in response to the command for that transition.

Command	H'40
Response	H'16

(14) Key-code checking command

In response to the key-code checking command, the chip compares the key code sent from the host to that in its memory mat and, if they match, sends the corresponding response (H'26) and enters the programming and erasure state. It sends the error response (H'E0) if they do not match.

Furthermore, if the key code in the chip is in its initial state (has not been programmed), the chip authenticates any key code and sends the key-code matched response (H'26). Then the chip enters the programming and erasure state.

Command	H'60	Size
	Key code	
	SUM	
Response	H'26	
Error response	H'E0	Error

[Legend]
Size (1 byte): Number of characters in the key code (fixed at 16)
Key code (16 bytes): Key code for authentication of boot mode
SUM (1 byte): Checksum (the setting is such that the sum of bytes from the command to the SUM byte is H'00)
Error (1 byte): Error code
H'11: Error in the sum check
H'61: Non-matching key code error

(15) Boot Program Status Inquiry

In response to a boot program status inquiry command sent from the host, this LSI returns its current status. The boot program status inquiry command can be issued in both inquiry/selection host command wait state and programming/erasure host command wait state.

Command	H'4F			
Response	H'5F	Size	Status	Error

[Legend]
Size (1 byte): Total number of bytes in the status and error fields (fixed at two)
Status (1 byte): Current status in this LSI (refer to Table 24.9 Status Code)
Error (1 byte): Error status in this LSI (refer to Table 24.10 Error Code)

Table 24.9 Status Code

Code	Description
H'11	Waiting for device selection
H'12	Waiting for clock mode selection
H'13	Waiting for bit rate selection
H'1F	Waiting for transition to the key-code determination state (bit rate has been selected)
H'3F	Waiting for a programming/erasure host command
H'4F	Waiting for reception of programming data
H'5F	Waiting for erasure block selection

Table 24.10 Error Code

Code	Description
H'00	No error
H'11	Checksum error
H'21	Incorrect device code error
H'22	Incorrect clock mode error
H'24	Bit rate selection error
H'25	Input frequency error
H'26	Multiplication ratio error
H'27	Operating frequency error
H'29	Block number error
H'2A	Address error
H'2B	Data size error
H'51	Erasure error
H'52	Incomplete erasure error
H'53	Programming error
H'54	Selection error
H'61	Non-matching key code error
H'80	Command error
H'FF	Bit rate adjustment verification error

24.5.5 Programming/Erasing Host Command Wait State

Table 24.11 shows the host commands available in programming/erasure host command wait state.
Figure 24.11 shows the Procedure for ROM Programming in Boot Mode.
Figure 24.12 shows the Procedure for ROM Erasure in Boot Mode.

Table 24.11 Programming/Erasure Host Commands

Host Command Name	Function
User boot MAT programming selection	Selects the program for user boot MAT programming
User MAT programming selection	Selects the program for user MAT programming
Simultaneous two-user MAT programming selection	Selects the program for simultaneous two-user MAT programming
256-byte programming	Programs 256 bytes of data
Erasure selection	Selects the erasure program
Block erasure	Erases block data
Memory read	Reads data from memory
User boot MAT checksum	Performs checksum verification for the user boot MAT
User MAT checksum	Performs checksum verification for the user MAT
User boot MAT blank check	Checks whether the user boot MAT is blank
User MAT blank check	Checks whether the user MAT is blank
Boot program status inquiry	Inquires regarding the state of this LSI

If the host has sent an undefined command, this LSI returns a response indicating a command error. For the format of this response, refer to section 24.5.4, Inquiry/Selection Host Command Wait State.
To program the ROM, issue a programming selection command (user boot MAT programming selection or user MAT programming selection command) and then a 256-byte programming command from the host. Upon reception of a programming selection command, this LSI enters programming data wait state (refer to section 24.5.5, Programming/Erasing Host Command Wait State). In response to a 256-byte programming command sent from the host in this state, this LSI starts programming the ROM. When the host sends a 256-byte programming command specifying H'FFFF FFFF as the programming start address, this LSI detects it as the end of programming and enters programming/erasure host command wait state.
To erase the ROM, issue an erasure selection command and then a block erasure command from the host. Upon reception of an erasure selection command, this LSI enters erasure block selection wait state (refer to section 24.5.2, State Transition in Boot Mode). In response to a block erasure command sent from the host in this state, this LSI erases the specified block in the ROM. When the host sends a block erasure command specifying H'FF as the block number, this LSI detects it as the end of erasure and enters programming/erasure host command wait state.

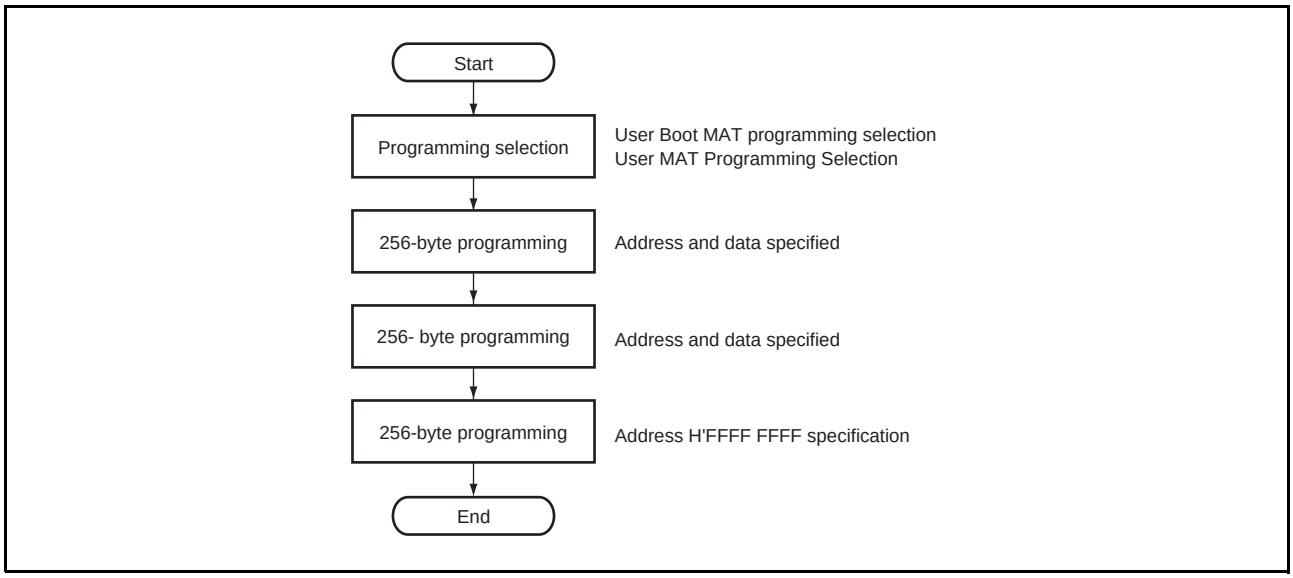


Figure 24.11 Procedure for ROM Programming in Boot Mode

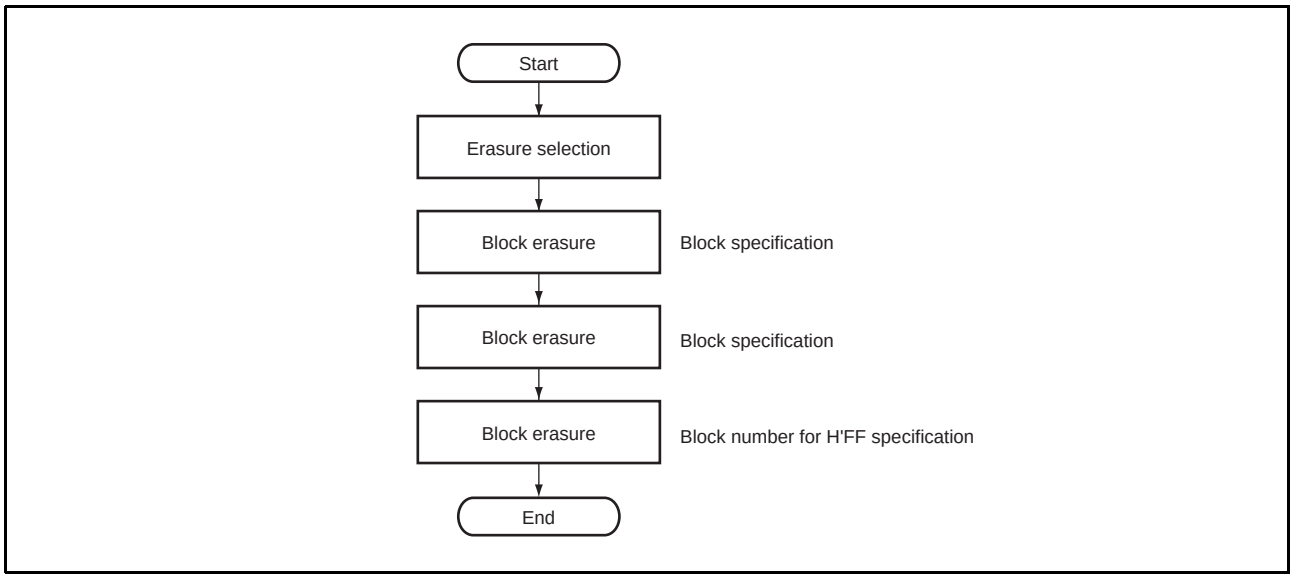


Figure 24.12 Procedure for ROM Erasure in Boot Mode

Each host command is described in detail below. The “command” in the description indicates a command sent from the host to this LSI and the “response” indicates a response sent from this LSI to the host. The “checksum” is byte-size data calculated so that the sum of all bytes to be sent by this LSI becomes H'00.

(1) User Boot MAT Programming Selection

In response to a user boot MAT programming selection command sent from the host, this LSI selects the program for user boot MAT programming and waits for programming data.

Command	H'42	
Response	H'06	

(2) User MAT Programming Selection

In response to a user MAT programming selection command sent from the host, this LSI selects the program for user MAT programming and waits for programming data.

Command	H'43	
Command	H'06	

(3) Simultaneous Two-User MAT Programming Selection

This LSI does not provide the simultaneous two-user MAT programming function. In response to a simultaneous two-user MAT programming selection command sent from the host, this LSI returns a command error response (sends H'80 and H'44 in that order).

Command	H'44	
Command	H'80	H'44

(4) 256-Byte Programming

In response to a 256-byte programming command sent from the host, this LSI programs the ROM. After completing ROM programming successfully, this LSI returns a response (H'06). If an error has occurred during ROM programming, this LSI returns an error response (H'D0).

Command	H'50	Programming Address		
	Data	Data	:	Data
	SUM			
Response	H'06			
Error response	H'D0	Error		

[Legend]
Programming address (4 bytes):Target address of programming
To program the ROM, a 256-byte boundary address should be specified.
To terminate programming, H'FFFF FFFF should be specified.
Data (256 bytes): Programming data
H'FF should be specified for the bytes that do not need to be programmed.
When terminating programming, no data needs to be specified
(only the programming address and SUM should be sent in that order).
SUM (1 byte): Checksum
Error (1 byte): Error code
H'11: Checksum error
H'2A: Address error (the specified address is not in the target MAT)
H'53: Programming cannot be done due to a programming error

(5) Erasure Selection

In response to an erasure selection command sent from the host, this LSI selects the erasure program and waits for erasure block specification.

Command	H'48
Response	H'06

(6) Block Erasure

In response to a block erasure command sent from the host, this LSI erases the ROM. After completing ROM erasure successfully, this LSI returns a response (H'06). If an error has occurred during ROM erasure, this LSI returns an error response (H'D8).

Command	H'58	Size	Block	SUM
Response	H'06			
Error response	H'D8	Error		

[Legend]
Size (1 byte): Number of bytes in the block specification field (fixed at 1)
Block (1 byte): Block number whose data is to be erased (specified in hexadecimal)
To terminate erasure, H'FF should be specified.
SUM (1 byte): Checksum
Error (1 byte): Error code
H'11: Checksum error
H'29: Block number error (an incorrect block number is specified)
H'51: Erasure cannot be done due to an erasure error

(7) Memory Read

In response to a memory read command sent from the host, this LSI reads data from the ROM. After completing ROM reading, this LSI returns the data stored in the address specified by the memory read command. If this LSI has failed to read the ROM, this LSI returns an error response (H'D2).

Command	H'52	Size	Area	Read start address
	Reading size			SUM
Response	H'52	Reading size		
	Data	Data	:	Data
	SUM			
Error response	H'D2	Error		

[Legend]
Size (1 byte): Total number of bytes in the area, read start address, and reading size fields
Area (1 byte): Target MAT to be read
H'00: User boot MAT
H'01: User MAT
Read start address (4 bytes): Start address of the area to be read
Reading size (4 bytes): Size of data to be read (bytes)
SUM (1 byte): Checksum
Data (1 byte): Data read from the ROM
Error (1 byte): Error code
H'11: Checksum error
H'2A: Address error
The value specified for area selection is neither H'00 nor H'01.
The specified read start address is outside the selected MAT.
H'2B: Data size error
H'00 is specified for the reading size.
The reading size is larger than the MAT.
The end address calculated from the read start address and the reading size is outside the selected MAT.

(8) User Boot MAT Checksum

In response to a user boot MAT checksum command sent from the host, this LSI sums the user boot MAT data in byte units and returns the result (checksum).

Command	H'4A			
Response	H'5A	Size	MAT checksum	SUM

[Legend]
Size (1 byte): Number of bytes in the MAT checksum field (fixed at 4)
MAT checksum (4 bytes): Checksum of the user boot MAT data
SUM (1 byte): Checksum (for the response data)

(9) User MAT Checksum

In response to a user MAT checksum command sent from the host, this LSI sums the user MAT data in byte units and returns the result (checksum).

Command	H'4B			
Response	H'5B	Size	MAT checksum	SUM

[Legend]
Size (1 byte): Number of bytes in the MAT checksum field (fixed at 4)
MAT checksum (4 bytes): Checksum of the user boot MAT data
The user MAT also stores the key code for debugging function authentication. Note that the checksum includes this key code value.
SUM (1 byte): Checksum (for the response data)

(10) User Boot MAT Blank Check

In response to a user boot MAT blank check command sent from the host, this LSI checks whether the user boot MAT is completely erased. When the user boot MAT is completely erased, this LSI returns a response (H'06). If the user boot MAT has an unerased area, this LSI returns an error response (sends H'CC and H'52 in that order).

Command	H'4D	
Response	H'06	
Error response	H'CC	H'52

(11) User MAT Blank Check

In response to a user MAT blank check command sent from the host, this LSI checks whether the user MAT is completely erased. When the user MAT is completely erased, this LSI returns a response (H'06). If the user MAT has an unerased area, this LSI returns an error response (sends H'CD and H'52 in that order).

Command	H'4D	
Response	H'06	
Error response	H'CD	H'52

(12) Boot Program Status Inquiry

For details, refer to section 24.5.4, Inquiry/Selection Host Command Wait State.

24.6 User Program Mode

24.6.1 FCU Command List

To program or erase the user MAT in user program mode, issue FCU commands to the FCU. Table 24.12 is a list of FCU commands for ROM programming and erasure.

Table 24.12 FCU Command List (ROM-Related Commands)

Command	Description
Normal mode transition	Moves to the normal mode (section 24.6.2, Conditions for FCU Command Acceptance)
Status read mode transition	Moves to the status read mode (section 24.6.2, Conditions for FCU Command Acceptance)
Lock bit read mode transition (lock bit read 1)	Moves to the lock bit read mode (section 24.6.2, Conditions for FCU Command Acceptance)
Peripheral clock A notification	Sets the frequency of peripheral clock A
Program	Programs ROM (in 256-byte units)
Block erase	Erases ROM (in block units; erasing the lock bit)
P/E suspend	Suspends programming or erasure
P/E resume	Resumes programming or erasure
Status register clear	Clears the ILGLERR, ERSERR, and PRGERR bits in the FSTATR0 register and cancels the command-locked state
Lock bit read 2	Reads the lock bit of a specified erasure block (updates the FLOCKST bit in the FSTATR1 register in register to reflect the lock bit state)
Lock bit program	Writes to the lock bit of a specified erasure block

FCU commands other than the lock bit read 2 program and lock bit program are also used for EEPROM programming and erasure. When a lock bit read 2 command is issued to the EEPROM, an EEPROM blank check is executed. When a lock bit program command is issued to the EEPROM, it is detected as an illegal command and generates an error (refer to section 25, EEPROM).
To issue a command to the FCU, write to a ROM program/erase address through the P bus. Table 24.13 shows the FCU Command Format. Performing P-bus write access as shown in Table 24.13 under specified conditions starts each command processing in the FCU. For the conditions for FCU command acceptance, refer to section 24.6.2, Conditions for FCU Command Acceptance. For details of each FCU command, refer to section 24.6.3, FCU Command Usage.

When H'71 is sent in the first cycle of an FCU command while the FRDMD bit in flash mode register is 0 (memory area read mode), the FCU accepts the lock bit read mode transition command (lock bit read 1). When a ROM program/erase address is read through the P bus after transition to the lock bit read mode, the FCU copies the lock bit of the erasure block corresponding to the accessed address into all bits in the read data. When H'71 is sent in the first cycle of the FCU command while the FRDMD bit in Flash Mode register is 1 (register read mode), the FCU waits for the second-cycle data (H'D0) of the lock bit read 2 command. When a ROM program/erase address is written to through the P bus in this state, the FCU copies the lock bit of the erasure block corresponding to the accessed address into the FLOCKST bit in the FSTATR1 register.
For details of the suspending operation to be initiated by the P/E suspend command, refer to section 24.6.4, Suspending Operation.

Table 24.13 FCU Command Format

Command	Number of Command Cycles*	First Cycle		Second Cycle		Third Cycle		Fourth and Fifth Cycles		Sixth Cycle		Seventh to 130th Cycles		131st Cycle	
		Address	Data	Address	Data	Address	Data	Address	Data	Address	Data	Address	Data	Address	Data
Normal mode transition	1	RA	H'FF	—	—	—	—	—	—	—	—	—	—	—	—
Status read mode transition	1	RA	H'70	—	—	—	—	—	—	—	—	—	—	—	—
Lock bit read mode transition (lock bit read 1)	1	RA	H'71	—	—	—	—	—	—	—	—	—	—	—	—
Peripheral clock A settings	6	RA	H'E9	RA	H'03	WA	H'0F0F	WA	H'0F0F	RA	H'D0	—	—	—	—
Program	131	RA	H'E8	RA	H'80	WA	WDn	WA	WDn	WA	WDn	RA	WDn	RA	H'D0
Block erase	2	RA	H'20	BA	H'D0	—	—	—	—	—	—	—	—	—	—
P/E suspend	1	RA	H'B0	—	—	—	—	—	—	—	—	—	—	—	—
P/E resume	1	RA	H'D0	—	—	—	—	—	—	—	—	—	—	—	—
Status register clear	1	RA	H'50	—	—	—	—	—	—	—	—	—	—	—	—
Lock bit read 2	2	RA	H'71	BA	H'D0	—	—	—	—	—	—	—	—	—	—
Lock bit program	2	RA	H'77	BA	H'D0	—	—	—	—	—	—	—	—	—	—

[Legend]
RA : ROM program/erase address
When FENTRY0 is 1: An address in the range from H'8080 0000 to H'8087 FFFF
WA : ROM program address:
Start address of 256-byte programming data
BA : ROM erasure block address:
An address in the target erasure block (specified by the ROM program/erase address)
WDn : n-th word of programming data (n = 1 to 128)
Note: * "Number of command cycles" indicates the number of cycles of write access to addresses for programming or erasure generated on the peripheral bus

24.6.2 Conditions for FCU Command Acceptance

The FCU determines whether to accept a command depending on the FCU mode or status. Figure 24.13 shows the FCU Mode Transition Diagram (ROM-Related Modes).

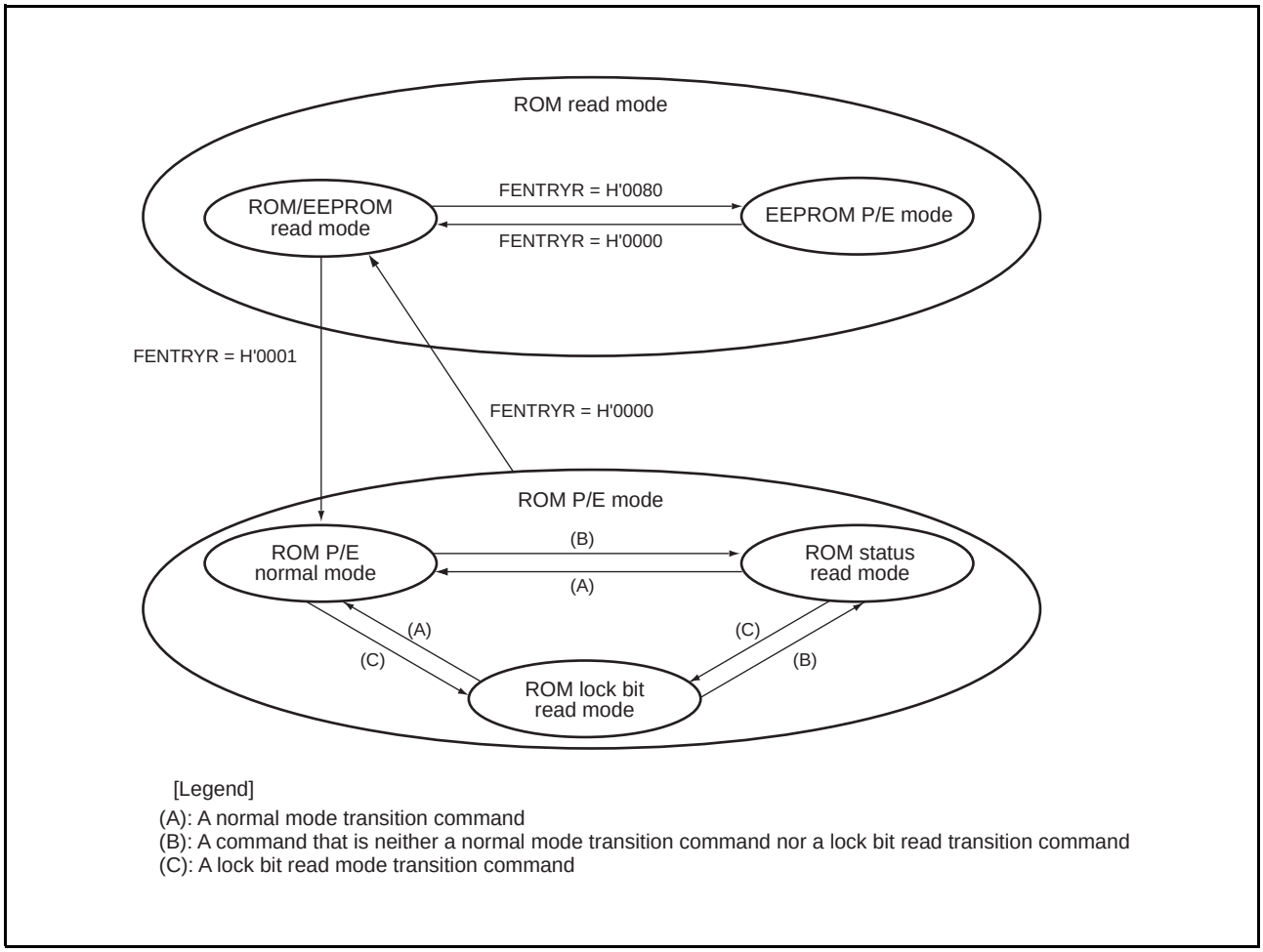


Figure 24.13 FCU Mode Transition Diagram (ROM-Related Modes)

Note: Care should be taken in causing a transition by writing to the FENTRYR register.
After writing to the FENTRYR register, execute a FENTRYR register read instruction, and five or more NOP instructions.

(1) ROM Read Mode

- ROM/EEPROM read mode

The ROM and EEPROM can be read through the ROM cache and peripheral bus A, respectively, at a high speed. The FCU does not accept commands. The FCU enters this mode when the FENTRY0 bit in FENTRYR register is set to 0 and the FENTRYD bit to 0 in FENTRYR register.

- EEPROM P/E mode

The ROM can be read through the ROM cache at a high speed. The FCU accepts commands for EEPROM, but does not accept commands for ROM. The FCU enters this mode when the FENTRY0 bit is set to 0 and the FENTRYD bit to 1. For details of the EEPROM P/E mode, refer to section 24.6.2, Conditions for FCU Command Acceptance.

(2) ROM P/E Mode

- ROM P/E normal mode

The FCU enters this mode when the FENTRYD bit in FENTRYR register is set to 0 and the FENTRY0 bit is set to 1 in ROM read mode, or when a normal mode transition command is accepted in ROM P/E mode. Table 24.14 shows the FCU Modes/States and Acceptable Commands. High-speed read operation is not available for the ROM. If an address in the range from H'8080 0000 to H'8087 FFFF is read through the P-bus while the FENTRY0 bit is set to 1, a ROM access error occurs and the FCU enters the command-locked state (refer to section 24.8.2, Error Protection).

- ROM status read mode

The FCU enters this mode when the FCU accepts a command that is neither a normal mode transition command nor a lock bit read mode transition command in ROM P/E mode. The ROM status read mode includes the state in which the FRDY bit in the FSTATR0 register is 0 and the command-locked state after an error has occurred. Table 24.14 shows the commands that can be accepted in this mode. High-speed read operation is not available for the ROM. If an address in the range from H'8080 0000 to H'8087 FFFF is read through the P-bus while the FENTRY0 bit is set to 1, the FSTATR0 register value is read.

- ROM lock bit read mode

The FCU enters this mode when the FCU accepts a lock bit read mode transition command in ROM P/E mode. Table 24.14 shows the commands that can be accepted in this mode. High-speed read operation is not available for the ROM. The FENTRYR value is the same as that in ROM P/E normal mode. If an address in the range from H'8080 0000 to H'8087 FFFF is read through the P-bus while the FENTRY0 bit is set to 1, the lock bit value of the target erasure block is returned through all bits in the read data.

Table 24.14 shows the acceptable commands in each FCU mode/state. When a command that cannot be accepted is issued, the FCU enters the command-locked state (refer to section 24.8.2, Error Protection).

To make sure that the FCU accepts a command, enter the mode in which the FCU can accept the target command, check the FRDY, ILGLERR, ERSERR, and PRGERR bit values in the FSTATR0 register, and the FCUERR, FRDTCT, and FRCRCT bit values in the FSTATR1 register, and then issue the target FCU command. The CMDLK bit in the 0FASTAT register holds a value obtained by logical ORing the ILGLERR, ERSERR, and PRGERR bit values in the FSTATR0 register and the FCUERR, FRDTCT, and FRCRCT bit values in the FSTATR1 register. Therefore the FCU's error occurrence state can be checked by reading the CMDLK bit. In Table 24.14, the CMDLK bit is used as the bit to indicate the error occurrence state. The FRDY bit in the FSTATR0 register is 0 during the programming/erasure, programming/erasure suspension, and lock bit read 2 processes. While the FRDY bit is 0, the P/E suspend command can be accepted only when the SUSRDY bit in the FSTATR0 register is 1.

Table 24.14 includes 0 and 1 in single cells of the ERSSPD, PRGSPD, and FRDY bit rows for the sake of simplification. The ERSSPD bits 1 and 0 indicate the erasure suspension and programming suspension processes, respectively. The PRGSPD bits 1 and 0 indicate the programming suspension and erasure suspension processes, respectively. The FRDY bit value can be either 1 or 0, which is a value held by the bit prior to a transition to the command lock state.

Table 24.14 FCU Modes/States and Acceptable Commands

Item	P/E Normal Mode			Status Read Mode									Lock Bit Read Mode		
	Programming-Suspended	Erasure-Suspended	Other State	Programming/Erasure Processing	Programming Processing in Erasure-Suspended State	Programming/Erasure Suspension Processing	Lock Bit Read 2 Processing	Programming-Suspended	Erasure-Suspended	Command-Locked (FRDY = 0)	Command-Locked (FRDY = 1)	Other State	Programming-Suspended	Erasure-Suspended	Other State
FRDY bit in FSTATR0	1	1	1	0	0	0	0	1	1	0	1	1	1	1	1
SUSRDY bit in FSTATR0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
ERSSPD bit in FSTATR0	0	1	0	0	1	0/1	0/1	0	1	0/1	0/1	0	0	1	0
PRGSPD bit in FSTATR0	1	0	0	0	0	0/1	0/1	1	0	0/1	0/1	0	1	0	0
CMDLK bit in FASTAT	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0
Normal mode transition	A	A	A	×	×	×	×	A	A	×	×	A	A	A	A
Status read mode transition	A	A	A	×	×	×	×	A	A	×	×	A	A	A	A
Lock bit read mode transition (lock bit read 1)	A	A	A	×	×	×	×	A	A	×	×	A	A	A	A
Peripheral clock A notification	×	×	A	×	×	×	×	×	×	×	×	A	×	×	A
Program	×	*	A	×	×	×	×	×	*	×	×	A	×	*	A
Block erase	×	×	A	×	×	×	×	×	×	×	×	A	×	×	A
P/E suspend	×	×	×	A	×	×	×	×	×	×	×	×	×	×	×
P/E resume	A	A	×	×	×	×	×	A	A	×	×	×	A	A	×
Status register clear	A	A	A	×	×	×	×	A	A	×	A	A	A	A	A
Lock bit read 2	A	A	A	×	×	×	×	A	A	×	×	A	A	A	A
Lock bit program	×	*	A	×	×	×	×	×	*	×	×	A	×	*	A

[Legend]
A: Acceptable
*: Only programming is acceptable for the areas other than the erasure-suspended block
×: Not acceptable

24.6.3 FCU Command Usage

This section shows examples of user processing procedures for firmware transfer to the FCU RAM and the issuing of FCU commands. In some procedures given in this section, the FCU state is not checked before an FCU command is issued but the command result is checked before the processing is completed. To make sure that the FCU accepts a command, check the FCU state before starting processing (refer to section 24.6.2, Conditions for FCU Command Acceptance).

In a flow used in this section, the current state of FCU command handling and error occurrence is checked via the FRDY, ILGLERR, ERSERR, PRGERR, SUSRDY, ERSSPD, and PRGSPD bits in the FSTATR0 register and the FCUERR, FRDTCT, and FRCRCT bits in the FSTATR1 register. Since both the FSTATR0 and FSTATR1 registers can be read in word access at a time, the FCU state can be checked by making register access only once. If the FCU state is checked via the FRDY bit in the FSTATR0 register and the CMDLK bit in FASTAT register, register access must be made twice. However, the state of error occurrence can checked via the CMDLK bit only.

The FRDY bit retains 0, if the FRDTCT and FRCRCT bits are set to 1 to put the FCU into a command-locked state in the middle of its command handling while the FCUERR bit is 1 or the FCDCLE and FRCCLE bits are 1. Since the FCU in a command-locked state halts its processes, the FRDY bit is never set to 1 from 0. If the FRDY retains 0 for a longer period than programming/erasing time or suspend delay time (refer to section 30, Electrical Characteristics), abnormal operation such as the FCU process halt may have occurred. In such case, initialize the FCU by a FCU reset.

If the FRDY bit is set to 1 upon completion of the FCU command handling, the FCUERR bit and the FRDTCT and FRCRCT bits are also 0. Therefore, the state of error occurrence can be checked via the ILGLERR, ERSERR, and PRGERR bits.

(1) Transferring Firmware to the FCU RAM

To use FCU commands, the FCU firmware must be stored in the FCU RAM. When this LSI is started, the FCU firmware is not stored in the FCU RAM; copy the firmware stored in the FCU firmware area to the FCU RAM. If the FCUERR, FRDTCT, or FRCRCT bit in the FSTATR1 register is 1, the firmware stored in the FCU RAM may have been damaged; reset the FCU and copy the FCU firmware again in this case.

Figure 24.14 shows the Procedure for Firmware Transfer to FCU RAM. Before writing data to the FCU RAM, clear the FENTRYR register to H'0000, and execute five or more NOP instructions after executing a FENTRYR register read instruction to stop the FCU. For details on the DMAC settings, refer to section 12, DMAC.

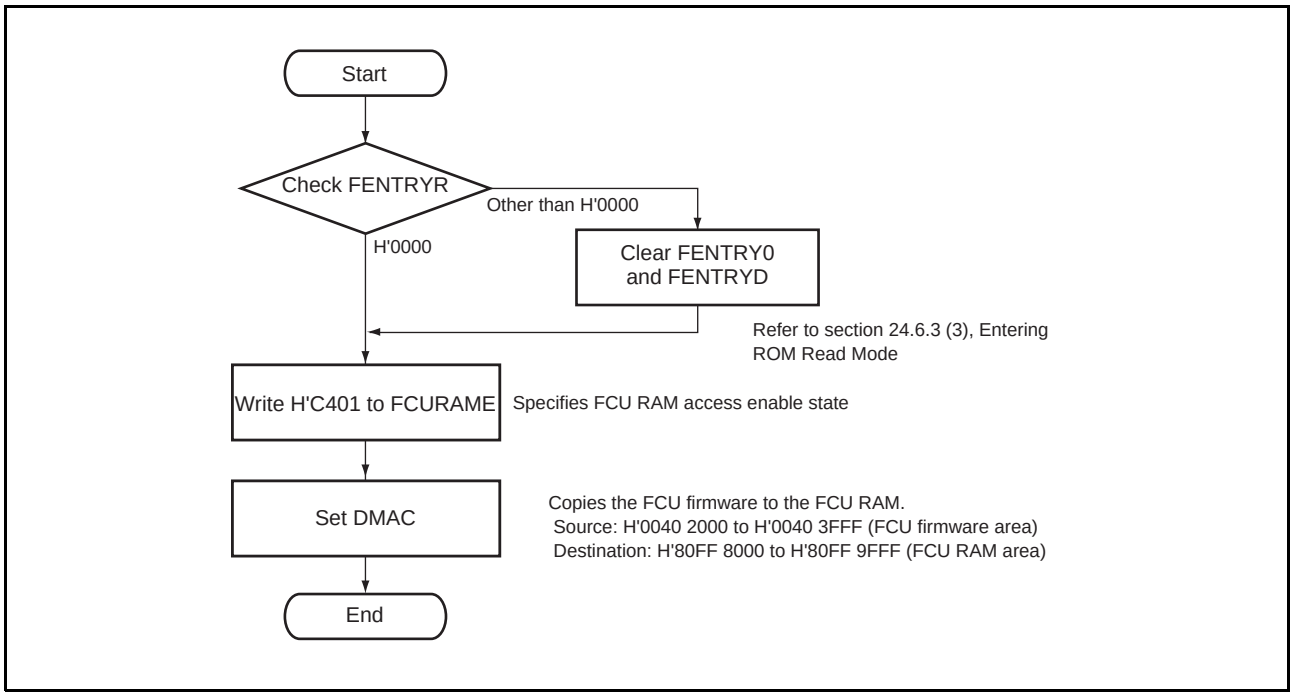


Figure 24.14 Procedure for Firmware Transfer to FCU RAM

(2) Entering ROM P/E Mode

To execute ROM-related FCU commands, set the FENTRY0 bit in FENTRYR register, and execute five or more NOP instructions after executing a FENTRYR register read instruction to make the FCU enter ROM P/E mode (refer to section 24.6.2, Conditions for FCU Command Acceptance). To execute FCU commands for the ROM, set the FENTRY0 bit to 1. For the conditions for writing to the FENTRY0 bit, refer to section 24.3.8, Flash P/E Mode Entry Register (FENTRYR).

After a transition from ROM read mode to ROM P/E mode, the FCU is in ROM P/E normal mode.

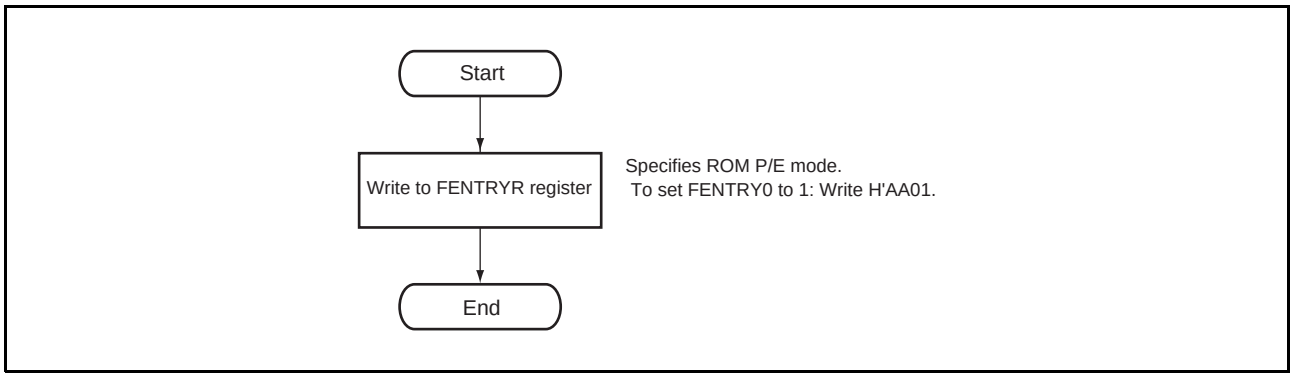
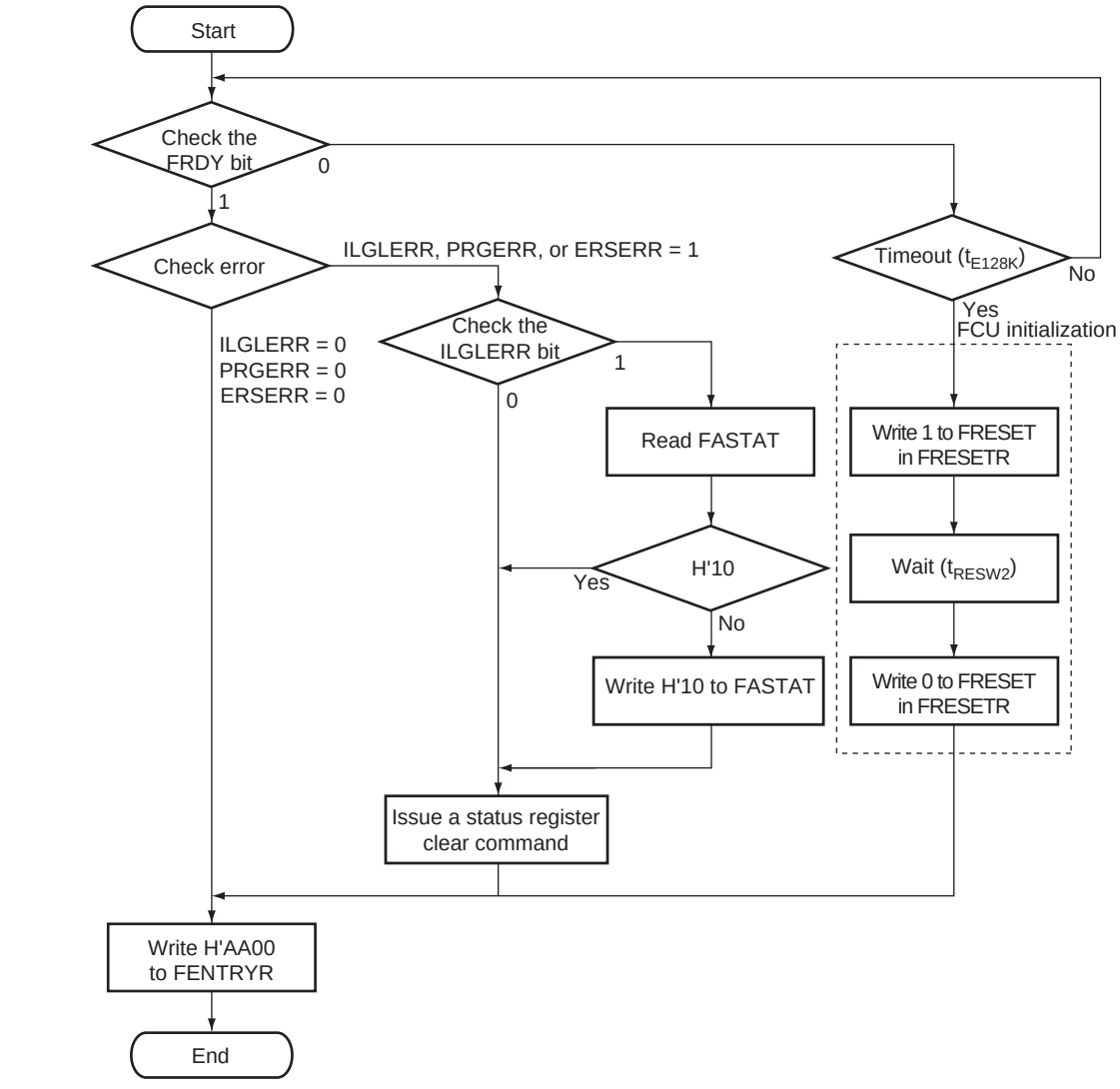


Figure 24.15 Procedure for Transition to ROM P/E Mode

(3) Entering ROM Read Mode

To enable high-speed ROM read access through the ROM cache, clear the FENTRY0 bit in FENTRYR register, and execute five or more NOP instructions after executing a FENTRYR register read instruction to make the FCU enter ROM read mode (refer to section 24.6.2, Conditions for FCU Command Acceptance). A transition from ROM P/E mode to ROM read mode must be made while no FCU error has been detected since FCU command processing is completed.



Notes. • t_{E128K}: Time required to erase a 128-Kbyte block (refer to section 30, Electrical Characteristics).
• t_{RESW2}: Reset pulse width during programming and erasure (refer to section 30, Electrical Characteristics).

Figure 24.16 Procedure for Transition to ROM Read Mode

(4) Using ROM P/E Normal Mode Transition Command

The FCU can be moved to ROM P/E normal mode in two ways: one is to set FENTRYR register appropriately in ROM read mode (refer to section 24.6.3, FCU Command Usage (1), Transferring Firmware to the FCU RAM) and the other is to issue a normal mode transition command in ROM P/E mode (Figure 24.17). The status read mode transition command and the lock bit read mode transition command can be used in the same way as the normal mode transition command.

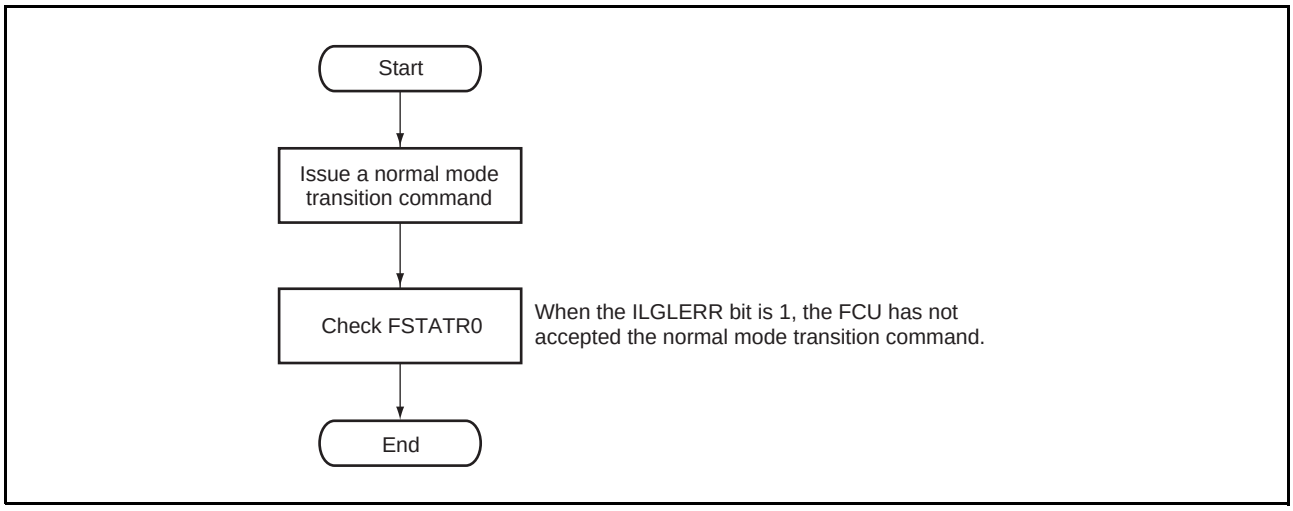


Figure 24.17 Procedure to Use ROM P/E Normal Mode Transition Command

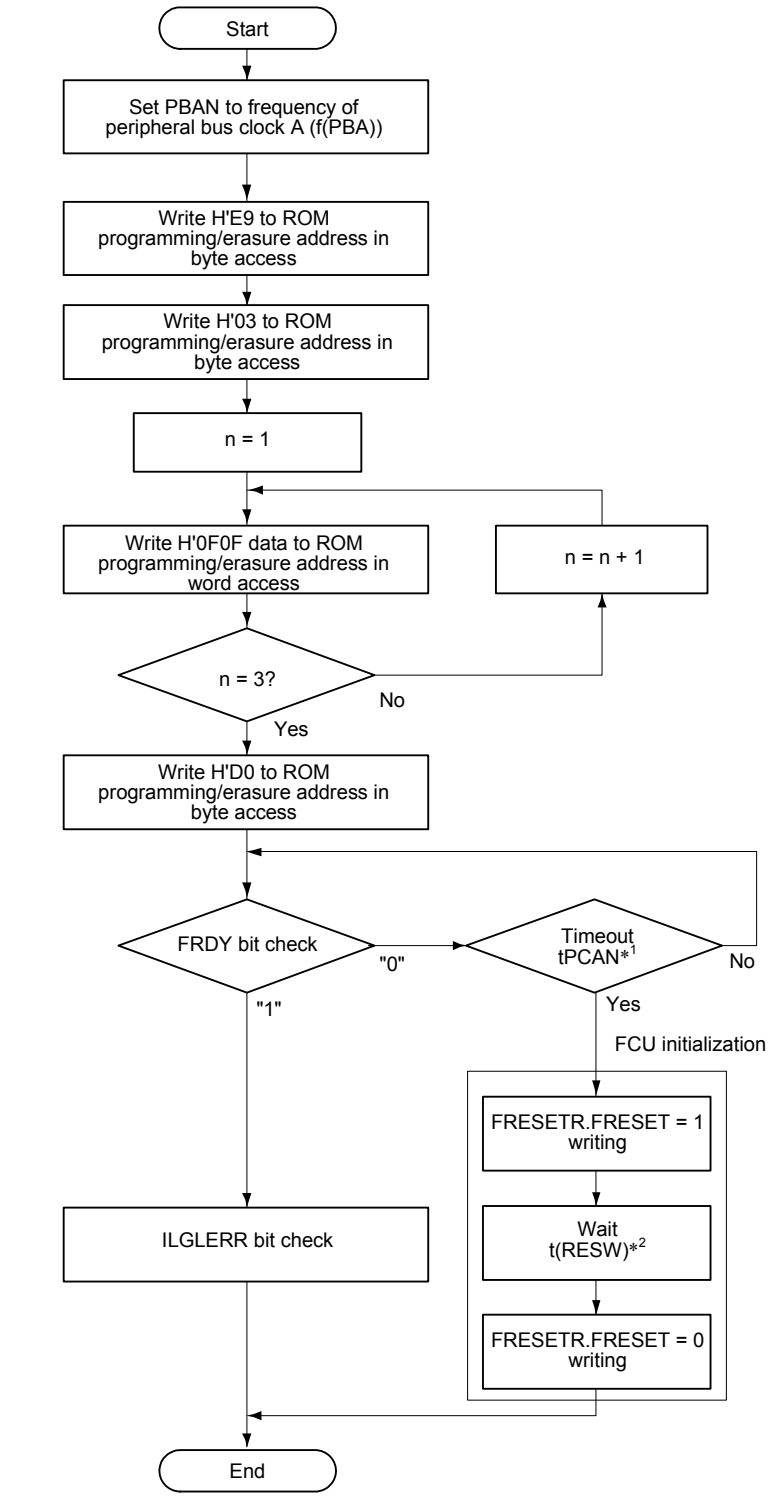
(5) Using the Peripheral Bus Clock A Notification Command

The peripheral bus clock A is used in programming and erasing the ROM, so the frequency of this clock has to be set in the PBAN. Frequencies in the range from 8 to 50 MHz are selectable. A frequency beyond this range should not be set.

The peripheral bus clock A notification command is used after the PBAN setting has been made. In the first and second cycles for the peripheral bus clock A notification command, respectively, the values H'E9 and H'03 are written as a byte to the address range for programming and erasure of the ROM. Word-unit writing is used in the third to fifth cycles of the command. Accordingly, make sure that the start addresses used are aligned with four-byte boundaries. After H'0F0F has been written three times (as a word) to the address range for programming and erasure of the ROM, the process of the FCU setting the frequency of the peripheral bus clock A starts once the value H'D0 has been written as a byte in the sixth cycle. The FRDY bit in FSTATR0 can be used to check whether or not the settings have been completed.

Addresses that can be used in the first to sixth cycles differ according to the settings of the FENTRY0 bit in FENTRYR. Ensure that the addresses suit the settings of this bit. If issuing of the command is attempted with an erroneous combination of the setting of the bit and specified addresses, the FCU will detect the error and enter the command-locked state (see section section 24.8.2, Error Protection).

Furthermore, if the setting for the peripheral bus clock A in use is not changed from this setting after release from the reset state, this setting is also valid for the next FCU command.



Notes: 1. tPCAN: 60 µs when f(PBA) = 50 MHz, 120 µs when f(PBA) = 25 MHz
2. t(RESW): Reset pulse width during programming/erasure (see section 30, Electrical Characteristics)

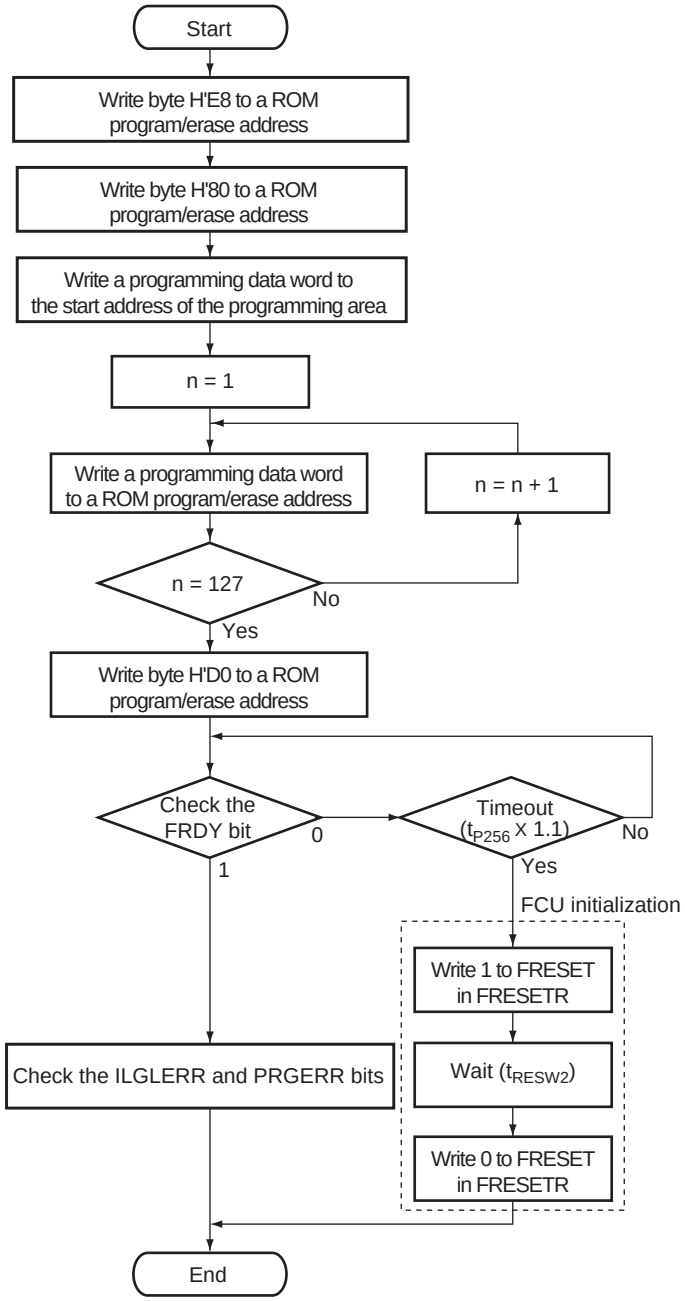
Figure 24.18 Using the Peripheral Bus Clock A Notification Command

(6) Programming

To program the ROM, use the program command. Write byte H'E8 to a ROM program/erase address in the first cycle of the program command and byte H'80 in the second cycle. Access the P bus in words from the third to 130th cycles of the command. In the third cycle, write the programming data to the start address of the target programming area. Here, the start address must be a 256-byte boundary address. After writing words to ROM program/erase addresses 127 times, write byte H'D0 to a ROM program/erase address in the 131st cycle; the FCU then starts ROM programming. Read the FRDY bit in the FSTATR0 register to confirm that ROM programming is completed.

An address that can be specified in the first to 131st cycles is in the range from H'8080 0000 to H'8087 FFFF when the FENTRY0 bit is set to 1. If a command is issued while an illegal combination of FENTRY0 bit values and addresses is specified, the FCU detects an error and enters command-locked state (refer to section 24.8.2, Error Protection).

If the area accessed in the third to 130th cycles includes addresses that do not need to be programmed, write H'FFFF as the programming data for those addresses. To ignore the protection provided by the lock bit during programming, set the FPROTCN bit in the FPROTR register to 1 before starting programming.



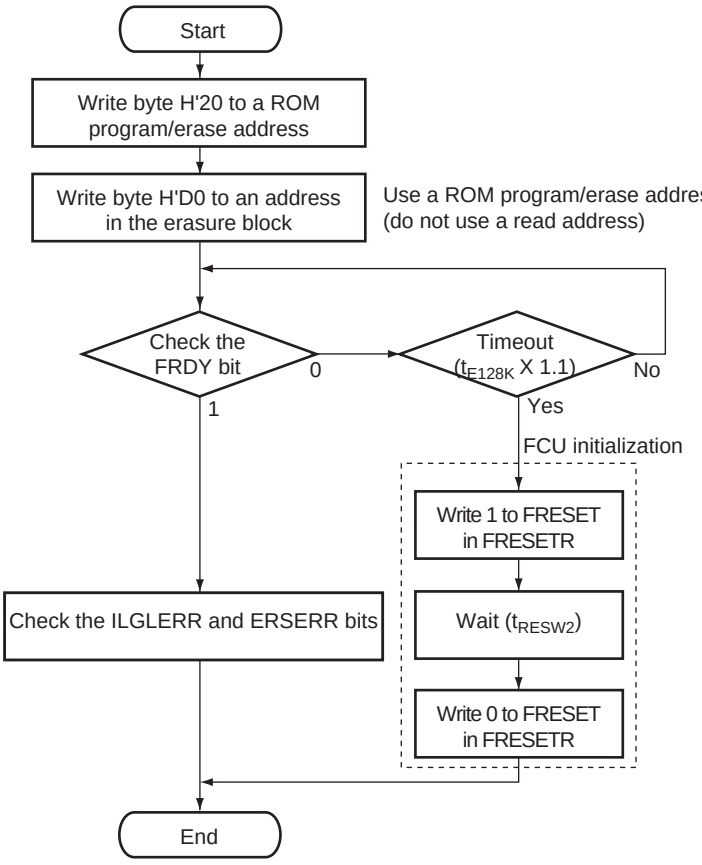
Notes. • t_{P256}: Time required to write 256-byte data (refer to section 30, Electrical Characteristics).
• t_{RESW2}: Reset pulse width during programming and erasure (refer to section 30, Electrical Characteristics).

Figure 24.19 Procedure for ROM Programming

(7) Erasure

To erase the ROM, use the block erase command. Write byte H'20 to a ROM program/erase address in the first cycle of the block erase command. Write byte H'D0 to an address in the target erasure block in the second cycle; the FCU then starts ROM erasure. Read the FRDY bit in the FSTATR0 register to confirm that ROM erasure is completed.

To ignore the protection provided by the lock bit during erasure, set the FPROTCN bit in FPROTR register to 1 before starting erasure.



Notes. • t_{E128K}: Time required to erase a 128-Kbyte block (refer to section 30, Electrical Characteristics).
• t_{RESW2}: Reset pulse width during programming and erasure (refer to section 30, Electrical Characteristics).

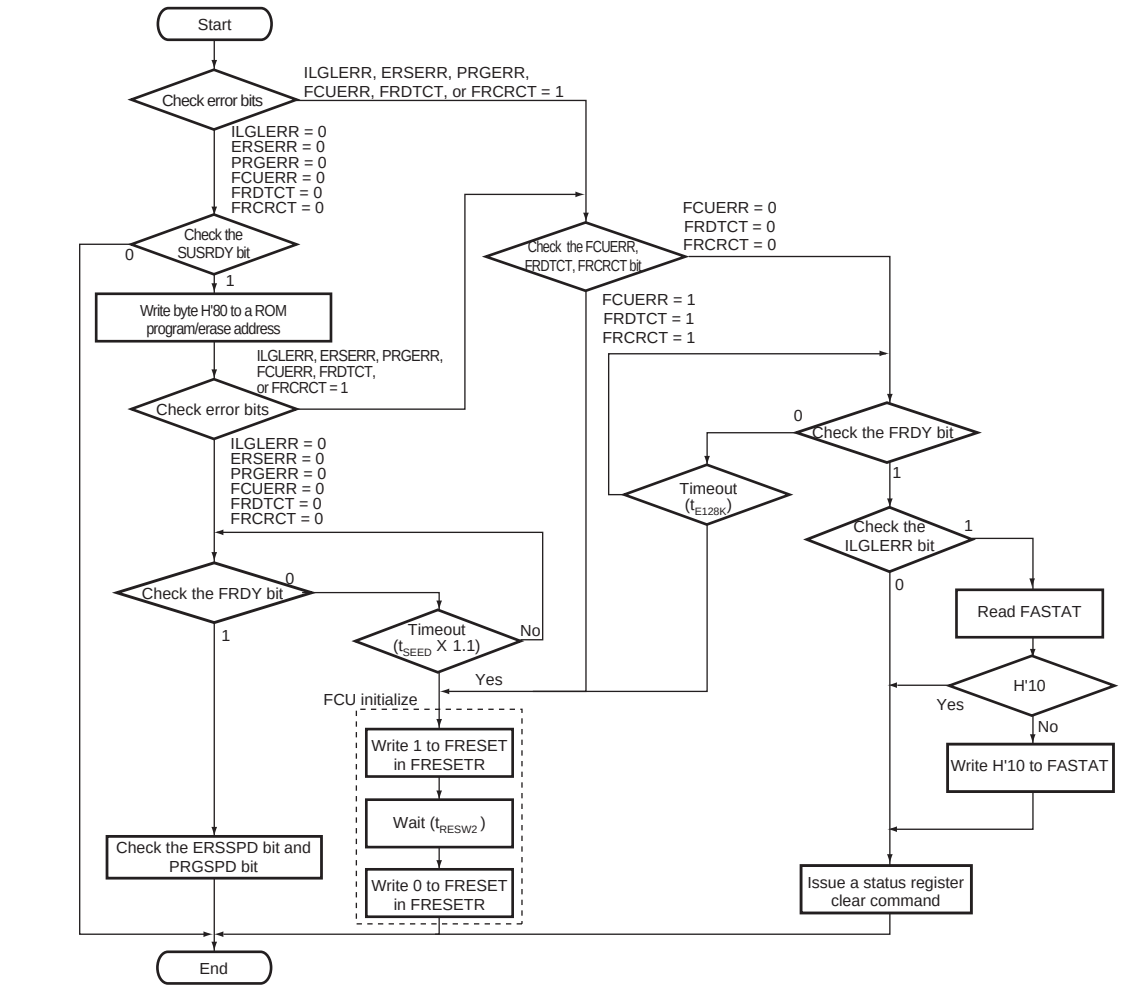
Figure 24.20 Procedure for ROM Erasure

(8) Suspending Programming or Erasure

To suspend programming or erasure of the ROM, use the P/E suspend command. Before issuing a P/E suspend command, check that the ILGLERR, ERSERR, and PRGERR bits in the FSTATR0 register and the FCUERR, FRDTCT, and FRCRCT bits in the FSTATR1 register are 0; that is, to ensure that programming or erasure processing is being performed correctly. Also, check that the SUSRDY bit in the FSTATR1 register is 1 to ensure that a suspend command is acceptable.

After issuing a P/E suspend command, read both the FSTATR0 and FSTATR1 registers to ensure no error has occurred. If an error has occurred, at least one of the ILGLERR, PRGERR, ERSERR, FCUERR, FRDTCT, and FRCRCT bits is set to 1. If programming/erasure is complete within the period from when the SUSRDY bit is ensured to be 1 until a P/E suspend command is accepted, the ILGLERR bit is set to 1 as the issued command is detected as illegal. If a P/E suspend command is accepted when programming/erasure is complete, no error occurs, hence no transition to a suspended state (the FRDY bit is 1 and both the ERSSPD and PRGSPD bits are 0).

Once a P/E suspend command is accepted and programming/erasure is normally suspended, the FCU enters a suspended state and that the FRDY bit is 1 and the ERSSPD or PRGSPD bit is 1. After issuing a P/E suspend and ensuring that the FCU has entered a suspend state, determine which operation to perform in the succeeding process.
If a P/E resume command is issued in the succeeding process while the FCU has not entered a suspended state, an illegal command error occurs and the FCU enters a command-locked state (refer to section 24.8.2, Error Protection).



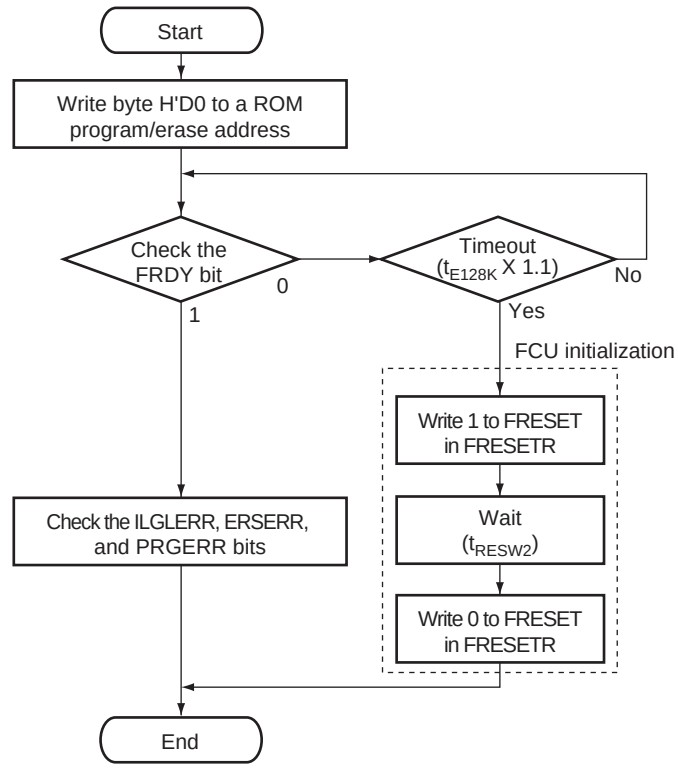
Notes. • t_{SEED}: Suspension delay time.
• t_{E128K}: Time required to erase a 128-Kbyte block (refer to section 30, Electrical Characteristics).

Figure 24.21 Procedure for Programming/Erasure Suspension

Once the FCU has entered the erasure-suspended state, blocks not for erasing can be written to. In both programming-suspended and erasure-suspended states, the FCU can be moved to ROM read mode by clearing the FENTRYR register.
For the operation when the FCU accepts a P/E suspend command, refer to section 24.6.4, Suspending Operation.

(9) Resuming Programming or Erasure

To resume programming or erasure that has been suspended, use the P/E resume command. If the FENTRYR register setting has been modified during suspension, issue a P/E resume command only after resetting the FENTRYR register to the previous value that was held before the P/E suspension command was issued.



Notes. • t_{E128K}: Time required to erase a 128-Kbyte block (refer to section 30, Electrical Characteristics).
• t_{RESW2}: Reset pulse width during programming and erasure (refer to section 30, Electrical Characteristics).

Figure 24.22 Procedure for Resuming Programming or Erasure

(10) Clearing Status Register 0 (FSTATR0)

To clear the ILGLERR, PRGERR, and ERSERR bits in the FSTATR0 register, use the status register clear command. When any one of the ILGLERR, PRGER, and ERSERR bits is 1, the FCU is in command-locked state, in which the FCU only accepts the status register clear command and does not accept other commands. When the ILGLERR bit is 1, check also the value of the ROMAE, EEPAE, EEPIFE, EEPRPE, and EEPWPE bits in the FASTAT register. If a status register clear command is issued without clearing these bits, the ILGLERR bit is not cleared.

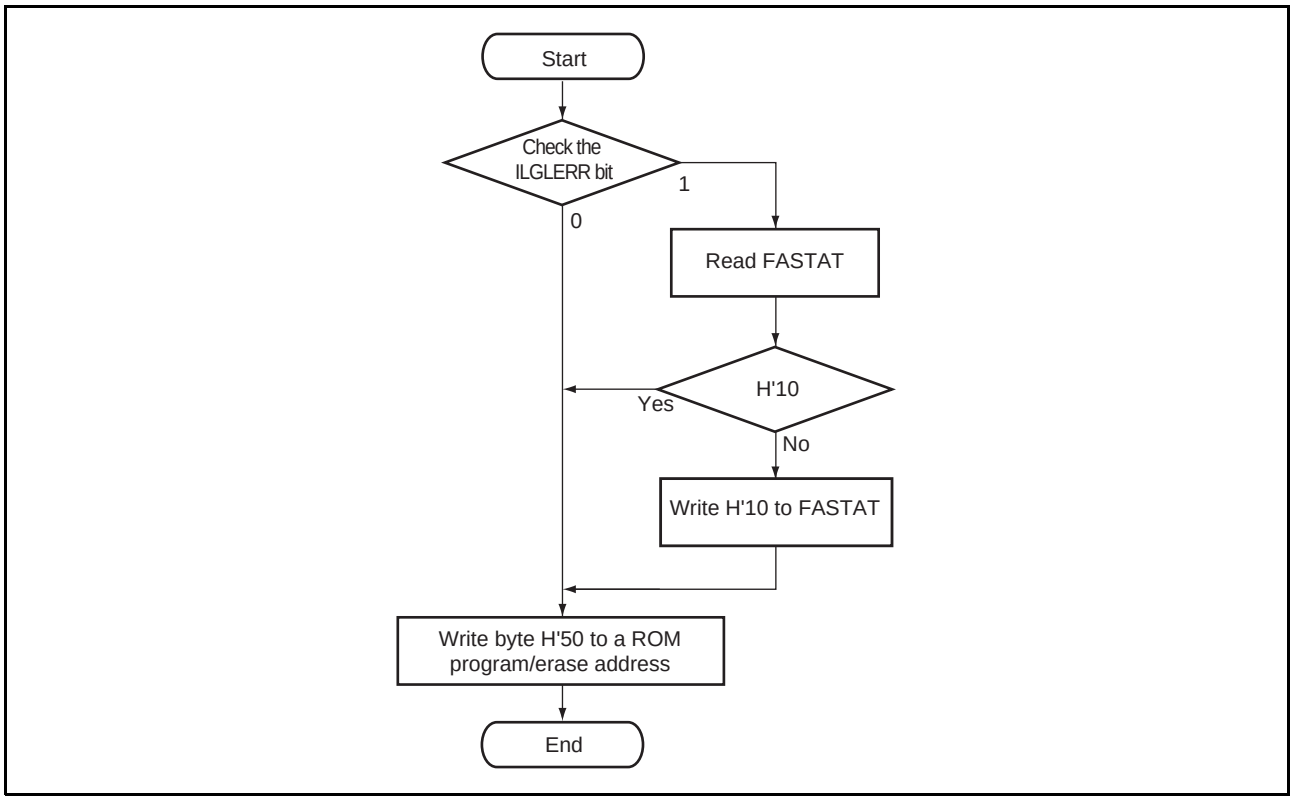


Figure 24.23 Procedure for Clearing Status Register 0

(11) Checking Status Register 0 (FSTATR0)

The FSTATR0 register value can be checked in two ways: one is to directly read the FSTATR0 register and the other is to read a ROM program/erase address in ROM status read mode. After an FCU command is issued that is neither a normal mode transition command nor a lock bit read mode transition command, the FCU is in ROM status read mode. In the example shown in Figure 24.24, a status read mode transition command is issued to enter ROM status read mode, and then a ROM program/erase address is read to check the FSTATR0 register value.

Figure 24.24 shows the Procedure for Checking Status Register 0.

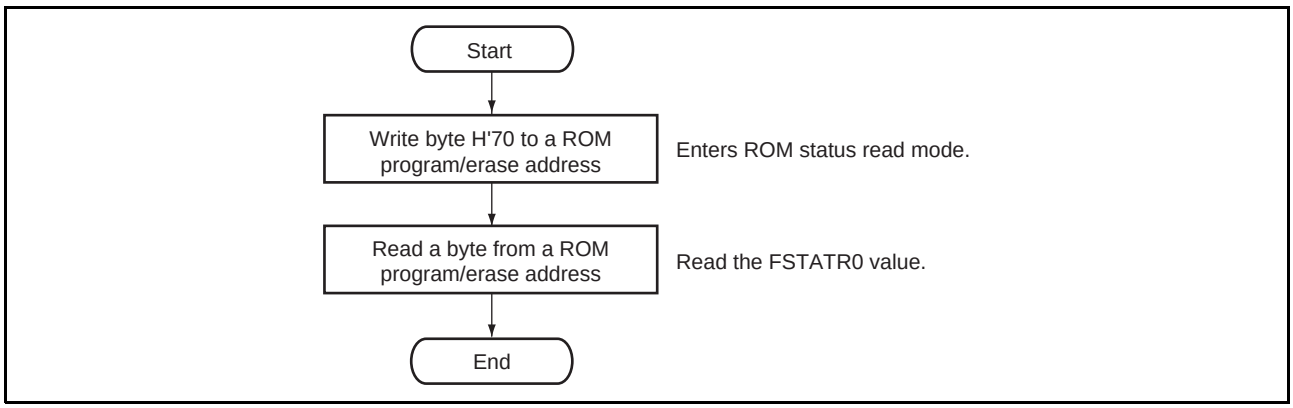


Figure 24.24 Procedure for Checking Status Register 0

(12) Reading Lock Bit

Each erasure block in the user MAT has a lock bit. While the FPROTCN bit in the FPROTR register is 0, the erasure block whose lock bit is set to 0 cannot be programmed or erased.

The lock bit status can be checked in either memory area read mode or register read mode. In memory area read mode (the FRDMD bit in the FMODR register is 0), read a ROM program/erase address in ROM lock bit read mode, and the lock bit value in the specified erasure block is copied to all bits in the data read through the P bus. In register read mode (the FRDMD bit in the FMODR register is 1), issue a lock bit read 2 command, and the lock bit value in the specified erasure block is copied to the FLOCKST bit in the FSTATR1 register.

Figure 24.25 shows the Procedure for Reading Lock Bit in Memory Area Read Mode, and Figure 24.26 shows the Procedure for Reading Lock Bit in Register Read Mode.

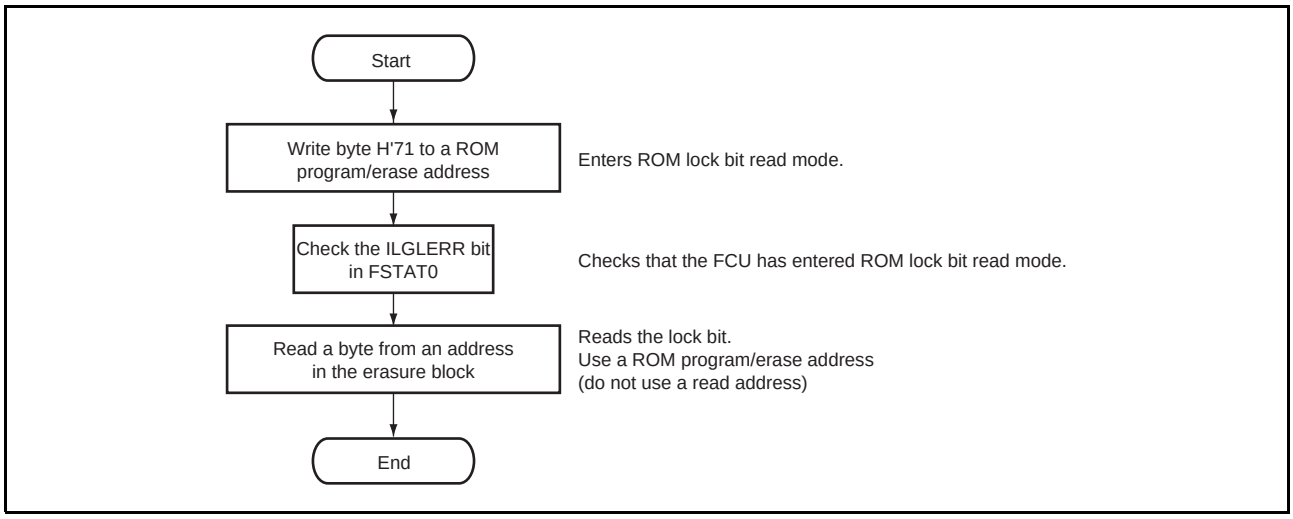
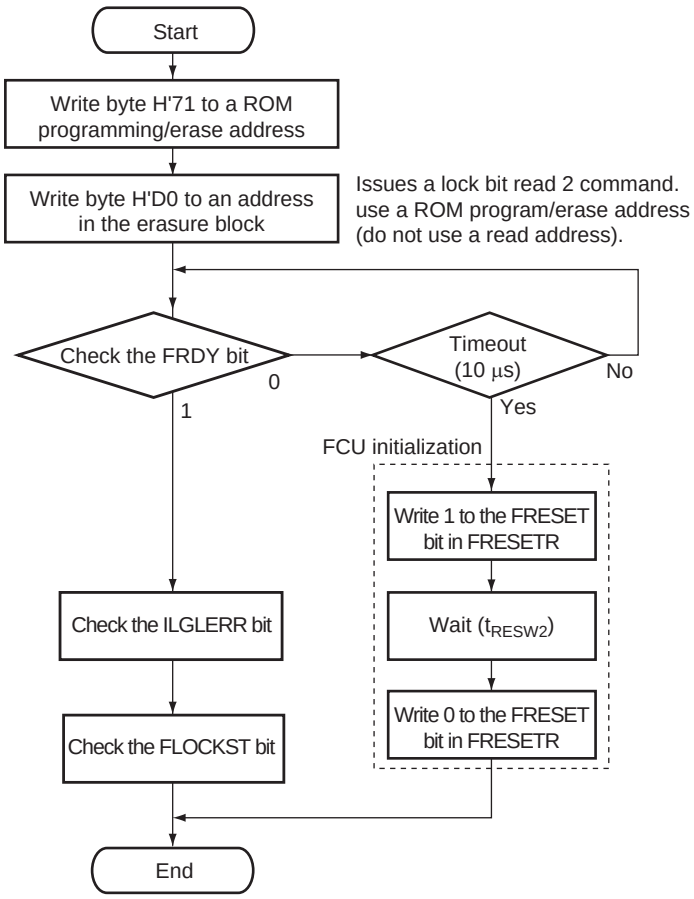


Figure 24.25 Procedure for Reading Lock Bit in Memory Area Read Mode



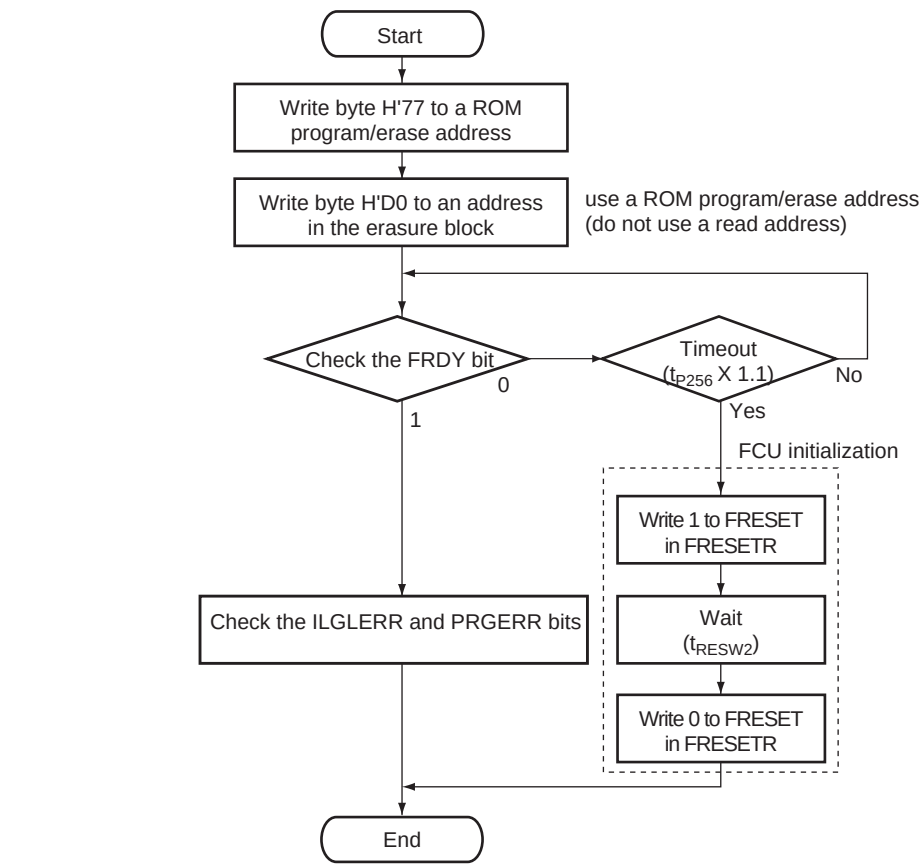
Note. • t_{RESW2}: Reset pulse width during programming and erasure (refer to section 30, Electrical Characteristics).

Figure 24.26 Procedure for Reading Lock Bit in Register Read Mode

(13) Writing to Lock Bit

Each erasure block in the user MAT has a lock bit. To write to a lock bit, use the lock bit program command. Write byte H'77 to a ROM program/erase address in the first cycle of the lock bit program command. Write byte H'D0 to an address in the target erasure block whose lock bit is to be written to in the second cycle; the FCU then starts writing to the lock bit. Read the FRDY bit in the FSTATR0 register to confirm that writing is completed.

Figure 24.27 shows the Procedure for Writing to the Lock Bit.



Notes. • t_{P256}: Time required to write 256-byte data (refer to section 30, Electrical Characteristics).
• t_{RESW2}: Reset pulse width during programming and erasure (refer to section 30, Electrical Characteristics).

Figure 24.27 Procedure for Writing to the Lock Bit

To erase a lock bit, use the block erase command. While the FPROTCN bit in the FPROTR register is 0, the erasure block whose lock bit is set to 0 cannot be erased. Set the FPROTCN bit to 1, and then issue a block erase command to erase a lock bit. The block erase command erases all data in the specified erasure block; it is not possible to erase only the lock bit.

24.6.4 Suspending Operation

When a P/E suspend command is issued while ROM is being programmed or erased, the FCU suspends the programming or erasure processing. Figure 24.28 gives an overview of operation for suspending programming. Upon accepting a programming command, the FCU clears the FRDY bit in the FSTATR0 register to 0 and starts programming. Once the FCU enters a state where it is ready to accept a command after the start of programming, the SUSRDY bit is set to 1. If a P/E suspend command is issued, the FCU accepts the command and clears the SUSRDY bit to 0. If the FCU accepts the command while reapplying a write pulse, the FCU continues applying the pulse. After a specified pulse application time has elapsed, the FCU completes applying the pulse, suspends programming, and sets the PRGSPD bit to 1. Once the process completes, the FCU sets the FRDY bit to 1 and enters a programming suspended state. If the FCU accepts a P/E resume command in this state, the FCU clears the FRDY and PRGSPD bits to 0 and restarts programming.

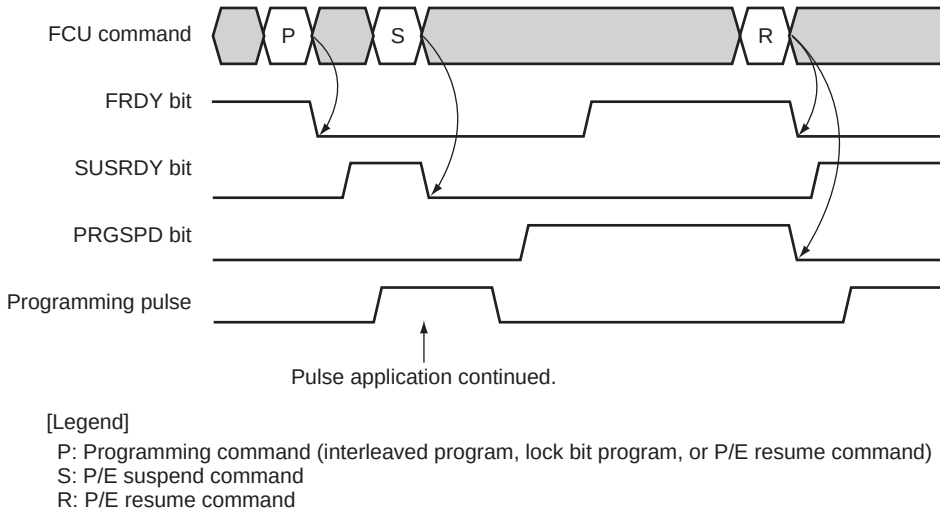


Figure 24.28 Suspending Programming Processing

Figure 24.29 shows the operation for suspending erasure processing. Upon accepting an erasing command, the FCU clears the FRDY bit to 0 and starts erasing. Once the FCU enters a state where it is ready to accept a command after the start of erasing, the SUSRDY bit is set to 1. If a P/E suspend command is issued, the FCU accepts the command and clears the SUSRDY bit. If the FCU accepts the command during its erasing operation, the FCU starts a suspending process even while applying a pulse and sets the ERSSPD bit to 1. Once the suspending process completes, the FCU sets the FRDY bit to 1 and enters an erasing suspended state. If the FCU accepts a P/E resume command in this state, the FCU clears the FRDY and PRGSPD bits to 0 and restarts erasing. The operations of the FRDY, SUSRDY, and ERSSPD bits are independent of the erasure-suspended mode.

The suspending erasure processing affects the control of erasure pulse. Issue a P/E suspend command 1.7 ms or more after a P/E resume command is issued.

After an erasure command is issued, if the FCU accepts the first P/E suspend command while applying erasing pulse, the FCU suspends the pulse application and enters an erasure-suspended state. The FCU resumes erasing by accepting a P/E resume command. Within 1.7 ms before the next P/E suspend command is issued, the FCU completes applying the erasing pulse. If the FCU accepts a P/E suspend command after 1.7 ms or more have elapsed, the FCU enters an erasure-suspended state.

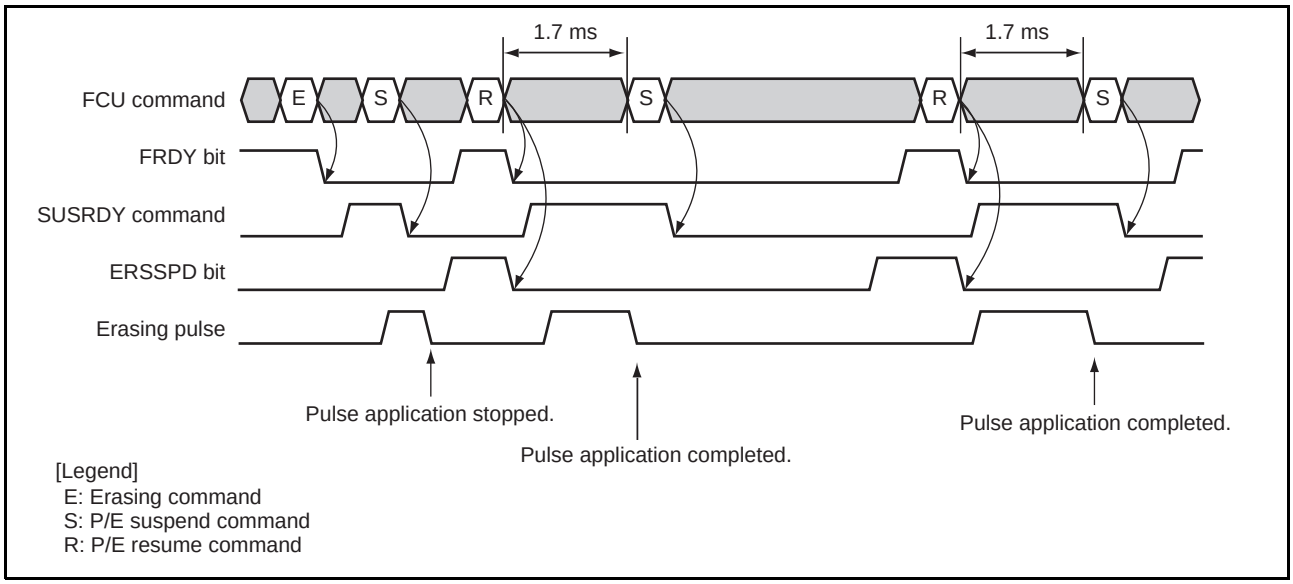


Figure 24.29 Suspending Erasure Processing

24.7 User Boot Mode

To program or erase the user MAT in user boot mode, issue FCU commands to the FCU. A user-defined boot mode can be implemented by writing to the user boot MAT a ROM programming/erasing routine that uses a desired communications interface; when this LSI is started in user boot mode after that, the user-defined boot mode is initiated. Programming/erasure of the user boot MAT is only enabled in boot mode.

24.7.1 User Boot Mode Initiation

When this LSI is started in user boot mode, execution starts in the embedded program stored MAT, necessary processing such as FCU firmware transfer to the FCU RAM is performed, and then execution jumps to the location indicated by the reset vector of the user boot MAT. Figure 24.30 gives an Overview of Boot Sequence in User Boot Mode.

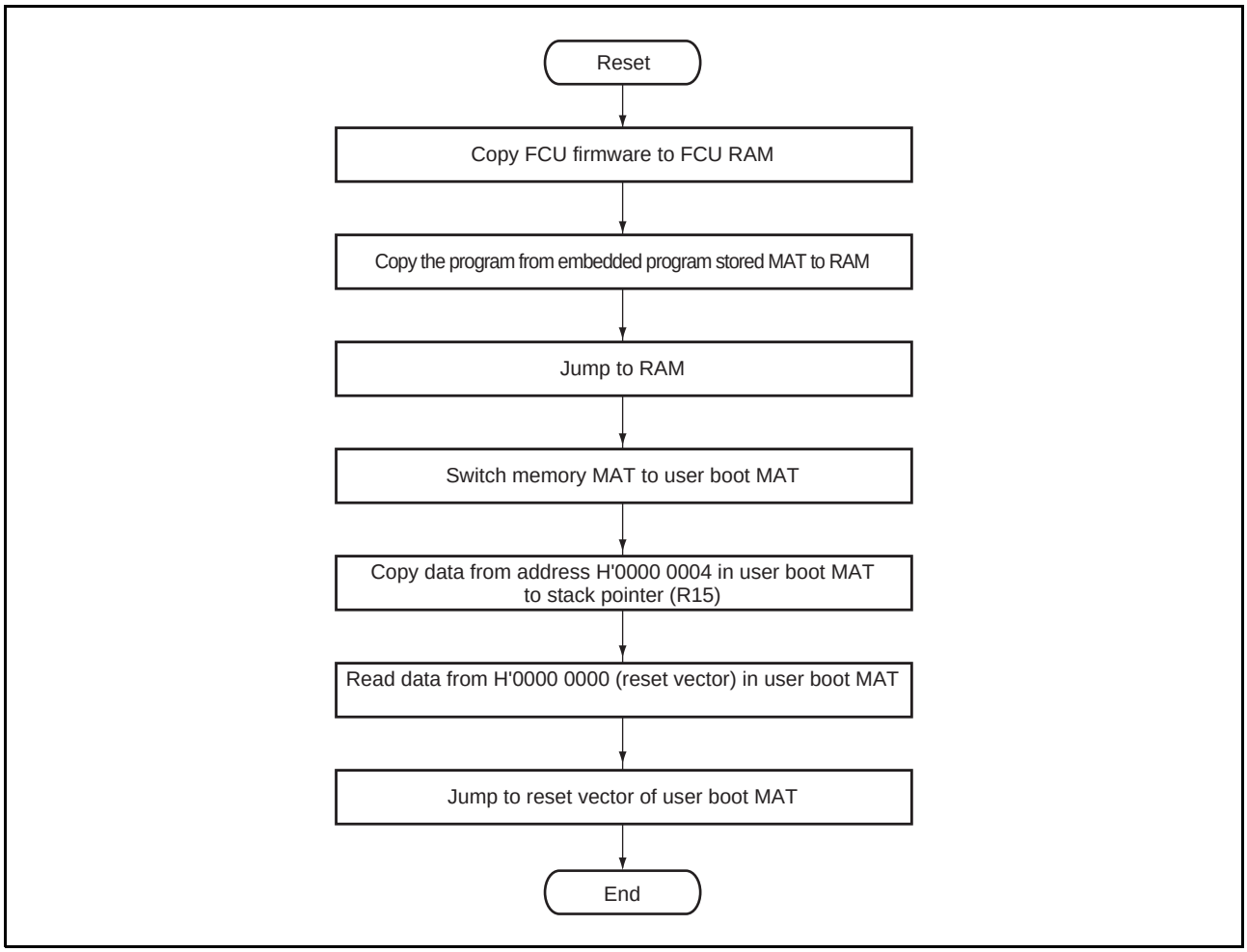


Figure 24.30 Overview of Boot Sequence in User Boot Mode

24.7.2 User MAT Programming

The user MAT can be programmed by starting this LSI in user boot mode while the user MAT programming/erasing routine created by the user is stored in the user boot MAT. Be sure to copy the user MAT programming/erasing routine to the RAM and execute it in the RAM. The user boot MAT is selected in the initial state in user boot mode; be sure to switch the memory MAT to the user MAT before starting programming. If an FCU command for ROM programming or erasure is issued while the user boot MAT is selected, the FCU does not program or erase the ROM. Figure 24.31 shows an Example of User MAT Programming.

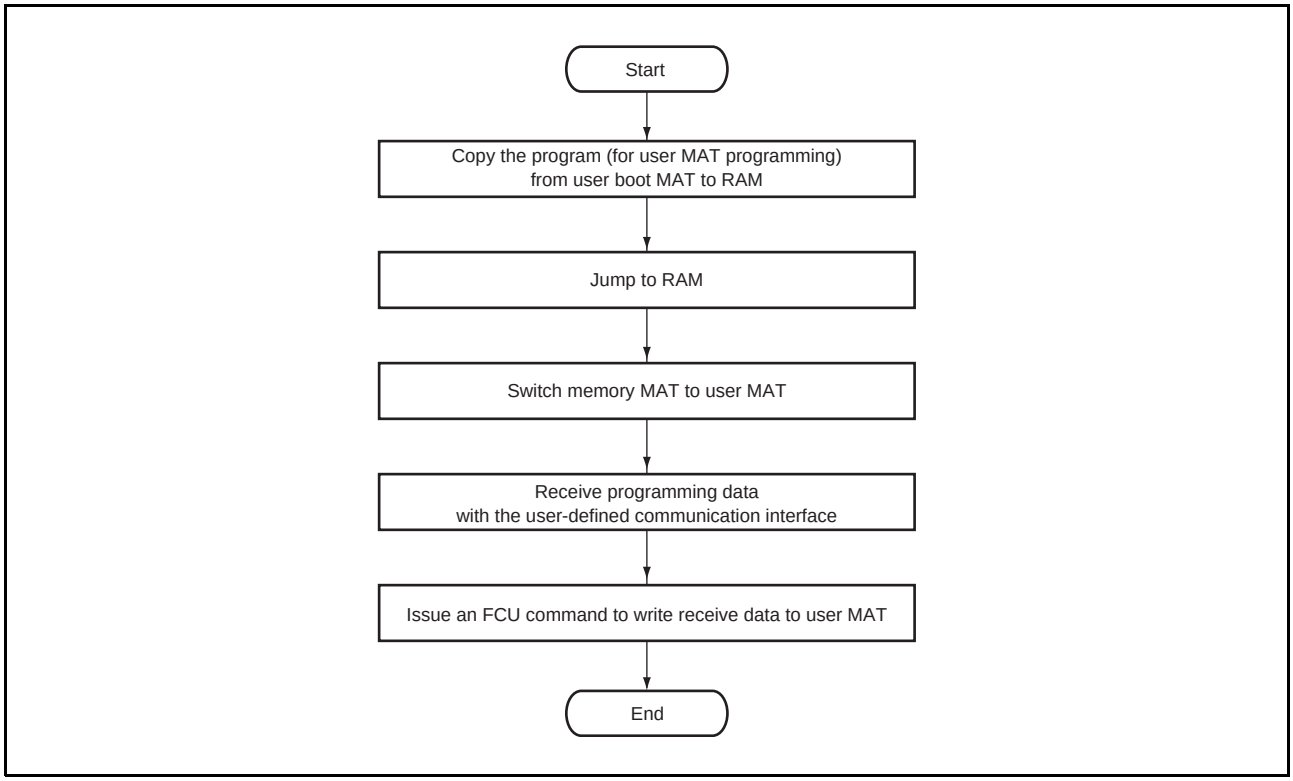


Figure 24.31 Example of User MAT Programming

24.8 Protection

There are two types of ROM programming/erasure protection: software, and error protection.

24.8.1 Software Protection

The software protection function disables ROM programming and erasure according to the control register settings or the lock bit settings in the user MAT. If an attempt is made to issue a programming or erasing command to the ROM against software protection, the FCU detects an error and enters command-locked state.

(1) Protection through the FENTRYR register

- When the FENTRY0 bit is 0,
  The 512-Kbyte ROM (read addresses: H'0000 0000 to H'0007 FFFF; program/erase addresses: H'8080 0000 to H'8087 FFFF) is set to ROM read mode.
  In ROM read mode, the FCU does not accept commands, so ROM programming and erasure are disabled. If an attempt is made to issue an FCU command in ROM read mode, the FCU detects an illegal command error and enters command-locked state (refer to section 24.8.2, Error Protection).

(2) Protection through Lock Bits

Each erasure block in the user MAT has a lock bit. When the FPROTCN bit in the FPROTR register is 0, the erasure block whose lock bit is set to 0 cannot be programmed or erased. To program or erase the erasure block whose lock bit is 0, set the FPROTCN bit to 1. If an attempt is made to issue a programming or erasing command against protection by lock bits, the FCU detects an programming/erasure error and enters command-locked state (refer to section 24.8.2, Error Protection).

24.8.2 Error Protection

The error protection function detects an illegal FCU command issued, an illegal access, or an FCU malfunction, and disables FCU command acceptance (command-locked state). While the FCU is in command-locked state, the ROM cannot be programmed or erased. To cancel command-locked state, issue a status register clear command while the FASTAT register is H'10.

While the CMDLKIE bit in the FAEINT register is 1, a flash interface error (FIFE) interrupt is generated if the FCU enters command-locked state (the CMDLK bit in FASTAT register becomes 1). While the ROMAEINT bit in the FAEINT register is 1, an FIFE interrupt is generated if the ROMAE bit in the FASTAT register becomes 1.

Table 24.15 and Table 24.16 shows the error protection types dedicated for the ROM, those used in common by the ROM and the EEPROM, and the status bit values (the ILGLERR, ERSERR, and PRGERR bits in the FSTATR0 register, the FCUERR, FRDTCT, and FRCRCT bits in the FSTATR1 register, and the ROMAE bit in the FASTST register) after each error detection. If the FCU enters command-locked state due to a command other than a suspend command issued during programming or erasure processing, the FCU continues programming or erasing the ROM. In this state, the P/E suspend command cannot suspend programming or erasure. If a command is issued in command-locked state, the ILGLERR bit becomes 1 and the other bits retain the values set due to the previous error detection.

Table 24.15 Error Protection Types (1)

Error	Description	ILGLERR	ERSERR	PRGERR	FCUERR	FRDTCT	FRCRCT	ROMAE
FENTRYR setting error	The value set in FENTRYR is not H'0001or H'0008.	1	0	0	0	0	0	0
	The FENTRYR setting for resuming operation does not match that for suspending operation.	1	0	0	0	0	0	0
Illegal command error	An undefined code has been specified in the first cycle of an FCU command.	1	0	0	0	0	0	0
	The value specified in the last of the multiple cycles of an FCU command is not H'D0.	1	0	0	0	0	0	0
	The command issued during programming or erasure is not a suspend command.	1	0	0	0	0	0	0
	A suspend command has been issued during operation that is programming nor erasure.	1	0	0	0	0	0	0
	A suspend command has been issued in suspended state.	1	0	0	0	0	0	0
	A resume command has been issued in a state that is not a suspended state.	1	0	0	0	0	0	0
	A programming or erasing command (program, lock bit program, block erase) has been issued in programming-suspended state.	1	0	0	0	0	0	0
	A block erase command has been issued in erasure-suspended state.	1	0	0	0	0	0	0
	A program, lock bit program, or non-interleaved program command has been issued for an erasure-suspended area in erasure-suspended state.	1	0	0	0	0	0	0
	The value specified in the second cycle of a program command is not H'80.	1	0	0	0	0	0	0
	A command has been issued in command-locked state.	1	0/1	0/1	0/1	0/1 *1	0/1 *2	0/1
Erasure error	An error has occurred during erasure processing.	0	1	0	0	0	0	0
	A block erase command has been issued for the erasure block whose lock bit is set to 0 while the FPROTCN bit in FPROTR is 0.	0	1	0	0	0	0	0
Programming error	An error has occurred during programming processing.	0	0	1	0	0	0	0
	A program, lock bit program, or program command has been issued for the erasure block whose lock bit is set to 0 while the FPROTCN bit in the FPROTR register is 0.	0	0	1	0	0	0	0
FCU error	An error has occurred during CPU processing in the FCU.	0	0	0	1	0	0	0

Notes: 1. When the FRDCLE bit in FRAMECCR is set to 1.
2. When the FRCCLE bit in FRAMECCR is set to 1.

Table 24.16 Error Protection Types (2)

Error	Description	ILGLERR	ERSERR	PRGERR	FCUERR	FRDTCT	FRCRCT	ROMAE
FCU RAM ECC error	An ECC 1-bit error has been corrected during FCU RAM reading.	0	0	0	0	0	1	0
	An ECC 2-bit has been detected during FCU RAM reading.	0	0	0	0	1	0	0
ROM access error	A read access command has been issued to addresses H'8080 0000 to H'8087 FFFF while FENTRY0 = 1 in ROM P/E normal mode.	1	0	0	0	0	0	1
	An access command has been issued to addresses H'8080 0000 to H'8087 FFFF while FENTRY0 = 0	1	0	0	0	0	0	1
	A read access command has been issued to addresses H'0000 0000 to H'0007 FFFF while the FENTRYR register value is not H'0000.	1	0	0	0	0	0	1
	A ROM programming or erasing command (program, lock bit program, or block erase command) has been issued while the user boot MAT is selected.	1	0	0	0	0	0	1
	An access command has been issued to an address other than the addresses for ROM programming/erasure H'8080 0000 to H'8080 7FFF while the user boot MAT is selected.	1	0	0	0	0	0	1

24.9 Notes on ROM

24.9.1 Switching Between User MAT and User Boot MAT

The user MAT and user boot MAT are allocated to the same address area. If the ROM area is accessed during switching between the user MAT and user boot MAT, an unexpected MAT may be accessed because the number of cycles required to access the ROM area depends on the internal bus status. When the ROM cache function is enabled, the previously stored data is left in the ROM cache even after MAT switching; note that a cache hit may occur when a newly selected MAT is accessed at the same address as the data stored in the cache. To avoid such unexpected behavior, take the following steps before and after MAT switching.

1. Modifying interrupt settings before MAT switching
   There are two ways to avoid ROM area access due to an interrupt during MAT switching: one is to specify the interrupt vector fetch destination outside the ROM area through the vector base register (VBR) setting in the CPU, and the other is to mask interrupts. Note that NMI interrupts cannot be masked in this LSI; when masking interrupts to avoid ROM area access in this LSI, design the system so that no NMI is generated during MAT switching.
2. Switching between MATs through a program outside the ROM area
   To avoid CPU instruction fetch in the ROM area during MAT switching, execute the MAT switching processing outside the ROM area.
3. Switching between MATs
   Write to the ROMMAT register to switch between MATs.
4. Flushing the ROM cache after MAT switching
   Flush all lines of the ROM cache by writing 1 to the RCF bit in the RCCR register (refer to section 26, ROM Cache (ROMC)).
5. Executing a ROMMAT register read instruction (dummy read)
   Execute a ROMMAT register read instruction (dummy read) to complete register rewrite (step 3).
6. Executing five or more NOP instructions
   Execute five or more NOP instructions after executing a ROMMAT register read instruction.

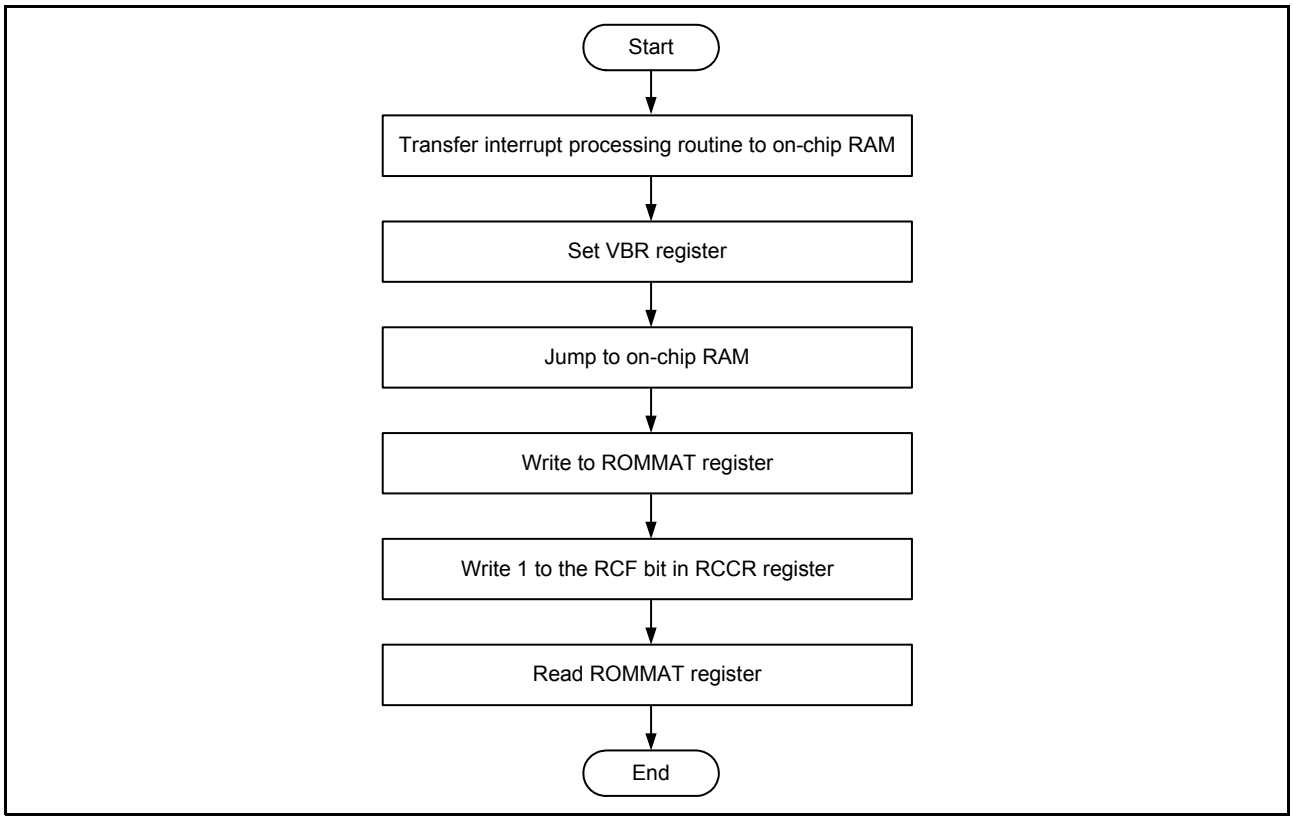


Figure 24.32 Example of MAT Switching Steps

24.9.2 Other Notes

(1) State in which AUD Operation Is Disabled and Interrupts Are Ignored

In the following mode or period, the AUD is in module standby mode and cannot operate. The NMI or maskable interrupt requests are ignored.

- Boot mode
- The program in the embedded program stored MAT is being executed immediately after the LSI is started in user boot mode

(2) Key Code Stored Area

Addresses H'0000 0050 to H'0000 005F in the user MAT store the key code for debugging function authentication to be used with the on-chip debugger. To restrict the debugging functions, write a key code in this area. After a key code is specified through the debugger, the code is stored in this area, which should be noted during checksum verification.

(3) Programming-/Erasure-Suspended Area

The data stored in the programming-suspended or erasure-suspended area is undetermined. To avoid malfunction due to undefined read data, ensure that no instruction is executed or no data is read from the programming-suspended or erasure-suspended area.

To avoid instruction fetch from the programming-suspended or erasure-suspended area, which may be caused by prefetch by the ROM cache, ensure that no instruction is fetched from within the 32 bytes immediately before the start address of the programming-suspended or erasure-suspended area.

During ROM cache prefetch, the destination of a branch instruction is also accessed. If the destination can be in the programming-suspended or erasure-suspended area, disable the prefetch function of the ROM cache.

(4) Compatibility with Programming/Erasing Program of Conventional F-ZTAT SH Microcomputers

The flash memory programming/erasing program used for conventional F-ZTAT SH microcontrollers does not work with this LSI.

(5) Reset during Programming or Erasure

During programming or erasure, do not generate the following resets: the hardware reset and the watchdog timer reset.

When a hardware reset is generated by asserting the RESET# pin during programming or erasure of the flash memory, hold the reset state for a period of t_{RESW2} (refer to section 30, Electrical Characteristics). In a hardware reset, not only does the voltage applied to the memory unit have to drop, but the power supply for the EEPROM and its internal circuitry also have to be initialized. Thus, the reset state must be maintained over a longer period than in the case of resetting the FCU.

To reset the FCU by setting the FRESET bit in the FRESETR register during programming or erasure, hold the FCU in the reset state for a period of t_{RESW2} (refer to section 30, Electrical Characteristics). Since a high voltage is applied to the ROM during programming and erasure, the FCU has to be held in the reset state long enough to ensure that the voltage applied to the memory unit has dropped. Do not read from the EEPROM while the FCU is in the reset state.

(6) Prohibition of Additional Programming

One area cannot be programmed twice in succession. To program an area that has already been programmed, be sure to erase the area before reprogramming.

(7) SLEEP Instructions in On-Board Programming Modes

Do not issue a SLEEP instruction while the ROM is in an on-board programming mode.

(8) Suspend by a Programming or Erasure Command

When the programming or erasure is suspended by a programming or erasure suspend command, be sure to resume the processing by issuing a resume command and complete it.

(9) Timing to Issue P/E Suspend Command during Erasure

During erasure, issue a P/E suspend command 1.7 ms or more after a P/E resume command is issued. There is no restriction on time from when an erasure command is issued to when the first P/E suspend command is issued.

25. EEPROM

25.1 Introduction

This LSI includes 32 Kbytes of flash memory (EEPROM) for storing data.
Table 25.1 list the EEPROM Specifications

Table 25.1 EEPROM Specifications

Item	Description
Flash-memory MATs	Data MAT: 32 Kbytes Product information MAT: 128 bytes
Read	Both the data MAT and Product information MAT can be read through the peripheral bus A. Byte access : 5 cycles of the CPU clock Word access : 9 cycles of the CPU clock
Programing and erasing methods	The data MAT can be programmed and erased by commands issued through the peripheral bus A to the ROM/EEPROM-dedicated sequencer (FCU).
Programing and erasing unit	Programming unit: 8 bytes or 128 bytes (User mode, User program mode, User boot mode) 256 bytes (boot mode) Erasing unit: block units (2 Kbytes)
Blank check function	Checking of the Erased State
On-board programming modes	Three types (boot mode, user program mode, user boot mode)
Protection mode	Software protection
Programming and erasing time and count	Refer to section 30, Electrical Characteristics

- Flash-memory MATs

The EEPROM has two types of memory areas (hereafter referred to as memory MATs) in the same address space. These two MATs can be switched by bank switching through the control register. For addresses H'8010 0080 to H'8010 7FFF, the data MAT contents will always be read even when the product information MAT is selected. The product information MAT cannot be programmed or erased.
Data MAT: 32 Kbytes
Product information MAT: 128 bytes

Figure 25.1 shows the Memory MAT Configuration in EEPROM.

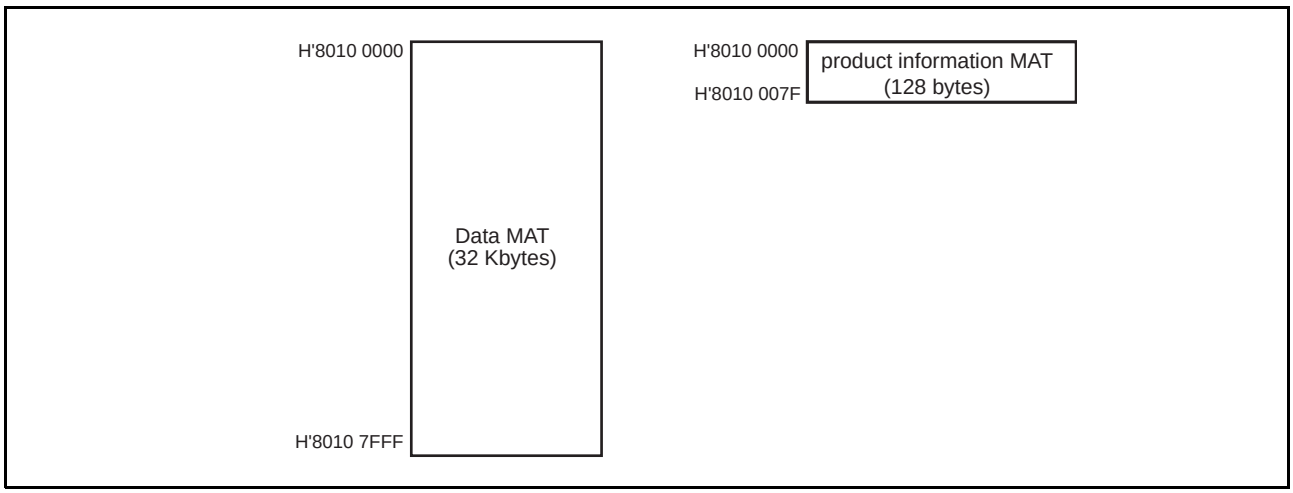


Figure 25.1 Memory MAT Configuration in EEPROM

- Reading through the peripheral bus A

Both the data MAT and product information MAT can be read through the peripheral bus A in five CPU clock cycles in bytes and in nine CPU clock cycles in words.

- Programming and erasing methods

The data MAT can be programmed and erased by commands issued through the peripheral bus A to the ROM/EEPROM-dedicated sequencer (FCU).
While the FCU is programming or erasing the data MAT, the CPU can execute a program located in the ROM, RAM, or external address space. While the FCU is programming or erasing the ROM or data MAT, data cannot be read form the data MAT. When the FCU suspends programming or erasure, the CPU can read data from the data MAT, and then the FCU can resume programming or erasure of the data MAT. While the FCU suspends erasure, areas other than the erasure-suspended area can be programmed.

- Programming/erasing unit

The data MAT is programmed in 8-byte or 128-byte units and erased in block units (2 Kbytes) in user mode, user program mode, and user boot mode. In boot mode, the data MAT is programmed in 256-byte units and erased in block units (2 Kbytes). The product information MAT is read-only memory and cannot be programmed or erased.
Figure 25.2 shows the Block Configuration of Data MAT of this LSI. The data MAT is divided into sixteen 2-Kbyte blocks (DB15 to DB30).

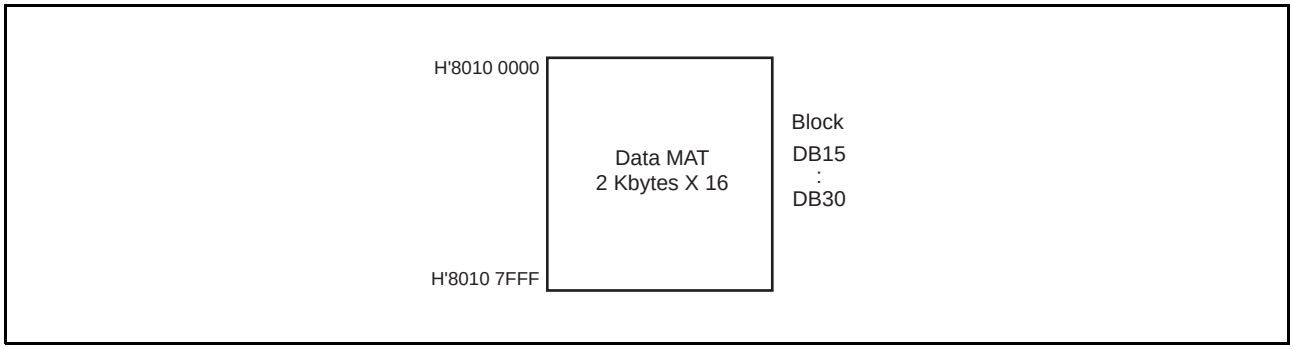


Figure 25.2 Block Configuration of Data MAT

- Blank check function

If data is read from erased EEPROM by the CPU, undefined values are read. Using blank check command of the FCU allows checking of whether the EEPROM is erased (in a blank state). Either a 2 Kbytes (1 erasure block) or 8 bytes of area can be checked by a single execution of the blank check command.

- Three types of on-board programming modes

[Boot mode]
The data MAT can be programmed using the SCI. The bit rate for SCI communications between the host and the LSI can be automatically adjusted.
[User mode/user program mode]
The data MAT can be programmed with a desired interface. The user mode includes MCU single-chip mode.
[User boot mode]
The data MAT can be programmed with a desired interface. To make a transition to this mode, a reset is needed.

- Protection modes

This LSI supports software protection mode to protect memory against programming, erasing, or reading by the setting of the FENTRYD bit in the Flash P/E Mode entry register (FENTRYR), EEPRE0/1 register, or EEPWE0/1

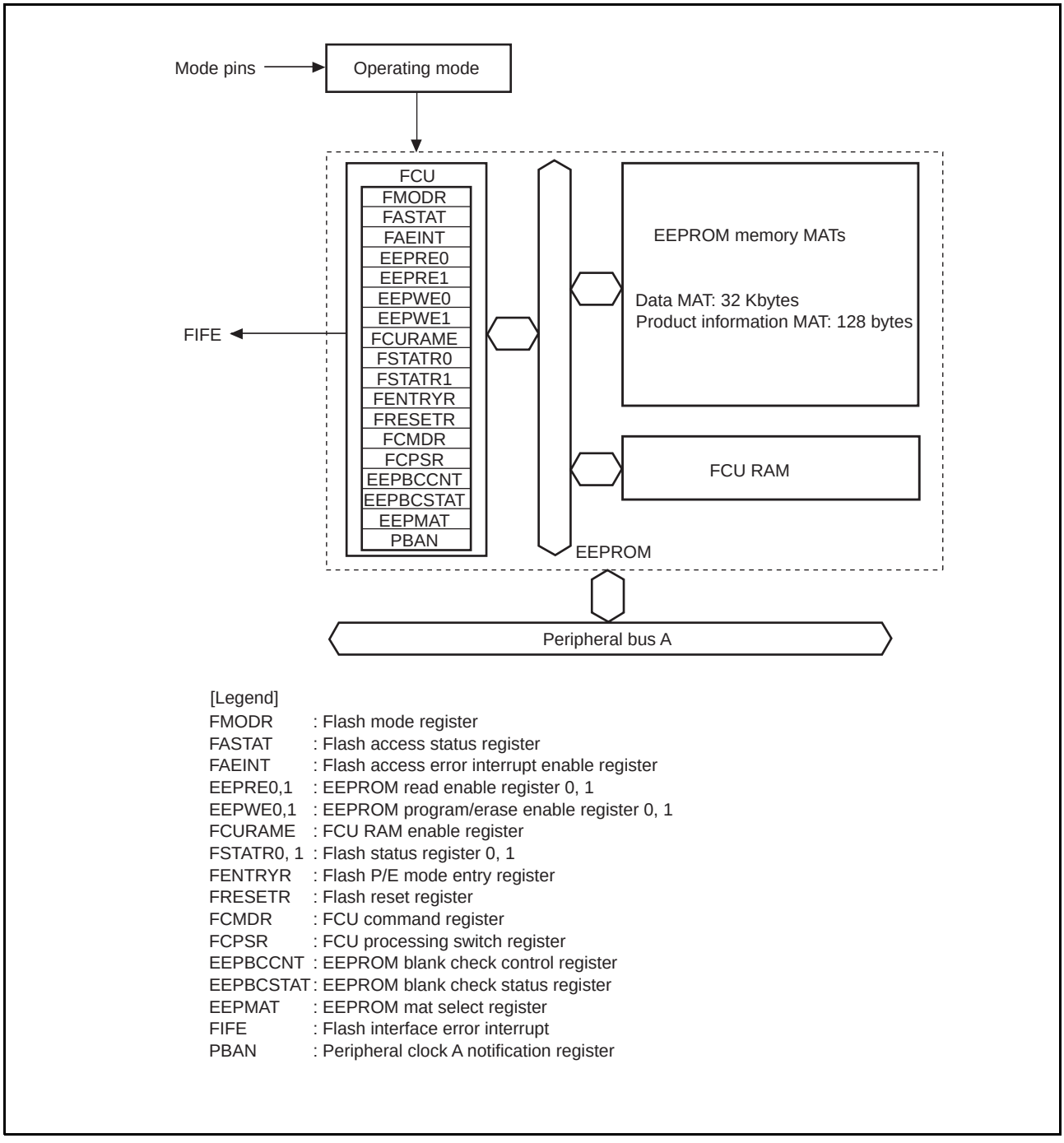
register. The FENTRYD bit enables or disables data MAT programming or erasure by the FCU. The EEPRE0/1 register control protection of each data MAT block against reading, and the EEPWE0/1 register control protection against programming and erasure.

The LSI also provides a function to suspend programming or erasure when abnormal operation is detected during programming or erasure. In addition, the LSI provides a function to protect the EEPROM against instruction fetch attempted by the CPU.

- Programming and erasing time and count

Refer to section 30, Electrical Characteristics

Figure 25.3 shows the Block Diagram of EEPROM.



[Legend]

FMODR	: Flash mode register
FASTAT	: Flash access status register
FAEINT	: Flash access error interrupt enable register
EEPRE0,1	: EEPROM read enable register 0, 1
EEPWE0,1	: EEPROM program/erase enable register 0, 1
FCURAME	: FCU RAM enable register
FSTATR0, 1	: Flash status register 0, 1
FENTRYR	: Flash P/E mode entry register
FRESETR	: Flash reset register
FCMDR	: FCU command register
FCPSR	: FCU processing switch register
EEPBCCNT	: EEPROM blank check control register
EEPBCSTAT	: EEPROM blank check status register
EEPMAT	: EEPROM mat select register
FIFE	: Flash interface error interrupt
PBAN	: Peripheral clock A notification register

Figure 25.3 Block Diagram of EEPROM

25.2 Input/Output Pins

Table 25.2 shows the input/output pins used for the EEPROM. The combination of the MD0 to MD1 pin levels, and the ASEMD pin levels determines the EEPROM programming mode (refer to section 25.4, Overview of EEPROM-Related Modes). In boot mode, programming and erasing the EEPROM can be performed by the host via the pins RxD1 and TxD1 (refer to section 25.5, Boot Mode).

Table 25.2 Pin Configuration

Pin Name	I/O	Function
RESET#	Input	This LSI enters the power-on reset state when this signal goes low.
MD0 to MD1, ASEMD	Input	These pins specify the operating mode.
RXD1	Input	Receives data through SCI1 (communications with host)
TXD1	Output	Transmits data through SCI1 (communications with host)

25.3 Registers

Table 25.3 list the EEPROM Registers. Some of these register have ROM-related bits. The EEPROM-related register are initialize by a reset.

Table 25.3 EEPROM Registers

Register Name	Symbol	After Reset	Address	Access Size
Flash mode register	FMODR	H'00	H'FFFF A802	8
Flash access status register	FASTAT	H'00	H'FFFF A810	8
Flash access error interrupt enable register	FAEINT	H'9F	H'FFFF A811	8
EEPROM read enable register 0	EEPRE0	H'0000	H'FFFF A840	8*2, 16
EEPROM read enable register 1	EEPRE1	H'0000	H'FFFF A842	8*2, 16
EEPROM program/erase enable register 0	EEPWE0	H'0000	H'FFFF A850	8*2, 16
EEPROM program/erase enable register 1	EEPWE1	H'0000	H'FFFF A852	8*2, 16
FCU RAM enable register	FCURAME	H'0000	H'FFFF A854	8*2, 16
Flash status register 0	FSTATR0	H'80 *1	H'FFFF A900	8, 16
Flash status register 1	FSTATR1	H'00 *1	H'FFFF A901	8, 16
Flash P/E mode entry register	FENTRYR	H'0000 *1	H'FFFF A902	8*2, 16
Flash reset register	FRESETR	H'0000	H'FFFF A906	8*2, 16
FCU command register	FCMDR	H'FFFF *1	H'FFFF A90A	8, 16
FCU processing switch register	FCPSR	H'0000 *1	H'FFFF A918	8, 16
EEPROM blank check control register	EEPBCCNT	H'0000 *1	H'FFFF A91A	8, 16
EEPROM blank check status register	EEPBCSTAT	H'0000 *1	H'FFFF A91E	8, 16
Peripheral clock A notification register	PBAN	H'00 *1	H'FFFF A938	8, 16
EEPROM mat select register	EEPMAT	H'0000	H'FFFF AB00	8*2, 16

Notes: 1. This register can be initialized by a reset, or by setting the FRESET bit in the FRESETR register to 1.
2. The 8-bit unit of access is only for reading this register.

25.3.1 Flash Mode Register (FMODR)

Address H'FFFF A802

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	FRDMD	—	—	—	—
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R
b4	FRDMD	FCU Read Mode Select Bit	Selects the read mode to read the ROM or EEPROM using FCU. This bit specifies the EEPROM lock bit read mode transition or blank check processings in the EEPROM (refer to section 25.6.1, FCU Command List, section 25.6.3, FCU Command Usage), whereas this bit must be set to specify the read method for the lock bits in the ROM (refer to section 24, ROM). 0: Memory area read mode This mode is selected to enter the EEPROM lock bit read mode. Since the EEPROM has no lock bits, reading an EEPROM area results in an undefined value. 1: Register read mode To make the blank check command available for use, register read mode is set.	R/W
b3 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R

The FMODR register specifies the FCU operation mode.

25.3.2 Flash Access Status Register (FASTAT)

Address H'FFFF A810

	b7	b6	b5	b4	b3	b2	b1	b0
	ROMAE	—	—	CMDLK	EEPAE	EEPIFE	EEPRPE	EEPWPEIE
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	ROMAE	ROM Access Error Bit	Refer to section 24, ROM.	R/(W) *
b6 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R
b4	CMDLK	FCU Command Lock Bit	Indicates whether the FCU is in command-locked state (refer to section 25.7.2, Error Protection). 0: The FCU is not in command-locked state 1: The FCU is in command-locked state [Setting condition] The FCU detects an error and enters command-locked state. [Clearing condition] The FCU completes the status-clear command processing.	R
b3	EEPAE	EEPROM Access Error Bit	Indicates whether an access error has been generated for the EEPROM. If this bit becomes 1, the ILGLERR bit in the FSTATR0 register is set to 1 and the FCU enters command-locked state. 0: No EEPROM access error has occurred 1: An EEPROM access error has occurred [Setting conditions] • A read access command is issued to the EEPROM area while the FENTRYD bit in the FENTRYR register is 1 in EEPROM P/E normal mode. • A write access command is issued to the EEPROM area while the FENTRYD bit in the FENTRYR register is 0. • An access command is issued to the EEPROM area while one of the FENTRY1 and FENTRY0 bits in the FENTRYR register is 1. [Clearing condition] • 0 is written to this bit after reading EEPAE = 1.	R/(W) *
b2	EEPIFE	EEPROM Instruction Fetch Error Bit	Indicates whether an instruction fetch error has been generated for the EEPROM. 0: No EEPROM instruction fetch error has occurred 1: An EEPROM instruction fetch error has occurred [Setting condition] • An attempt is made to fetch an instruction from the EEPROM. [Clearing condition] • 0 is written to this bit after reading EEPIFE = 1.	R/(W) *
b1	EEPRPE	EEPROM Read Protect Error Bit	Indicates whether an error has been generated against the EEPROM read protection provided by the EEPRE0/1 register settings. 0: The EEPROM has not been read against the EEPRE0/1 register settings 1: An attempt has been made to read data from the EEPROM against the EEPRE0/1 register setting [Setting condition] • An attempt is made to read data from the EEPROM area that has been read-protected through the EEPRE0/1 register setting. [Clearing condition] • 0 is written to this bit after reading EEPRPE = 1.	R/(W) *

Bit	Symbol	Bit Name	Description	R/W
b0	EEPWPE	EEPROM Program/Erase Protect Error Bit	Indicates whether an error has been generated against the EEPROM program/erasure protection provided by the EEPWE0/1 register setting. 0: No programming or erasing command has been issued to the EEPROM against the EEPWE0/1 register setting 1: A programming or erasing command has been issued to the EEPROM against the EEPWE0/1 register setting [Setting condition] • A programming or erasing command is issued to the EEPROM area that has been program/erase-protected through the EEPWE0/1 register setting. [Clearing condition] • 0 is written to this bit after reading EEPWPE = 1.	R/(W) *

Note: * Only 0 can be written to clear the flag after 1 is read.

The FASTAT register indicates the access error status for the ROM and EEPROM. If any bit in the FASTAT register is set to 1, the FCU enters command-locked state (refer to section 25.7.2, Error Protection). To cancel command-locked state, set the FASTAT register to H'10, and then issue a status-clear command to the FCU.

25.3.3 Flash Access Error Interrupt Enable Register (FAEINT)

Address H'FFFF A811

	b7	b6	b5	b4	b3	b2	b1	b0
	ROMAEIE	—	—	CMDLKIE	EEPAEIE	EEPIFEIE	EEPRPEIE	EEPWPEIE
After Reset	1	0	0	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b7	ROMAEIE	ROM Access Error Interrupt Enable Bit	Refer to section 24, ROM.	R/W
b6, b5	—	Reserved	These bits are read as 0. The write value should be 0.	R
b4	CMDLKIE	FCU Command Lock Interrupt Enable	Enables or disables an FIFE interrupt request when FCU command-locked state is entered and the CMDLK bit in the FASTAT register becomes 1. 0: Does not generate an FIFE interrupt request when CMDLK = 1 1: Generates an FIFE interrupt request when CMDLK = 1	R/W
b3	EEPAEIE	EEPROM Access Error Interrupt Enable	Enables or disables an FIFE interrupt request when an EEPROM access error occurs and the EEPAE bit in FASTAT becomes 1. 0: Does not generate an FIFE interrupt request when EEPAE = 1 1: Generates an FIFE interrupt request when EEPAE = 1	R/W
b2	EEPIFEIE	EEPROM Instruction Fetch Error Interrupt Enable	Enables or disables an FIFE interrupt request when an EEPROM instruction fetch error occurs and the EEPIFE bit in the FASTAT register becomes 1. 0: Does not generate an FIFE interrupt request when EEPIFE = 1 1: Generates an FIFE interrupt request when EEPIFE = 1	R/W
b1	EEPRPEIE	EEPROM Read Protect Error Interrupt Enable	Enables or disables an FIFE interrupt request when an EEPROM read protect error occurs and the EEPRPE bit in the FASTAT register becomes 1. 0: Does not generate an FIFE interrupt request when EEPRPE = 1 1: Generates an FIFE interrupt request when EEPRPE = 1	R/W
b0	EEPWPEIE	EEPROM Program/Erase Protect Error Interrupt Enable	Enables or disables an FIFE interrupt request when an EEPROM program/erase protect error occurs and the EEPWPE bit in the FASTAT register becomes 1. 0: Does not generate an FIFE interrupt request when EEPWPE = 1 1: Generates an FIFE interrupt request when EEPWPE = 1	R/W

Note: After writing to the FAEINT register, execute a FAEINT register read instruction, and five or more NOP instructions.

The FAEINT register enables or disables output of flash interface error (FIFE) interrupt requests.

25.3.4 EEPROM Read Enable Register 0 (EEPRE0)

Address H'FFFF A840

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	KEY[7:0]								DBREi							
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 b8	KEY[7:0]	Key Code Bits	These bits enable or disable DBREi bit modification. The data written to these bits are not stored.	R/(W) *
b7 to b0	DBREi	DB22 To DB15 Block Read Enable Bits	Enables or disables read access to blocks DB22 to DB15 in the data MAT. The DBREi bit controls read access to block DBi. Writing to these bits is enabled only when this register is accessed in word size and H'2D is written to the KEY bits. 0: Disables read access 1: Enables read access	R/W

Notes: * Write data is not retained.
• i = 07 to 00

The EEPRE0 register enables or disables read access to blocks DB22 to DB15 (refer to Figure 25.2 Block Configuration of Data MAT) in the data MAT.
Writing to the upper byte is enabled only when this register is accessed in word size and specific value is written. The data written to the upper byte is not retained in the register.

25.3.5 EEPROM Read Enable Register 1 (EEPRE1)

Address H'FFFF A842

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	KEY[7:0]								DBREi							
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 b8	KEY[7:0]	Key Code Bits	These bits enable or disable DBREi bit modification. The data written to these bits are not stored.	R/(W) *
b7 to b0	DBREi	DB30 To DB23 Block Read Enable Bits	Enables or disables read access to blocks DB30 to DB23 in the data MAT. The DBREi bit controls read access to block DBi. Writing to these bits is enabled only when this register is accessed in word size and H'D2 is written to the KEY bits. 0: Disables read access 1: Enables read access	R/W

Notes: * Write data is not retained.
• i = 15 to 08

The EEPRE1 register enables or disables read access to blocks DB30 to DB23 (refer to Figure 25.2 Block Configuration of Data MAT) in the data MAT.
Writing to the upper byte is enabled only when this register is accessed in word size and specific value is written. The data written to the upper byte is not retained in the register.

25.3.6 EEPROM Program/Erase Enable Register 0 (EEPWE0)

Address H'FFFF A850

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	KEY[7:0]								DBWEi							
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 b8	KEY[7:0]	Key Code Bits	These bits enable or disable DBWEi bit modification. The data written to these bits are not stored.	R/(W) *
b7 to b0	DBWEi	DB22 To DB15 Block Program/ Erase Enable	Enables or disables programming and erasure of blocks DB22 to DB15 in the data MAT. The DBWEi bit controls programming and erasure of block DBi. Writing to these bits is enabled only when this register is accessed in word size and H'1E is written to the KEY bits. 0: Disables programming and erasure 1: Enables programming and erasure	R/W

Notes: * Write data is not retained.
• i = 07 to 00

The EEPWE0 register enables or disables programming and erasure of blocks DB22 to DB15 (refer to Figure 25.2 Block Configuration of Data MAT) in the data MAT.
Writing to the upper byte is enabled only when this register is accessed in word size and specific value is written. The data written to the upper byte is not retained in the register.

25.3.7 EEPROM Program/Erase Enable Register 1 (EEPWE1)

Address H'FFFF A852

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	KEY[7:0]								DBWEi							
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 b8	KEY[7:0]	Key Code Bits	These bits enable or disable DBWEi bit modification. The data written to these bits are not stored.	R/(W) *
b7 to b0	DBWEi	DB30 To DB23 Block Program/ Erase Enable	Enables or disables programming and erasure of blocks DB30 to DB23 in the data MAT. The DBWEi bit controls programming and erasure of block DBi. Writing to these bits is enabled only when this register is accessed in word size and H'E1 is written to the KEY bits. 0: Disables programming and erasure 1: Enables programming and erasure	R/W

Notes: * Write data is not retained.
• i = 15 to 08

The EEPWE1 register enables or disables programming and erasure of blocks DB30 to DB23 (refer to Figure 25.2 Block Configuration of Data MAT) in the data MAT.
Writing to the upper byte is enabled only when this register is accessed in word size and specific value is written. The data written to the upper byte is not retained in the register.

25.3.8 FCU RAM Enable Register (FCURAME)

Address H'FFFF A854

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	KEY[7:0]								—	—	—	—	—	—	—	FCRME
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 b8	KEY[7:0]	Key Code Bits	These bits enable or disable FCRME bit modification. The data written to these bits are not stored.	R/(W) *
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R
b0	FCRME	FCU RAM Enable Bit	Enables or disables access to the FCU RAM. Writing to this bit is enabled only when this register is accessed in word size and H'C4 is written to the KEY bits. Before writing to the FCU RAM, clear FENTRYR to H'0000 to stop the FCU. 0: Disables access to FCU RAM 1: Enables access to FCU RAM	R/W

Note: * Write data is not retained.

The FCURAME register enables or disables access to the FCU RAM area.
Writing to the upper byte is enabled only when this register is accessed in word size and specific value is written. The data written to the upper byte is not retained in the register.

25.3.9 Flash Status Register 0 (FSTATR0)

Address H'FFFF A900

	b7	b6	b5	b4	b3	b2	b1	b0
	FRDY	ILGLERR	ERSERR	PRGERR	SUSRDY	—	ERSSPD	PRGSPD
After Reset	1	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	FRDY	Flash Ready Bit	Indicates the processing state in the FCU. 0: Programming or erasure processing, programming or erasure suspension processing, lock bit read 2 command processing, or EEPROM blank check is in progress. 1: None of the above is in progress.	R
b6	ILGLERR	Illegal Command Error Bit	Indicates that the FCU has detected an illegal command or illegal ROM or EEPROM access. When this bit is 1, the FCU is in command-locked state (refer to section 25.7.2, Error Protection). 0: The FCU has not detected any illegal command or illegal ROM/EEPROM access 1: The FCU has detected an illegal command or illegal ROM/EEPROM access [Setting conditions] • The FCU has detected an illegal command. • The FCU has detected an illegal ROM/EEPROM access (the ROMAE, EEPAE, EEPIFE, EEPRPE, or EEPWPE bit in FASTAT register is 1). • The FENTRYR setting is illegal. [Clearing condition] • The FCU completes the status-clear command processing while FASTAT register is H'10.	R
b5	ERSERR	Erasure Error Bit	Indicates the result of ROM or EEPROM erasure by the FCU. When this bit is 1, the FCU is in command-locked state (refer to section 25.7.2, Error Protection). 0: Erasure processing has been completed successfully 1: An error has occurred during erasure [Setting conditions] • An error has occurred during erasure. • A block erase command has been issued for the area protected by a lock bit. [Clearing condition] • The FCU completes the status-clear command processing.	R
b4	PRGERR	Programming Error Bit	Indicates the result of ROM or EEPROM programming by the FCU. When this bit is 1, the FCU is in command-locked state (refer to section 25.7.2, Error Protection). 0: Programming has been completed successfully 1: An error has occurred during programming [Setting conditions] • An error has occurred during programming. • A programming command has been issued for the area protected by a lock bit. [Clearing condition] • The FCU completes the status-clear command processing.	R
b3	SUSRDY	Suspend Ready Bit	Indicates whether the FCU is ready to accept a P/E suspend command. 0: The FCU cannot accept a P/E suspend command 1: The FCU can accept a P/E suspend command [Setting condition] • After initiating programming/erasure, the FCU has entered a state where it is ready to accept a P/E suspend command. [Clearing conditions] • The FCU has accepted a P/E suspend command. • The FCU has entered a command-locked state during programming or erasure.	R
b2	—	Reserved	This bit is read as 0. The write value should be 0.	R

Bit	Symbol	Bit Name	Description	R/W
b1	ERSSPD	Erasure-Suspended Status Bit	Indicates that the FCU has entered an erasure suspension process or an erasure-suspended status (refer to section 24, ROM). 0: The FCU is in a status other than the below-mentioned. 1: The FCU is in an erasure suspension process or an erasure-suspended status. [Setting condition] • The FCU has initiated an erasure suspend command. [Clearing condition] • The FCU has accepted a resume command.	R
b0	PRGSPD	Programming-Suspended Status Bit	Indicates that the FCU has entered a programming suspension process or a programming suspend status (refer to section 24, ROM). 0: The FCU is in a status other than the below-mentioned. 1: The FCU is in a write suspension process or a write-suspended status. [Setting condition] • The FCU has initiated a write suspend command. [Clearing condition] • The FCU has accepted a resume command.	R

The FSTATR0 register indicates the FCU status.

The FRTATR0 register is initialized by a reset, or by setting the FRESET bit of the FRESETR register is set to 1.

25.3.10 Flash Status Register 1 (FSTATR1)

Address H'FFFF A901

	b7	b6	b5	b4	b3	b2	b1	b0
	FCUERR	—	—	FLOCKST	—	—	FRDTCT	FRCRCT
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7	FCUERR	FCU Error Bit	Indicates an error has occurred during the CPU processing in the FCU. 0: No error has occurred during the CPU processing in the FCU 1: An error has occurred during the CPU processing in the FCU [Clearing condition] • The FRESET bit in FRESETR register is set to 1. When the FCUERR bit is 1, set the FRESET bit to 1 to initialize the FCU, and then copy the FCU firmware again from the FCU firmware area to the FCU RAM area.	R
b6, b5	—	Reserved	These bits are read as 0. The write value should be 0.	R
b4	FLOCKST	Lock Bit Status Bit	Reflects the lock bit data read through lock bit read 2 command execution. When the FRDY bit becomes 1 after the lock bit read 2 command is issued, valid data is stored in this bit. This bit value is retained until the next lock bit read 2 command is completed. 0: Protected state 1: Non-protected state	R
b3, b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b1	FRDTCT	FCU RAM ECC 2-Bit Error Detection Monitoring Bit	Indicates that a 2-bit error has been detected when the FCU is reading RAM. 0: No 2-bit error has been detected. 1: A 2-bit error has been detected. When the FRDTCT bit is 1, set the FRESET bit to 1 to initialize the FCU, and then copy the FCU firmware again from the FCU firmware area to the FCU RAM area.	R
b0	FRCRCT	FCU RAM ECC 1-Bit Error Correction Monitoring Bit	Indicates that a 1-bit error has been corrected when the FCU is reading RAM. 0: No 1-bit error has been corrected. 1: A 1-bit error has been corrected. When the FRCRCT bit is 1, set the FRESET bit to 1 to initialize the FCU, and then copy the FCU firmware again from the FCU firmware area to the FCU RAM area.	R

The FSTATR1 register indicates the FCU status.

The FSTATR1 register is initialized by a reset, or by setting the FRESET bit in the FRESETR register to 1.

25.3.11 Flash P/E Mode Entry Register (FENTRYR)

Address H'FFFF A902

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	FEKEY[7:0]								FENTRYD	—	—	—	—	—	—	FENTRY0
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 b8	FEKEY[7:0]	Key Code Bits	These bits enable or disable rewriting of the FENTRYD and FENTRY0 bits. Write data to these bits are not retained.	R/(W) *
b7	FENTRYD	EEPROM P/E Mode Entry Bit	Specify the P/E mode for the EEPROM. 0: The EEPROM is in read mode 1: The EEPROM is in P/E mode [Write enabling conditions] When the following conditions are all satisfied: • The FRDY bit in the FSTATR0 register is 1. • H'AA is written to FEKEY in word access. [Setting condition] • 1 is written to FENTRYD while the write enabling conditions are satisfied and FENTRYR register is H'0000. [Clearing conditions] • This register is written to in byte access. • A value other than H'AA is written to the FEKEY in word access. • 0 is written to the FENTRYD while the write enabling conditions are satisfied. • FENTRYR register is written to while FENTRYR register is not H'0000 and the write enabling conditions are satisfied.	R/W
b6 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R
b0	FENTRY0	ROM P/E Mode Entry Bit 0 Bit	Refer to section 24, ROM.	R/W

Notes: After writing to the FENTRYR register, execute a FENTRYR register read instruction, and five or more NOP instructions.
* Write data is not retained.

The FENTRYR register specifies the P/E mode for the ROM or EEPROM. To specify the P/E mode for the ROM or EEPROM so that the FCU can accept commands, set either of the FENTRYD and FENTRY0 bits to 1.
The FENTRYR register can be initialized by a reset, or by setting the FRESET bit in the FRESETR register to 1.

Note: * This register can be written to only when a specified value is written to the upper byte in word access; the register is initialized when a value not allowed for the register is written to the upper byte. The data written to the upper byte is not stored in the register.

25.3.12 Flash Reset Register (FRESETR)

Address H'FFFF A906

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	FRKEY[7:0]								—	—	—	—	—	—	—	FRESET
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 b8	FRKEY[7:0]	Key Code Bits	These bits enable or disable FRESET bit modification. The data written to these bits are not stored.	R/(W) *
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R
b0	FRESET	Flash Reset Bit	Setting this bit to 1 forcibly terminates programming/erasure of ROM or EEPROM and initializes the FCU. A high voltage is applied to the ROM/EEPROM memory units during programming and erasure. To ensure sufficient time for the voltage applied to the memory unit to drop, keep the value of the FRESET bit at 1 for a period of t_{RESW2} (refer to section 30, Electrical Characteristics) when the FCU is initialized. Do not read from the ROM/EEPROM units while the value of the FRESET bit is kept at 1. The FCU commands are unavailable for use while the FRESET bit is set to 1, since this initializes the FENTRYR register. This bit can be written only when H'CC is written to the FRKEY bits in word access. 0: Issue no reset to the FCU. 1: Issues a reset to the FCU.	R/W

Note: * Write data is not retained.

The FRESETR register is used for the initialization of FCU.

Writing to the upper byte is enabled only when this register is accessed in word size and specific value is written. The data written to the upper byte is not retained in the register.

25.3.13 FCU Command Register (FCMDR)

Address H'FFFF A90A

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	CMDR[7:0]								PCMDR[7:0]							
After Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b15 to b8	CMDR[7:0]	Command Register Bits	These bits store the latest command accepted by the FCU.	R
b7 to b0	PCMDR[7:0]	Precommand Register Bits	These bits store the previous command accepted by the FCU.	R

The FCMDR register stores the commands that the FCU has accepted.
The FCMDR register is initialized by a reset, or by setting the FRESET bit in FRESETR register to 1.

Table 25.4 shows the States of FCMDR after acceptance of the various commands. For details on the blank check, Refer to section 25.4, Overview of EEPROM-Related Modes.

Table 25.4 FCMDR Status after a Command is Accepted

Command	CMDR	PCMDR
Normal mode transition	H'FF	Previous command
Status read mode transition	H'70	Previous command
Lock bit read mode transition (lock bit read 1)	H'71	Previous command
Program	H'E8	Previous command
Block erase	H'D0	H'20
P/E suspend	H'B0	Previous command
P/E resume	H'D0	Previous command
Status register clear	H'50	Previous command
Lock bit read 2 blank check	H'D0	H'71
Lock bit program	H'D0	H'77

25.3.14 FCU Processing Switch Register (FCPSR)

Address H'FFFF A918

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ESUSPMD
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 b1	—	Reserved	These bits are read as 0. The write value should be 0.	R
b0	ESUSPMD	Erasure-Suspended Mode Bit	Erasure-suspended mode This LSI does not use this bit. The write value should always be 0.	R/W

The FCPSR register selects a function to make the FCU suspend erasure.

The FCPSR register is initialized by a reset, or by setting the FRESET bit in FRESETR register to 1.

25.3.15 EEPROM Blank Check Control Register (EEPBCCNT)

Address H'FFFF A91A

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	BCADR [10:0]											—	—	BCSIZE
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, b14	—	Reserved	These bits are read as 0. The write value should be 0.	R
b13 to b3	BCADR [10:0]	Blank Check Address Setting Bit	Use these bits to specify the address of the target area when the size of the target area to be checked by the blank check command is 8 bytes (the BCSIZE bit is set to 0). When the BCSIZE bit is set to 0, the start address of the target area is the value obtained by summing the EEPBCCNT value (the value obtained by shifting the set BCADR value by 3 bits) and the start address of an erased block specified when a blank check command is issued.	R/W
b2, b1	—	Reserved	These bits are read as 0. The write value should be 0.	R
b0	BCSIZE	Blank Check Size Setting Bit	This bit selects the size of the target area to be checked by the blank check command. 0: Selects 8 bytes as the size of a blank check target area. 1: Selects 2 Kbytes as the size of a blank check target area.	R/W

The EEPBCCNT register specifies the addresses and sizes of the target areas to be checked by the blank check command.

The EEPBCCNT register is initialized by a reset, or by setting the FRESET bit in the FRESETR to 1.

25.3.16 EEPROM Blank Check Status Register (EEPBCSTAT)

Address H'FFFF A91E

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	BCST
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 b1	—	Reserved	These bits are read as 0. The write value should be 0.	R
b0	BCST	Blank Check Status Bit	Indicates the result of a blank check. 0: The target area is erased (blank). 1: The target area is filled with 0s and/or 1s.	R

The EEPBCSTAT register stores check results by executing the blank check command.
The EEPBCSTAT register is initialized by a reset, or by setting the FRESET bit in the FRESETR register to 1.

25.3.17 Peripheral Clock A Notification Register (PBAN)

Address H'FFFF A938

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	PBAN [7:0]							
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 b8	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b7 to b0	PBAN [7:0]	Peripheral clock A notification bit	Notifies the ROM/EEPROM-dedicated sequencer (FCU) of the frequency of peripheral bus clock A.	R/W

PBAN is initialized by a reset or by setting the FRESET bit of FRESETR to 1.

PBAN Bits

Before programming or erasing the ROM, set the PBAN bits to match the frequency of peripheral bus clock A. When the peripheral bus clock A notification command is issued, the FCU is notified of the specified value. Do not change the frequency while programming or erasing the ROM/EEPROM.
The value to be specified is the result of converting the operating frequency in MHz units to binary.
Example: When the operating frequency of peripheral clock A is 50 MHz:
Convert 50 to binary, so set the PBAN bits to H'32 (B'0011 0010).

Note: Set the PBAN bits to the frequency of peripheral clock A in MHz. If the specified frequency is different from the actual frequency, data in the ROM/EEPROM may be corrupted.

25.3.18 EEPROM Mat Select Register (EEPMAT)

Address H'FFFF AB00

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	KEY[7:0]								—	—	—	—	—	—	—	EEPSEL
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 b8	KEY[7:0]	Key Code Bits	These bits enable or disable EEPSEL bit modification. The data written to these bits are not stored.	R/(W) *
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R
b0	EEPSEL	EEPROM MAT Select Bit	Selects a memory MAT in the EEPROM. Writing to this bit is enabled only when this register is accessed in word size and H'B3 is written to the KEY bits. 0: Selects the data MAT 1: Selects the product information MAT	R/W

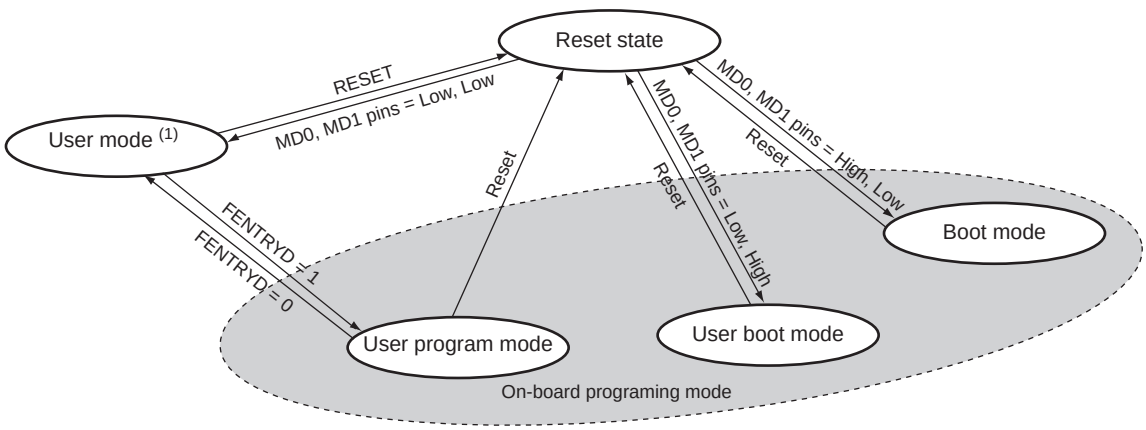
Note: * The data written to are not retained.

The EEPMAT switches memory MATs in the EEPROM.

Writing to the upper byte is enabled only when this register is accessed in word size and specific value is written. The data written to the upper byte is not retained in the register.

25.4 Overview of EEPROM-Related Modes

Figure 25.4 shows the EEPROM-Related mode transition in this LSI.



Note 1. MCU single-chip mode of this LSI is a user mode.

Figure 25.4 EEPROM-Related Mode Transition

- The data MAT can be read, programmed, and erased on the board in user mode, user program mode, user boot mode, and boot mode.
- In user mode, the ROM cannot be programmed or erased but the EEPROM can be programmed and erased. While the EEPROM is being programmed or erased, the ROM can be read. Therefore, the user can program the EEPROM while executing an application program in the ROM protected against programming and erasure.

Table 25.5 compares programming-and erasure-related items for the boot mode, user mode, user program mode, and user boot mode.

Table 25.5 Comparison of Programming Modes

Item	Boot Mode	User Mode	User Program Mode	User Boot Mode
Programming/erasure environment	On-board programming			
Programming/erasure enabled MAT	Data MAT	Data MAT	Data MAT	Data MAT
Programming/erasure control	Host	FCU	FCU	FCU
Entire area erasure	Available (automatic)	Available	Available	Available
Block erasure	Available *1	Available	Available	Available
Programming data transfer	From host via SCI	From any device via RAM	From any device via RAM	From any device via RAM
Reset-start MAT	Embedded program stored MAT	User MAT	User MAT	User boot MAT *2

Notes: 1. The entire area is erased when the LSI is started. After that, a specified block can be erased.
2. After the LSI is started in the embedded program stored MAT and the boot program provided by Renesas Corp. is executed, execution starts from the location indicated by the reset vector of the user boot MAT.

- In boot mode, the user MAT and user boot MAT in the ROM and the data MAT are all erased immediately after the LSI is started. The data MAT can then be programmed from the host via the SCI. The data MAT can also be read after this entire area erasure.
- In user boot mode, a boot operation with a desired interface can be implemented through mode pin settings different from those in user mode or user program mode.
- Both in boot mode and user boot mode, the boot program uses the internal RAM. Therefore, once the RAM is disabled via the RAM Enable Control register (RAMEN) and a reset is issued, the data prior to the reset is no longer stored in the RAM after booting is initiated in boot mode or boot mode. (refer to section 27, RAM Control)

25.5 Boot Mode

To program or erase the data MAT in boot mode, send control commands and programming data from the host. For the system configuration and settings in boot mode, refer to section 24, ROM. This section describes only the commands dedicated for the EEPROM.

25.5.1 Inquiry/Selection Host Commands

Table 25.6 shows the inquiry/selection host commands dedicated to the EEPROM. The data MAT inquiry and data MAT information inquiry commands are used in the step for inquiry regarding the MAT programming information shown in Table 24.9 in section 24.5.4, Inquiry/Selection Host Command Wait State.

Table 25.6 Inquiry/Selection Host Commands (for EEPROM only)

Host Command Name	Function
Data MAT inquiry	Inquires regarding the availability of user MAT
Data MAT information inquiry	Inquires regarding the number of data MATs and the start and end addresses

Each host command is described in detail below. The “command” in the description indicates a command sent from the host to this LSI and the “response” indicates a response sent from this LSI to the host. The “checksum” is byte-size data calculated so that the sum of all bytes to be sent by this LSI becomes H'00.

(1) Data MAT Inquiry

In response to a data MAT inquiry command sent from the host, this LSI returns the information concerning the availability of data MATs.

Command	H'2A			
Response	H'3A	Size	Availability	SUM

[Legend]
Size (1 byte): Total number of characters in the availability field (fixed at 1)
Availability (1 byte): Availability of data MATs (fixed at H'01)
H'00: No data MAT is available
H'01: Data MAT is available
SUM (1 byte): Checksum

(2) Data MAT Information Inquiry

In response to a data MAT information inquiry command sent from the host, this LSI returns the number of data MATs and their addresses.

Command	H'2B		
Response	H'3B	Size	MAT count
	MAT start address		
	MAT end address		
	MAT start address		
	MAT end address		
	:		
	MAT start address		
	MAT end address		
	SUM		

[Legend]
Size (1 byte): Total number of bytes in the MAT count, MAT start address, and MAT end address fields
MAT count (1 byte): Number of data MATs (consecutive areas are counted as one MAT)
MAT start address (4 bytes): Start address of a data MAT
MAT end address (4 bytes): End address of a data MAT
SUM (1 byte): Checksum

The information concerning the block configuration in the data MAT is included in the response to the erasure block information inquiry command (refer to section 24.5.4, Inquiry/Selection Host Command Wait State).

25.5.2 Programming/Erasing Host Commands

Table 25.7 shows the Programming/Erasure Host Commands (for EEPROM). EEPROM-dedicated host commands are provided only for checksum and blank check; the programming, erasing, and reading commands are used in common for the ROM and EEPROM.

To program the data MAT, issue from the host a user MAT programming selection command and then a 256-byte programming command specifying a data MAT address as the programming address. To erase the data MAT, issue an erasure selection command and then a block erasure command specifying an erasure block in the data MAT. The information concerning the erasure block configuration in the data MAT is included in the response to the erasure block information inquiry command. To read data from the data MAT, select the user MAT through a memory read command specifying a data MAT address as the read address.

For the user MAT programming selection, user boot MAT programming selection, 256-byte programming, erasure selection, block erasure selection, and memory read commands, refer to section 24.5.5, Programming/Erasing Host Command Wait State. For the erasure block information inquiry command, refer to section 24.5.4, Inquiry/Selection Host Command Wait State.

Table 25.7 Programming/Erasure Host Commands (for EEPROM)

Host Command Name	Function
Data MAT checksum	Performs checksum verification for the data MAT
Data MAT blank check	Checks whether the data MAT is blank

Each host command is described in detail below. The “command” in the description indicates a command sent from the host to this LSI and the “response” indicates a response sent from this LSI to the host. The “checksum” is byte-size data calculated so that the sum of all bytes to be sent by this LSI becomes H'00.

(1) Data MAT Checksum

In response to a data MAT checksum command sent from the host, this LSI sums the data MAT data in byte units and returns the result (checksum).

Command	H'61			
Response	H'71	Size	MAT checksum	SUM

[Legend]
Size (1 byte): Number of bytes in the MAT checksum field (fixed at 4)
MAT checksum (4 bytes): Checksum of the data MAT data
SUM (4 bytes): Checksum (for the response data)

(2) Data MAT Blank Check

In response to a data MAT blank check command sent from the host, this LSI checks whether the data MAT is completely erased. When the data MAT is completely erased, this LSI returns a response (H'06). If the user MAT has an unerased area, this LSI returns an error response (sends H'E2 and H'52 in that order).

Command	H'62	
Response	H'06	
Error Response	H'E2	H'52

25.6 User Mode, User Program Mode, and User Boot Mode

25.6.1 FCU Command List

To program or erase the data MAT in user mode, user program mode, or user boot mode, issue FCU commands to the FCU. Table 25.8 is a list of FCU commands for EEPROM programming and erasure.

Table 25.8 FCU Command List (EEPROM-Related Commands)

Command Name	Function
Normal mode transition	Moves to the normal mode (refer to section 25.6.2, Conditions for FCU Command Acceptance).
Status read mode transition	Moves to the status read mode (refer to section 25.6.2, Conditions for FCU Command Acceptance).
Lock bit read mode transition (lock bit read 1)	Moves to the lock bit read mode (refer to section 25.6.2, Conditions for FCU Command Acceptance).
Peripheral clock A notification	Sets the frequency of peripheral clock A
Program	Programs EEPROM (in 8-byte or 128-byte units).
Block erase	Erases EEPROM (in block units).
P/E suspend	Suspends programming or erasure.
P/E resume	Resumes programming or erasure.
Status register clear	Clears the ILGLERR, ERSERR, and PRGERR bits in FSTATR0 register and cancels the command-locked state.
Blank check	Checks if a specified area is erased (blank).

FCU commands other than the program command and blank check command are also used for ROM programming and erasure. When the blank check command is issued to the ROM, the lock bits in the ROM are read out.
To issue a command to the FCU, access the EEPROM area through the peripheral bus A. Table 25.9 shows the FCU command formats for the program command and blank check command. For the other command formats, refer to section 24.6.1, FCU Command List. When a peripheral bus A access, as shown in Table 25.8, is made under specified conditions, the FCU performs processing specified by a selected command. For the conditions for the FCU command acceptance, refer to section 25.6.2, Conditions for FCU Command Acceptance. For details of command usage, refer to section 25.6.3, FCU Command Usage.

When the FRDMD bit is set to 0 (memory area read mode), if the data in the first cycle of an FCU command is determined as H'71, the FCU accepts the lock bit read mode transition command. Since the EEPROM has no lock bits, making peripheral bus A read access after a transition to the lock bit read mode results in undefined read data. The FCU detects no access violation error when the undefined data is read. When the FRDMD bit is set to 1 (register read mode), if the data in the first cycle of an FCU command is determined as H'71, the FCU enters a waiting state to wait for the command in the second cycle (H'D0) of the blank check command. At this stage, if H'D0 is written into an EEPROM area by a peripheral bus A write access, the FCU detects it and starts performing the blank check processes specified by the set values in the EEPBCCNT register, and once the check completes the FCU writes check results into the EEPBCSTAT register.
For details of the suspending operation to be initiated by the P/E suspend command, refer to section 24.6.4, Suspending Operation.

Table 25.9 FCU Command Formats (for EEPROM only)

Command	Number of Command Cycles*	First Cycle		Second Cycle		Third Cycle		Fourth Cycle to Cycle N + 2		Cycle N + 3	
		Address	Data	Address	Data	Address	Data	Address	Data	Address	Data
Program (8-byte programming: N = 4)	7	EA	H'E8	EA	H'04	WA	WD1	EA	WDn	EA	H'D0
Program (128-byte programming: N = 64)	67	EA	H'E8	EA	H'40	WA	WD1	EA	WDn	EA	H'D0
Blank check	2	EA	H'71	BA	H'D0	—	—	—	—	—	—

[Legend]
EA : EEPROM area address
An arbitrary address within the range of H'8010 000 to H'8010 7FFF
WA : The start address of write data
BA : The address of an EEPROM erasure block
(An arbitrary address in the erase target block)
WDn : n-th word of programming data (n = 1 to N)
Note: * The number of command cycles indicates the number of instances of write access to programming or erasing addresses through the peripheral bus (P bus).

25.6.2 Conditions for FCU Command Acceptance

The FCU determines whether to accept a command depending on the FCU mode or status. Figure 25.5 is an FCU Mode Transition Diagram (EEPROM-Related Modes)

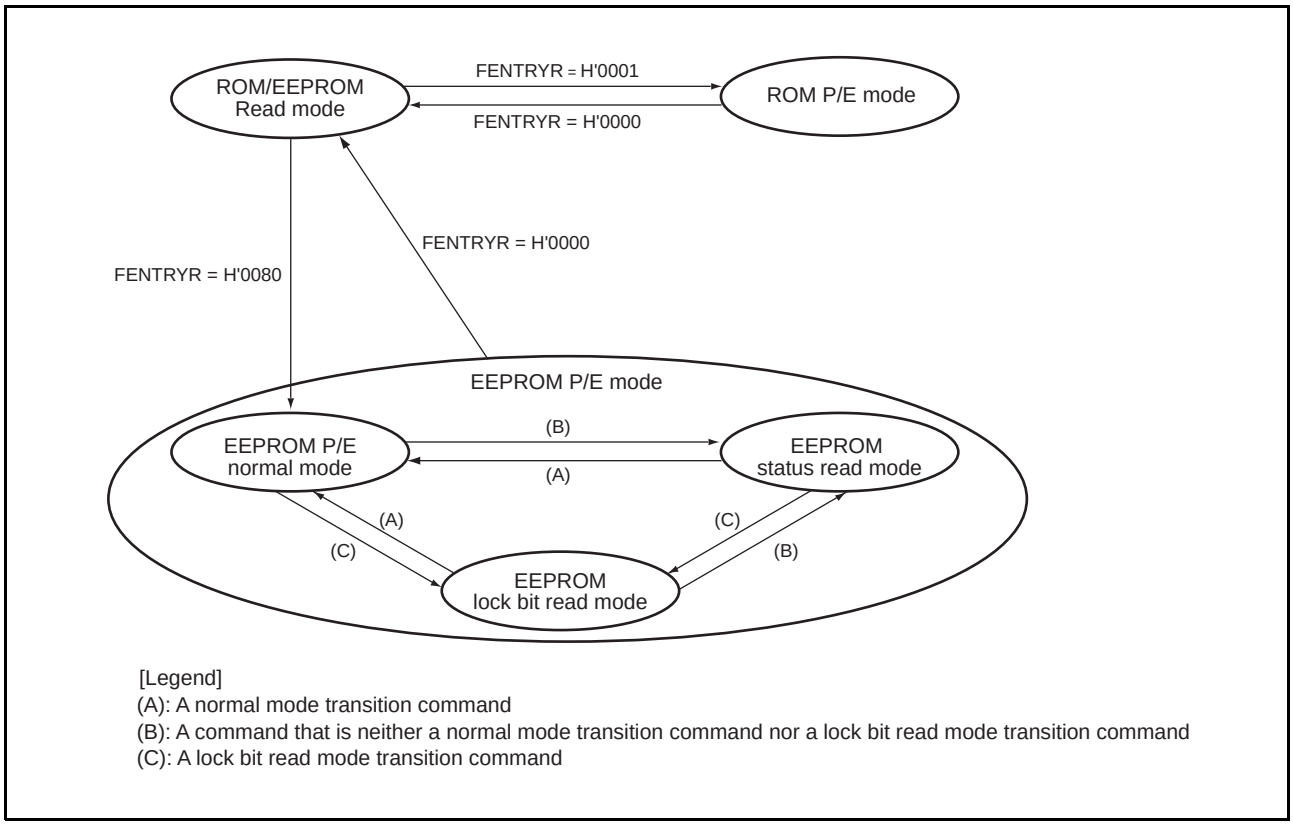


Figure 25.5 FCU Mode Transition Diagram (EEPROM-Related Modes)

Note: Care should be taken in causing a transition by writing to the FENTRYR register.
After writing to the FENTRYR register, execute a FENTRYR register read instruction, and five or more NOP instructions.

(1) ROM P/E Mode

The FCU can accept ROM programming and erasing commands in this mode. The EEPROM cannot be read. The FCU enters this mode when the FENTRYD bit is set to 0 and the FENTRY0 bit is set to 1 in the FENTRYR register. For details of ROM P/E mode, refer to section 24.6.2, Conditions for FCU Command Acceptance.

(2) ROM/EEPROM Read Mode

The EEPROM can be read through the peripheral bus A, and the ROM can be read through the ROM cache at a high speed. The FCU does not accept commands. The FCU enters this mode when the FENTRY0 bit is set to 0 and the FENTRYD bit is set to 0 in the FENTRYR register.

(3) EEPROM P/E Mode

- EEPROM P/E normal mode
  The FCU enters this mode when the FENTRYD bit is set to 1 and the FENTRY0 bit is set to 0 in ROM/EEPROM read mode or ROM P/E mode, or when a normal mode transition command is accepted in EEPROM P/E mode. If the EEPROM area is read through the peripheral bus A, an EEPROM access error occurs and the FCU enters the command-locked state. The ROM can be read through the ROM cache at a high speed.

- EEPROM status read mode
  The FCU enters this mode when the FCU accepts a command that is neither the normal mode transition command nor the lock bit read mode transition command in EEPROM P/E mode. The EEPROM status read mode includes the state in which the FRDY bit in the FSTATR0 register is 0 and the command-locked state after an error has occurred. If the EEPROM area is read through the P bus, the FSTATR0 register value is read. The ROM can be read through the ROM cache at a high speed.

- EEPROM lock bit read mode
  The FCU enters this mode when the FCU accepts a lock bit read mode transition command in EEPROM P/E mode. Since the EEPROM has no lock bits, reading an EEPROM area via the peripheral bus A results in an undefined value. However, no access violation occurs in this case. The ROM can be read through the ROM cache at a high speed.

Table 25.10 shows the correlation between each FCU mode and its register /state and its acceptable commands. When an unacceptable command is issued, the FCU enters the command-locked state (refer to section 25.7.2, Error Protection).
To make sure that the FCU accepts a command, enter the mode in which the FCU can accept the target command, check the FRDY, ILGLERR, ERSERR, and PRGERR bit values in the FSTATR0 register, and the FCUERR, FRDTCT, and FRCRCT bit values in the FSTATR1 register, and then issue the target FCU command. The CMDLK bit in the FASTAT register holds a value obtained by logical ORing the ILGLERR, ERSERR, and PRGERR bit values in the FSTATR0 and the FCUERR, FRDTCT, and FRCRCT bit values in the FSTATR1. Therefore the FCU's error occurrence state can be checked by reading the CMDLK bit. In Table 25.10, the CMDLK bit is used as the bit to indicate the error occurrence state. The FRDY bit of FSTATR0 register is 0 during the programming/erasure, programming/erasure suspension, and blank check processes. While the FRDY bit is 0, the P/E suspend command can be accepted only when the SUSRDY bit in the FSTATR0 register is 1.
Table 25.10 includes 0 and 1 in single cells of the ERSSPD, PRGSPD, and FRDY bit rows for the sake of simplification. The ERSSPD bits 1 and 0 indicate the erasure suspension and programming suspension processes, respectively. The PRGSPD bits 1 and 0 indicate the programming suspension and erasure suspension processes, respectively. The FRDY bit value can be either 1 or 0, which is a value held by the bit prior to a transition to the command lock state.

Table 25.10 FCU Modes/States and Acceptable Commands

Item	P/E Normal Mode			Status Read Mode									Lock Bit Read Mode		
	Programming-Suspended	Erasure-Suspended	Other State	Programming/Erasure Processing	Programming Processing in Erasure-Suspended State	Programming/Erasure Suspension Processing	Blank Check Processing	Programming-Suspended	Erasure-Suspended	Command-Locked (FRDY = 0)	Command-Locked (FRDY = 1)	Other State	Programming-Suspended	Erasure-Suspended	Other State
FRDY bit in FSTATR0	1	1	1	0	0	0	0	1	1	0	1	1	1	1	1
SUSRDY bit in FSTATR0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
ERSSPD bit in FSTATR0	0	1	0	0	1	0/1	0/1	0	1	0/1	0/1	0	0	1	0
PRGSPD bit in FSTATR0	1	0	0	0	0	0/1	0/1	1	0	0/1	0/1	0	1	0	0
CMDLK bit in FASTAT	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0
Normal mode transition	A	A	A	×	×	×	×	A	A	×	×	A	A	A	A
Status read mode transition	A	A	A	×	×	×	×	A	A	×	×	A	A	A	A
Lock bit read mode transition (lock bit read 1)	A	A	A	×	×	×	×	A	A	×	×	A	A	A	A
Peripheral clock A notification	×	×	A	×	×	×	×	×	×	×	×	A	×	×	A
Program	×	*	A	×	×	×	×	×	*	×	×	A	×	*	A
Block erase	×	×	A	×	×	×	×	×	×	×	×	A	×	×	A
P/E suspend	×	×	×	A	×	×	×	×	×	×	×	×	×	×	×
P/E resume	A	A	×	×	×	×	×	A	A	×	×	×	A	A	×
Status register clear	A	A	A	×	×	×	×	A	A	×	A	A	A	A	A
Blank check	A	A	A	×	×	×	×	A	A	×	×	A	A	A	A

[Legend]
A: Acceptable
* : Only programming is acceptable for the areas other than the erasure-suspended block
×: Not acceptable

25.6.3 FCU Command Usage

This section shows how to program and erase the EEPROM using the program command and block erase command, respectively, and how to check the erasure status of the EEPROM using the blank check command. For the firmware transfer to the FCU RAM and the other FCU command usage, refer to section 24.6.3, FCU Command Usage.

If the FCU enters the command lock state in the middle of its handling of commands by setting the FCUERR, FRDTCT, or FRCRCT bit in the FSTATR1 register to 1, the FRDY bit in the FSTATR0 register retains 0. Since the FCU halts its operation in the command lock state, the FRDY bit is not set to 1 from 0.

If the FRDY bit retains 0 for longer than the programming/erasure time or suspend delay time (refer to section 30, Electrical Characteristics), an abnormal operation may have occurred. In such case, initialize the FCU by issuing an FCU reset.

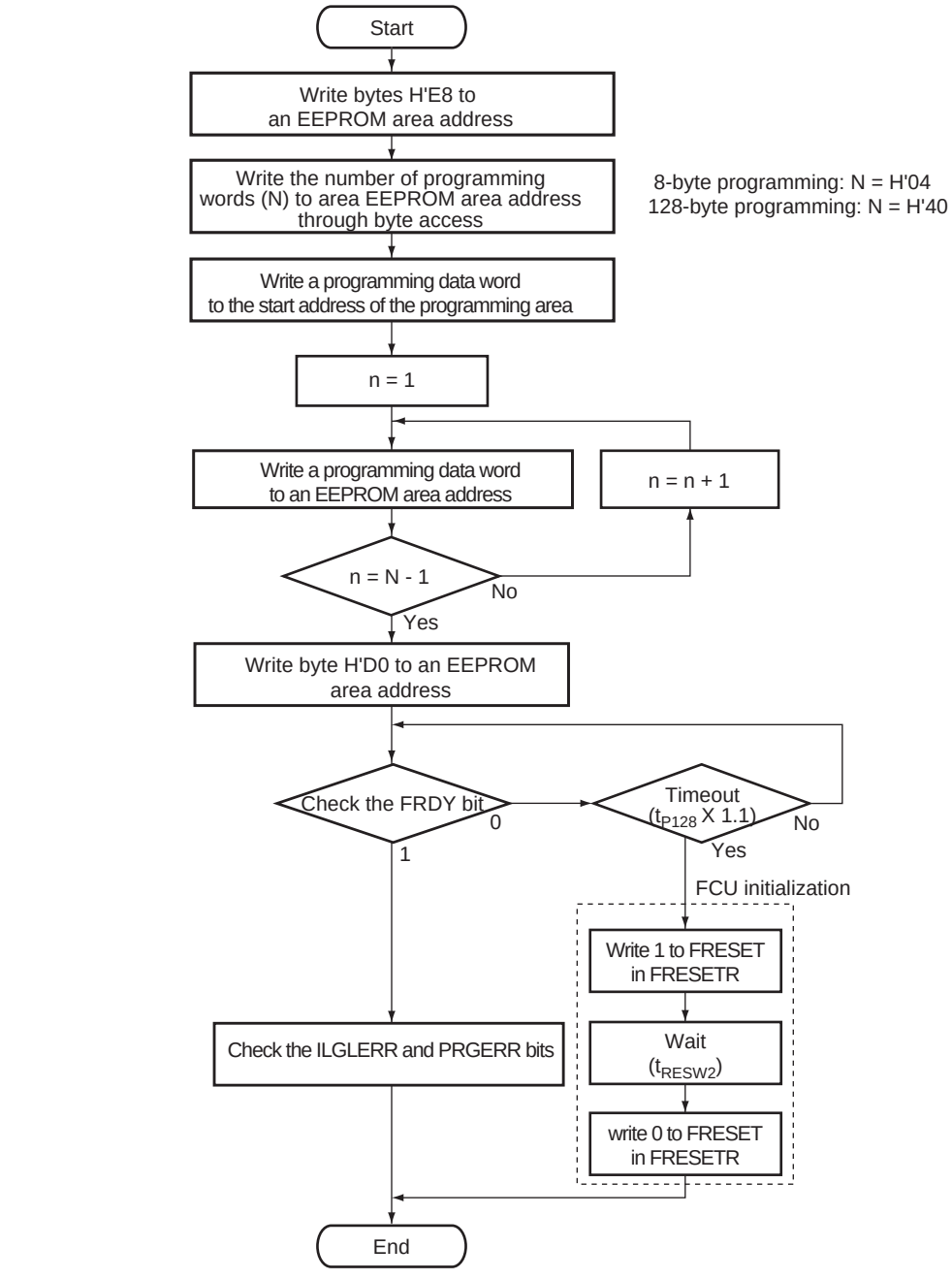
If the FRDY bit is set to 1 upon the termination of an FCU command operation, both the FCUERR, FRDTCT, and FRCRCT bits are cleared to 0. On the other hand, it can be checked via the ILGLERR, ERSERR, or PRGERR bit whether or not an error has occurred after a command operation terminates.

(1) Programming

To program the EEPROM, use the program command. Write byte H'E8 to an EEPROM area address in the first cycle of the program command and the number of words (N)* to be programmed through byte access in the second cycle. Access the peripheral bus in words from the third cycle to cycle N + 2 of the command. In the third cycle, write the programming data to the start address of the target programming area. Here, the start address must be an 8-byte boundary address for 8-byte programming or a 128-byte boundary address for 128-byte programming. After writing words to EEPROM area addresses N times, write byte H'D0 to an EEPROM area address in cycle N + 3; the FCU then starts EEPROM programming. Read the FRDY bit in FSTATR0 register to confirm that EEPROM programming is completed.

If the area accessed in the third cycle to cycle N + 2 includes addresses that do not need to be programmed, write H'FFFF as the programming data for those addresses. To ignore the programming and erasure protection provided by the EEPWE0/1 register settings, set the program/erase enable bit for the target block to 1 before starting programming. Figure 25.6 shows the Procedure for EEPROM Programming.

Note: * N = H'04 for 8-byte programming or N = H'40 for 128-byte programming.



Notes. • t_{P128}: Time required for programming 128-byte data (refer to section 30, Electrical Characteristics).
• t_{RESW2}: Reset pulse width during programming and erasure (refer to section 30, Electrical Characteristics).

Figure 25.6 Procedure for EEPROM Programming

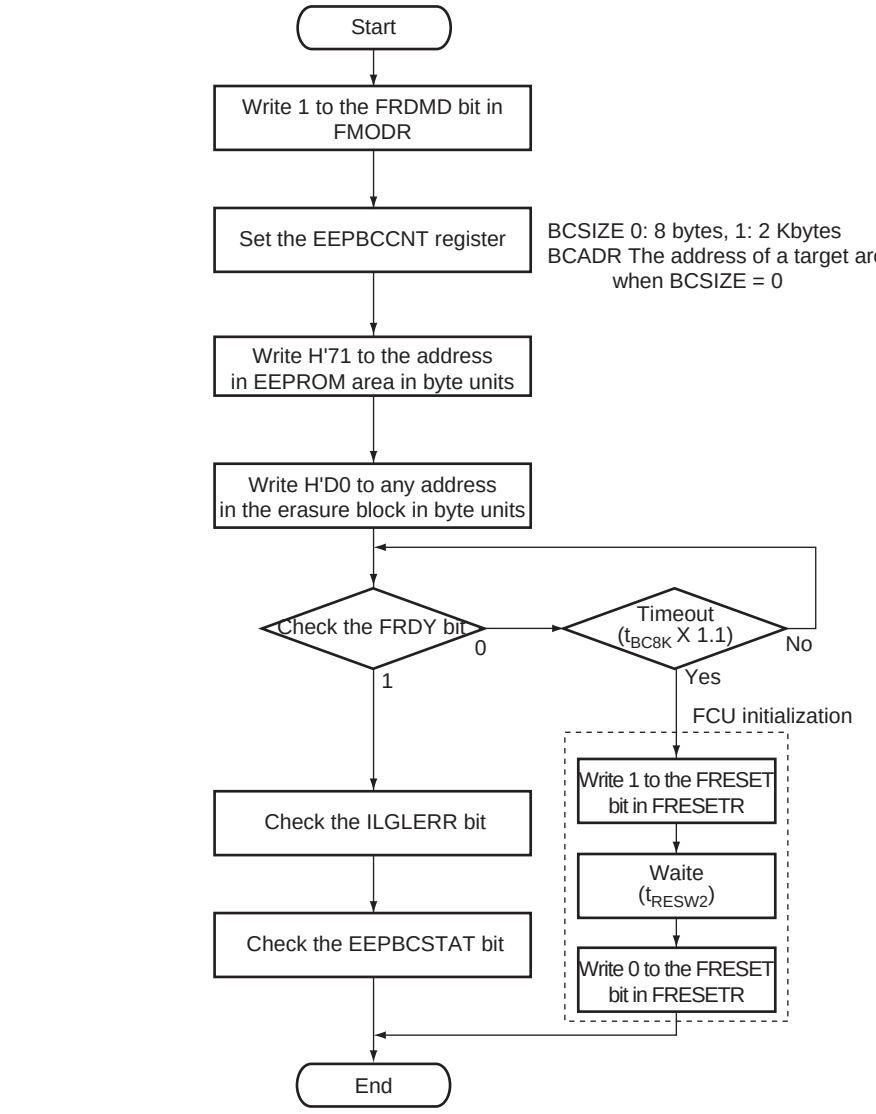
(2) Erasure

To erase the ROM, use the block erase command. The EEPROM can be erased in the same way as ROM erasure (refer to section 24, ROM). Note that the EEPROM has a programming and erasure protection function through EEPWE0/1 register. To ignore the programming and erasure protection provided by the EEPWE0/1 register settings, set the program/erase enable bit for the target block to 1 before starting erasure.

(3) Checking of the Erased State

Since reading the EEPROM erased by the CPU results in undefined values, the blank check command should be used to check the erased state of the EEPROM. To make the blank check command available for use, set the FRDMD bit in the FMODR register to 1 to enable the command first, and then specify the size and start address of a target area via the EEPBCCNT register. When the BCSIZE bit of the EEPBCCNT register is set to 1, a check can be performed on the entire erased block (2 Kbytes) specified in the second cycle of the command. When the BCSIZE bit is set to 0, a check can be performed on an 8-byte area starting from the address obtained by summing the start address of the erased area specified in the second cycle of the command and the value held by the EEPBCCNT register. In the first cycle of the command, a value of H'71 is written in byte into an address of the EEPROM. In the second cycle, once a value of H'D0 is written into a specified address included in the target area, the FCU starts the blank check on the EEPROM. It can be checked whether or not the check is complete via the FRDY bit in the FSTATR0 register. After the blank check is complete, it can be checked whether the target area is erased or filled with 0s and/or 1s via the BCST bit of the EEPBCSTAT register.

Figure 25.7 shows the Procedure of the EEPROM Blank Check.



Notes. • t_{BC8K}: Time required for blank check (refer to section 30, Electrical Characteristics).
• t_{RESW2}: Reset pulse width during programming and erasure (refer to section 30, Electrical Characteristics).

Figure 25.7 Procedure of the EEPROM Blank Check

25.7 Protection

There are three types of EEPROM programming/erasure protection: hardware, software, and error protection.

25.7.1 Software Protection

The software protection function disables EEPROM programming and erasure according to the control register settings. If an attempt is made to issue a programming or erasing command to the EEPROM against software protection, the FCU detects an error and enters command-locked state.

(1) Protection through FENTRYR

When the FENTRYD bit in the FENTRYR register is 0, the FCU does not accept commands for the EEPROM, so EEPROM programming and erasure are disabled. If an attempt is made to issue an FCU command for the EEPROM while the FENTRYD bit is 0, the FCU detects an illegal command error and enters command-locked state (refer to section 25.7.2, Error Protection).

(2) Protection through EEPWE0

When the DBWEi (i = 00 to 15) bit in the EEPWE0 register is 0, programming and erasure of block DBi in the data MAT is disabled. If an attempt is made to program or erasure block DBi while the DBWEi bit is 0, the FCU detects a program/erase protect error and enters command-locked state (refer to section 25.7.2, Error Protection).

25.7.2 Error Protection

The error protection function detects an illegal FCU command issued, an illegal access, or an FCU malfunction, and disables FCU command acceptance (command-locked state). While the FCU is in command-locked state, the EEPROM cannot be programmed or erased. To cancel command-locked state, issue a status register clear command while the FASTAT register is H'10.

While the CMDLKIE bit in the FAEINT register is 1, a flash interface error (FIFE) interrupt is generated if the FCU enters command-locked state (the CMDLK bit in the FASTAT register becomes 1). While an EEPROM-related interrupt enable bit (EEPAEIE, EEPIFEIE, EEPRPEIE, or EEPWPEIE) in the FAEINT is 1, an FIFE interrupt is generated if the corresponding status bit (EEPAE, EEPIFE, EEPRPE, or EEPWPE) in the FASTAT register becomes 1. Table 25.11 shows the error protection types for the EEPROM and the status bit values (the ILGLERR, ERSERR, and PRGERR bits in the FSTATR0 register and the EEPAE, EEPIFE, EEPRPE, and EEPWPE bits in the FASTST register) after each error detection. For the error protection types used in common by the ROM and EEPROM (FENTRYR setting error, most of illegal command errors, erasing error, programming error, FCU error, and FCU RAM ECC error), refer to section 24.8.2, Error Protection. If the FCU enters command-locked state due to a command other than a suspend command issued during programming or erasure processing, the FCU continues programming or erasing the EEPROM. In this state, the P/E suspend command cannot suspend programming or erasure. If a command is issued in command-locked state, the ILGLERR bit becomes 1 and the other bits retain the values set due to the previous error detection.

Table 25.11 Error Protection Types (for EEPROM only)

Error	Description	ILGLERR	ERSERR	PRGERR	EEPAE	EEPIFE	EEPRPE	EEPWPE
Illegal command error	The value specified in the second cycle of a program command is neither H'04 nor H'40.	1	0	0	0	0	0	0
	A lock bit program command has been issued to an area in the EEPROM while the FENTRYD bit of the FENTRYR register is set to 1.	1	0	0	0	0	0	0
EEPROM access error	A read access command has been issued to the EEPROM area while FENTRYD = 1 in the FENTRYR register in EEPROM P/E normal mode.	1	0	0	1	0	0	0
	A write access command has been issued to the EEPROM area while FENTRYD = 0.	1	0	0	1	0	0	0
	An access command has been issued to the EEPROM area while one of the FENTRY1, and FENTRY0 bits in the FENTRYR register is 1.	1	0	0	1	0	0	0
EEPROM instruction fetch error	An instruction fetch has been made in the EEPROM area.	1	0	0	0	1	0	0
EEPROM read protect error	A read access command has been issued to the EEPROM area protected against reading through the EEPRE0/1 register.	1	0	0	0	0	1	0
EEPROM program/erase protect error	A program command or block erase command has been issued to the EEPROM area protected against programming and erasure through the EEPWE0/1 register.	1	0	0	0	0	0	1

25.8 Product Information MAT

The product information MAT stores the device name, device revision number, and embedded program revision number information in ASCII code. The embedded program is stored in the reset-start MAT used in boot mode and user boot mode (refer to section 24.4, Overview of ROM-Related Modes). Table 25.12 and Table 25.13 show the addresses to store the information and an example of information data. In the product information MAT (H'8010 0000 to H'8010 007F), the addresses not shown in these tables are reserved areas. Undefined data will be read from the reserved areas.

Table 25.12 Data Stored in Product Information MAT (SH72A2 Group)

Information	Address	Example of Data
Device name	H'8010 0000 to H'8010 0007	H'5235463732413238 = R5F72A28
Device revision number	H'8010 0010 to H'8010 0011	H'3031 = 01
Embedded program revision number	H'8010 0020 to H'8010 0022	H'313030 = 100 (1.00)

Table 25.13 Data Stored in Product Information MAT (SH72A0 Group)

Information	Address	Example of Data
Device name	H'8010 0000 to H'8010 0007	H'5235463732413038 = R5F72A08
Device revision number	H'8010 0010 to H'8010 0011	H'3031 = 01
Embedded program revision number	H'8010 0020 to H'8010 0022	H'313030 = 100 (1.00)

25.9 Notes on EEPROM

25.9.1 Protection of Data MAT Immediately after a Reset

As the initial value of EEPRE0 and EEPWE0 is H'0000, data MAT programming, erasure, and reading are disabled immediately after a reset. To read data from the data MAT, set the EEPRE0 register appropriately before accessing the data MAT. To program or erase the data MAT, set EEPWE0 register appropriately before issuing an FCU command for programming or erasure. If an attempt is made to read, program, or erase the data MAT without setting the registers, the FCU detects an error and enters command-locked state.

25.9.2 State in which AUD Operation Is Disabled and Interrupts Are Ignored

In the following modes or period, the AUD is in module standby mode and cannot operate. The NMI or maskable interrupt requests are ignored.

- Boot mode
- The program in the embedded program stored MAT is being executed immediately after the LSI is started in user boot mode.

25.9.3 Programming-/Erasure-Suspended Area

The data stored in the programming-suspended or erasure-suspended area is undetermined. To avoid malfunction due to undefined read data, ensure that no data is read from the programming-suspended or erasure-suspended area.

25.9.4 Compatibility with Programming/Erasing Program of Conventional F-ZTAT SH Microcontrollers.

The flash memory programming/erasing program used for conventional F-ZTAT SH microcontrollers does not work with this LSI.

25.9.5 Reset during Programming or Erasure

During programming or erasure, do not generate the following resets: the hardware reset, the power-on reset, the voltage monitor reset, and the watchdog timer reset.

When a hardware reset is generated by asserting the RESET# pin during programming or erasure of the flash memory, hold the reset state for a period of t_{RESW2} (refer to section 30, Electrical Characteristics). In a hardware reset, not only does the voltage applied to the memory unit have to drop, but the power supply for the EEPROM and its internal circuitry also have to be initialized. Thus, the reset state must be maintained over a longer period than in the case of resetting the FCU.
To reset the FCU by setting the FRESET bit in the FRESETR register during programming or erasure, hold the FCU in the reset state for a period of t_{RESW2} (refer to section 30, Electrical Characteristics). Since a high voltage is applied to the EEPROM during programming and erasure, the FCU has to be held in the reset state long enough to ensure that the voltage applied to the memory unit has dropped. Do not read from the EEPROM while the FCU is in the reset state.

25.9.6 Prohibition of Additional Programming

One area cannot be programmed twice or more in succession. To program an area that has already been programmed, be sure to erase the area before reprogramming.

25.9.7 SLEEP instruction in on-board programming mode

Do not execute the SLEEP instruction in on-board programming mode.

25.9.8 Programming or Erasing Product Information MAT

The product information MAT is read-only memory and cannot be programmed or erased. If programming or erasure is performed by setting the EEPSEL bit in the EEPMAT register to 1, the data MAT is programmed or erased. No errors, such as EEPROM access error, occur. Do not attempt to program or erase the product information MAT.

25.9.9 Suspend by a Programming or Erasure Command

When the programming or erasure is suspended by a programming or erasure suspend command, be sure to resume the processing by issuing a resume command and complete it.

26. ROM Cache (ROMC)

26.1 Overview

The ROM cache is designed to cache the instructions and data stored in the ROM, permitting high-speed access to these instructions and data.

26.1.1 Features

Table 26.1 shows the ROM cache specification.

Table 26.1 ROM Cache Specification

Item	Specification
Configuration	Separate caches for instructions and data
Prefetch cache	8-line, 4-way set associative, LRU method*
Prefetch-miss cache	4-line, fully associative, LRU method*
Data cache	4-line, fully associative, LRU method* Write method: always write-through and automatic line invalidation
Line size	16 bytes (128 bits)
Hardware prefetching	Instructions are read from the ROM and stored in the prefetch cache prior to instruction fetching by the CPU

Note: * LRU: Least Recently Used

26.2 Cache Configuration

The ROM-cache module has separate units for instructions and data. The instruction caches consist of a prefetch cache (for instruction read-ahead) and prefetch-miss cache (where data read from the ROM are stored when neither the prefetch cache nor the prefetch-miss cache currently contains the desired data). The data cache is for the storage of data.

Figure 26.1 shows the configuration of the caches. The prefetch cache is 8-line and 4-way set associative, while both the prefetch-miss cache and the data cache are 4-line full associative. All of the caches have 16-byte lines, and the LRU (Least Recently Used) method is the principle of the line-replacement algorithm. For detailed descriptions of the line-replacement policies for the individual caches, refer to section section 26.4.1, Data Cache Lookup.

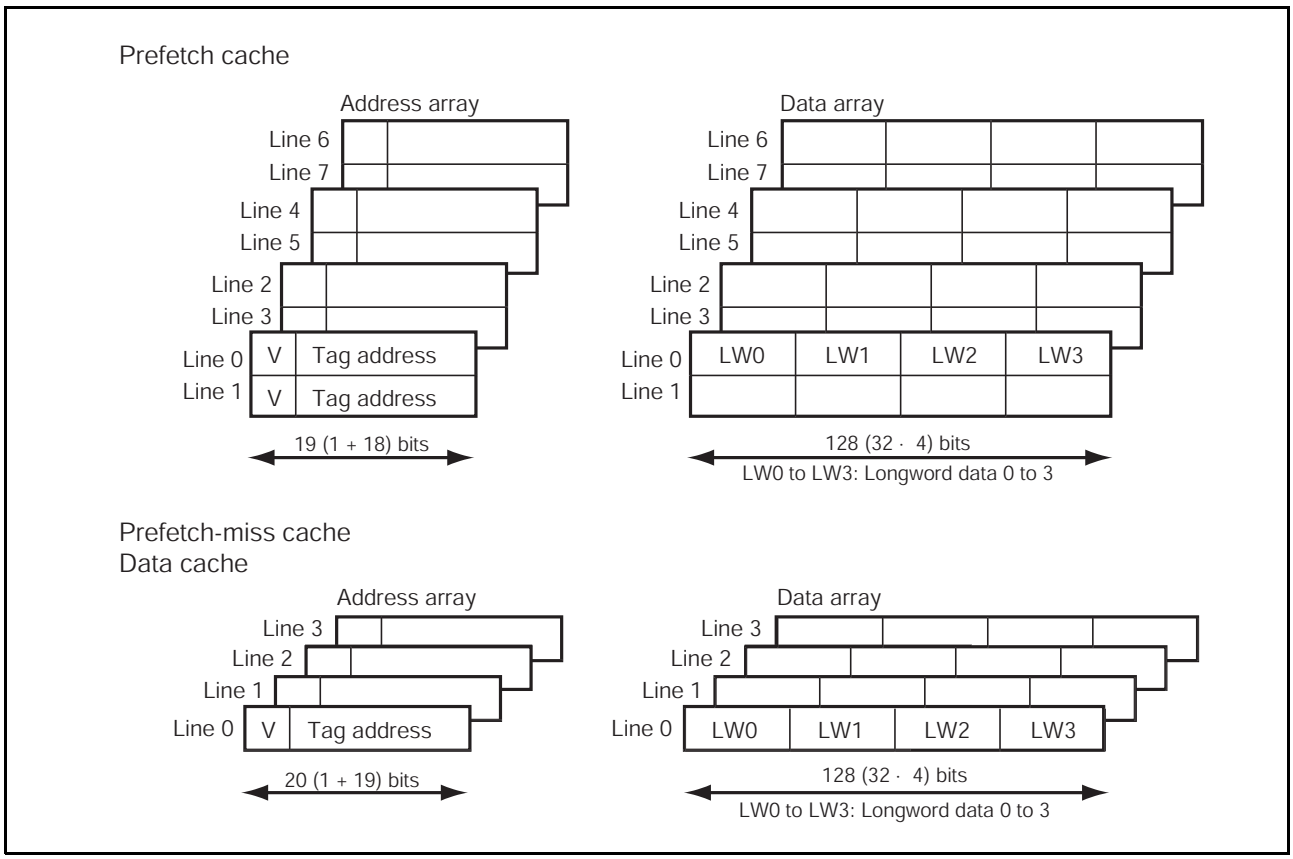


Figure 26.1 Cache Configuration

(1) Address Arrays

The address array of the prefetch cache consists of eight lines, lines 0 to 7. The address arrays of the data and prefetch-miss caches consist of four lines, lines 0 to 3. Each line is composed of a V bit and a Tag address.

The V bit indicates whether or not the line data is valid; the data line is valid when the V bit is 1 and invalid when the V bit is 0. The write method is always write-through and when the target address in the ROM is identified as already cached, the corresponding line is invalidated by setting the V bit to 0.

The Tag addresses hold the addresses for reference in the cache lookup process. Each tag consists of 18 bits (corresponding to bits 22 to 5 of the access address in the ROM) in the prefetch cache and 19 bits (bits 22 to 4 of the access address) in the prefetch-miss cache and the data cache. Bits 31 to 23 of the access address are used on the address bus to identify the memory space, and are thus not relevant to the cache lookup process.

The V bits are initialized to 0 by writing to the corresponding flush bit in the ROM cache control register or by a reset.

(2) Data Array

The data array retains 16-byte instructions or data. The 16 bytes of instructions or data held by a single line is the smallest unit of caching. The line size (size of the part corresponding to the data line of each line) is set to 16 bytes (128 bits) for this MCU.

Data stored in the data array are undefined after a reset.

26.3 Register Descriptions

The ROM cache has the following registers. These registers can only be accessed as longwords. Table 26.2 shows the configuration of the cache-related registers.

Table 26.2 ROM Cache Registers

Register Name	Symbol	After Reset	Address	Access Size
ROM cache control register	RCCR	H'0000 0001	H'FFFC 1400	32
ROM cache control register 2	RCCR2	H'0000 00F5	H'FFFC 1408	32

26.3.1 ROM Cache Control Register (RCCR)

Address H'FFFC 1400

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
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	RCFI	RCFD	RCE
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
b31 to b4	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b3	RCF	ROM Cache Flush Bit	Writing 1 to this bit clears the V bits of all lines in the ROM cache to 0 (flushes the cache). This bit is read as 0. 0: Does not clear the V bits in the ROM cache lines. 1: Clears the V bits in the ROM cache lines. [Condition to become 1] • Writing 1. [Condition to become 0] • Reset	R/W
b2	RCFI	Instruction Cache Flush Bit	Writing 1 to this bit clears the V bits of all lines in the prefetch and prefetch-miss caches to 0 (flushes the prefetch and prefetch-miss caches). This bit is read as 0. 0: Does not clear the V bits in the instruction cache lines. 1: Clears the V bits in the instruction cache lines. [Condition to become 1] • Writing 1. [Condition to become 0] • Reset	R/W
b1	RCFD	Data Cache Flush Bit	Writing 1 to this bit clears the V bits of all lines in the data cache to 0 (flushes the data cache). This bit is read as 0. 0: Does not clear the V bits in the data cache lines. 1: Clears the V bits in the data cache lines. [Condition to become 1] • Writing 1. [Condition to become 0] • Reset	R/W
b0	RCE	ROM Cache Enable Bit	Specifies usage or non-usage of ROM caching. 0: The ROM cache is not used 1: The ROM cache is used [Conditions to become 1] • Reset • Writing 1. [Condition to become 0] • Writing 0.	R/W

The RCCR register contains the RCF bit, which can be used to invalidate all lines in the ROM cache, the RCFI bit, which can be used to invalidate all lines of the instruction caches (the prefetch and prefetch-miss caches), the RCFD bit, which can be used to invalidate all lines in the data cache, and the RCE bit, which can be used to specify enabling or disabling of the ROM cache.

26.3.2 ROM Cache Control Register 2 (RCCR2)

Address H'FFFC 1408

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
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
	—	—	—	—	—	—	—	—	PFECB	PFENB	PFECF	PFE	—	PCE2	—	PCE0
After Reset	0	0	0	0	0	0	0	0	1	1	1	1	0	1	0	1

Bit	Symbol	Bit Name	Description	R/W
b31 to b8	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b7	PFECB	Conditional Branch Prefetch Enable Bit	Specifies whether or not data is to be prefetched from the destinations of conditional branches. 0: No prefetching from the destinations of conditional branches. 1: Prefetching from the destinations of conditional branches. [Conditions to become 1] • Reset • Writing 1. [Condition to become 0] • Writing 0.	R/W
b6	PFENB	Unconditional Branch Prefetch Enable Bit	Specifies whether or not data is to be prefetched from the destinations of unconditional branches. 0: No prefetching from the destinations of unconditional branches. 1: Prefetching from the destinations of unconditional branches. [Conditions to become 1] • Reset • Writing 1. [Condition to become 0] • Writing 0.	R/W
b5	PFECF	Consecutive Prefetch Enable Bit	Specifies whether or not prefetching is applied to instructions for consecutive execution (consecutive instructions). 0: Consecutive instructions are not prefetched. 1: Consecutive instructions are prefetched. [Conditions to become 1] • Reset • Writing 1. [Condition to become 0] • Writing 0.	R/W
b4	PFE	Prefetch Cache Enable Bit	Specifies usage or non-usage of the prefetch cache. 0: The prefetch cache function is not used. 1: The prefetch cache function is used. [Conditions to become 1] • Reset • Writing 1. [Condition to become 0] • Writing 0.	R/W
b3	—	Reserved	This bit is always read as 0. The write value should always be 0.	R
b2	PCE2	Prefetch-Miss Cache Enable Bit	Specifies usage or non-usage of the prefetch-miss cache. 0: The prefetch-miss cache function is not used. 1: The prefetch-miss cache function is used. [Conditions to become 1] • Reset • Writing 1. [Condition to become 0] • Writing 0.	R/W
b1	—	Reserved	This bit is always read as 0. The write value should always be 0.	R

Bit	Symbol	Bit Name	Description	R/W
b0	PCE0	Data Cache Enable Bit	Specifies usage or non-usage of the data cache. 0: The data cache is not used. 1: The data cache is used. [Conditions to become 1] • Reset • Writing 1. [Condition to become 0] • Writing 0.	R/W

The PCE2 and PCE0 bits of the RCCR2 register specify enabling or disabling the prefetch-miss cache and data cache, respectively. The PFE bit specifies enabling or disabling the prefetch cache.

With regard to the prefetching function, the PFECF, PFENB, and PFECB bits are used to specify prefetching of consecutive instructions, prefetching from the destinations of unconditional branches, and prefetching from the destinations of conditional branches, respectively.

26.4 Operation

26.4.1 Data Cache Lookup

When the data cache is enabled and data is read from the ROM area, the data cache is checked to see if it holds valid target data. The tag addresses of all four lines are simultaneously fed to four comparators, each of which compares bits 22 to 4 of one tag address with the access address. If the result of comparison is a match and the compared line is valid (the V bit = 1), the cache has been "hit" and LW0 to LW3 in the corresponding line of the data array are read out. Otherwise, the result is regarded as a cache miss

Bits 3 and 2 of the access address then indicate the target longword for output to the CPU: 00 corresponds to LW0, 01 to LW1, 10 to LW2, and 11 to LW3.

Figure 26.2 shows the concept of looking up the data cache, with line 1 being hit.

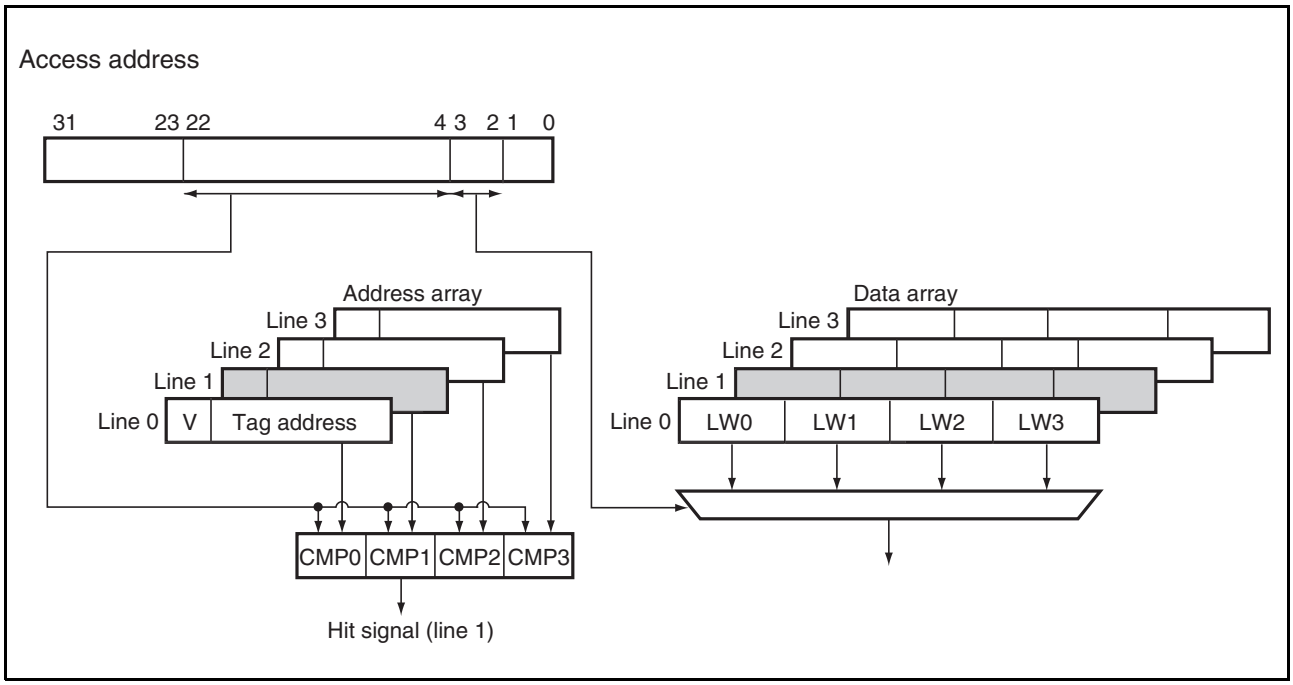


Figure 26.2 Concept of Looking up the Data Cache (with Line 1 being Hit)

If the data cache has not been hit (the cache was missed), data for the cache line is read out from the corresponding actual addresses in the ROM and bits 22 to 4 of the target address replace the tag address for the least recently used line (LRU method) of the address array. At the same time, the V bit is set and data from the target address is also output to the CPU.

26.4.2 Instruction Cache Lookup

In looking up the prefetch cache, whether the value of bit 4 of the access address is 0 or 1 determines whether the even- (0, 2, 4, or 6) or odd-numbered (1, 3, 5, or 7) lines should be checked, respectively. The tag addresses of the four selected lines are then simultaneously fed to four comparators, each of which compares bits 22 to 5 of one tag address with the access address. If the result of comparison is a match and the compared line is valid (the V bit = 1), the cache has been hit and LW0 to LW3 in the corresponding line of the data array are read out. Otherwise, the result is regarded as a cache miss.

Bits 3 and 2 of the access address then indicate the target longword for output to the CPU: 00 corresponds to LW0, 01 to LW1, 10 to LW2, and 11 to LW3.

Figure 26.3 shows the concept of looking up the prefetch cache, with line 2 being hit.

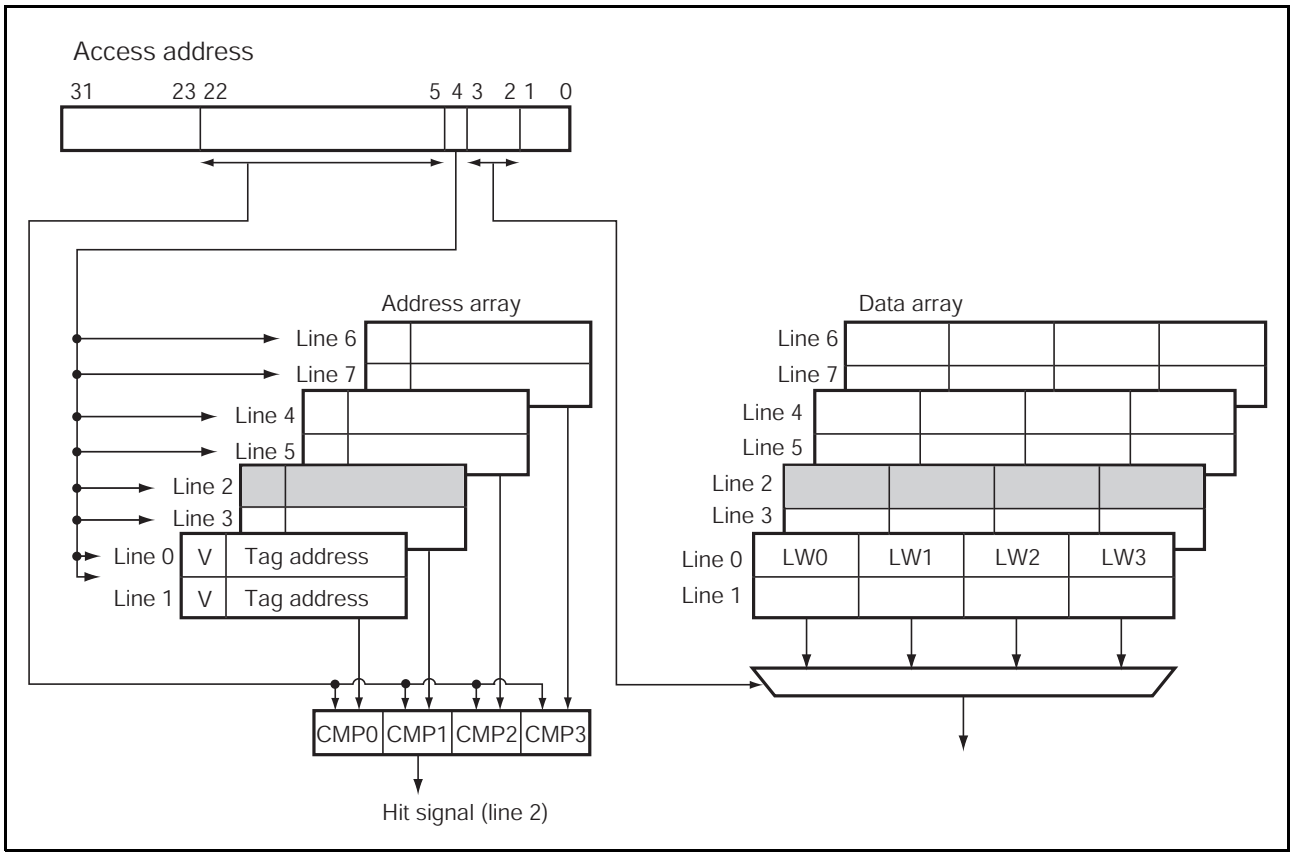


Figure 26.3 Concept of Looking up the Prefetch Cache (with Line 2 being Hit)

The prefetch-miss cache is looked up in the same way as the data cache. When a cache miss is produced by looking up both the prefetch and prefetch-miss cache, data for the cache line is read out from the corresponding actual addresses in the ROM and bits 22 to 5 of the target address replace the tag address for the least recently used line (LRU method) of the address array. At the same time, the V bit is set and the instruction from the target address is also output to the CPU.

The prefetch cache is constantly updated if hardware prefetching is enabled. Whether bit 4 of the address containing the prefetched instruction is 0 or 1 determines whether the line to be updated is the least recently used of the even- or odd-numbered lines, respectively. Updating is done by replacing the tag address of the least recently used line with bits 22 to 5 of the read address and reading the data for the corresponding line of the data array from the ROM.

26.4.3 Hardware Prefetching

The ROM cache improves the hit rate of its component caches by hardware prefetching. There are two types of prefetching: consecutive prefetching and branch prefetching.

(1) Consecutive Prefetching
Instructions that are subject to consecutive access are prefetched and placed in the prefetch cache.

(2) Branch Prefetching
In branch prefetching, branch instructions are prefetched and decoded, their target addresses are predicted, and the instructions at those addresses are prefetched. Branch prefetching is subdivided into prefetching from the targets of conditional branches and prefetching from the targets of unconditional branches.

(a) Conditional-branch prefetching
On encountering a conditional branch instruction (BF, BT, BF/S, or BT/S), prefetching from the predicted destination address is performed.

(b) BRA and BSR instructions
If unconditional-branch prefetching has been selected, prefetching always proceeds on encountering these unconditional branch instructions.

(c) JMP, JSR, and JSR/N instructions
If unconditional-branch prefetching has been selected, prefetching from the destinations of branch instructions of the above type is performed when the instruction has a register index and the index is fully predictable.

The following is an example of instructions that will lead to prefetching from the destination address of the JMP instruction when consecutive-instructive and unconditional-branch prefetching have been selected (JMP):

```
MOVI20 # imm20, Rn
JMP @Rn
```

27. RAM Control

27.1 Introduction

This LSI incorporates 64- or 32-Kbyte RAM, which is connected to the F (Fetch), M (Memory), and CPU (Internal) buses. This on-chip RAM can be accessed via any of these buses independently. Figure 27.1 shows the RAM Block Diagram and Figure 27.2 shows Bus Connections in RAM.

The on-chip 64-Kbyte RAM is allocated in addresses H'FFF8 0000 to H'FFF8 FFFF (pages 0 to 3) and the 32-Kbyte RAM is allocated to addresses H'FFF8 0000 to H'FFF8 7FFF (pages 0 and 1), as shown in Table 27.1.

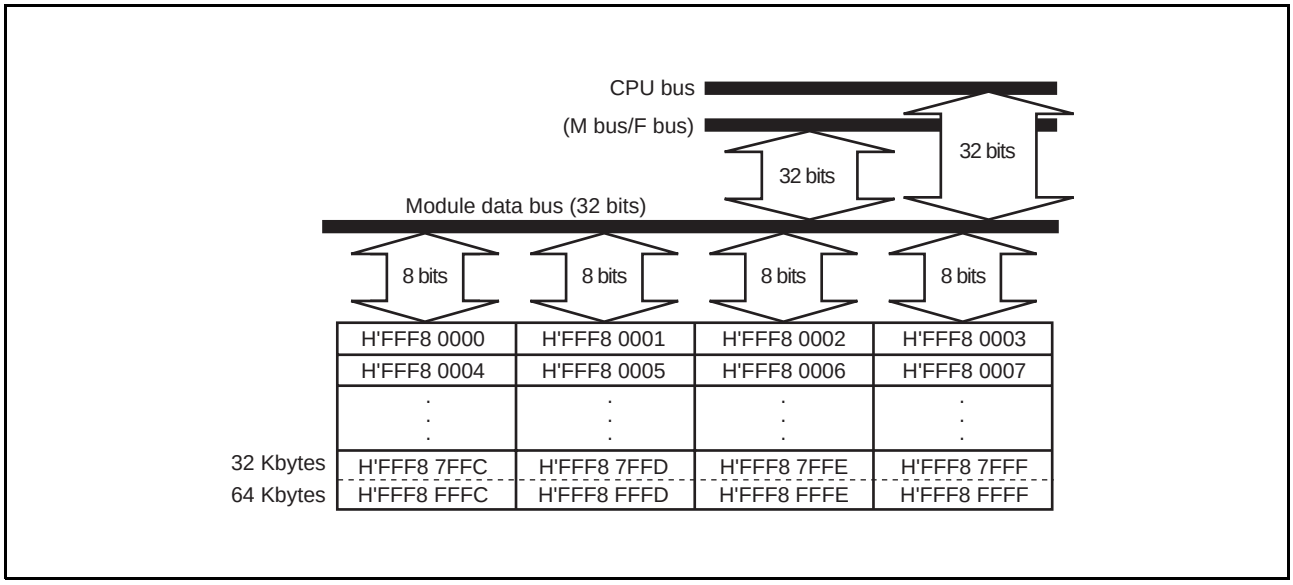


Figure 27.1 RAM Block Diagram

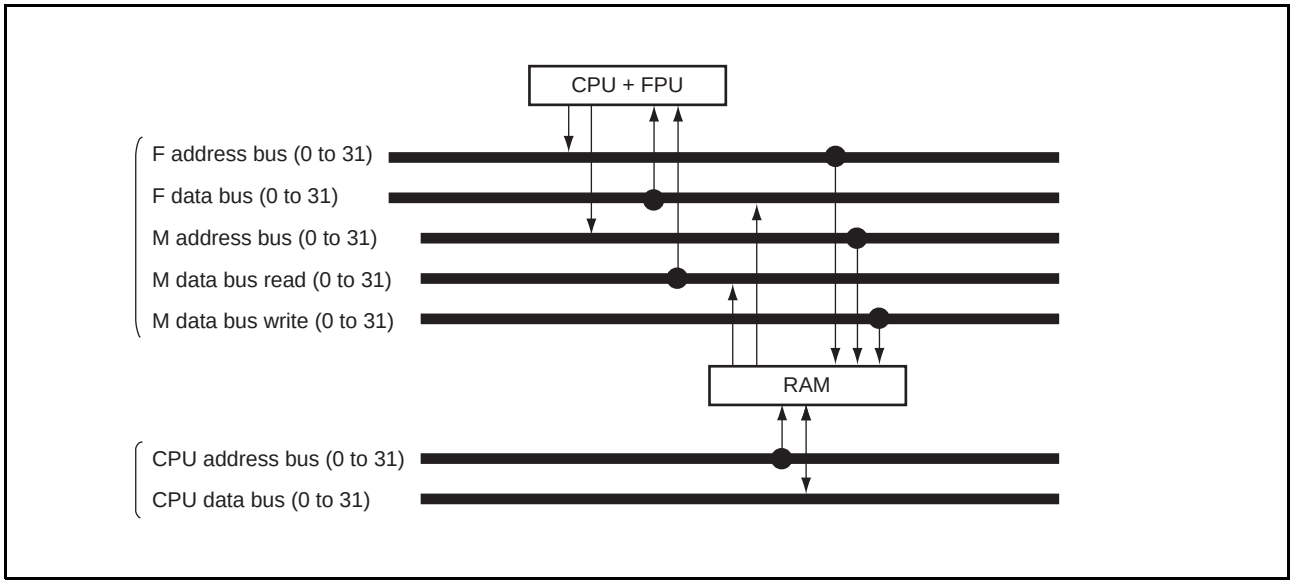


Figure 27.2 Bus Connections in RAM

Table 27.1 On-chip RAM Address Space

Page	Address
Page 0	H'FFF8 0000 to H'FFF8 3FFF
Page 1	H'FFF8 4000 to H'FFF8 7FFF
Page 2	H'FFF8 8000 to H'FFF8 BFFF
Page 3	H'FFF8 C000 to H'FFF8 FFFF

27.1.1 Specifications

Table 27.2 lists the RAM Specifications.

Table 27.2 RAM Specifications

Item	Specification
Access	The CPU/FPU, DMAC, and AUD-II can access on-chip RAM in 8, 16, or 32 bits. Data in the on-chip RAM can be effectively used as program area or stack area data necessary for access at high speed. According to on/off of the ECC error correction function and operating frequency (PLL multiplication rate), 1 or 2 cycles should be specified by a register when the on-chip RAM is read, and 2 or 3 cycles must be specified by a register when the on-chip RAM is written.
ECC	The ECC error correction function can be enabled or disabled by register settings. The function is initially enabled. On reading/writing RAM data, correction of one erroneous bit and detection of 2 erroneous bits are possible for 32 bits of data when the function is enabled, and parity error detection is available when the function is disabled. These ECC error detection and correction and parity errors are collectively called RAM errors. Flags that indicate occurrence of these RAM errors are provided.
Interrupts	Whether or not an interrupt is requested upon occurrence of a RAM error can be selected by register settings.
Ports	Each page in the on-chip RAM has two independent read and write ports. The read port is connected to CPU, F, and M buses and the write port is connected to I and M buses. The F and M buses are used for accesses from the CPU. The CPU bus is used for accesses from external address spaces.
Priority	If the same page is accessed from multiple buses simultaneously, the access is performed according to the bus priority. The bus priority is as follows: CPU bus (highest), M bus (middle), F bus (lowest).

27.2 Registers

Table 27.3 lists the RAM Registers.

Table 27.3 RAM Registers

Register Name	Symbol	After Reset	Address	Access Size
System protect register 0	SPR0	H'00	H'FF46 E063	8
RAM enable control register 0	RAMEN0	H'00FF	H'FF46 E102	8, 16
RAM write enable control register 0	RAMWEN0	H'00FF	H'FF46 E106	8, 16
RAM ECC enable control register	RAMECC	H'0000	H'FF46 E10A	8, 16
RAM error status register	RAMERR	H'00	H'FF46 E10F	8
RAM error interrupt control register	RAMINT	H'00	H'FF46 E117	8
RAM access cycle set register	RAMACYC	H'0000	H'FF46 E11A	8, 16

27.2.1 System Protect Register 0 (SPR0)

Address H'FF46 E063

	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	0	0	0	0	0	0	0	0

Bit	Description	R/W
b7 to b0	When writing: B'1111 0001: Protect canceled Other than B'1111 0001: Protect When reading: b0 bit 0: Protect 1: Protect canceled Bits b7 to b1 are always read as 0.	R/W

The SPR0 register is used to set the protect function in which registers LOCR, SLCR0, VMCR, VD1LSL, RAMECC, RAMACYC, and AUDEN are not easily modified. When modifying these register values, take the following steps.

(1) Write H'F1 to the SPR0 register (writing to the registers above is enabled).
(2) Modify the values of registers LOCR, SLCR0, VMCR, VD1LSL, RAMECC, RAMACYC, and AUDEN.
(3) Write a value other than H'F1 to the SPR0 register (writing to the registers above is disabled).

27.2.2 RAM Enable Control Register 0 (RAMEN0)

Address H'FF46 E102

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RNKEY [7:0]								—	—	—	—	RAME3 *2	RAME2 *2	RAME1	RAME0
After Reset	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b15 to b8	RNKEY [7:0]	RAMEn Write Enable Bits (n = 0 to 3)	These bits enable or disable write to the RAMEn (n = 0 to 3) bit. The write data is not retained and these bits are always read as 0. H'96: Enable write to bits RAMEn (n = 0 to 3) Other than H'96: Disable write to bits RAMEn (n = 0 to 3)	R/(W) *1
b7 to b4	—	Reserved	These bits are read as 1. The write value should be 1.	R
b3	RAME3*2	Enable Control Bits	These bits enable or disable access to each of spaces page 0 to page 3 which allocates the RAMEn (n = 0 to 3) bit in the on-chip RAM. H'96 should be written to the upper byte (RNKEY) simultaneously when writing. The initial value that is all set to 1 enables access to all the RAM pages. 0: Disables access to Page n in the on-chip RAM (n = 0 to 3) 1: Enables access to Page n in the on-chip RAM (n = 0 to 3)	R/W
b2	RAME2*2			R/W
b1	RAME1			R/W
b0	RAME0			R/W

Notes: To avoid erroneous rewriting, the way of writing data to this register is different from that to other general registers. For details, see section 27.2.8, Notes on Register Access.
1. Write data is not retained.
2. This bit is reserved in the use of 32-Kbyte RAM.

The RAMEN0 register is a 16-bit readable/writable register that enables or disables the access to the on-chip RAM. The RAMEN0 register is initialized to H'00FF by a reset. The RAMEN0 register can be written to in words, and can be read in bytes or words.

If a RAMEn bit (n = 0 to 3) corresponding to the access page is set to 1, accessing the on-chip RAM becomes enabled; while if RAMEn bit (n = 0 to 3) is cleared to 0, the on-chip RAM cannot be accessed. In the access disabled state, an undefined data is read if the page is read or if an instruction in the page is fetched, and a write to the page is ignored. The initial value of the RAMEn bit is 1.

To rewrite the RAMEn bit, word size data with upper byte as H'96 and lower byte as write data must be written.

When the upper byte of the RAMEN0 register is read, H'00 is always read.

An instruction to access the on-chip RAM must not be placed immediately after an instruction to write to the RAMEN0 register. Otherwise, correct access to the on-chip RAM cannot be guaranteed.

When rewriting the RAMEn bit, execute an instruction to read from the RAMEN0 register and five or more NOP instructions immediately after the instruction for writing.

27.2.3 RAM Write Enable Control Register 0 (RAMWEN0)

Address H'FF46 E106

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RWNKEY [7:0]								—	—	—	—	RAMWE3 *2	RAMWE2 *2	RAMWE1	RAMWE0
After Reset	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b15 to b8	RWNKEY [7:0]	RAMWEn Write Enable Bits (n = 0 to 3)	These bits enable or disable write to the RAMWEn (n= 0 to 3) bit. The write data is not retained and these bits are always read as 0. H'69: Enable write to bits RAMWEn (n = 0 to 3) Other than H'69: Disable write to bits RAMWEn (n = 0 to 3)	R/(W) *1
b7 to b4	—	Reserved	These bits are read as 1. The write value should be 1.	R
b3	RAMWE3 *2	Enable Control Bits	These bits enable or disable access to each of spaces page 0 to page 3 which allocates the RAMWEn (n = 0 to 3) bit in the on-chip RAM. H'69 should be written to the upper byte (RWNKEY) simultaneously when writing. The initial value that is all set to 1 enables access to all the RAM pages. 0: Disables access to Page n in the on-chip RAM (n = 0 to 3) 1: Enables access to Page n in the on-chip RAM (n = 0 to 3)	R/W
b2	RAMWE2 *2			R/W
b1	RAMWE1			R/W
b0	RAMWE0			R/W

Notes: To avoid erroneous rewriting, the way of writing data to this register is different from that to other general registers. For details, see section 27.2.8, Notes on Register Access.
1. Write data is not retained.
2. This bit is reserved in the use of 32-Kbyte RAM.

The RAMWEN0 register is a 16-bit readable/writable register that enables or disables the access to the on-chip RAM. The RAMWEN0 register is initialized to H'00FF by a reset. The RAMEN0 register can be written to in words, and can be read in bytes or words.

If a RAMWEn bit (n = 0 to 3) is corresponding to the page to be accessed is set to 1, writing to the on-chip RAM becomes enabled; while if RAMWEn bit (n = 0 to 3) is cleared to 0, the on-chip RAM cannot be written to.

In the write disabled state, writing to the on-chip RAM is ignored. The initial value of the RAMWE bit is 1.

To rewrite the RAMWEn bit, word size data with upper byte as H'69 and lower byte as write data must be written.

When the upper byte (bits 15 to 8) of the RAMWEN0 register is read, the read value is always H'00.

An instruction to access the on-chip RAM must not be placed immediately after an instruction to write to the RAMWEN0 register. Otherwise, correct access to the on-chip RAM cannot be guaranteed.

To enable the access to the on-chip RAM by setting the RAMWEn bit to 1, an instruction to read the RAMWEN register must be placed immediately after an instruction to write to the RAMWEN0 register.

When rewriting the RAMWEn bit, execute an instruction to read from the RAMWEN0 register and five or more NOP instructions immediately after the instruction for writing to the RAWEN0 register.

27.2.4 RAM ECC Enable Control Register (RAMECC)

Address H'FF46 E10A

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	REKEY [7:0]								—	—	—	—	—	—	—	RECCA
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 b8	REKEY [7:0]	RECCA Write Enable Bits	These bits enable or disable write to the RECCA bit. The write data is not retained and these bits are always read as 0. H'76: Enable write to the RECCA bit. Other than H'76: Disable write to the RECCA bit.	R/(W) *
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R
b0	RECCA	ECC Enable Control Bit	Enables or disables the ECC correction. H'76 is written to the upper byte (REKEY) simultaneously. The initial value that is all set to 0 enables ECC correction. 0: Enables ECC correction 1: Disables ECC correction	R/W

Notes: To avoid erroneous rewriting, the way of writing data to this register is different from that to other general registers. For details, see section 27.2.8, Notes on Register Access.
* Write data is not retained.

Before changing the values in the RAMECC register, first disable the protection with the SPR0 register. The RAMECC register enables or disables the ECC correction function.

The RAMECC register can be written to in words, and can be read in bytes or words. To rewrite the RAMECC register, word size data with upper byte as H'76 and lower byte as write data must be written.

When the upper byte (bits 15 to 8) of the RAMECC register is read, the read value is always H'00.

Do not place an instruction to access to the on-chip RAM immediately after an instruction to write to the RAMECC register; otherwise correct access to the on-chip RAM cannot be guaranteed.

When rewriting the RECCA bit, execute an instruction to read from the RAMECC register and five or more NOP instructions immediately after the instruction for writing to the RAMECC register.

27.2.5 RAM Error Status Register (RAMERR)

Address H'FF46 E10F

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	RPARI	—	—	RDTCT	RCRCT
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	—
b4	RPARI	RAM Parity Error Monitor Bit	Monitors whether or not a parity error occurs when the ECC error correction is disabled. 0: No parity error has occurred. 1: A parity error has occurred. [Conditions to become 0] • Reset • Writing 0 to this bit after reading 1 from it [Condition to become 1] • A parity error has occurred	R/W
b3, b2	—	Reserved	These bits are read as 0. The write value should be 0.	—
b1	RDTCT	RAM 2-bit Error Detection Monitor Bit	Monitors whether or not a 2-bit error detection occurs when the ECC error correction is enabled. 0: No 2-bit error detection has occurred. 1: A 2-bit error detection has occurred. [Conditions to become 0] • Reset • Writing 0 to this bit after reading 1 from it [Condition to become 1] • A 2-bit error detection has occurred	R/W
b0	RCRCT	RAM 1-bit Error Correction Monitor Bit	Monitors whether or not a 1-bit error correction occurs when the ECC error correction is enabled. 0: No 1-bit error correction has occurred. 1: A 1-bit error correction has occurred. [Conditions to become 0] • Reset • Writing 0 to this bit after reading 1 from it [Condition to become 1] • A 1-bit error correction has occurred	R/W

The RAMERR register monitors RAM error occurrence.

The RAMERR register is initialized by a reset. The RAMERR register can be read or written to in bytes.

When the ECC error correction function is enabled (the RECCA bit in RAMECC register = 0), the occurrence of a 1-bit error correction during an on-chip RAM read operation sets 1 to the RCRCT bit, whereas a 2-bit error detection sets 1 to the RDTCT bit. Also, when the ECC error correction function disabled (the RECCA bit in RAMECC register = 1), the occurrence of a parity error during an on-chip RAM read operation sets 1 to the RPARI bit.

If the ECC error correction function is disabled via the RAMECC register after the RDTCT and RCRCT bits are set, these bits still remain set. Also, if the ECC error correction function is enabled via the RAMECC register after the RPARI bit is set, this bit still remains set to 1.

Write H'00 to clear the RAMERR register. Immediately after the instruction for writing to the RAMERR register, execute an instruction to read from the RAMERR register and five or more NOP instructions. The status bits that have been set can be cleared by reading them while they are set and then writing 0 to them. Note that clearing a status bit by writing 0 is only possible after 1 has been read from it.

27.2.6 RAM Error Interrupt Control Register (RAMINT)

Address H'FF46 E117

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	RPEIE	—	—	REDIE	RECIE
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R
b4	RPEIE	RAM Parity Error Interrupt Bit	Enables/disables an interrupt upon occurrence of a parity error when the ECC error correction is disabled. 0: Disables an interrupt upon occurrence of a parity error. 1: Enables an interrupt upon occurrence of a parity error.	R/W
b3, b2	—	Reserved	These bits are read as 0. The write value should be 0.	R
b1	REDIE	RAM 2-bit Error Correction Interrupt Bit	Enables/disables an interrupt upon detection of a 2-bit error when the ECC error correction is enabled. 0: Disables an interrupt upon detection of a 2-bit error. 1: Enables an interrupt upon detection of a 2-bit error.	R/W
b0	RECIE	RAM 1-bit Error Correction Interrupt Bit	Enables/disables an interrupt upon occurrence of a 1-bit error correction when the ECC error correction is enabled. 0: Disables an interrupt upon correction of a 1-bit error. 1: Enables an interrupt upon correction of a 1-bit error. (An interrupt is also generated upon detection of a 2-bits error)	R/W

The RAMINT register enables or disables the RAM error interrupt.

If the RECIE bit in RAMINT register is set enabled while the ECC error correction function is set enabled (the RECCA bit in the RAMECC register = 0) via the RAMECC register, an interrupt is generated upon the correction of a 1-bit error or detection of a 2-bit error. Also, if the REDIE bit in RAMINT register is set enabled an interrupt is generated upon detection of a 2-bit error. Table 27.4 shows the Conditions for Interrupt Occurrence When ECC Error Correction is Enabled. If the RPEIE bit of RAMINT register is set enabled while the ECC error correction function is set enabled (the RECCA bit = 1) via the RAMECC register, an interrupt is generated upon occurrence of a parity error.

The RAMINT register is initialized by a reset.

The RAMINT register can only be read or written to in bytes.

When rewriting the RAMINT register, execute an instruction to read from the RAMINT register and five or more NOP instructions immediately after the instruction for writing to the RAMINT register.

Table 27.4 Conditions for Interrupt Occurrence When ECC Error Correction is Enabled

REDIE Bit	RECIE Bit	RAM Errors to Trigger an Interrupt
0	0	None
0	1	Upon occurrence of a 1-bit error correction or 2-bit error detection
1	0	Upon occurrence of 2-bit error correction
1	1	Upon occurrence of a 1-bit error correction or 2-bit error detection

27.2.7 RAM Access Cycle Set Register (RAMACYC)

Address H'FF46 E11A

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RAKEY [7:0]								—	—	WRCYC1	WRCYC0	—	—	—	RDCYC
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 b8	RAKEY [7:0]	WRCYC1, WRCYC0, RDCYC Write Enable Bits	These bits enable or disable write to bits WRCYC1, WRCYC0, and RDCYC bits. The write data is not retained and these bits are always read as 0. H'78: Enable write to bits WRCYC1, WRCYC0, and RDCYC bits. Other than H'78: Disable write to bits WRCYC1, WRCYC0, and RDCYC bits.	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R
b5	WRCYC1	RAM Write Cycle Set Bits	These bits set the RAM write cycle of the RAM M bus (the memory bus). H'78 is written to the upper byte (RAKEY) simultaneously. b5 b4 0 0 : Set write cycle to 4 cycles (initial setting) 0 1 : Set write cycle to 3 cycles 1 0 : Set write cycle to 2 cycles 1 1 : Setting prohibited	R/W
b4	WRCYC0			
b3 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	—
b0	RDCYC	RAM Read Cycle Set Bit	Sets the RAM read cycle of the RAM F bus (the fetch bus) and the RAM M bus (the memory bus). H'78 is written to the upper byte (RAKEY) simultaneously. 0: Set read cycle to 2 cycles (initial setting) 1: Set read cycle to 1 cycle	R/W

Note: To avoid erroneous rewriting, the way of writing data to this register is different from that to other general registers. For details, see section 27.2.8, Notes on Register Access.

Before changing the values in the RAMACYC register, first disable the protection with the SPR0 register. The RAMACYC register sets the read or write cycle of the RAM F bus (the fetch bus) and the RAM M bus (the memory bus). The WRCYC and RDCYC bits are initialized by a reset.

To write the RAMACYC register, word size data with upper byte as H'78 and lower byte as write data must be written. The RAMACYC register can be written to in words, and can be read in bytes or words.

Do not write to the RAMACYC register while RAM is being accessed. When rewriting the RAMACYC register, all the RAMEn bits in the RAM enable control register (RAMEN0) must be cleared to 0 and access to RAM must be disabled.

Also, do not place an instruction to access to the on-chip RAM immediately after an instruction to write to the RAMACYC register; otherwise normal access is not guaranteed.

When rewriting bits WRCYC1, WRCYC0, and RDCYC, execute an instruction to read from the RAMACYC register and five or more NOP instructions immediately after the instruction for writing to the RAMACYC register.

27.2.8 Notes on Register Access

The way of writing data to the RAM enable control register (RAMEN0), RAM write enable control register (RAMWEN0), RAM ECC enable control register (RAMECC), and RAM access cycle set register (RAMACYC) is different from that to other general registers. This is because these registers are not to be easily rewritten.
To write these registers, use the following ways.

(1) To write data to the RAMEN0 register, transfer data with upper byte as H'96 and lower byte as write data.
(2) To write data to the RAMWEN0 register, transfer data with upper byte as H'69 and lower byte as write data.
(3) To write data to the RAMECC register, transfer data with upper byte as H'76 and lower byte as write data.
(4) To write data to the RAMACYC register, transfer data with upper byte as H'78 and lower byte as write data.

In addition, note that the RAMEN0, RAMWEN0, RAMECC, and RAMACYC registers must be written in words. These registers cannot be written in byte or longword instructions. As shown in Figure 27.3, Write data to the upper bytes in the RAMEN0, RAMWEN0, RAMECC, and RAMACYC.

When the upper bytes (bits 15 to 8) in the RAMEN0, RAMWEN0, RAMECC, and RAMACYC registers are read, the read value is always H'00.

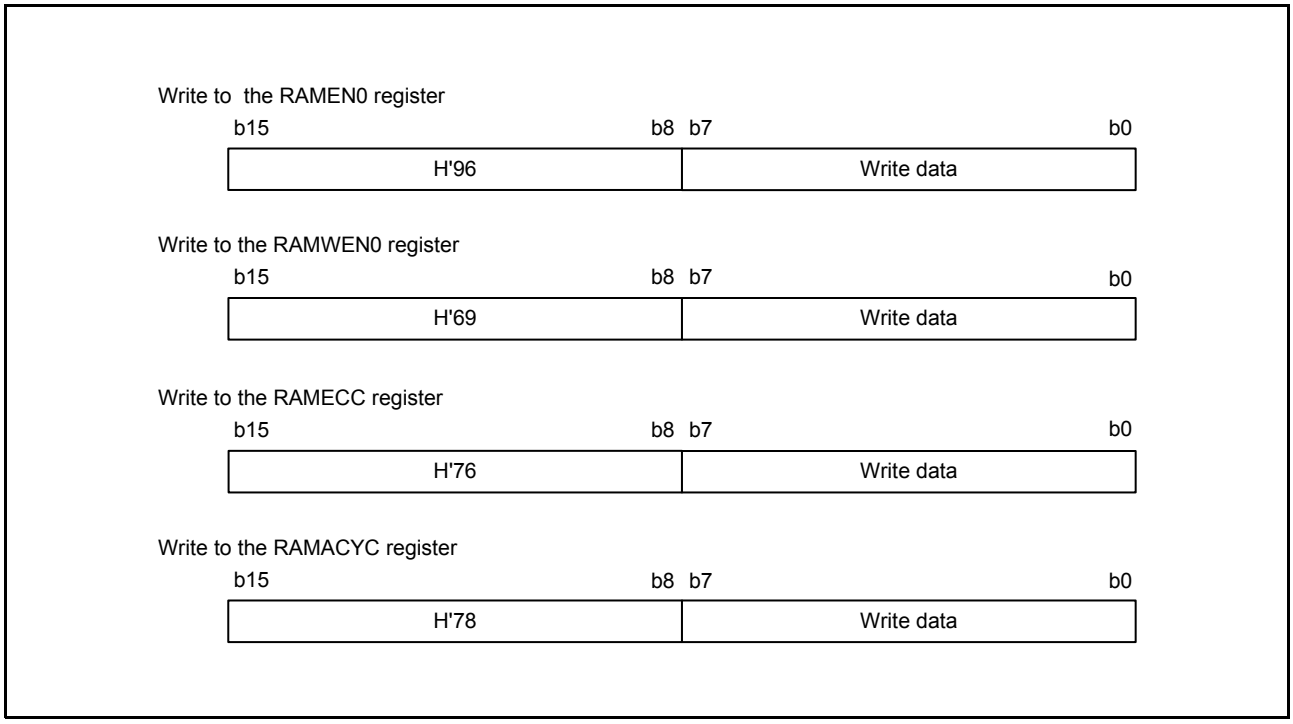


Figure 27.3 Write Data to RAMEN0, RAMWEN0, RAMECC, and RAMACYC

27.3 Operations

(1) Access to the on-chip RAM is controlled by the RAM enable control register (RAMEN0) and RAM write enable control register (RAMWEN0).

(2) Accessing each area of the on-chip RAM is enabled or disabled by the RAMEn (n = 0 to 3) bit in the RAM enable control register (RAMEN0).

(3) When the RAMEn bit in the RAMEN0 register is cleared to 0, the on-chip RAM cannot be accessed. In this case, values read from the on-chip RAM are undefined and the on-chip RAM cannot be modified.

(4) Writing to each area of the on-chip RAM is enabled or disabled by the RAMWEn bits in RAM write enable control register (RAMWEN0).

(5) The ECC error correction function can be set enabled or disabled by register settings. The function is initially set enabled. (the RECCA bit in the RAMECC register = 0)
When the ECC error correction function is set enabled (the RECCA bit in the RAMECC register = 0), 1-bit error correction and 2-bit error detection can be performed. Upon correction of a 1-bit error and/or detection of a 2-bit error, flags (in the RAM error status register) are set to indicate their occurrence. When the ECC error correction is set disabled (the RECCA bit in the RAMECC register = 1), a flag (in the RAM error status register) is set upon occurrence of a parity error.

(6) Whenever the RAM error status register is set, an interrupt can be generated. The interrupt generation can be set enabled or disabled via the RAM error interrupt control register (RAMINT).

27.4 RAM Data Retention

27.4.1 Data Retention at Reset

If a low level signal is input on the RESET# pin from an external device while this LSI is in operation, this LSI enters the hardware reset. In this case, if the on-chip RAM is accessed, data in the RAM address being accessed may be destroyed because the bus cycle cannot be completed normally.

Since it is difficult to input reset signals from the external devices while avoiding accesses to the on-chip RAM. Accordingly, to retain all data items in the on-chip RAM at reset, invalidate the RAM by the RAM enable control register (RAMEN).

However, data in the on-chip RAM is not retained when the operation enters boot mode or user boot mode from the hardware reset since the RAM is cannot be disabled as occupied by programs embedded in the LSI.

27.5 Notes on RAM Control

27.5.1 Page Conflict

If the same page is accessed by the different buses simultaneously, a page conflict occurs. Each of those accesses is handled in such priority scheme as: CPU bus (highest), M bus (middle), F bus (lowest).
In this case, each access is completed normally but this conflict degrades the memory access efficiency. To avoid this conflict, it is recommended to take preventative measures by software. For example, accessing different memory or different pages using different buses can avoid page conflict.

27.5.2 State After Turning on Power

After turning on the power, all data items in the on-chip RAM including ECC correction data and parity are undefined. Accordingly, correspondence among RAM data, error correction data, and parity may not be correct.
To enable correct (initialize) correspondence among RAM data, error correction data, and parity after power-on, write longword (32 bits) data to all on-chip RAM areas. If RAM is read without initialization, a RAM error may occur. Note that no RAM error occurs in RAM writes.

28. Advanced User Debugger-II (AUD-II)

The AUD-II provides functionality that aids the user to perform simple debugging of user programs in the state in which product chips are installed. The AUD-II can access the modules connected to the internal bus by controlling special pins.

28.1 Overview

- When addresses are written externally to the AUDATA3 to AUDATA0 pins, the data existing at the addresses is output.
- When addresses and data are written externally to the AUDATA3 to AUDATA0 pins, the data is forwarded to the addresses.

Table 28.1 describes the AUD-II I/O pins.

Table 28.1 AUD-II I/O Pins

Pin Name	I/O	Description
AUDMD	Input	For this pin, supply a high-level input. This pin is pulled up internally when nothing is connected.
AUDRST#	Input	If a low-level input is supplied, the AUD-II is reset, and the buffers and processing in the AUD-II are initialized.
AUDCK	Input	This pin is used to input the external clock. Input the clock used for debugging. Make sure that the input frequency does not exceed 20 MHz and the 1/2 frequency of the bus clock. This pin is pulled up internally when nothing is connected.
AUDSYNC#*	Input	The AUD bus command enabling signal is supplied to this pin. While a low-level input is being supplied to this pin, input the DIR command, write/read start addresses, and data to be written to the AUDATA3 to AUDATA0 pins. After the AUD-II has become ready, the read data is output from the AUDATA3 to AUDATA0 pins by supplying a high-level input. This pin is pulled up internally when nothing is connected.
AUDATA3 to AUDATA0	Input/output	The following information is input on a time-division basis: • DIR command • Write/read start addresses • Data to be written (For reading, the read data is output after the level of input to the AUDSYNC# pin has changed to high.) When a read operation is performed, if the DIR command and read start addresses are input externally, the data at the addresses is output after Ready is sent. The output starts after the AUDSYNC# pin input is set high. This pin is pulled up internally when nothing is connected.

Note: * Do not supply a high-level input to this pin until a command is externally input to the AUDATA3 to AUDATA0 pins and necessary data is prepared (until B'0001 is output as the Ready flag).

28.2 Register Descriptions

Table 28.2 lists the AUD-II monitoring function registers.

Table 28.2 List of AUD-II Monitoring Function Registers

Register Name	Symbol	After Reset	Address	Access Size
System protect register 0	SPR0	H'00	H'FF46 E063	8
AUD pin enable register	AUDEN	H'00	H'FF46 E182	8

28.2.1 System Protect Register 0 (SPR0)

Address H'FF46 E063

	b7	b6	b5	b4	b3	b2	b1	b0
After Reset	0	0	0	0	0	0	0	0

Bit	Description	R/W
b7 to b0	When writing: B'1111 0001: Protection disabled Other than B'1111 0001: Protection enabled When reading: b0 bit 0: Protection enabled 1: Protection disabled The b7 to b1 bits are always read as 0.	R/W

The SPR0 register is an 8-bit register that configures the protection function so that registers LOCR, SLCR0, VMCR, VD1LSL, RAMECC, RAMACYC, and AUDEN are not overwritten inadvertently. To change the values of these registers, use the following procedure:

(1) Write H'F1 to the SPR0 register (each register is write-enabled).
(2) Change the values of registers LOCR, SLCR0, VMCR, VD1LSL, RAMECC, RAMACYC, and AUDEN.
(3) Write a value other than H'F1 to the SPR0 register (each register is write-disabled).

28.2.2 AUD Pin Enable Register (AUDEN)

Address H'FF46 E182

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	AUDEN0
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b1	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b0	AUDEN0	AUD Pin Enable Bit 0	0: AUD-II pin functionality disabled 1: AUD-II pin functionality enabled	R/W

Note: To change the value in the AUDEN register, first disable the protection with the SPR0 register.

AUDEN0 Bit

This bit is used to set whether to enable or disable the AUD-II pin functionality.
If this bit is set to 0, the functionality is disabled.
If this bit is set to 1, the functionality is enabled, and the other functionality multiplexed with this functionality is disabled forcibly.
[Applicable pins]

- AUDMD, AUDRST#, AUDCK, AUDSYNC#, and AUDATA3 to AUDATA0 pins

28.3 RAM Monitoring Function

The module connected to the internal bus can be accessed by controlling the special pin.

28.3.1 Communication Protocol

The AUD-II receives AUDATA from the AUDATA3 to AUDATA0 pins when the level of input to the AUDSYNC# pin changes from high to low. The following figure shows the format of AUDATA to be input.

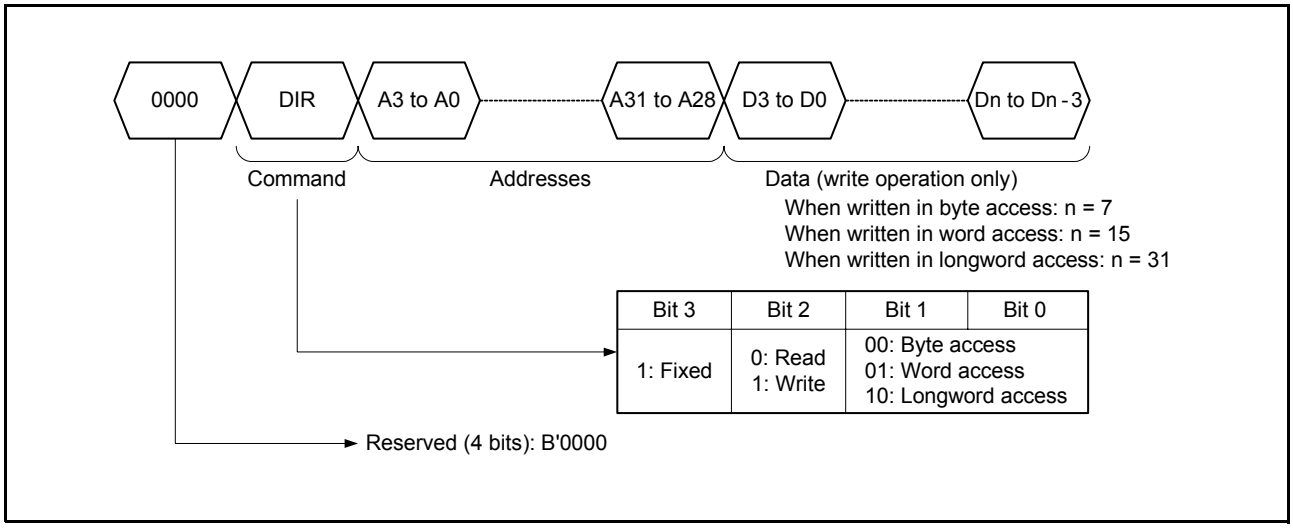


Figure 28.1 AUDATA Input Format

28.3.2 Operation

To use the RAM monitoring function, use the AUD pin enable register to enable the AUD-II pin functionality. After that, the AUD-II functions as the RAM monitoring function when the level of input to the AUDRST# pin is changed from low (AUD-II module is reset) to high (AUD-II module is not reset). Figure 28.2 shows an example of a read operation. Figure 28.3 shows an example of a write operation.

When the level of input to the AUDSYNC# pin changes from high to low, the AUD-II starts receiving data from the AUDATA3 to AUDATA0 pins. When the DIR command, addresses, and data (for a write operation only) are input in the format shown in figure 28.1, the AUD-II starts reading data at the specified addresses or writing data to the specified addresses. While the internal processing is in progress, the AUD-II returns Not Ready (B'0000). When the operation is completed, it returns the Ready (B'0001) flag (figure 28.2 and figure 28.3). Table 28.3 shows the format of the Ready flag.

During a read operation, after this flag is detected, the AUD-II starts outputting data of the specified size when the level of input to the AUDSYNC# pin changes from low to high (figure 28.2). If an invalid command is input as the DIR command, the AUD-II performs no processing, assuming a command error, and sets bit 1 in the Ready flag to 1. If a read or write operation attempted by a command specified in the DIR command causes a bus error, the AUD-II performs no processing, and sets bit 2 in the Ready flag to 1 (figure 28.4).

For read and write operations, the level of input to the AUDSYNC# pin can change from high to low only when certain conditions are satisfied. For details on these conditions, see section 28.3.3 (3) AUDSYNC# Pin Input Timing.

The following shows when a bus error occurs:

1. When address 4n + 1 or 4n + 3 is accessed in word access.
2. When address 4n + 1, 4n + 2, or 4n + 3 is accessed in longword access.

Table 28.3 Ready Flag Format

Bit 3	Bit 2	Bit 1	Bit 0
0: Fixed	0: Normal state 1: Bus error occurred	0: Normal state 1: Command error occurred	0: Not Ready 1: Ready

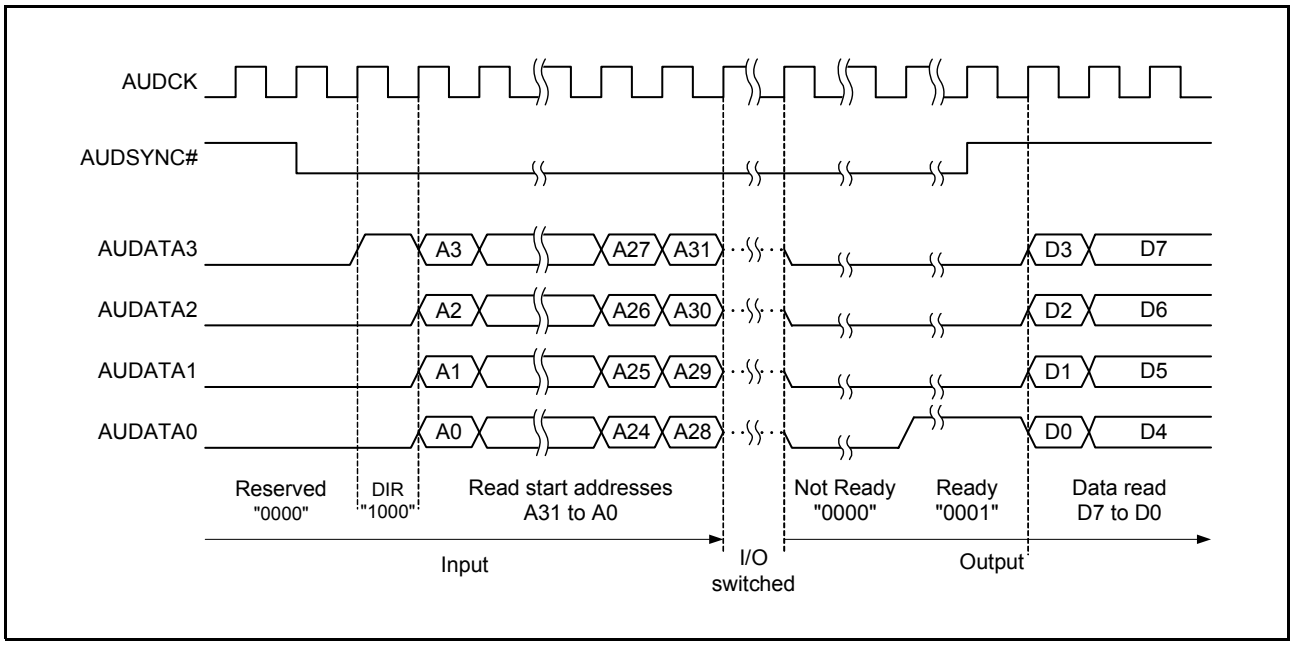


Figure 28.2 Example of Read Operation (in Byte Access)

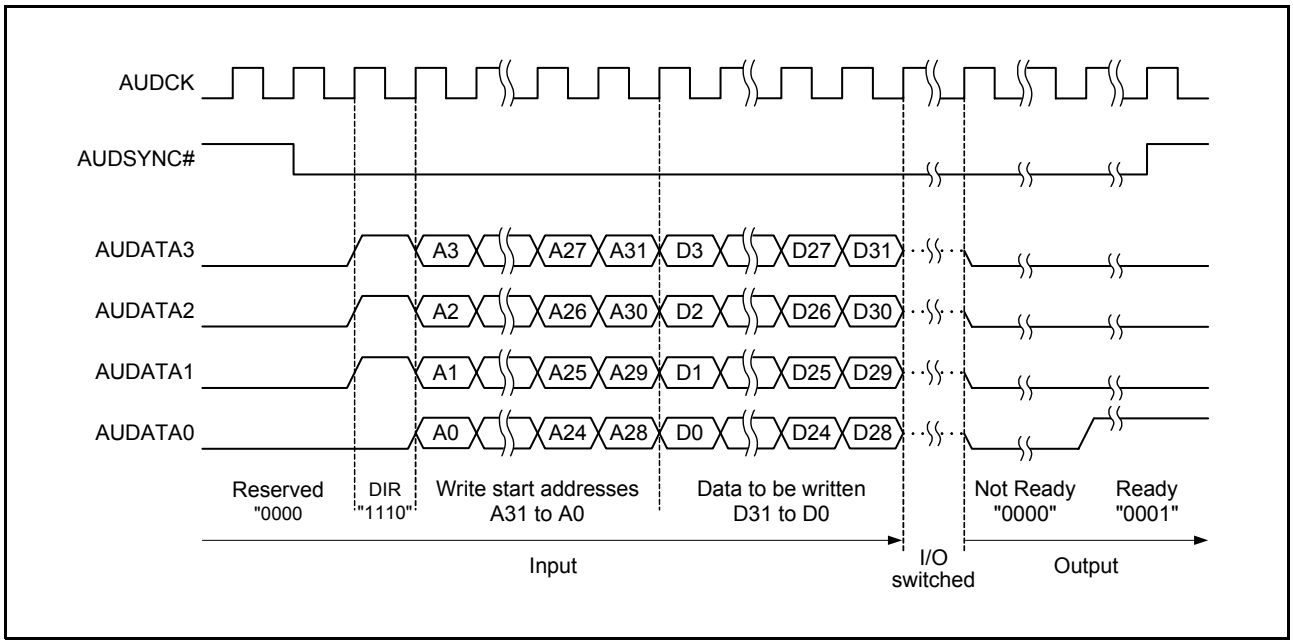


Figure 28.3 Example of Write Operation (in Longword Access)

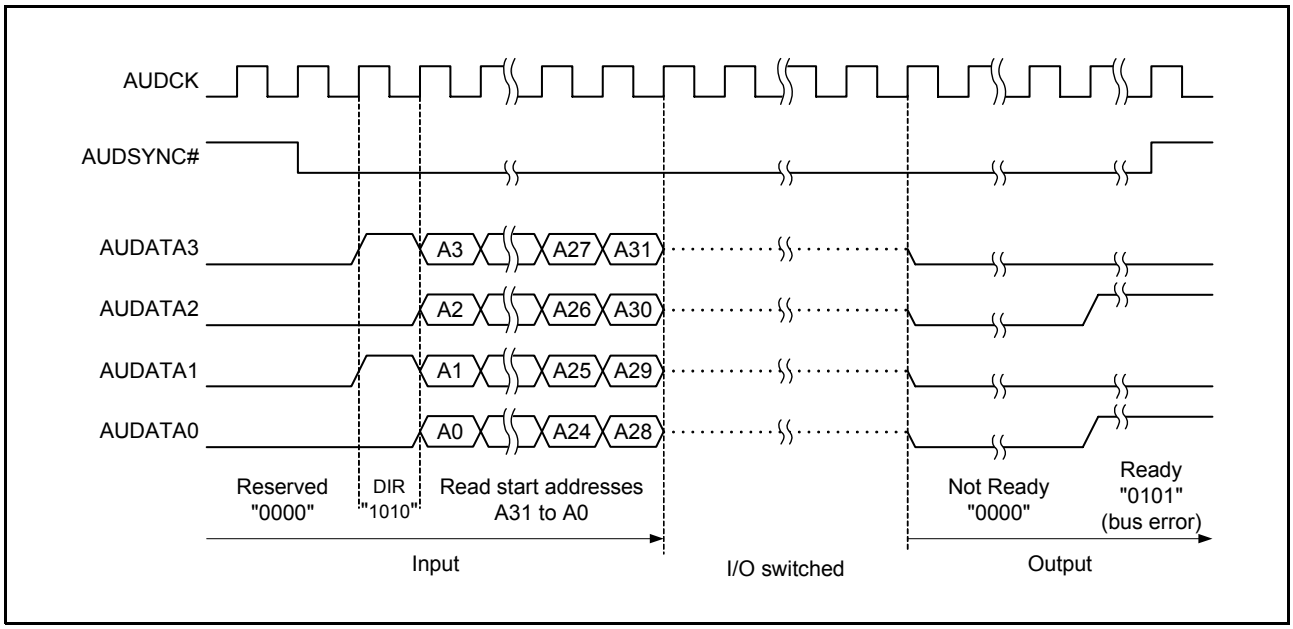


Figure 28.4 Example of Read Operation that Causes Error (in Longword Access)

28.3.3 Notes on the RAM Monitoring Function

(1) Regulations on Initialization of RAM Monitoring Function

The buffers and processing in this debugger are initialized in any of the following conditions:

- When a hardware reset is performed.
- When the watchdog timer is reset.
- When a software reset is performed.
- When low-level input is supplied to the AUDRST# pin.

(2) Regulations on Frequency for AUDCK Pin

The AUDCK pin is the external clock input pin. The frequency input to the pin must not exceed 20 MHz and 1/2 frequency of the bus clock.

(3) AUDSYNC# Pin Input Timing

- After the level of input to the AUDRST# pin has been changed from low to high, supply a high-level input to the AUDSYNC# pin for the period of at least two AUDCK clock cycles.
- After a write operation has ended, supply a high-level input to the AUDSYNC# pin for the period of at least two AUDCK clock cycles.
- When a read operation is performed, supply a high-level input to the AUDSYNC# pin until the last read data unit is sent.
- The level of input to the AUDSYNC# pin must not be changed from low to high until the command is input in AUDATA and Ready is returned.

(4) Other Notes

- The RAM monitoring function can also be used in CPU sleep mode.

29. Memory Protection Unit (MPU)

29.1 Overview

This LSI has a memory protection unit (MPU) that allows setting of 16 areas in the entire address space (H'0000 0000 to H'FFFF FFFF) for each of the bus masters (CPU and DMAC) and specifying the protection attributes for each area. The protection attributes supported for the areas are as follows: Read enabled or disabled, write enabled or disabled, and instruction execution enabled or disabled (for CPU only).

In this section, *n* indicates a value in the range from 0 to 15.

29.2 Specifications

Table 29.1 lists the MPU specifications.

Table 29.1 MPU Specifications

Item	Specification
Areas subject to access protection	H'0000 0000 to H'FFFF FFFF
Number of areas	16 (per bus master)
Minimum area size	4 bytes
Address specification for each area	Specified by the start address and end address
Validity setting for each area	Each area can be set to be valid or invalid.
Protection attribute setting for each area	Operand access: Read enabled or disabled, write enabled or disabled CPU instruction access: Instruction execution enabled or disabled
Background area setting	The protection attributes can be set for background areas (entire address space).
Handling of overlapping areas	The protection attribute of overlapping areas is obtained by logical ORing of the access control bits of the overlapping areas (including the background area).
MPU error handling	CPU operand access: A CPU operand access MPU error exception occurs. CPU instruction access: An exception that is the same as general illegal instruction or slot illegal instruction occurs. DMAC access: A DMAC access MPU error interrupt occurs.
Address at which the MPU error occurred	The address is stored in the error address register for each bus master.
Determination of the MPU error source	The source is stored in the error status register for each bus master.

29.3 Register Description

Table 29.2 to table 29.4 list the MPU registers.

Table 29.2 List of MPU Registers (1)

Register Name	Symbol	After Reset	Address	Access Size
MPUC enable register	MPCMPEN	H'0000 0000	H'FFF7 8000	32
MPUC read access control register	MPCRACR	H'0000 0000	H'FFF7 8010	32
MPUC write access control register	MPCWACR	H'0000 0000	H'FFF7 8014	32
MPUC instruction access control register	MPCIACR	H'0000 0000	H'FFF7 8018	32
MPUC area setting validity register	MPCVLD	H'0000 0000	H'FFF7 801C	32
MPUC background area access control register	MPCACBCR	H'00	H'FFF7 8020	8
MPUC error status clear register	MPCECLR	H'00	H'FFF7 8024	8
MPUC error status register	MPCESR	H'00	H'FFF7 8028	8
MPUC instruction access error address register	MPCERADRI	Undefined	H'FFF7 802C	32
MPUC operand access error address register	MPCERADRO	Undefined	H'FFF7 8030	32
MPUC instruction access hit area register	MPCHITI	H'0000 0000	H'FFF7 8034	32
MPUC operand access hit area register	MPCHITO	H'0000 0000	H'FFF7 8038	32
MPUC area search address register	MPCRSADR	Undefined	H'FFF7 803C	32
MPUC area search operation register	MPCRSOP	H'00	H'FFF7 8040	8
MPUC area 0 start address register	MPCSADR0	Undefined	H'FFF7 8100	32
MPUC area 0 end address register	MPCEADR0	Undefined	H'FFF7 8104	32
MPUC area 0 access control register	MPCACR0	H'00	H'FFF7 8108	8
MPUC area 1 start address register	MPCSADR1	Undefined	H'FFF7 8110	32
MPUC area 1 end address register	MPCEADR1	Undefined	H'FFF7 8114	32
MPUC area 1 access control register	MPCACR1	H'00	H'FFF7 8118	8
MPUC area 2 start address register	MPCSADR2	Undefined	H'FFF7 8120	32
MPUC area 2 end address register	MPCEADR2	Undefined	H'FFF7 8124	32
MPUC area 2 access control register	MPCACR2	H'00	H'FFF7 8128	8
MPUC area 3 start address register	MPCSADR3	Undefined	H'FFF7 8130	32
MPUC area 3 end address register	MPCEADR3	Undefined	H'FFF7 8134	32
MPUC area 3 access control register	MPCACR3	H'00	H'FFF7 8138	8
MPUC area 4 start address register	MPCSADR4	Undefined	H'FFF7 8140	32
MPUC area 4 end address register	MPCEADR4	Undefined	H'FFF7 8144	32
MPUC area 4 access control register	MPCACR4	H'00	H'FFF7 8148	8
MPUC area 5 start address register	MPCSADR5	Undefined	H'FFF7 8150	32
MPUC area 5 end address register	MPCEADR5	Undefined	H'FFF7 8154	32
MPUC area 5 access control register	MPCACR5	H'00	H'FFF7 8158	8
MPUC area 6 start address register	MPCSADR6	Undefined	H'FFF7 8160	32
MPUC area 6 end address register	MPCEADR6	Undefined	H'FFF7 8164	32
MPUC area 6 access control register	MPCACR6	H'00	H'FFF7 8168	8
MPUC area 7 start address register	MPCSADR7	Undefined	H'FFF7 8170	32
MPUC area 7 end address register	MPCEADR7	Undefined	H'FFF7 8174	32

Table 29.3 List of MPU Registers (2)

Register Name	Symbol	After Reset	Address	Access Size
MPUC area 7 access control register	MPCACR7	H'00	H'FFF7 8178	8
MPUC area 8 start address register	MPCSADR8	Undefined	H'FFF7 8180	32
MPUC area 8 end address register	MPCEADR8	Undefined	H'FFF7 8184	32
MPUC area 8 access control register	MPCACR8	H'00	H'FFF7 8188	8
MPUC area 9 start address register	MPCSADR9	Undefined	H'FFF7 8190	32
MPUC area 9 end address register	MPCEADR9	Undefined	H'FFF7 8194	32
MPUC area 9 access control register	MPCACR9	H'00	H'FFF7 8198	8
MPUC area 10 start address register	MPCSADR10	Undefined	H'FFF7 81A0	32
MPUC area 10 end address register	MPCEADR10	Undefined	H'FFF7 81A4	32
MPUC area 10 access control register	MPCACR10	H'00	H'FFF7 81A8	8
MPUC area 11 start address register	MPCSADR11	Undefined	H'FFF7 81B0	32
MPUC area 11 end address register	MPCEADR11	Undefined	H'FFF7 81B4	32
MPUC area 11 access control register	MPCACR11	H'00	H'FFF7 81B8	8
MPUC area 12 start address register	MPCSADR12	Undefined	H'FFF7 81C0	32
MPUC area 12 end address register	MPCEADR12	Undefined	H'FFF7 81C4	32
MPUC area 12 access control register	MPCACR12	H'00	H'FFF7 81C8	8
MPUC area 13 start address register	MPCSADR13	Undefined	H'FFF7 81D0	32
MPUC area 13 end address register	MPCEADR13	Undefined	H'FFF7 81D4	32
MPUC area 13 access control register	MPCACR13	H'00	H'FFF7 81D8	8
MPUC area 14 start address register	MPCSADR14	Undefined	H'FFF7 81E0	32
MPUC area 14 end address register	MPCEADR14	Undefined	H'FFF7 81E4	32
MPUC area 14 access control register	MPCACR14	H'00	H'FFF7 81E8	8
MPUC area 15 start address register	MPCSADR15	Undefined	H'FFF7 81F0	32
MPUC area 15 end address register	MPCEADR15	Undefined	H'FFF7 81F4	32
MPUC area 15 access control register	MPCACR15	H'00	H'FFF7 81F8	8
MPUD enable register	MPDMPEN	H'0000 0000	H'FFF7 8200	32
MPUD read access control register	MPDRACR	H'0000 0000	H'FFF7 8210	32
MPUD write access control register	MPDWACR	H'0000 0000	H'FFF7 8214	32
MPUD area setting validity register	MPDVLD	H'0000 0000	H'FFF7 821C	32
MPUD background area access control register	MPDACBCR	H'00	H'FFF7 8220	8
MPUD error status clear register	MPDECLR	H'00	H'FFF7 8224	8
MPUD error status register	MPDESR	H'00	H'FFF7 8228	8
MPUD error address register	MPDERADR	Undefined	H'FFF7 8230	32
MPUD access hit area register	MPDHIT	H'0000 0000	H'FFF7 8238	32
MPUD area 0 start address register	MPDSADR0	Undefined	H'FFF7 8300	32
MPUD area 0 end address register	MPDEADR0	Undefined	H'FFF7 8304	32
MPUD area 0 access control register	MPDACR0	H'00	H'FFF7 8308	8
MPUD area 1 start address register	MPDSADR1	Undefined	H'FFF7 8310	32
MPUD area 1 end address register	MPDEADR1	Undefined	H'FFF7 8314	32
MPUD area 1 access control register	MPDACR1	H'00	H'FFF7 8318	8
MPUD area 2 start address register	MPDSADR2	Undefined	H'FFF7 8320	32
MPUD area 2 end address register	MPDEADR2	Undefined	H'FFF7 8324	32

Table 29.4 List of MPU Registers (3)

Register Name	Symbol	After Reset	Address	Access Size
MPUD area 2 access control register	MPDACR2	H'00	H'FFF7 8328	8
MPUD area 3 start address register	MPDSADR3	Undefined	H'FFF7 8330	32
MPUD area 3 end address register	MPDEADR3	Undefined	H'FFF7 8334	32
MPUD area 3 access control register	MPDACR3	H'00	H'FFF7 8338	8
MPUD area 4 start address register	MPDSADR4	Undefined	H'FFF7 8340	32
MPUD area 4 end address register	MPDEADR4	Undefined	H'FFF7 8344	32
MPUD area 4 access control register	MPDACR4	H'00	H'FFF7 8348	8
MPUD area 5 start address register	MPDSADR5	Undefined	H'FFF7 8350	32
MPUD area 5 end address register	MPDEADR5	Undefined	H'FFF7 8354	32
MPUD area 5 access control register	MPDACR5	H'00	H'FFF7 8358	8
MPUD area 6 start address register	MPDSADR6	Undefined	H'FFF7 8360	32
MPUD area 6 end address register	MPDEADR6	Undefined	H'FFF7 8364	32
MPUD area 6 access control register	MPDACR6	H'00	H'FFF7 8368	8
MPUD area 7 start address register	MPDSADR7	Undefined	H'FFF7 8370	32
MPUD area 7 end address register	MPDEADR7	Undefined	H'FFF7 8374	32
MPUD area 7 access control register	MPDACR7	H'00	H'FFF7 8378	8
MPUD area 8 start address register	MPDSADR8	Undefined	H'FFF7 8380	32
MPUD area 8 end address register	MPDEADR8	Undefined	H'FFF7 8384	32
MPUD area 8 access control register	MPDACR8	H'00	H'FFF7 8388	8
MPUD area 9 start address register	MPDSADR9	Undefined	H'FFF7 8390	32
MPUD area 9 end address register	MPDEADR9	Undefined	H'FFF7 8394	32
MPUD area 9 access control register	MPDACR9	H'00	H'FFF7 8398	8
MPUD area 10 start address register	MPDSADR10	Undefined	H'FFF7 83A0	32
MPUD area 10 end address register	MPDEADR10	Undefined	H'FFF7 83A4	32
MPUD area 10 access control register	MPDACR10	H'00	H'FFF7 83A8	8
MPUD area 11 start address register	MPDSADR11	Undefined	H'FFF7 83B0	32
MPUD area 11 end address register	MPDEADR11	Undefined	H'FFF7 83B4	32
MPUD area 11 access control register	MPDACR11	H'00	H'FFF7 83B8	8
MPUD area 12 start address register	MPDSADR12	Undefined	H'FFF7 83C0	32
MPUD area 12 end address register	MPDEADR12	Undefined	H'FFF7 83C4	32
MPUD area 12 access control register	MPDACR12	H'00	H'FFF7 83C8	8
MPUD area 13 start address register	MPDSADR13	Undefined	H'FFF7 83D0	32
MPUD area 13 end address register	MPDEADR13	Undefined	H'FFF7 83D4	32
MPUD area 13 access control register	MPDACR13	H'00	H'FFF7 83D8	8
MPUD area 14 start address register	MPDSADR14	Undefined	H'FFF7 83E0	32
MPUD area 14 end address register	MPDEADR14	Undefined	H'FFF7 83E4	32
MPUD area 14 access control register	MPDACR14	H'00	H'FFF7 83E8	8
MPUD area 15 start address register	MPDSADR15	Undefined	H'FFF7 83F0	32
MPUD area 15 end address register	MPDEADR15	Undefined	H'FFF7 83F4	32
MPUD area 15 access control register	MPDACR15	H'00	H'FFF7 83F8	8

29.3.1 MPUC Enable Register (MPCMPEN)

Address H'FFF7 8000

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	KEY [15:0]															
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
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MPEN
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 b16	KEY [15:0]	Key Code Bits	The MPEN bit can be modified only when H'AAAA is written.	W*1
b15 to b1	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b0	MPEN	MPU Enable Bit*2	0 : MPU disabled, register protection canceled 1 : MPU enabled, register protection enabled	R/W

Notes: 1. Data written to this bit is not retained.
2. If an MPU error exception occurs due to CPU access, this bit is automatically set to 0 (MPU disabled, register protection canceled).

Use the following procedure to change the values in registers MPCACBCR, MPCECLR, MPCRSADR, MPCRSOP, MPCSADRn, MPCEADRn, and MPCACRn.

(1) Write H'AAAA 0000 to the MPCMPEN register (to disable the MPU and enable writing to each register).
(2) Change the values in registers MPCACBCR, MPCECLR, MPCRSADR, MPCRSOP, MPCSADRn, MPCEADRn, and MPCACRn.
(3) Write H'AAAA 0001 to the MPCMPEN register (to enable the MPU and disable writing to each register).

MPEN Bit

This bit enables and disables the MPU for the CPU. This bit also sets the register protection function to protect the MPU registers for the CPU from being inadvertently overwritten.

29.3.2 MPUC Read Access Control Register (MPCRACR)

Address H'FFF7 8010

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	KEY [15:0]															
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
	R15	R14	R13	R12	R11	R10	R9	R8	R7	R6	R5	R4	R3	R2	R1	R0
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 b16	KEY [15:0]	Key Code Bits	The R15 to R0 bits can be modified only when H'AAAA is written.	W*
b15	R15	Area 15 Read Access Control Bit	0 : Read disabled 1 : Read enabled	R/W
b14	R14	Area 14 Read Access Control Bit		R/W
b13	R13	Area 13 Read Access Control Bit		R/W
b12	R12	Area 12 Read Access Control Bit		R/W
b11	R11	Area 11 Read Access Control Bit		R/W
b10	R10	Area 10 Read Access Control Bit		R/W
b9	R9	Area 9 Read Access Control Bit		R/W
b8	R8	Area 8 Read Access Control Bit		R/W
b7	R7	Area 7 Read Access Control Bit		R/W
b6	R6	Area 6 Read Access Control Bit		R/W
b5	R5	Area 5 Read Access Control Bit		R/W
b4	R4	Area 4 Read Access Control Bit		R/W
b3	R3	Area 3 Read Access Control Bit		R/W
b2	R2	Area 2 Read Access Control Bit		R/W
b1	R1	Area 1 Read Access Control Bit		R/W
b0	R0	Area 0 Read Access Control Bit		R/W

Note: * Data written to this bit is not retained.

R15 to R0 Bits

These bits enable and disable read access to areas 0 to 15 in the MPU for the CPU. The entity of these bits is the same as the R bit in the MPCACRn register.

29.3.3 MPUC Write Access Control Register (MPCWACR)

Address H'FFF7 8014

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	KEY [15:0]															
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
	W15	W14	W13	W12	W11	W10	W9	W8	W7	W6	W5	W4	W3	W2	W1	W0
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 b16	KEY [15:0]	Key Code Bits	The W15 to W0 bits can be modified only when H'AAAA is written.	W*
b15	W15	Area 15 Write Access Control Bit	0 : Write disabled 1 : Write enabled	R/W
b14	W14	Area 14 Write Access Control Bit		R/W
b13	W13	Area 13 Write Access Control Bit		R/W
b12	W12	Area 12 Write Access Control Bit		R/W
b11	W11	Area 11 Write Access Control Bit		R/W
b10	W10	Area 10 Write Access Control Bit		R/W
b9	W9	Area 9 Write Access Control Bit		R/W
b8	W8	Area 8 Write Access Control Bit		R/W
b7	W7	Area 7 Write Access Control Bit		R/W
b6	W6	Area 6 Write Access Control Bit		R/W
b5	W5	Area 5 Write Access Control Bit		R/W
b4	W4	Area 4 Write Access Control Bit		R/W
b3	W3	Area 3 Write Access Control Bit		R/W
b2	W2	Area 2 Write Access Control Bit		R/W
b1	W1	Area 1 Write Access Control Bit		R/W
b0	W0	Area 0 Write Access Control Bit		R/W

Note: * Data written to this bit is not retained.

W15 to W0 Bits

These bits enable and disable write access to areas 0 to 15 in the MPU for the CPU. The entity of these bits is the same as the W bit in the MPCACRn register.

29.3.4 MPUC Instruction Access Control Register (MPCIACR)

Address H'FFF7 8018

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	KEY [15:0]															
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
	I15	I14	I13	I12	I11	I10	I9	I8	I7	I6	I5	I4	I3	I2	I1	I0
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 b16	KEY [15:0]	Key Code Bits	The I15 to I0 bits can be modified only when H'AAAA is written.	W*
b15	I15	Area 15 Instruction Access Control Bit	0 : Instruction execution disabled 1 : Instruction execution enabled	R/W
b14	I14	Area 14 Instruction Access Control Bit		R/W
b13	I13	Area 13 Instruction Access Control Bit		R/W
b12	I12	Area 12 Instruction Access Control Bit		R/W
b11	I11	Area 11 Instruction Access Control Bit		R/W
b10	I10	Area 10 Instruction Access Control Bit		R/W
b9	I9	Area 9 Instruction Access Control Bit		R/W
b8	I8	Area 8 Instruction Access Control Bit		R/W
b7	I7	Area 7 Instruction Access Control Bit		R/W
b6	I6	Area 6 Instruction Access Control Bit		R/W
b5	I5	Area 5 Instruction Access Control Bit		R/W
b4	I4	Area 4 Instruction Access Control Bit		R/W
b3	I3	Area 3 Instruction Access Control Bit		R/W
b2	I2	Area 2 Instruction Access Control Bit		R/W
b1	I1	Area 1 Instruction Access Control Bit		R/W
b0	I0	Area 0 Instruction Access Control Bit		R/W

Note: * Data written to this bit is not retained.

I15 to I0 Bits

These bits enable and disable instruction access to areas 0 to 15 in the MPU for the CPU. The entity of these bits is the same as the I bit in the MPCACRn register.

29.3.5 MPUC Area Setting Validity Register (MPCVLD)

Address H'FFF7 801C

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	KEY [15:0]															
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
	V15	V14	V13	V12	V11	V10	V9	V8	V7	V6	V5	V4	V3	V2	V1	V0
After Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Symbol	Bit Name	Description	R/W
b31 to b16	KEY [15:0]	Key Code Bits	The V15 to V0 bits can be modified only when H'AAAA is written.	W*
b15	V15	Area 15 Validity Bit	0 : Settings invalid 1 : Settings valid	R/W
b14	V14	Area 14 Validity Bit		R/W
b13	V13	Area 13 Validity Bit		R/W
b12	V12	Area 12 Validity Bit		R/W
b11	V11	Area 11 Validity Bit		R/W
b10	V10	Area 10 Validity Bit		R/W
b9	V9	Area 9 Validity Bit		R/W
b8	V8	Area 8 Validity Bit		R/W
b7	V7	Area 7 Validity Bit		R/W
b6	V6	Area 6 Validity Bit		R/W
b5	V5	Area 5 Validity Bit		R/W
b4	V4	Area 4 Validity Bit		R/W
b3	V3	Area 3 Validity Bit		R/W
b2	V2	Area 2 Validity Bit		R/W
b1	V1	Area 1 Validity Bit		R/W
b0	V0	Area 0 Validity Bit		R/W

Note: * Data written to this bit is not retained.

V15 to V0 Bits

These bits validate or invalidate the settings of areas 0 to 15 in the MPU for the CPU. The entity of these bits is the same as the V bit in the MPCACRn register.

29.3.6 MPUC Background Area Access Control Register (MPCACBCR)

Address H'FFF7 8020

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	R	W	I	—
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b4	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b3	R	Read Access Control Bit	0 : Read disabled 1 : Read enabled	R/W
b2	W	Write Access Control Bit	0 : Write disabled 1 : Write enabled	R/W
b1	I	Instruction Access Control Bit	0 : Instruction execution disabled 1 : Instruction execution enabled	R/W
b0	—	Reserved	This bit is always read as 0. The write value should always be 0.	R

Before changing the value in the MPCACBCR register, use the MPCMPEN register to cancel the protection.

R Bit

This bit enables and disables read access to the background area in the MPU for the CPU.

W Bit

This bit enables and disables write access to the background area in the MPU for the CPU.

I Bit

This bit enables and disables instruction access to the background area in the MPU for the CPU.

29.3.7 MPUC Error Status Clear Register (MPCECLR)

Address H'FFF7 8024

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	CLR
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b1	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b0	CLR	Error Status Clear Bit	Writing: 0 : Nothing is performed. 1 : All bits in the MPCESR, MPCHITI, and MPCHITO registers are set to 0. Reading: 0 is read.	R/W

Before changing the value in the MPCECLR register, use the MPCMPEN register to cancel the protection.

29.3.8 MPUC Error Status Register (MPCESR)

Address H'FFF7 8028

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	RER	WER	IER	—
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b4	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b3	RER	Read Access Error Flag*	0 : No read access error occurred. 1 : A read access error occurred.	R
b2	WER	Write Access Error Flag*	0 : No write access error occurred. 1 : A write access error occurred.	R
b1	IER	Instruction Access Error Flag*	0 : No instruction execution error occurred. 1 : An instruction execution error occurred.	R
b0	—	Reserved	This bit is always read as 0. The write value should always be 0.	R

Note: * This bit is set to 0 by setting the CLR bit in the MPCECLR register to 1.

The MPCESR register is an 8-bit read-only register.

RER Flag

This flag indicates whether an MPU error has occurred due to a CPU operand read access.
If the WER flag is set to 1, this flag is not set to 1 even if an MPU error occurs due to a CPU operand read access.

WER Flag

This flag indicates whether an MPU error has occurred due to a CPU operand write access.
If the RER flag is set to 1, this flag is not set to 1 even if an MPU error occurs due to a CPU operand write access.

IER Flag

This flag indicates whether an MPU error has occurred due to a CPU instruction access.

29.3.9 MPUC Instruction Access Error Address Register (MPCERADRI)

Address H'FFF7 802C

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ERADR[31:16]															
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
	ERADR[15:0]															
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
b31 to b0	ERADR [31:0]	Error Address Bits	These bits retain the address of the CPU instruction access that caused an MPU error for the first time after the CLR bit in the MPCECLR register was set to 1 and the IER bit in the MPCESR register was set to 0.	R

The MPCERADRI register is a 32-bit read-only register.

29.3.10 MPUC Operand Access Error Address Register (MPCERADRO)

Address H'FFF7 8030

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ERADR[31:16]															
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
	ERADR[15:0]															
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
b31 to b0	ERADR [31:0]	Error Address Bits	These bits retain the address of the CPU operand access (read or write) that caused an MPU error for the first time after the CLR bit in the MPCECLR register was set to 1 and the RER and WER flags in the MPCESR register were set to 0.	R

The MPCERADRO register is a 32-bit read-only register.

29.3.11 MPUC Instruction Access Hit Area Register (MPCHITI)

Address H'FFF7 8034

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	H15	H14	H13	H12	H11	H10	H9	H8	H7	H6	H5	H4	H3	H2	H1	H0
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
	—	—	—	—	—	—	—	—	—	—	—	—	R	W	I	—
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	H15	Area 15 Instruction Access Hit Indication Flag*	0 : No instruction access error 1 : Instruction access error	R
b30	H14	Area 14 Instruction Access Hit Indication Flag*		R
b29	H13	Area 13 Instruction Access Hit Indication Flag*		R
b28	H12	Area 12 Instruction Access Hit Indication Flag*		R
b27	H11	Area 11 Instruction Access Hit Indication Flag*		R
b26	H10	Area 10 Instruction Access Hit Indication Flag*		R
b25	H9	Area 9 Instruction Access Hit Indication Flag*		R
b24	H8	Area 8 Instruction Access Hit Indication Flag*		R
b23	H7	Area 7 Instruction Access Hit Indication Flag*		R
b22	H6	Area 6 Instruction Access Hit Indication Flag*		R
b21	H5	Area 5 Instruction Access Hit Indication Flag*		R
b20	H4	Area 4 Instruction Access Hit Indication Flag*		R
b19	H3	Area 3 Instruction Access Hit Indication Flag*		R
b18	H2	Area 2 Instruction Access Hit Indication Flag*		R
b17	H1	Area 1 Instruction Access Hit Indication Flag*		R
b16	H0	Area 0 Instruction Access Hit Indication Flag*		R
b15 to b4	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b3	R	Read Access Control Bit Retention Flag for Instruction Access Hit Area*	0 : Read disabled 1 : Read enabled	R
b2	W	Write Access Control Bit Retention Flag for Instruction Access Hit Area*	0 : Write disabled 1 : Write enabled	R
b1	I	Instruction Access Control Bit Retention Flag for Instruction Access Hit Area*	0 : Instruction execution disabled 1 : Instruction execution enabled	R
b0	—	Reserved	This bit is always read as 0. The write value should always be 0.	R

Note: * This bit is set to 0 by setting the CLR bit in the MPCECLR register to 1.

The MPCHITI register is a 32-bit read-only register.

H15 to H0 Flags

These bits indicate the area in which an CPU instruction access error occurred for the first time after the CLR bit in the MPCECLR register was set to 1 and the IER flag in the MPCESR register was set to 0. These bits are not set to 1 if the error occurred only in the background area.

R Flag

This bit retains the read access control bit of the area in which a CPU instruction access error occurred. If the error occurred in overlapping areas, this bit retains the logical OR of the read access control bits of the relevant areas (including the background area).

W Flag

This bit retains the write access control bit of the area in which a CPU instruction access error occurred. If the error occurred in overlapping areas, this bit retains the logical OR of the write access control bits of the relevant areas (including the background area).

I Flag

This bit retains the instruction access control bit of the area in which a CPU instruction access error occurred. If the error occurred in overlapping areas, this bit retains the logical OR of the instruction access control bits of the relevant areas (including the background area).

29.3.12 MPUC Operand Access Hit Area Register (MPCHITO)

Address H'FFF7 8038

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	H15	H14	H13	H12	H11	H10	H9	H8	H7	H6	H5	H4	H3	H2	H1	H0
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
	—	—	—	—	—	—	—	—	—	—	—	—	R	W	I	—
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	H15	Area 15 Operand Access Hit Indication Flag*	0 : No operand access error or no area search hit 1 : Operand access error or area search hit	R
b30	H14	Area 14 Operand Access Hit Indication Flag*		R
b29	H13	Area 13 Operand Access Hit Indication Flag*		R
b28	H12	Area 12 Operand Access Hit Indication Flag*		R
b27	H11	Area 11 Operand Access Hit Indication Flag*		R
b26	H10	Area 10 Operand Access Hit Indication Flag*		R
b25	H9	Area 9 Operand Access Hit Indication Flag*		R
b24	H8	Area 8 Operand Access Hit Indication Flag*		R
b23	H7	Area 7 Operand Access Hit Indication Flag*		R
b22	H6	Area 6 Operand Access Hit Indication Flag*		R
b21	H5	Area 5 Operand Access Hit Indication Flag*		R
b20	H4	Area 4 Operand Access Hit Indication Flag*		R
b19	H3	Area 3 Operand Access Hit Indication Flag*		R
b18	H2	Area 2 Operand Access Hit Indication Flag*		R
b17	H1	Area 1 Operand Access Hit Indication Flag*		R
b16	H0	Area 0 Operand Access Hit Indication Flag*		R
b15 to b4	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b3	R	Read Access Control Bit Retention Flag for Operand Access Hit Area*	0 : Read disabled 1 : Read enabled	R
b2	W	Write Access Control Bit Retention Flag for Operand Access Hit Area*	0 : Write disabled 1 : Write enabled	R
b1	I	Instruction Access Control Bit Retention Flag for Operand Access Hit Area*	0 : Instruction execution disabled 1 : Instruction execution enabled	R
b0	—	Reserved	This bit is always read as 0. The write value should always be 0.	R

Note: * This bit is set to 0 by setting the CLR bit in the MPCECLR register to 1.

The MPCHITO register is a 32-bit read-only register.

H15 to H0 Flags

These bits indicate either the area in which an CPU operand access error occurred for the first time after the CLR bit in the MPCECLR register was set to 1 and the RER and WER flags in the MPCESR register were set to 0, or the area in which a search hit occurred. These bits are not set to 1 if the access error occurred only in the background area.

R Flag

This bit retains the read access control bit of the area in which a CPU operand access error or a search hit occurred. If the error or search hit occurred in overlapping areas, this bit retains the logical OR of the read access control bits of the

relevant areas (including the background area).

W Flag

This bit retains the write access control bit of the area in which a CPU operand access error or a search hit occurred. If the error or search hit occurred in overlapping areas, this bit retains the logical OR of the write access control bits of the relevant areas (including the background area).

I Flag

This bit retains the instruction access control bit of the area in which a CPU operand access error or a search hit occurred. If the error or search hit occurred in overlapping areas, this bit retains the logical OR of the instruction access control bits of the relevant areas (including the background area).

29.3.13 MPUC Area Search Address Register (MPCRSADR)

Address H'FFF7 803C

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	RSADA[31:16]															
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
	RSADA[15:0]															
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
b31 to b0	RSADA [31:0]	Area Search Address Bits	These bits set the address that will be compared with the start address and end address of each MPU area for the CPU area during area search operation.	R/W

Before changing the value in the MPCRSADR register, use the MPCMPEN register to cancel the protection.

29.3.14 MPUC Area Search Operation Register (MPCRSOP)

Address H'FFF7 8040

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	S
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b1	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b0	S	Area Search Operation Bit	Writing: 0 : Nothing is performed. 1 : Area search operation is performed. Reading: 0 is read.	R/W

Before changing the value in the MPCRSOP register, use the MPCMPEN register to cancel the protection.

S Bit

This bit searches for the MPU area for the CPU that includes the address set in the MPCRSADR register. The search result is stored in the MPCHITO register.

29.3.15 MPUC Area n Start Address Register (MPCSADRn) (n = 0 to 15)

Address: MPCSADR0: H'FFF7 8100, MPCSADR1: H'FFF7 8110, MPCSADR2: H'FFF7 8120, MPCSADR3: H'FFF7 8130, MPCSADR4: H'FFF7 8140, MPCSADR5: H'FFF7 8150, MPCSADR6: H'FFF7 8160, MPCSADR7: H'FFF7 8170, MPCSADR8: H'FFF7 8180, MPCSADR9: H'FFF7 8190, MPCSADR10: H'FFF7 81A0, MPCSADR11: H'FFF7 81B0, MPCSADR12: H'FFF7 81C0, MPCSADR13: H'FFF7 81D0, MPCSADR14: H'FFF7 81E0, MPCSADR15: H'FFF7 81F0

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	SADR[31:16]															
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
	SADR[15:2]														—	—
After Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b2	SADR [31:2]	Area n Start Address Bits	These bits set the upper 30 bits of the start address of area n in the MPU for the CPU.	R/W
b1, b0	—	Reserved	These bits are always read as 0. The write value should always be 0.	R

Before changing the value in the MPCSADRn register, use the MPCMPEN register to cancel the protection.

29.3.16 MPUC Area n End Address Register (MPCEADRn) (n = 0 to 15)

Address: MPCEADR0: H'FFF7 8104, MPCEADR1: H'FFF7 8114, MPCEADR2: H'FFF7 8124, MPCEADR3: H'FFF7 8134, MPCEADR4: H'FFF7 8144, MPCEADR5: H'FFF7 8154, MPCEADR6: H'FFF7 8164, MPCEADR7: H'FFF7 8174, MPCEADR8: H'FFF7 8184, MPCEADR9: H'FFF7 8194, MPCEADR10: H'FFF7 81A4, MPCEADR11: H'FFF7 81B4, MPCEADR12: H'FFF7 81C4, MPCEADR13: H'FFF7 81D4, MPCEADR14: H'FFF7 81E4, MPCEADR15: H'FFF7 81F4

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	EADR[31:16]															
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
	EADR[15:2]														—	—
After Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b2	EADR [31:2]	Area n End Address	These bits set the upper 30 bits of the end address of area n in the MPU for the CPU.	R/W
b1, b0	—	Reserved	These bits are always read as 0. The write value should always be 0.	R

Before changing the value in the MPCEADRn register, use the MPCMPEN register to cancel the protection.

29.3.17 MPUC Area n Access Control Register (MPCACRn) (n = 0 to 15)

Address: MPCACR0: H'FFF7 8108, MPCACR1: H'FFF7 8118, MPCACR2: H'FFF7 8128, MPCACR3: H'FFF7 8138, MPCACR4: H'FFF7 8148, MPCACR5: H'FFF7 8158, MPCACR6: H'FFF7 8168, MPCACR7: H'FFF7 8178, MPCACR8: H'FFF7 8188, MPCACR9: H'FFF7 8198, MPCACR10: H'FFF7 81A8, MPCACR11: H'FFF7 81B8, MPCACR12: H'FFF7 81C8, MPCACR13: H'FFF7 81D8, MPCACR14: H'FFF7 81E8, MPCACR15: H'FFF7 81F8

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	R	W	I	V
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b4	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b3	R	Read Access Control Bit	0 : Read disabled 1 : Read enabled	R/W
b2	W	Write Access Control Bit	0 : Write disabled 1 : Write enabled	R/W
b1	I	Instruction Access Control Bit	0 : Instruction execution disabled 1 : Instruction execution enabled	R/W
b0	V	Validity Bit	0 : Area settings invalid 1 : Area settings valid	R/W

Before changing the value in the MPCACRn register, use the MPCMPEN register to cancel the protection.

R Bit

This bit enables and disables read access to area n in the MPU for the CPU.

W Bit

This bit enables and disables write access to area n in the MPU for the CPU.

I Bit

This bit enables and disables instruction access to area n in the MPU for the CPU.

V Bit

This bit validates or invalidates the settings of area n in the MPU for the CPU.

29.3.18 MPUD Enable Register (MPDMPEN)

Address H'FFF7 8200

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	KEY [15:0]															
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
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MPEN
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 b16	KEY [15:0]	Key Code Bits	The MPEN bit can be modified only when H'AAAA is written.	W*
b15 to b1	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b0	MPEN	MPU Enable Bit	0 : MPU disabled, register protection canceled 1 : MPU enabled, register protection enabled	R/W

Note: * Data written to this bit is not retained.

Use the following procedure to change the values in registers MPDACBCR, MPDECLR, MPDSADRn, MPDEADRn, and MPDACRn.

(1) Write H'AAAA 0000 to the MPDMPEN register (to disable MPU and enable writing to each register).
(2) Change the values in registers MPDACBCR, MPDECLR, MPDSADRn, MPDEADRn, and MPDACRn.
(3) Write H'AAAA 0001 to the MPDMPEN register (to enable MPU and disable writing to each register).

MPEN Bit

This bit enables and disables the MPU for the DMAC. This bit also sets the register protection function to protect the MPU registers for the DMAC from being inadvertently overwritten.

29.3.19 MPUD Read Access Control Register (MPDRACR)

Address H'FFF7 8210

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	KEY [15:0]															
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
	R15	R14	R13	R12	R11	R10	R9	R8	R7	R6	R5	R4	R3	R2	R1	R0
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 b16	KEY [15:0]	Key Code Bits	The R15 to R0 bits can be modified only when H'AAAA is written.	W*
b15	R15	Area 15 Read Access Control Bit	0 : Read disabled 1 : Read enabled	R/W
b14	R14	Area 14 Read Access Control Bit		R/W
b13	R13	Area 13 Read Access Control Bit		R/W
b12	R12	Area 12 Read Access Control Bit		R/W
b11	R11	Area 11 Read Access Control Bit		R/W
b10	R10	Area 10 Read Access Control Bit		R/W
b9	R9	Area 9 Read Access Control Bit		R/W
b8	R8	Area 8 Read Access Control Bit		R/W
b7	R7	Area 7 Read Access Control Bit		R/W
b6	R6	Area 6 Read Access Control Bit		R/W
b5	R5	Area 5 Read Access Control Bit		R/W
b4	R4	Area 4 Read Access Control Bit		R/W
b3	R3	Area 3 Read Access Control Bit		R/W
b2	R2	Area 2 Read Access Control Bit		R/W
b1	R1	Area 1 Read Access Control Bit		R/W
b0	R0	Area 0 Read Access Control Bit		R/W

Note: * Data written to this bit is not retained.

R15 to R0 Bits

These bits enable and disable read access to areas 0 to 15 in the MPU for the DMAC. The entity of these bits is the same as the R bit in the MPDACRn register.

29.3.20 MPUD Write Access Control Register (MPDWACR)

Address H'FFF7 8214

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	KEY [15:0]															
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
	W15	W14	W13	W12	W11	W10	W9	W8	W7	W6	W5	W4	W3	W2	W1	W0
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 b16	KEY [15:0]	Key Code Bits	The W15 to W0 bits can be modified only when H'AAAA is written.	W*
b15	W15	Area 15 Write Access Control Bit	0 : Write disabled 1 : Write enabled	R/W
b14	W14	Area 14 Write Access Control Bit		R/W
b13	W13	Area 13 Write Access Control Bit		R/W
b12	W12	Area 12 Write Access Control Bit		R/W
b11	W11	Area 11 Write Access Control Bit		R/W
b10	W10	Area 10 Write Access Control Bit		R/W
b9	W9	Area 9 Write Access Control Bit		R/W
b8	W8	Area 8 Write Access Control Bit		R/W
b7	W7	Area 7 Write Access Control Bit		R/W
b6	W6	Area 6 Write Access Control Bit		R/W
b5	W5	Area 5 Write Access Control Bit		R/W
b4	W4	Area 4 Write Access Control Bit		R/W
b3	W3	Area 3 Write Access Control Bit		R/W
b2	W2	Area 2 Write Access Control Bit		R/W
b1	W1	Area 1 Write Access Control Bit		R/W
b0	W0	Area 0 Write Access Control Bit		R/W

Note: * Data written to this bit is not retained.

W15 to W0 Bits

These bits enable and disable write access to areas 0 to 15 in the MPU for the DMAC. The entity of these bits is the same as the W bit in the MPDACRn register.

29.3.21 MPUD Area Setting Validity Register (MPDVLD)

Address H'FFF7 821C

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	KEY [15:0]															
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
	V15	V14	V13	V12	V11	V10	V9	V8	V7	V6	V5	V4	V3	V2	V1	V0
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 b16	KEY [15:0]	Key Code Bits	The V15 to V0 bits can be modified only when H'AAAA is written.	W*
b15	V15	Area 15 Validity Bit	0 : Settings invalid 1 : Settings valid	R/W
b14	V14	Area 14 Validity Bit		R/W
b13	V13	Area 13 Validity Bit		R/W
b12	V12	Area 12 Validity Bit		R/W
b11	V11	Area 11 Validity Bit		R/W
b10	V10	Area 10 Validity Bit		R/W
b9	V9	Area 9 Validity Bit		R/W
b8	V8	Area 8 Validity Bit		R/W
b7	V7	Area 7 Validity Bit		R/W
b6	V6	Area 6 Validity Bit		R/W
b5	V5	Area 5 Validity Bit		R/W
b4	V4	Area 4 Validity Bit		R/W
b3	V3	Area 3 Validity Bit		R/W
b2	V2	Area 2 Validity Bit		R/W
b1	V1	Area 1 Validity Bit		R/W
b0	V0	Area 0 Validity Bit		R/W

Note: * Data written to this bit is not retained.

V15 to V0 Bits

These bits validate or invalidate the settings of areas 0 to 15 in the MPU for the DMAC. The entity of these bits is the same as the V bit in the MPDACRn register.

29.3.22 MPUD Background Area Access Control Register (MPDACBCR)

Address H'FFF7 8220

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	R	W	—	—
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b4	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b3	R	Read Access Control Bit	0 : Read disabled 1 : Read enabled	R/W
b2	W	Write Access Control Bit	0 : Write disabled 1 : Write enabled	R/W
b1, b0	—	Reserved	These bits are always read as 0. The write value should always be 0.	R

Before changing the value in the MPDACBCR register, use the MPDMPEN register to cancel the protection.

R Bit

This bit enables and disables read access to the background area in the MPU for the DMAC.

W Bit

This bit enables and disables write access to the background area in the MPU for the DMAC.

29.3.23 MPUD Error Status Clear Register (MPDECLR)

Address H'FFF7 8224

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	CLR
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b1	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b0	CLR	Error Status Clear Bit	Writing: 0 : Nothing is performed. 1 : All bits in the MPDESR and MPDHIT registers are set to 0. Reading: 0 is read.	R/W

Before changing the value in the MPDECLR register, use the MPDMPEN register to cancel the protection.

29.3.24 MPUD Error Status Register (MPDESR)

Address H'FFF7 8228

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	RER	WER	—	—
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b4	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b3	RER	Read Access Error Flag*	0 : No read access error occurred. 1 : A read access error occurred.	R
b2	WER	Write Access Error Flag*	0 : No write access error occurred. 1 : A write access error occurred.	R
b1, b0	—	Reserved	These bits are always read as 0. The write value should always be 0.	R

Note: * This bit is set to 0 by setting the CLR bit in the MPDECLR register to 1.

The MPDESR register is an 8-bit read-only register.

RER Flag

This flag indicates whether an MPU error occurred due to DMAC read access.
If the WER flag is set to 1, this flag is not set to 1 even if an MPU error occurs due to DMAC read access.

WER Flag

This flag indicates whether an MPU error occurred due to DMAC write access.
If the RER flag is set to 1, this flag is not set to 1 even if an MPU error occurs due to DMAC write access.

29.3.25 MPUD Error Address Register (MPDERADR)

Address H'FFF7 8230

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ERADR[31:16]															
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
	ERADR[15:0]															
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
b31 to b0	ERADR [31:0]	Error Address Bits	These bits retain the address of the DMAC access (read or write) that caused an MPU error for the first time after the CLR bit in the MPDECLR register was set to 1 and the RER and WER flags in the MPDESR register were set to 0.	R

29.3.26 MPUD Access Hit Area Register (MPDHIT)

Address H'FFF7 8238

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	H15	H14	H13	H12	H11	H10	H9	H8	H7	H6	H5	H4	H3	H2	H1	H0
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
	—	—	—	—	—	—	—	—	—	—	—	—	R	W	—	—
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	H15	Area 15 Access Hit Indication Flag*	0 : No access error 1 : Access error	R
b30	H14	Area 14 Access Hit Indication Flag*		R
b29	H13	Area 13 Access Hit Indication Flag*		R
b28	H12	Area 12 Access Hit Indication Flag*		R
b27	H11	Area 11 Access Hit Indication Flag*		R
b26	H10	Area 10 Access Hit Indication Flag*		R
b25	H9	Area 9 Access Hit Indication Flag*		R
b24	H8	Area 8 Access Hit Indication Flag*		R
b23	H7	Area 7 Access Hit Indication Flag*		R
b22	H6	Area 6 Access Hit Indication Flag*		R
b21	H5	Area 5 Access Hit Indication Flag*		R
b20	H4	Area 4 Access Hit Indication Flag*		R
b19	H3	Area 3 Access Hit Indication Flag*		R
b18	H2	Area 2 Access Hit Indication Flag*		R
b17	H1	Area 1 Access Hit Indication Flag*		R
b16	H0	Area 0 Access Hit Indication Flag*		R
b15 to b4	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b3	R	Read Access Control Bit Retention Flag for Access Hit Area*	0 : Read disabled 1 : Read enabled	R
b2	W	Write Access Control Bit Retention Flag for Access Hit Area*	0 : Write disabled 1 : Write enabled	R
b1, b0	—	Reserved	These bits are always read as 0. The write value should always be 0.	R

Note: * This bit is set to 0 by setting the CLR bit in the MPDECLR register to 1.

The MPDHIT register is a 32-bit read-only register.

H15 to H0 Flags

These bits indicate the area in which a DMAC access error occurred for the first time after the CLR bit in the MPDECLR register was set to 1 and the RER and WER flags in the MPDESR register were set to 0. These bits are not set to 1 if the access error occurred only in the background area.

R Flag

This bit retains the read access control bit of the area in which a DMAC access error occurred. If the error occurred in overlapping areas, this bit retains the logical OR of the read access control bits of the relevant areas (including the background area).

W Flag

This bit retains the write access control bit of the area in which a DMAC access error occurred. If the error occurred in overlapping areas, this bit retains the logical OR of the write access control bits of the relevant areas (including the background area).

29.3.27 MPUD Area n Start Address Register (MPDSADRn) (n = 0 to 15)

Address: MPDSADR0: H'FFF7 8300, MPDSADR1: H'FFF7 8310, MPDSADR2: H'FFF7 8320, MPDSADR3: H'FFF7 8330, MPDSADR4: H'FFF7 8340, MPDSADR5: H'FFF7 8350, MPDSADR6: H'FFF7 8360, MPDSADR7: H'FFF7 8370, MPDSADR8: H'FFF7 8380, MPDSADR9: H'FFF7 8390, MPDSADR10: H'FFF7 83A0, MPDSADR11: H'FFF7 83B0, MPDSADR12: H'FFF7 83C0, MPDSADR13: H'FFF7 83D0, MPDSADR14: H'FFF7 83E0, MPDSADR15: H'FFF7 83F0

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	SADR[31:16]															
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
	SADR[15:2]														—	—
After Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b2	SADR [31:2]	Area n Start Address Bits	These bits set the upper 30 bits of the start address of area n in the MPU for the DMAC.	R/W
b1, b0	—	Reserved	These bits are always read as 0. The write value should always be 0.	R

Before changing the value in the MPDSADRn register, use the MPDMPEN register to cancel the protection.

29.3.28 MPUD Area n End Address Register (MPDEADRn) (n = 0 to 15)

Address: MPDEADR0: H'FFF7 8304, MPDEADR1: H'FFF7 8314, MPDEADR2: H'FFF7 8324, MPDEADR3: H'FFF7 8334, MPDEADR4: H'FFF7 8344, MPDEADR5: H'FFF7 8354, MPDEADR6: H'FFF7 8364, MPDEADR7: H'FFF7 8374, MPDEADR8: H'FFF7 8384, MPDEADR9: H'FFF7 8394, MPDEADR10: H'FFF7 83A4, MPDEADR11: H'FFF7 83B4, MPDEADR12: H'FFF7 83C4, MPDEADR13: H'FFF7 83D4, MPDEADR14: H'FFF7 83E4, MPDEADR15: H'FFF7 83F4

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	EADR[31:16]															
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
	EADR[15:2]														—	—
After Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b2	EADR [31:2]	Area n End Address Bits	These bits set the upper 30 bits of the end address of area n in the MPU for the DMAC.	R/W
b1, b0	—	Reserved	These bits are always read as 0. The write value should always be 0.	R

Before changing the value in the MPDEADRn register, use the MPDMPEN register to cancel the protection.

29.3.29 MPUD Area n Access Control Register (MPDACRn) (n = 0 to 15)

Address: MPDACR0: H'FFF7 8308, MPDACR1: H'FFF7 8318, MPDACR2: H'FFF7 8328, MPDACR3: H'FFF7 8338, MPDACR4: H'FFF7 8348, MPDACR5: H'FFF7 8358, MPDACR6: H'FFF7 8368, MPDACR7: H'FFF7 8378, MPDACR8: H'FFF7 8388, MPDACR9: H'FFF7 8398, MPDACR10: H'FFF7 83A8, MPDACR11: H'FFF7 83B8, MPDACR12: H'FFF7 83C8, MPDACR13: H'FFF7 83D8, MPDACR14: H'FFF7 83E8, MPDACR15: H'FFF7 83F8

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	R	W	—	V
After Reset	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b4	—	Reserved	These bits are always read as 0. The write value should always be 0.	R
b3	R	Read Access Control Bit	0 : Read disabled 1 : Read enabled	R/W
b2	W	Write Access Control Bit	0 : Write disabled 1 : Write enabled	R/W
b1	—	Reserved	This bit is always read as 0. The write value should always be 0.	R
b0	V	Validity Bit	0 : Area settings invalid 1 : Area settings valid	R/W

Before changing the value in the MPDACRn register, use the MPDMPEN register to cancel the protection.

R Bit

This bit enables and disables read access to area n in the MPU for the DMAC.

W Bit

This bit enables and disables write access to area n in the MPU for the DMAC.

V Bit

This bit validates or invalidates the settings of area n in the MPU for the DMAC.

29.4 Operation

29.4.1 Enabling MPUs

(1) MPU for the CPU

Setting the MPEN bit in the MPCMPEN register to 1 enables the MPU for the CPU. However, because the CPU pre-reads instructions, several instructions after the instruction that enables the MPU for the CPU (sets the MPEN bit in the MPCMPEN register to 1) have already been read by the CPU when the MPU was disabled. The MPU settings must match the CPU operation.

Figure 29.1 shows an example of a program that enables the MPU for the CPU.

```
        MOV.L   #H'FFF7 8000, R0    ;  R0 = Address of the MPCMPEN register
        MOV.L   #H'AAAA 0001, R1    ;  R1 = Data written to the MPCMPEN register
        MOV.L   R1, @R0             ;  Set the MPEN bit in the MPCMPEN register to 1
        NOP                         ;
        NOP                         ;
        NOP                         ;  Execute five or more NOPs
        NOP                         ;
        NOP                         ;
        BRA     L1                  ;  Execute the branch instruction
        NOP                         ;
         :                          ;
L1:
         :
```

Do not set this area as an instruction execution disabled area.

Figure 29.1 Example of a Program that Enables the MPU for the CPU

(2) MPU for the DMAC

Setting the MPEN bit in the MPDMPEN register to 1 enables the MPU for the DMAC.

29.4.2 Setting MPU Registers

All MPU registers can be accessed only from the CPU, and cannot be accessed from any other bus master such as the DMAC. All MPU registers are subject to access protection with the MPU for the CPU. Some registers are write protected until the MPEN bit in both the MPCMPEN and MPDMPEN registers is set to 0 (MPU disabled, register protection canceled). For details, see sections section 29.3.1, MPUC Enable Register (MPCMPEN) and section 29.3.18, MPUD Enable Register (MPDMPEN).

29.4.3 Setting Area Information

Area information items, such as start address, end address, protection attributes, and whether the area is valid or invalid, are specified in the following registers for each bus master: MPU area n start address registers (MPCSADRn and MPDSADRn), MPU area n end address registers (MPCEADRn and MPDEADRn), and MPU area n access control registers (MPCACRn and MPDACRn).

The protection attributes for each area and whether the area is valid or invalid can also be specified in the following registers for each bus master: MPU read access control registers (MPCRACR and MPDRACR), MPU write access control registers (MPCWACR and MPDWACR), MPU instruction access control register (MPCIACR), MPU area setting validity registers (MPCVLD and MPDVLD). These registers allow concurrent specification of the protection attributes for multiple areas and whether the areas are valid. For details, see section 29.3, Register Description.

29.4.4 Area Overlap

The protection attribute of overlapping areas is the logical OR of the access control bits of these areas (including the background area).

29.4.5 Background Area

The target area for background area covers the entire address space (H'0000 0000 to H'FFFF FFFF). The protection attributes of the background area are set in the MPU background area access control registers (MPCACBCR and MPDACBCR) for each bus master. When the protection attributes of the background area are set, access to addresses that are not included in the 16 areas is protected.

The background area overlaps all 16 areas. Therefore, if the protection attributes of the background area are set to enable read, write, and instruction execution, the protection attributes for the entire space are also set to enable read, write, and instruction execution, irrespective of the protection attribute settings of the 16 areas.

29.4.6 MPU Errors

(1) CPU Operand Access MPU Error

If a CPU operand access violates the protection attribute of an area to be accessed, an CPU operand access MPU error occurs. If the MPU error is caused by a write access, write processing is not performed. If the MPU error is caused by a read access, an undefined value is returned to the CPU as the read data. The source of the MPU error is stored in the MPCESR register. The address of the operand access that caused the MPU error is stored in the MPCERADRO register. The information about the area in which the MPU error occurred is stored in the MPCHITO register. The MPU error information stored in these registers will be used as the information about the CPU operand access MPU error that occurs for the first time after the CLR bit in the MPCECLR register is set to 1 and the RER and WER flags in the MPCESR register are set to 0.

When a CPU operand access MPU error occurs, exception handling for the CPU operand access MPU error starts. If this error occurs more than once in succession, exception handling for the error might be performed several times in succession. For details of exception handling, see section 6, Exception Handling.

(2) CPU Instruction Access MPU Error

If the CPU decodes an instruction in an area for which instruction execution is disabled, a CPU instruction access MPU error occurs. The source of the MPU error is stored in the MPCESR register. The address of the instruction access that caused the MPU error is stored in the MPCERADRI register. The information about the area in which the MPU error occurred is stored in the MPCHITI register. The MPU error information stored in these registers will be used as the information about the CPU instruction access MPU error that occurs for the first time after the CLR bit in the MPCECLR register is set to 1 and the IER flag in the MPCESR register is set to 0.

When a CPU instruction access MPU error occurs and the instruction is not located immediately after the delayed branch instruction, the CPU performs exception handling for general illegal instruction. If the instruction is located immediately after the delayed branch instruction, the CPU performs exception handling for slot illegal instruction. For details of exception handling, see section 6, Exception Handling.

If the first-half 16-bit instruction code of the MOVI20 instruction or MOVI20S instruction is located in an area for which instruction execution is enabled and the second-half 16-bit instruction code is located in an area for which instruction execution is disabled, exception handling for general illegal instruction does not start immediately after the the CPU decodes this MOVI20 instruction or MOVI20S instruction. Instead, exception handling for general illegal instruction starts immediately after the CPU decodes the next instruction of the MOVI20 instruction or MOVI20S instruction. An undefined value is stored in the destination register of this MOVI20 instruction or MOVI20S instruction. The start address of the second-half 16-bit instruction code of the MOVI20 instruction or MOVI20S instruction is stored in the MPCERADRI register.

(3) DMAC Access MPU Error

If DMAC access violates the protection attribute of an area to be accessed, a DMAC access MPU error occurs. If the MPU error is caused by a write access, write processing is not performed. If the MPU error is caused by a read access, an undefined value is returned to the DMAC as the read data. The source of the MPU error is stored in the MPDESR register. The address of the access that caused the MPU error is stored in the MPDERADR register. The information about the area in which the MPU error occurred is stored in the MPDHIT register. The MPU error information stored in these registers will be used as the information about the DMAC access MPU error that occurs for the first time after the CLR bit in the MPDECLR register is set to 1 and the RER and WER flags in the MPDESR register are set to 0.
The CPU handles the DMAC access MPU error as an interrupt. For details of interrupt processing, see sections 6, Exception Handling and 8, Interrupt Controller (INTC).

29.4.7 Area Search

Set the S bit in the MPCRSOP register to 1 to search for the MPU area for the CPU that includes the address set in the MPCRSADR register. The search result is stored in the MPCHITO register.
Before searching for an area, set the MPEN bit in the MPCMPEN register to 0.

29.5 Notes on the MPU

29.5.1 MPU Setting Changes

All MPU registers are subject to access protection by the MPU for the CPU. Accordingly, if the protection attributes of an area that includes the addresses of the MPU registers for the CPU or DMAC are set to prohibit write access from the CPU, the settings of the given register can no longer be changed.

29.5.2 Changing DMAC MPU Settings

If the MPU settings for the DMAC are to be changed, only do so while the DMAC is stopped.

29.5.3 Notes on Usage

When the MPU enable (MPEN) bit in the MPUC enable register (MPCMPEN) is set to 1, the MPEN bit does not disable writing to the MPCRACR, MPCWACR, MPCIACR, and MPCVLD registers. If the upper bits of the data written to these registers are specific values (key codes), the values in the registers can be changed. Because the CPU pre-reads instructions, several instructions after the instruction that changes the values in the MPCRACR, MPCWACR, MPCIACR, and MPCVLD registers have already been read by the CPU before the values are reflected in the corresponding registers. To match the MPU settings with the CPU operation, change the value in the MPCRACR, MPCWACR, MPCIACR, and MPCVLD registers in the following procedure.

(1) To change the value in the MPCRACR or MPCWACR register (to change the protection attribute for an operand access) with the area setting enabled, execute three or more NOP instructions after a register value change instruction.

```
MOV.L   #H'FFF78010,R0    ; R0 = Address of the MPCRACR register
MOV.L   #H'AAAAFFFE,R1    ; R1 = Data written to the MPCRACR register
MOV.L   R1,@R0            ; Change the value in the MPCRACR register (Area 0 is set to disable reading)
NOP                       ;
NOP                       ;                          } No CPU operand access MPU error occurs
NOP                       ; Execute three or more NOPs } even if read access is performed in a disabled area
 :                                                     } during the instruction execution.
```

Note: Change the value in the MPUC write access control register (MPCWACR) by a similar procedure.

Figure 29.2 Procedure to Change the Value in the MPUC Read Access Control Register (MPCRACR)

(2) To change the value in the MPCIACR register (to change the protection attribute for an instruction access) with the area setting enabled or to change the value in the MPCVLD register with the protection attribute for the area set (to enable or disable the area setting), execute five or more NOP instructions and the branch instruction after a register value change instruction.

```
    MOV.L  #H'FFF78018,R0   ; R0 = Address of the MPCIACR register
    MOV.L  #H'AAAAFFFE,R1 ; R1 = Data written to the MPCIACR register
    MOV.L  R1,@R0           ; Change the value in the MPCIACR register (Area 0 is set to disable instruction execution)
    NOP                     ;
    NOP                     ;
    NOP                     ;
    NOP                     ;
    NOP                     ; Execute five or more NOPs
    BRA    L1               ; Execute the branch instruction
    NOP                     ;
     :
L1:
     :
```

The CPU does not perform exception handling for general illegal instruction even if instruction execution is set to be disabled during the instruction execution.
Note: The MPCESR and MPCERADRI registers are set.

Note: Change the value in the MPUC area setting validity register (MPCVLD) by a similar procedure.

Figure 29.3 Procedure to Change the Value in the MPUC Instruction Access Control Register (MPCIACR)

30. Electrical Characteristics

30.1 Absolute Maximum Ratings

Table 30.1 shows the absolute maximum ratings.

Table 30.1 Absolute Maximum Ratings

Item		Symbol	Rating	Unit	Remarks
Power supply voltage	VCC	VCC	-0.3 to +6.5	V	
Input voltage (ports other than ports K, L, and N)*2		V_{in}	-0.3 to VCC + 0.3	V	
Input voltage (ports K and L)*2		V_{in}	-0.3 to AVCC1 + 0.3	V	
Input voltage (port N)		V_{in}	-0.3 to AVCC0 + 0.3	V	
Analog supply voltage	AVCC0, AVCC1	AVCC	-0.3 to +6.5V	V	AVCC1 = VCC±0.3V
Analog reference voltage		VREFH0	-0.3 to AVCC0 + 0.3	V	VREFH0 > VREFL0
		VREFL0	-0.3 to AVSS0 + 0.3	V	
Analog input voltage (ports J, K, and L)*2		V_{AN}	-0.3 to AVCC1 + 0.3	V	
Analog input voltage (port N)		V_{AN}	-0.3 to AVCC0 + 0.3	V	
VSS differential voltage		VSS-AVSS0	-0.1 to 0.1	V	
		VSS-AVSS1	-0.1 to 0.1	V	
		AVSS0-AVSS1	-0.1 to 0.1	V	
Maximum input current per pin	Digital input pins	Imax	-20 to +20	mA	One pin at a time
	Analog input pins	Imax	-20 to +20	mA	
Operating temperature*1		Topr	-40 to +85	°C	J version
			-40 to +125	°C	K version
Storage temperature		tstg	-55 to +125	°C	Before assembly

[Operating precautions]
Operating the LSI in excess of the absolute maximum ratings may result in permanent damage.
Be sure to use the LSI in compliance with the connection of power pins, combination conditions of applicable power supply voltages, voltage applicable to each pin, and conditions of output voltage, as specified in the manual. Connecting a non-specified power supply or using the LSI at an incorrect voltage may result in permanent damage of the LSI or the system that contains the LSI.

Notes: 1. When this LSI is used in the range of 85 °C to 125 °C, the accumulated operating time must be within 3000 hours.
2. The port L pins are not available in the SH72A0 Group.

30.2 DC Characteristics

Tables 30.2 to 30.10 show the DC characteristics.

Table 30.2 DC Characteristics (Input Level Voltage)

Conditions: VCC = 5.0 V ± 0.5 V, AVCC1 = 5.0 V ± 0.5 V, AVCC0 = 5.0 V ± 0.5 V, VREFH0 = 4.5 V to AVCC0,
VSS = AVSS1 = AVSS0 = VREFL0 = 0 V, Ta = -40 °C to 125 °C

Item	Target Pins	Symbol	Min.	Typ.	Max.	Unit	Measurement Conditions
Schmitt trigger input voltage	Peripheral function input pins, AUDRST, AUDMD, ASEMD, RESET#, MD0, MD1, and XIN	V_{T+} (VIH)	VCC × 0.8 *1	—	VCC + 0.3 *1	V	
		V_{T-} (VIL)	-0.3	—	VCC × 0.2 *1	V	
		V_{HS} (V_{T+}-V_{T-})	VCC × 0.08 *1	—	—	V	
TTL input voltage	Port input pins, NMI, INTi, POE0 to POE2 (When a TTL level is selected) AUDCK, AUDSYNC, and AUDATA0 to AUDATA3	V_{IH}	2.2	—	VCC + 0.3 *1*2	V	
		V_{IL}	-0.3	—	0.8	V	
CMOS input voltage	Port input pins, NMI, INTi, and POE0 to POE2 (When the (VCC × 0.5) level is selected)	V_{IH}	VCC × 0.70 *1*2	—	VCC + 0.3 *1*2	V	
		V_{IL}	–0.3	—	VCC × 0.30 *1*2	V	
	Port input pins, NMI, INTi, and POE0 to POE2 (When the (VCC × 0.7) level is selected)	V_{IH}	VCC × 0.85 *1*2	—	VCC + 0.3 *1*2	V	
		V_{IL}	-0.3	—	VCC × 0.50 *1*2	V	

Notes: 1. The input reference voltage across ports K and L is AVCC1. (The port L pins are not available in the SH72A0 Group.)
2. The input reference voltage across port N is AVCC0.
• INTi (i = 0 to 13: SH72A2 Group, i = 0 to 8 and 10: SH72A0 Group)

Table 30.3 DC Characteristics (Input Leak Current)

Conditions: VCC = 5.0 V ± 0.5 V, AVCC1 = 5.0 V ± 0.5 V, AVCC0 = 5.0 V ± 0.5 V, VREFH0 = 4.5 V to AVCC0,
VSS = AVSS1 = AVSS0 = VREFL0 = 0 V, Ta = -40 °C to 125 °C

Item	Target Pins	Symbol	Min.	Typ.	Max.	Unit	Measurement Conditions
Input leak current	ASEMD, MD0, and MD1	See table 30.4, DC Characteristics (Input Pull-Down MOS Current).					
	Schmitt- and TTL-type input pins other than ASEMD, MD0, and MD1	$\lvert I_{in} \rvert$	—	—	2.0	µA	V_{in} = 0.3 V to VCC - 0.3 V
	A/D port pins (ports J, K, and L)*		—	—	0.2	µA	V_{in} = 0.3 V to AVCC1 - 0.3 V
	A/D port pins (port N)		—	—	0.2	µA	V_{in} = 0.3 V to AVCC0 - 0.3 V

Note: * The port L pins are not available in the SH72A0 Group.

Table 30.4 DC Characteristics (Input Pull-Up MOS Current)

Conditions: VCC = 5.0 V ± 0.5 V, AVCC1 = 5.0 V ± 0.5 V, AVCC0 = 5.0 V ± 0.5 V, VREFH0 = 4.5 V to AVCC0,
VSS = AVSS1 = AVSS0 =VREFL0 = 0 V, Ta = -40 °C to 125 °C

Item	Target Pins	Symbol	Min.	Typ.	Max.	Unit	Measurement Conditions
Input pull-up MOS current	Port input pins	$-I_{pu}$	50	—	300	µA	Vin = 0 V

Table 30.5 DC Characteristics (Input Pull-Down MOS Current)

Conditions: VCC = 5.0 V ± 0.5 V, AVCC1= 5.0 V ± 0.5 V, AVCC0 = 5.0 V ± 0.5 V, VREFH0 = 4.5 V to AVCC0,
VSS = AVSS1 = AVSS0 = VREFL0 = 0 V, Ta = -40 °C to 125 °C

Item	Target Pins	Symbol	Min.	Typ.	Max.	Unit	Measurement Conditions
Input pull-down MOS current	ASEMD, MD0, and MD1	I_{pd}	—	—	300	µA	Vin = VCC
	Analog input pins	I_{pd}	—	—	50	µA	Vin = AVCC0 or AVCC1

Table 30.6 DC Characteristics (Output Voltage)

Conditions: VCC = 5.0 V ± 0.5 V, AVCC1 = 5.0 V ± 0.5 V, AVCC0 = 5.0 V ± 0.5 V, VREFH0 = 4.5 V to AVCC0,
VSS = AVSS1 = AVSS0 = VREFL0 = 0 V, Ta = -40 °C to 125 °C

Item	Target Pins	Symbol	Min.	Typ.	Max.	Unit	Measurement Conditions
Output high-level voltage	All output pins other than port N pins	V_{OH}	VCC - 0.5 *	—	—	V	I_{OH} = -200 µA
			VCC - 1.0 *	—	—	V	I_{OH} = -2 mA
Output low-level voltage	All output pins other than port N pins	V_{OL}	—	—	0.5	V	I_{OL} = 200 µA
				—	1	V	I_{OL} = 2 mA

Note: * The input reference voltage across ports K and L is AVCC1. (The port L pins are not available in the SH72A0 Group.)

Table 30.7 DC Characteristics (Permissible Output Current)

Conditions: VCC = 5.0 V ± 0.5 V, AVCC1 = 5.0 V ± 0.5 V, AVCC0 = 5.0 V ± 0.5 V, VREFH0 = 4.5 V to AVCC0,
VSS = AVSS1 = AVSS0 = VREFL0 = 0 V, Ta = -40 °C to 125 °C

Item	Symbol	Min.	Typ.	Max.	Unit	Measurement Conditions
Output low-level permissible current (per pin)	I_{OL}	—	—	2	mA	
Output low-level permissible current (total)	ΣI_{OL}	—	—	12	mA	
Output high-level permissible current (per pin)	I_{OH}	—	—	2	mA	
Output high-level permissible current (total)	ΣI_{OH}	—	—	12	mA	

[Operating precautions]
To assure LSI reliability, do not exceed the output values listed in table 30.7.

Table 30.8 DC Characteristics (Permissible Input Current)

Conditions: VCC = 5.0 V ± 0.5 V, AVCC1 = 5.0 V ± 0.5 V, AVCC0 = 5.0 V ± 0.5 V, VREFH0 = 4.5 V to AVCC0,
VSS = AVSS1 = AVSS0 = VREFL0 = 0 V, Ta = -40 °C to 125 °C

Item		Symbol	Min.	Typ.	Max.	Unit	Measurement Conditions
DC injection current (per pin)	Logic pin	I_{IC}	-1.0	—	1.0	mA	
	Analog pin		-3.0	—	3.0	mA	
DC injection current (total)		$\Sigma \mid I_{IC} \mid$	—	—	50	mA	

Table 30.9 DC Characteristics (Input Capacitance)

Conditions: VCC = 5.0 V ± 0.5 V, AVCC1= 5.0 V ± 0.5 V, AVCC0 = 5.0 V ± 0.5 V, VREFH0 = 4.5 V to AVCC0,
VSS = AVSS1 = AVSS0 = VREFL0 = 0 V

Item		Symbol	Min.	Typ.	Max.	Unit	Measurement Conditions
Input capacitance	All input pins	C_{in}	—	—	20	pF	Vin = 0 V, f = 1 MHz, Ta = 25 °C

Table 30.10 DC Characteristics (Supply Current)

Conditions: VCC = 5.0 V ± 0.5 V, AVCC1 = 5.0 V ± 0.5 V, AVCC0 = 5.0 V ± 0.5 V, VREFH0 = 4.5 V to AVCC0,
VSS = AVSS1 = AVSS0 = VREFL0 = 0 V, Ta = -40 °C to 125 °C

Item		Symbol	Min.	Typ.	Max.	Unit	Measurement Conditions
VCC supply current	Normal operation	I_{VCC}	—	—	85	mA	f = 100 MHz*
			—	—	75		f = 80 MHz
	CPU sleep mode	I_{VCC_SLP}	—	—	70	mA	f = 100 MHz*
			—	—	60		f = 80 MHz
	Standby mode	I_{VCC_STBY}	—	0.1	1	mA	Ta ≤ 50 °C
			—	—	15		50 °C < Ta
	Power-down mode 0		—	30	100	µA	Ta ≤ 50 °C
			—	—	500		50 °C < Ta
	Power-down mode 1		—	25	100		Ta ≤ 50 °C
			—	—	500		50 °C < Ta
ADC supply current (AVCC0 supply current)	During A/D conversion	I_{AVCC0}	—	6.1	8.5	mA	When the AD0 is in use
	Awaiting A/D conversion		—	—	100	µA	
	Standby mode		—	—	100		When the AD0 is not in use
	Power-down mode 0/1		—	—	10		
ADC supply current (AVCC1 supply current)	During A/D conversion	I_{AVCC1}	—	3.5	5	mA	When the AD1 is in use
	Awaiting A/D conversion		—	—	100	µA	
	Standby mode		—	—	10		When the AD1 is not in use
	Power-down mode 0/1		—	—	10		
ADC supply current (VREFH0 supply current)	During A/D conversion	I_{AVREF}	—	1.25	2	mA	When the AD0 is in use
	Awaiting A/D conversion		—	—	1.3		
	Standby mode		—	—	1.3		When the AD0 is not in use
	Power-down mode 0/1		—	—	10	µA	

[Operating precautions]
Values of the supply current are for when there are no loads on any output pins and all input pins are at VIHmin = VCC or VIL = 0 V.
To increase the radiation of this LSI, use a substrate having four or more layers.

Note: * Only the SH72A2 Group.

30.3 AC Characteristics

30.3.1 Power-On and Reset Timing

Table 30.11 AC Characteristics (Power-On and Reset Timing)

Conditions: VCC = 5.0 V ± 0.5 V, AVCC1 = 5.0 V ± 0.5 V, AVCC0 = 5.0 V ± 0.5 V, VREFH0 = 4.5 V to AVCC0,
VSS = AVSS1 = AVSS0 =VREFL0 = 0 V, Ta = -40 °C to 125 °C

Item	Symbol	Min.	Max.	Unit	Reference Figure
Power-supply power-on time	tVCCR	50	—	µs	Figures 30.1 and 30.5
Hold time until release from reset once power is turned on	tRESETH	10	—	ms	

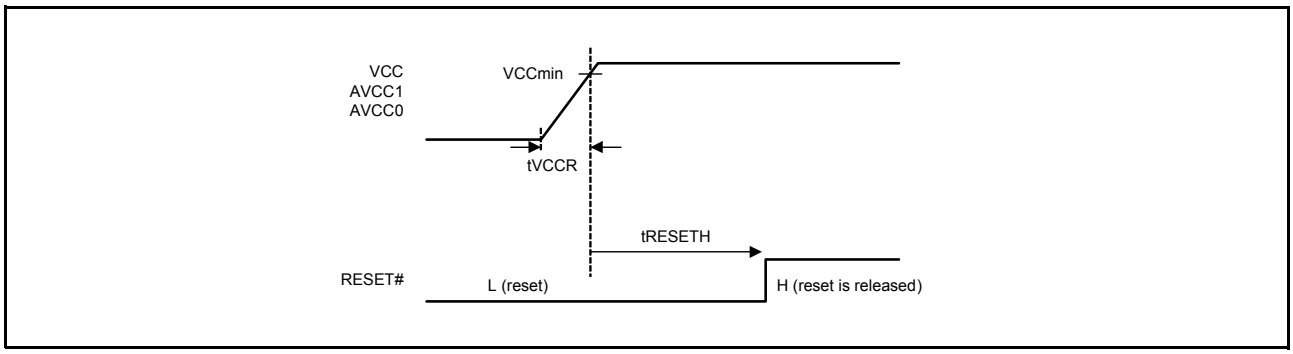


Figure 30.1 Power-On and Reset Timing

Note: VREFH0 ≤ AVCC0 + 0.3 V must be satisfied both when power is turned on and when power is turned off.

30.3.2 Power-Down and Reset Timing

Table 30.12 AC Characteristics (Power-Down and Reset Timing)

Conditions: VCC = 5.0 V ± 0.5 V, AVCC1 = 5.0 V ± 0.5 V, AVCC0 = 5.0 V ± 0.5 V, VREFH0 = 4.5 V to AVCC0,
VSS = AVSS1 = AVSS0 = VREFL0 = 0 V, Ta = -40 °C to 125 °C

Item		Symbol	Min.	Max.	Unit	Reference Figure
Reset period before power-supply voltage is falling	Any time other than during programming or erasure of the on-chip flash memory	tRESETS	0	—	ms	Figures 30.2 and 30.5
	During programming or erasure of the on-chip flash memory		20	—	µs	

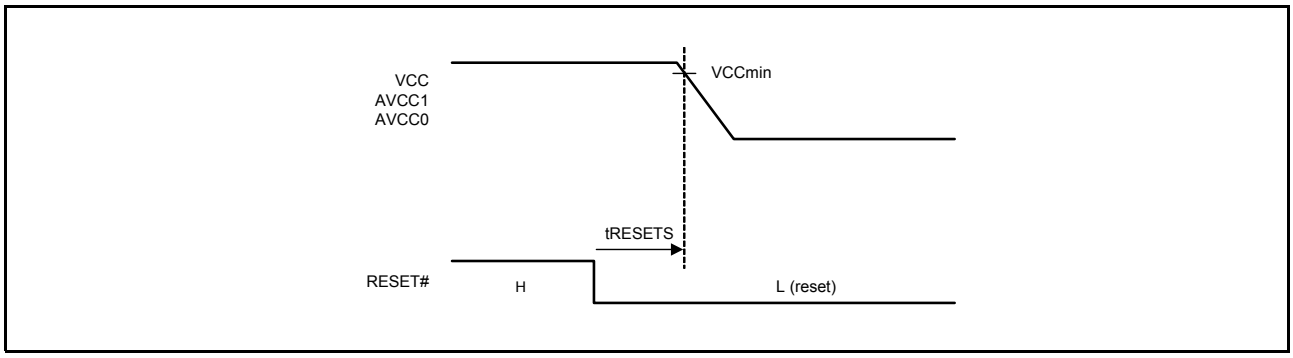


Figure 30.2 Power-Down and Reset Timing

Note: VREFH0 ≤ AVCC0 + 0.3 V must be satisfied both when power is turned on and when power is turned off.

30.3.3 Clock Timing

Table 30.13 shows the clock timing.

Table 30.13 Clock Timing

Conditions: VCC = 5.0 V ± 0.5 V, AVCC1 = 5.0 V ± 0.5 V, AVCC0 = 5.0 V ± 0.5 V, VREFH0 = 4.5 V to AVCC0,
VSS = AVSS1 = AVSS0 = VREFL0 = 0 V, Ta = -40 °C to 125 °C

Item	Symbol	Min.	Max.	Unit	Reference Figure
External input clock frequency	f_{EX}	8	10	MHz	Figure 30.3
External input clock high-level pulse width	t_{EXH}	40		ns	
External input clock low-level pulse width	t_{EXL}	40	—	ns	
External input clock rising time	t_{EXr}	—	5	ns	
External input clock falling time	t_{EXf}	—	5	ns	
CLKOUT clock high-level pulse width*	t_{CH}	1/2 t_{cyc} - 7.5	—	ns	Figure 30.4*
CLKOUT clock low-level pulse width*	t_{CL}	1/2 t_{cyc} - 7.5	—	ns	
CLKOUT clock rising time*	t_{Cr}	—	5	ns	
CLKOUT clock falling time*	t_{Cf}	—	5	ns	

Note: * The CLKOUT pin is not available in the SH72A0 Group.

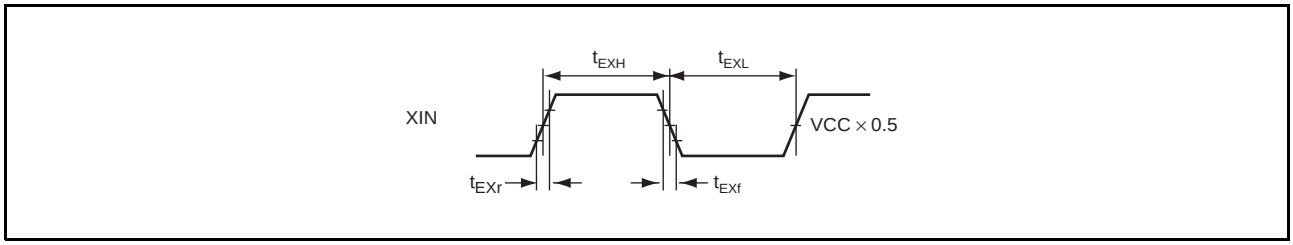


Figure 30.3 External Input Clock Timing

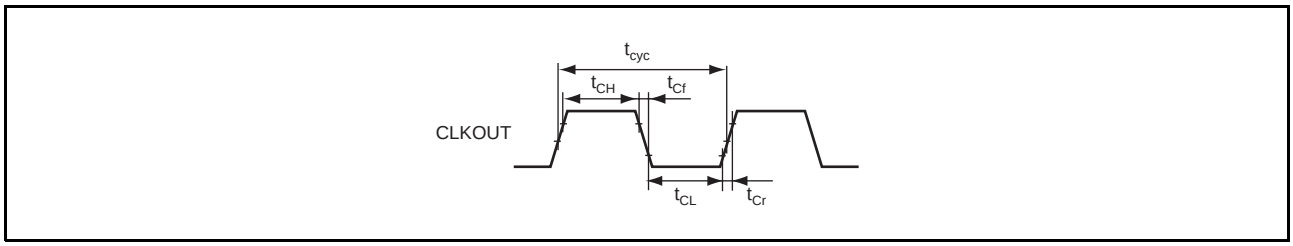


Figure 30.4 External Bus Clock Timing

30.3.4 Operating Mode and Oscillation Timing

Table 30.14 AC Characteristics (Operating Mode and Oscillation Timing)

Conditions: VCC = 5.0 V ± 0.5 V, AVCC1 = 5.0 V ± 0.5 V, AVCC0 = 5.0 V ± 0.5 V, VREFH0 = 4.5 V to AVCC0,
VSS = AVSS1 = AVSS0 = VREFL0 = 0 V, Ta = -40 °C to 125 °C

Item		Symbol	Min.	Max.	Unit	Reference Figure
Hold time until release from reset once power is turned on		tRESETH	10	—	ms	Figure 30.1 Figure 30.5
Reset period before power-supply voltage is falling	Any time other than during programming or erasure of the on-chip flash memory	tRESETS	0	—	ms	Figure 30.2
	During programming or erasure of the on-chip flash memory		20	—	µs	Figure 30.5
VCC set-up time in response to setting of an operating mode pin		tVCCMDS	0	—	ms	Figure 30.5
VCC hold time in response to release of an operating mode pin setting		tVCCMDH	0	—	ms	
Operating mode set-up time until setting of the RESET# pin to the high level		tMDS	10	—	ms	
Oscillation stabilization time	From the point where power supplies have been turned on and the states of the mode pins have settled	tOSC1	—	10	ms	Figure 30.5
	When standby mode is exited	tOSC2	—	10	ms	Figure 30.6

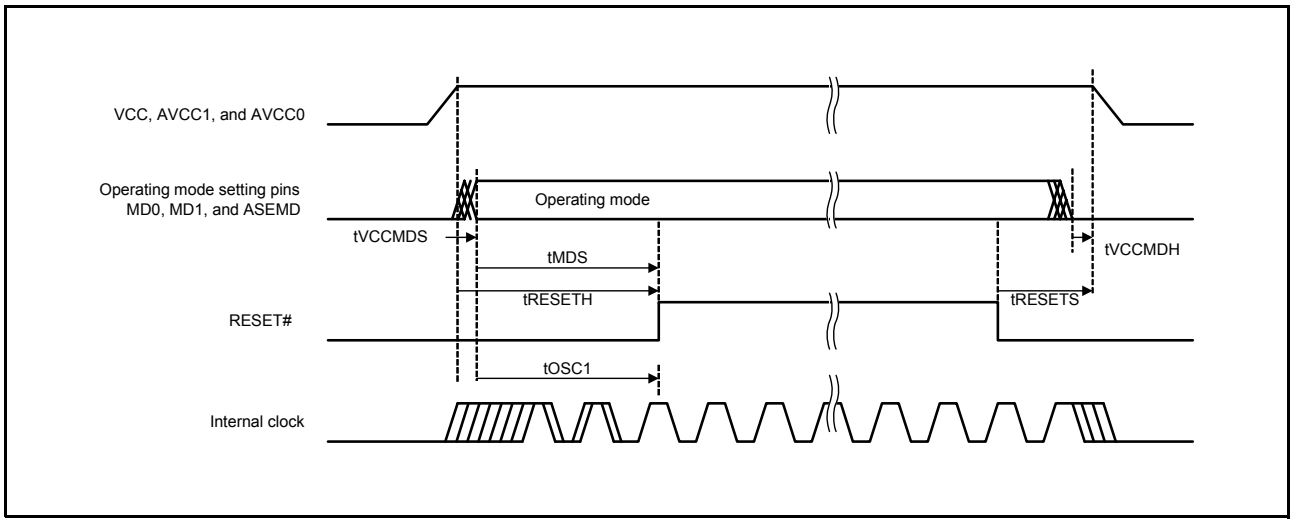


Figure 30.5 Operating Mode and Oscillation Timing

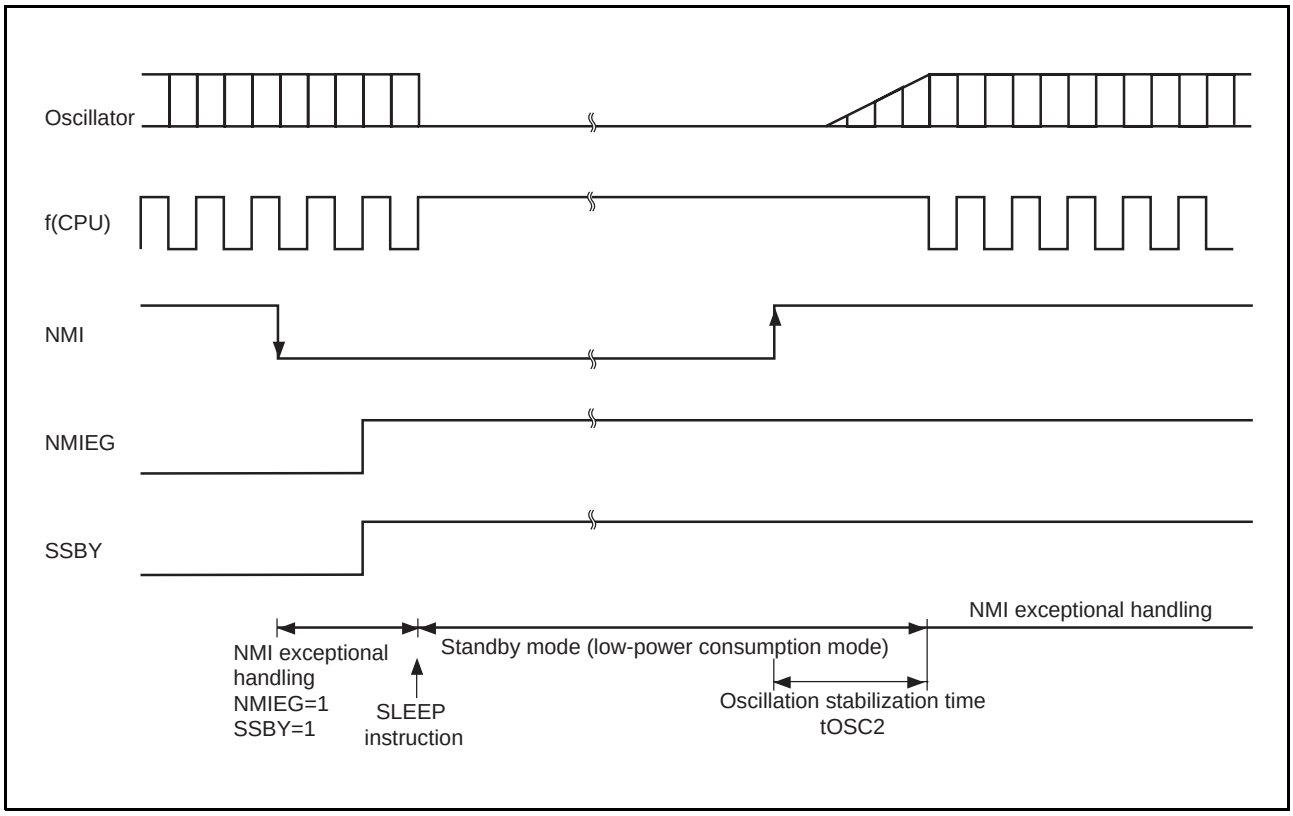


Figure 30.6 Oscillation Stabilization Timing in Standby Mode

30.3.5 Reset Signal Timing

Table 30.15 AC Characteristics (Reset Signal Timing)

Conditions: VCC = 5.0 V ± 0.5 V, AVCC1 = 5.0 V ± 0.5 V, AVCC0 = 5.0 V ± 0.5 V, VREFH0 = 4.5 V to AVCC0,
VSS = AVSS1 = AVSS0 = VREFL0 = 0 V, Ta = -40 °C to 125 °C

Item	Symbol	Min.	Max.	Unit	Reference Figure
RESET# signal low-level width	tRESW	100	—	µs	Figure 30.7
RESET# signal noise cancellation width	tRESNCW	50	400	ns	

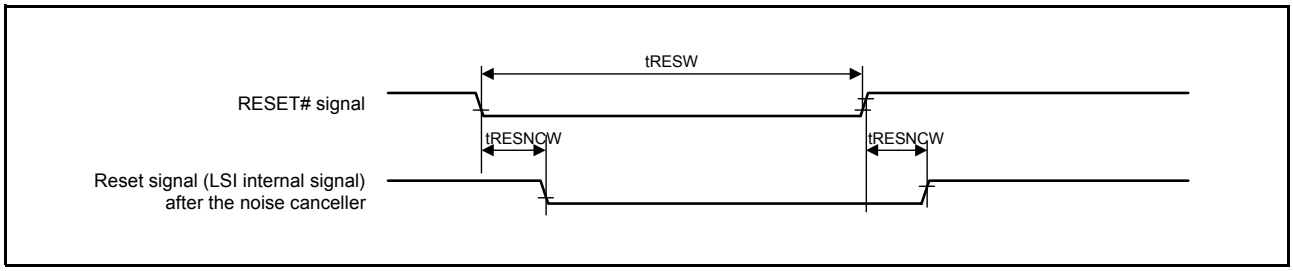


Figure 30.7 Reset Signal Timing

30.3.6 NMI and External INT Timing

Table 30.16 shows the NMI and external INT input timing.

Table 30.16 NMI and External INT Input Timing (NMI and INTi (i = 0 to 13: SH72A2 Group, i = 0 to 8, 10: SH72A0 Group))

Conditions: VCC = 5.0 V ± 0.5 V, AVCC1 = 5.0 V ± 0.5 V, AVCC0 = 5.0 V ± 0.5 V, VREFH0 = 4.5 V to AVCC0,
VSS = AVSS1 = AVSS0 = VREFL0 = 0 V, Ta = –40 °C to 125 °C

Item	Symbol	Min.	Max.	Unit	Reference Figure
External interrupt input pulse width (digital filter is not used)	tINTWL/H	400	—	ns	Figure 30.8
External interrupt input pulse width (digital filter is used, with division by 8)	tINTWL/H	25 × tPBA	—	ns	
External interrupt input pulse width (digital filter is used, with division by 16)	tINTWL/H	49 × tPBA	—	ns	
External interrupt input pulse width (digital filter is used, with division by 32)	tINTWL/H	97 × tPBA	—	ns	
External interrupt input pulse width (digital filter is used, with division by 64)	tINTWL/H	193 × tPBA	—	ns	

Notes: • “tPBA” is the period of peripheral function clock A.
• NMI does not have digital filter function. For NMI, see the column saying that digital filter is not used.

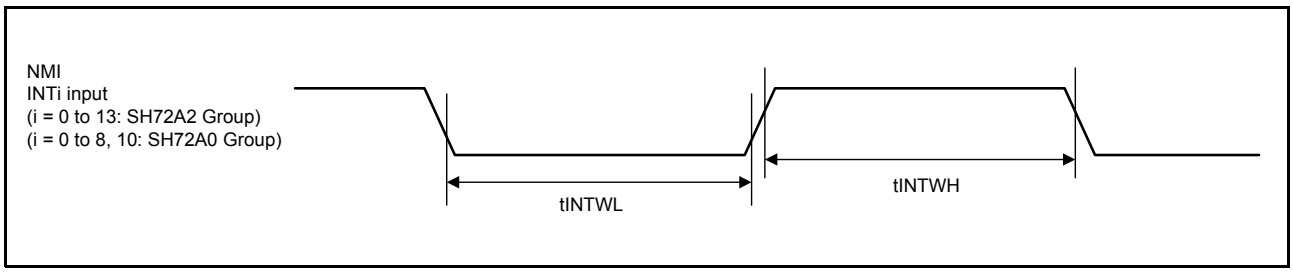


Figure 30.8 NMI and INTi Input Timing

30.3.7 TPU Timing

Table 30.17 shows the TPU timing.

Table 30.17 TPU Timing

Conditions: VCC = 5.0 V ± 0.5 V, AVCC1 = 5.0 V ± 0.5 V, AVCC0 = 5.0 V ± 0.5 V, VREFH0 = 4.5 V to AVCC0,
VSS = AVSS1 = AVSS0 = VREFL0 = 0 V, Ta = -40 °C to 125 °C

Item		Symbol	Min.	Max.	Unit	Reference Figure
Input capture input pulse width	Single edge specified	tTICWH	1.5	—	tBPA	Figure 30.9
	Dual edge specified	tTICWL	2.5	—	tBPA	

Note: “tPBA” is the period of peripheral function clock A.

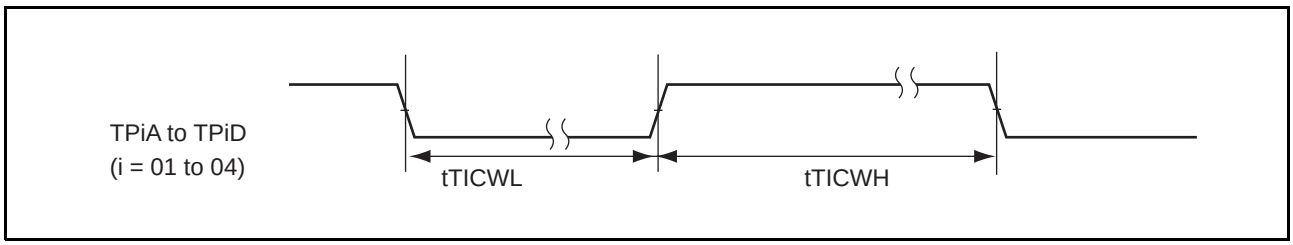


Figure 30.9 TPU Input Capture Input Timing

30.3.8 MTU-III Timing

Table 30.18 shows the MTU-III timing.

Table 30.18 MTU-III Timing

Conditions: VCC = 5.0 V ± 0.5 V, AVCC1 = 5.0 V ± 0.5 V, AVCC0 = 5.0 V ± 0.5 V, VREFH0 = 4.5 V to AVCC0,
VSS = AVSS1 = AVSS0 = VREFL0 = 0 V, Ta = -40 °C to 125 °C

Item	Symbol	Min.	Max.	Unit	Reference Figure
Input capture input pulse width (single edge specified)	tTICW	1.5	—	tMTU	Figure 30.10
Input capture input pulse width (dual edge specified)	tTICW	2.5	—	tMTU	
Timer clock pulse width (single edge specified)	tTCKWH/L	1.5	—	tMTU	Figure 30.11
Timer clock pulse width (dual edge specified)	tTCKWH/L	2.5	—	tMTU	
Timer clock pulse width (in phase counting mode)	tTCKWH/L	2.5	—	tMTU	

Note: "tMTU" is the period of MTU operating clock.

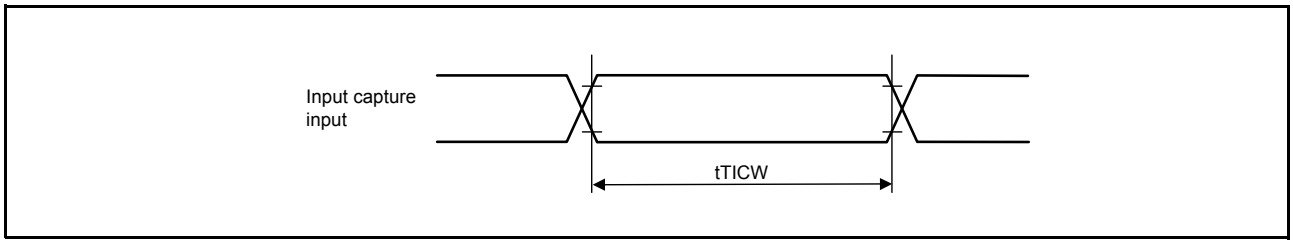


Figure 30.10 MTU-III Input Capture Input Timing

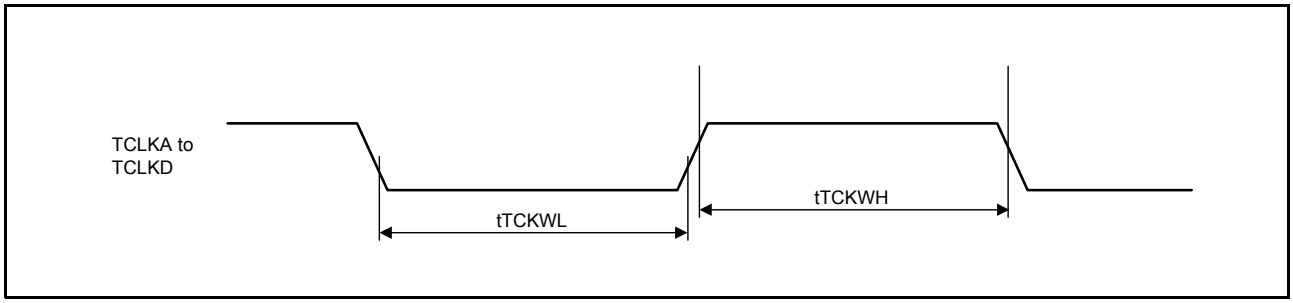


Figure 30.11 MTU-III Input Timing

30.3.9 Port Output Enable (POE) Timing

Table 30.19 shows the POE input timing.

Table 30.19 POE Input Timing (POEi#(i = 0 to 2))

Conditions: VCC = 5.0 V ± 0.5 V, AVCC1 = 5.0 V ± 0.5 V, AVCC0 = 5.0 V ± 0.5 V, VREFH0 = 4.5 V to AVCC0,
VSS = AVSS1 = AVSS0 = VREFL0 = 0 V, Ta = -40 °C to 125 °C

Item	Symbol	Min.	Max.	Unit	Reference Figure
POE input pulse width (digital filter is not used)	tPOEWL	400	—	ns	Figure 30.12
POE input pulse width (digital filter is used with the sampling period set to 128 cycles)	tPOEWL	129 × tPBA	—	ns	
POE input pulse width (digital filter is used with the sampling period set to 256 cycles)	tPOEWL	257 × tPBA	—	ns	
POE input pulse width (digital filter is used with the sampling period set to 2048 cycles)	tPOEWL	2049 × tPBA	—	ns	

Note: “tPBA” is the period of peripheral function clock A.

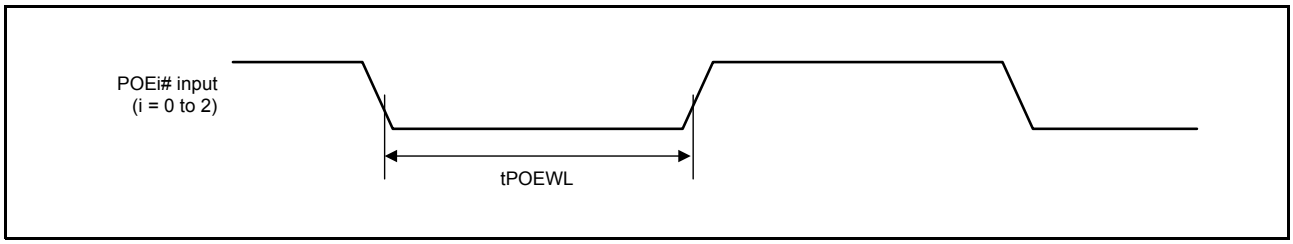


Figure 30.12 POEi# Input Timing

30.3.10 A/D Converter Timing

Table 30.20 shows the timing of input of an A/D converter external trigger (ADTRG#).

Note: The SH72A0 group does not have the ADTRG# pin.

Table 30.20 Timing of Input of an A/D Converter External Trigger (ADTRG#)

Conditions: VCC = 5.0 V ± 0.5 V, AVCC1 = 5.0 V ± 0.5 V, AVCC0 = 5.0 V ± 0.5 V, VREFH0 = 4.5 V to AVCC0,
VSS = AVSS1 = AVSS0 = VREFL0 = 0 V, Ta = -40 °C to 125 °C

Item	Symbol	Min.	Max.	Unit	Reference Figure
ADC external trigger input pulse width	tADTRGWL/H	2	—	tPBA	Figure 30.13

Note: “tPBA” is the period of peripheral function clock A.

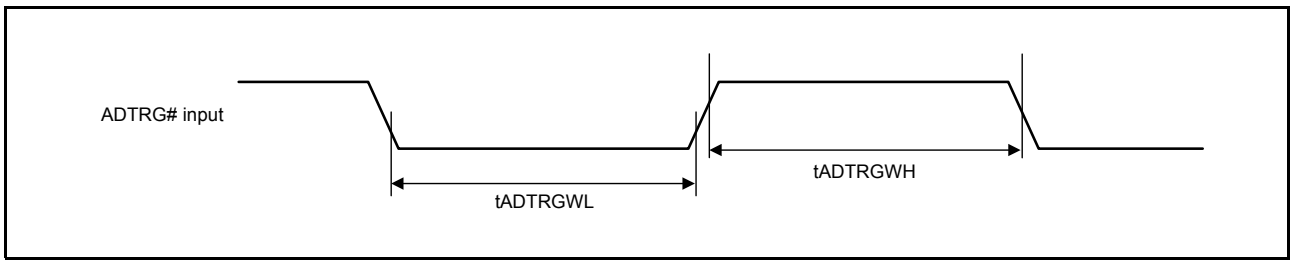


Figure 30.13 ADTRG# Input Timing

30.3.11 SCI Timing

Table 30.21 shows the SCI timing.

Table 30.21 SCI Timing

Conditions: VCC = 5.0 V ± 0.5 V, AVCC1 = 5.0 V ± 0.5 V, AVCC0 = 5.0 V ± 0.5 V, VREFH0 = 4.5 V to AVCC0,
VSS = AVSS1 = AVSS0 = VREFL0 = 0 V, Ta = -40 °C to 125 °C

Item		Symbol	Min.	Max.	Unit	Reference Figure
Input clock pulse width		tSCKW	0.4	0.6	tScyc	Figure 30.14
Input clock cycle	Asynchronous	tScyc	4	—	tSCI	
	Clock synchronous		8	—		
Input clock rising time		tSCKr	—	15	ns	Figure 30.14
Input clock falling time		tSCKf	—	15	ns	
Output clock cycle	Asynchronous	tScyc	32	—	tSCI	Figure 30.14
	Clock synchronous		8	—		
Output clock pulse width		tSCKW	0.4	0.6	tScyc	
Output clock rising time		tSCKr	—	15	ns	Figure 30.14
Output clock falling time		tSCKf	—	15	ns	
Transmit data delay time		tTXD	—	40	ns	Figure 30.15
Receive data set-up time (clock synchronous)		tRXS	30	—	ns	
Receive data hold time (clock synchronous)		tRXH	30	—	ns	

Note: “tSCI” is the period of SCI operating clock.

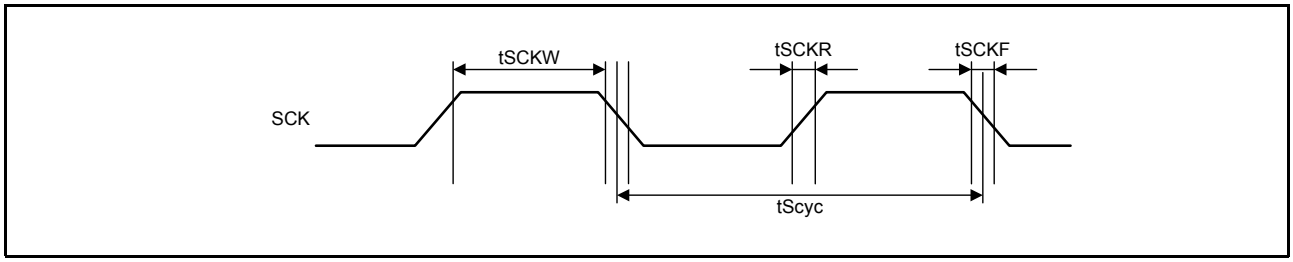


Figure 30.14 SCK Clock Input/Output Timing

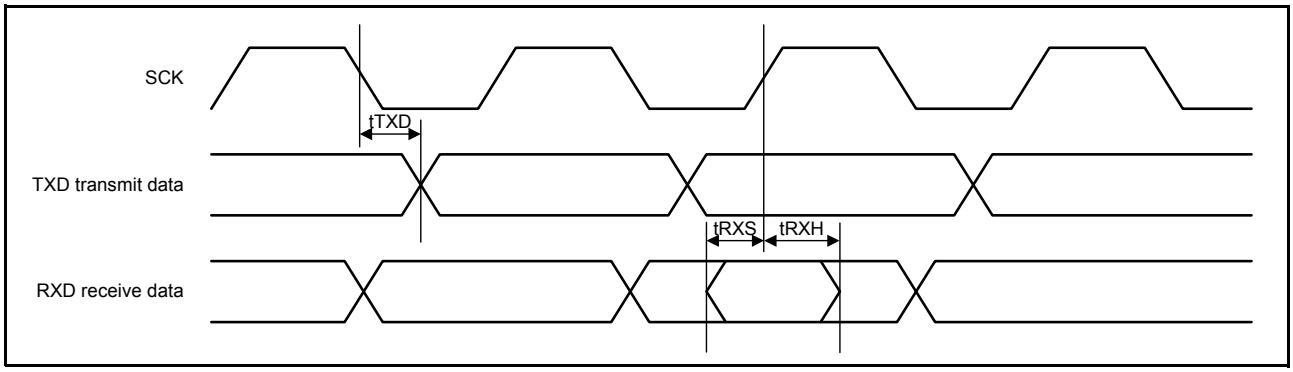


Figure 30.15 SCI Input/Output Timing (in Clock Synchronous Mode)

30.3.12 SBI Timing

Table 30.22 shows the SBI timing.

Table 30.22 SBI Timing

Conditions: VCC = 5.0 V ± 0.5 V, AVCC1 = 5.0 V ± 0.5 V, AVCC0 = 5.0 V ± 0.5 V, VREFH0 = 4.5 V to AVCC0,
VSS = AVSS1 = AVSS0 = VREFL0 = 0 V, Ta = -40 °C to 125 °C

Item		Symbol	Min.	Max.	Unit	Reference Figure
RSPCK cycle	Master	tSPcyc	8	4096	tPBB	Figure 30.16
	Slave		8	4096		
RSPI clock high-level pulse width	Master	tSPCKWH	(tSPcyc - tSPCKR - tSPCKF) / 2 - 3	—	ns	
	Slave		(tSPcyc - tSPCKR - tSPCKF) / 2	—		
RSPI clock low-level pulse width	Master	tSPCKWL	(tSPcyc - tSPCKR - tSPCKF) / 2 - 3	—	ns	
	Slave		(tSPcyc - tSPCKR - tSPCKF) / 2	—		
RSPCK clock rising/ falling time	Master	tSPCKR, tSPCKF	—	15	ns	
	Slave		—	1	µs	
Data input set-up time	Master	tSU	30	—	ns	Figures 30.17 to 30.20
	Slave		20 - 1 × tPBB	—		
Data input hold time	Master	tH	10	—	ns	
	Slave		20 + 2 × tPBB	—		
SSL set-up time	Master	tLEAD	1	8	tSPcyc	
	Slave		4	—	tPBB	
SSL hold time	Master	tLAG	1	8	tSPcyc	
	Slave		4	—	tPBB	
Data output delay time	Master	tOD	—	25	ns	
	Slave		—	3 × tPBB + 40		
Data output hold time	Master	tOH	–10	—	ns	
	Slave		0	—		
Continuous transition delay time	Master	tTD	tSPcyc + 2 × tPBB	8 × tSPcyc + 2 × tPBB	ns	
	Slave		4 × tPBB	—		
MOSI/MISO rising/ falling time	Output	tDR, tDF	—	15	ns	
	Input		—	1	µs	
SSL rising/falling time	Output	tSSLR, tSSLF	—	15	ns	
	Input		—	1	µs	
Slave access time		tSA	—	4	tPBB	Figures 30.19 and 30.20
Slave output release timing		tREL	—	3	tPBB	

Note: “tPBB” is the period of peripheral function clock B.

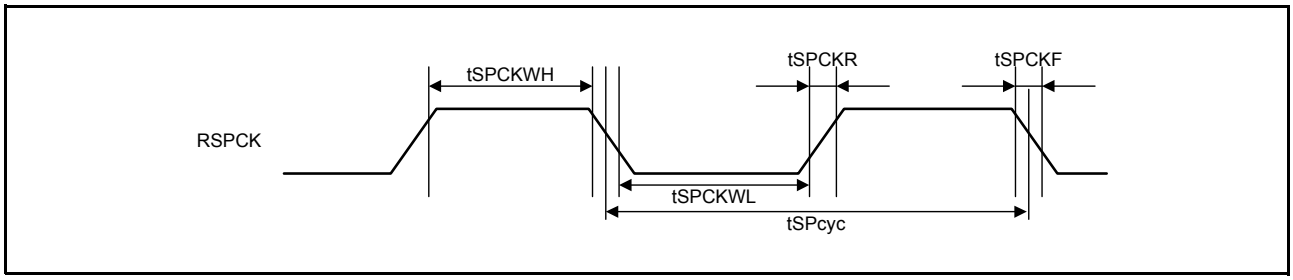


Figure 30.16 SBI Clock Timing

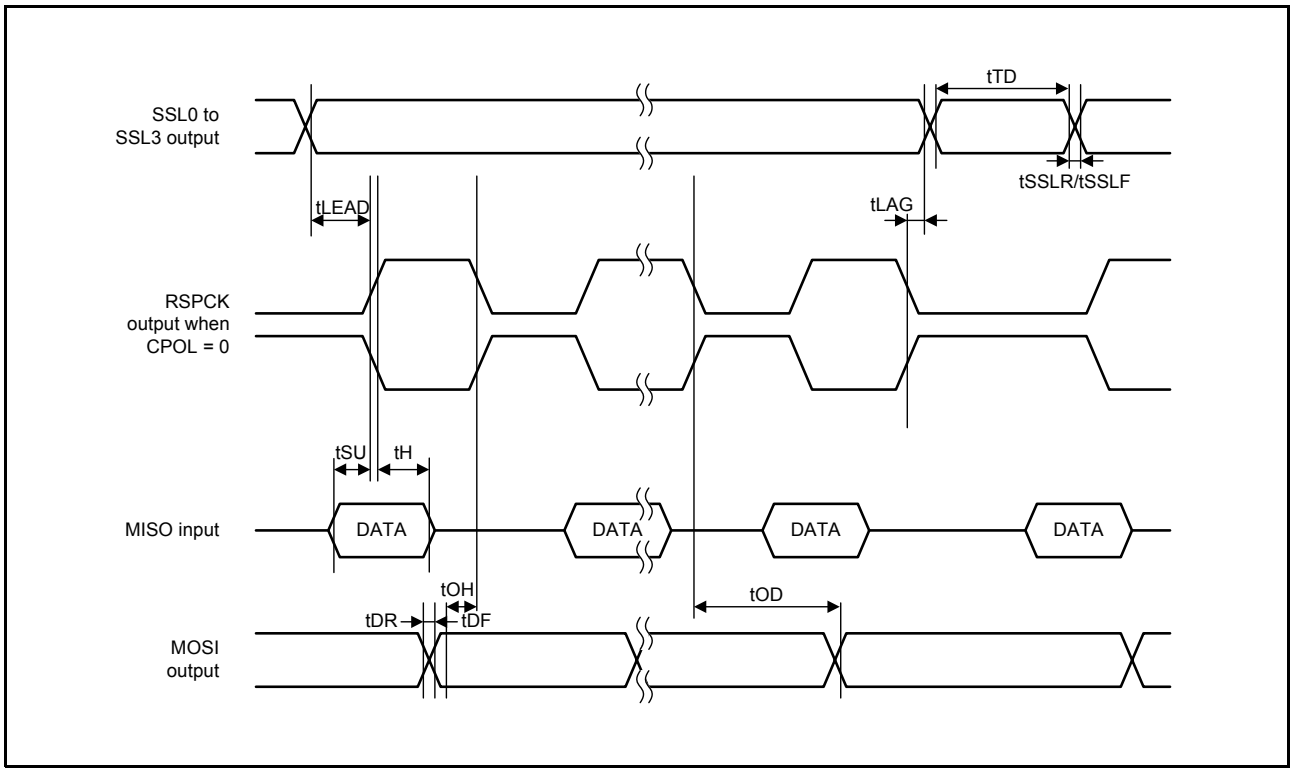


Figure 30.17 SBI Timing (in Master Mode with CPHA = 0)

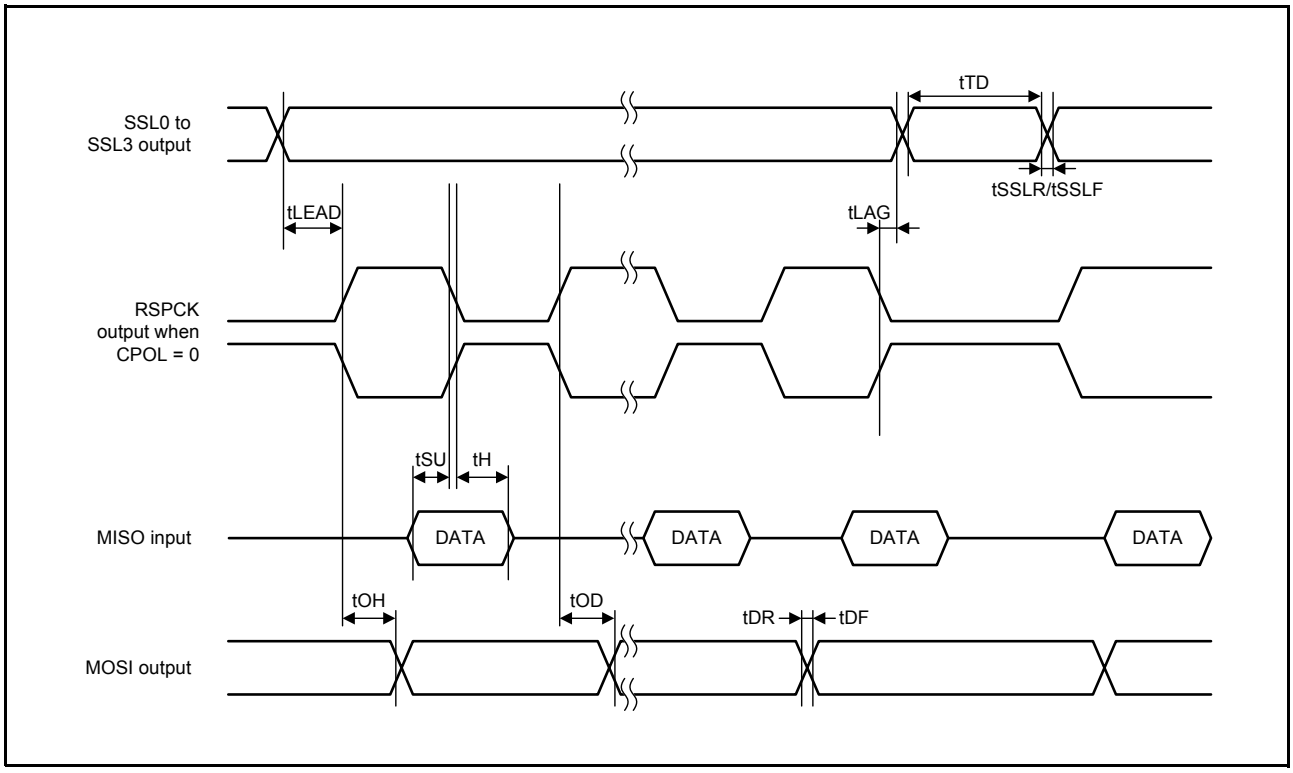


Figure 30.18 SBI Timing (in Master Mode with CPHA = 1)

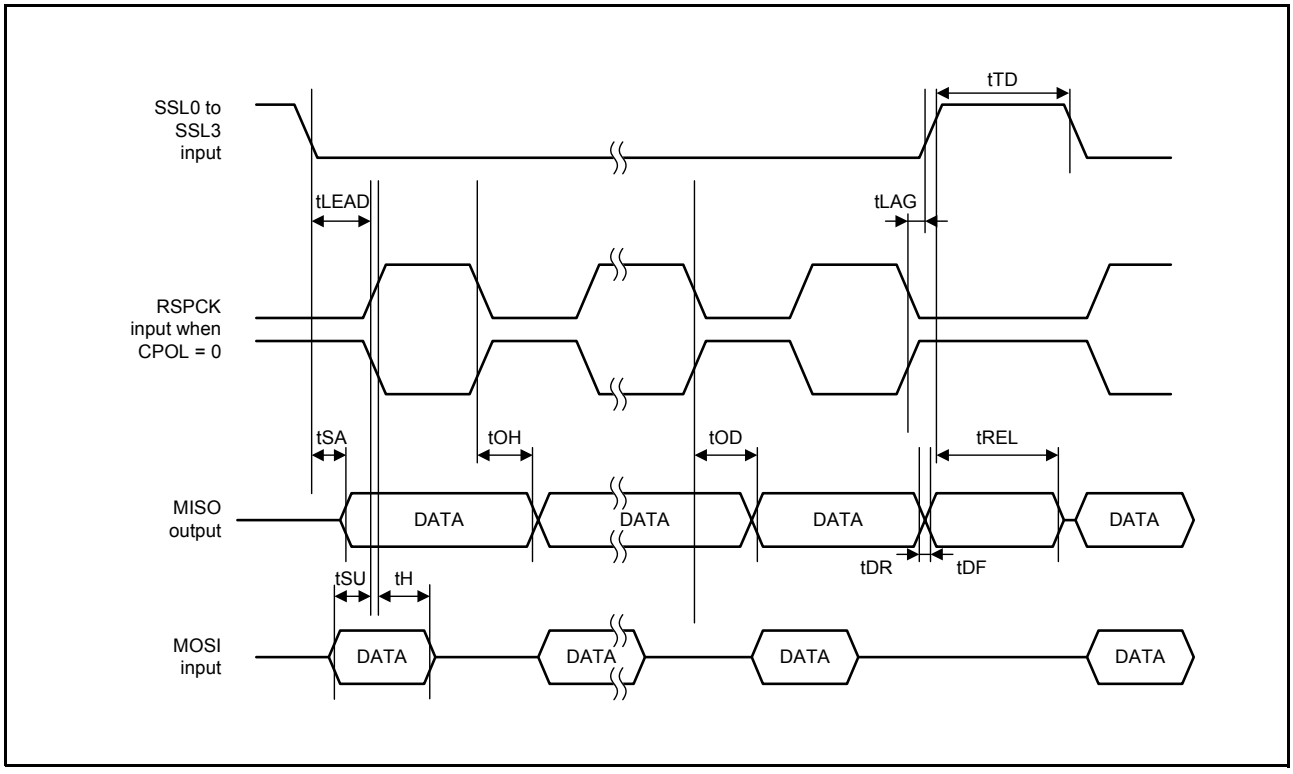


Figure 30.19 SBI Timing (in Slave Mode with CPHA = 0)

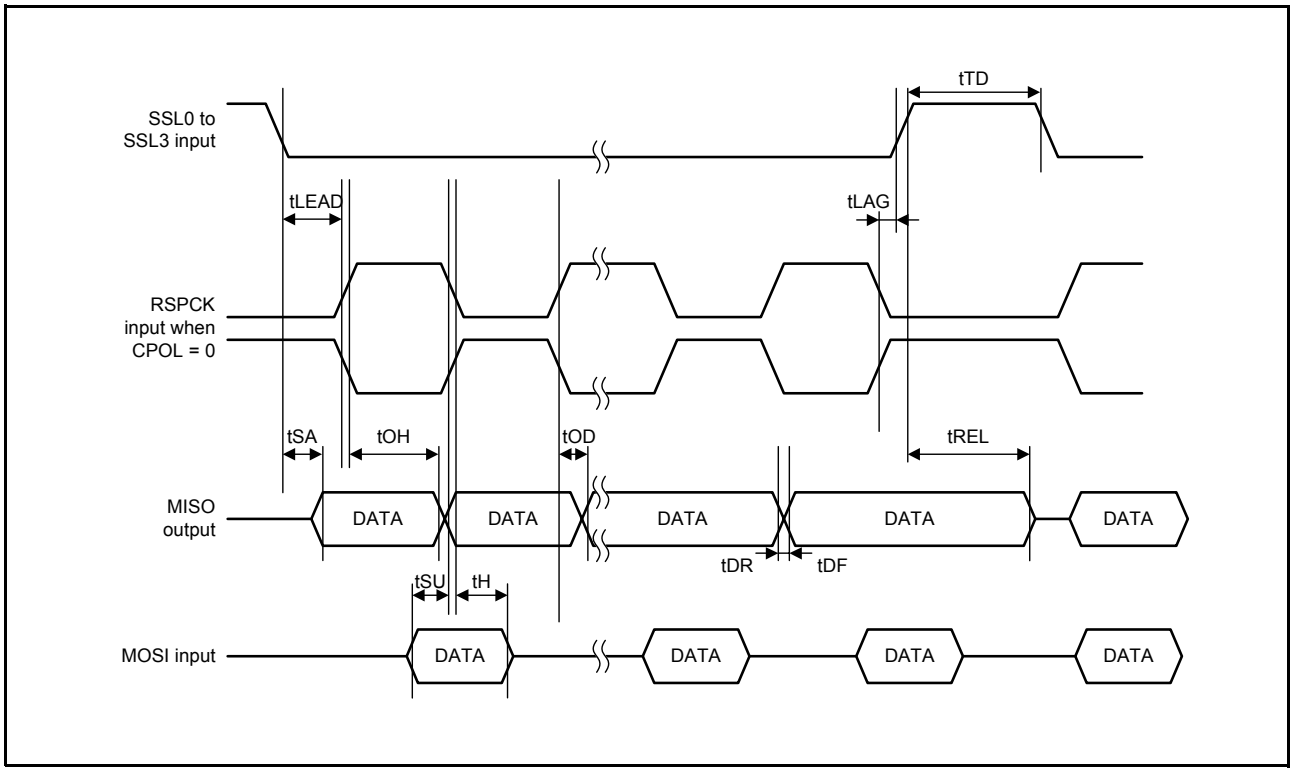


Figure 30.20 SBI Timing (in Slave Mode with CPHA = 1)

30.3.13 CAN Timing

Table 30.23 shows the CAN timing.

Table 30.23 CAN Timing

Conditions: VCC = 5.0 V ± 0.5 V, AVCC1 = 5.0 V ± 0.5 V, AVCC0 = 5.0 V ± 0.5 V, VREFH0 = 4.5 V to AVCC0,
VSS = AVSS1 = AVSS0 = VREFL0 = 0 V, Ta = -40 °C to 125 °C

Item	Symbol	Min.	Max.	Unit	Reference Figure
Internal delay time	tnode	—	100	ns	Figure 30.21
Transmission rate		—	1	Mbps	

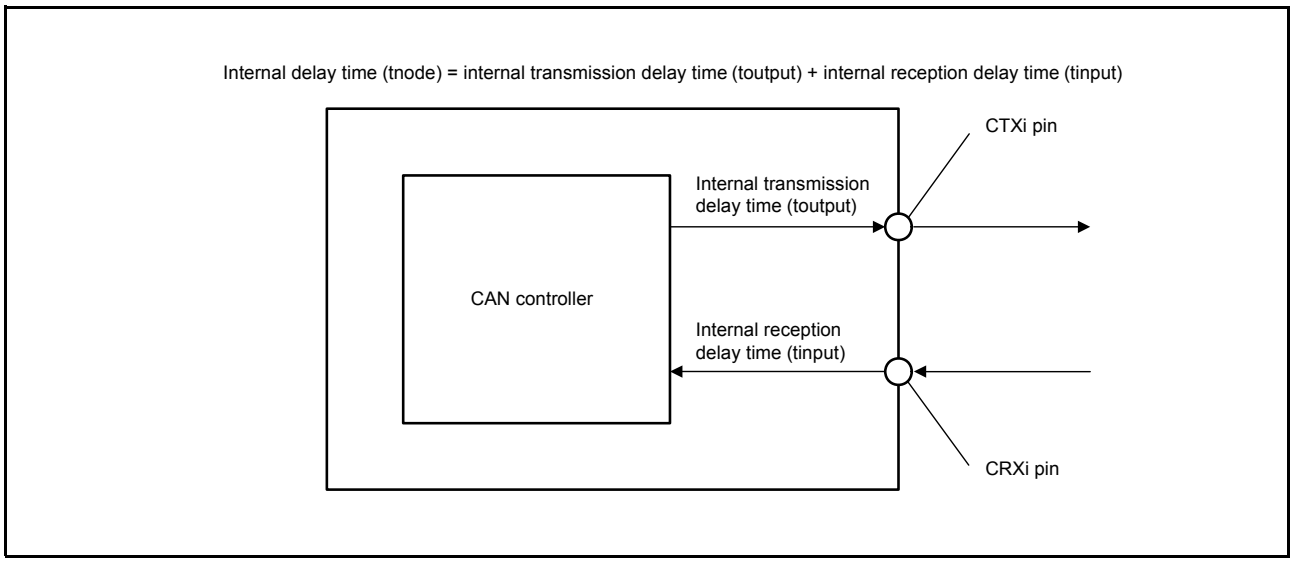


Figure 30.21 CAN Timing

30.3.14 AUD-II (Monitor Mode) Timing

Table 30.24 shows the AUD-II (monitor mode) timing.

Table 30.24 AUD-II (Monitor Mode) Timing

Conditions: VCC = 5.0 V ± 0.5 V, AVCC1 = 5.0 V ± 0.5 V, AVCC0 = 5.0 V ± 0.5 V, VREFH0 = 4.5 V to AVCC0,
VSS = AVSS1 = AVSS0 = VREFL0 = 0 V, Ta = -40 °C to 125 °C

Item	Symbol	Min.	Max.	Unit	Reference Figure
AUDCK cycle time	tAUCKMcyc	50	—	ns	Figure 30.22
AUDCK high-level width	tAUCKMH	0.4	—	tAUCKMcyc	
AUDCK low-level width	tAUCKML	0.4	—	tAUCKMcyc	
AUDRST set-up time	tAURSTMS	30	—	ns	
AUDRST pulse width	tAURSTMW	5	—	tAUCKMcyc	
AUDMD set-up time	tAUMDMS	5	—	tAUCKMcyc	
Monitor data output delay time	tAUDTMD	—	35	ns	Figure 30.23
Monitor data input set-up time	tAUDTMS	15	—	ns	
Monitor data input hold time	tAUDTMH	5	—	ns	
AUDSYNC input set-up time	tAUDSYS	15	—	ns	
AUDSYNC input hold time	tAUDSYH	5	—	ns	

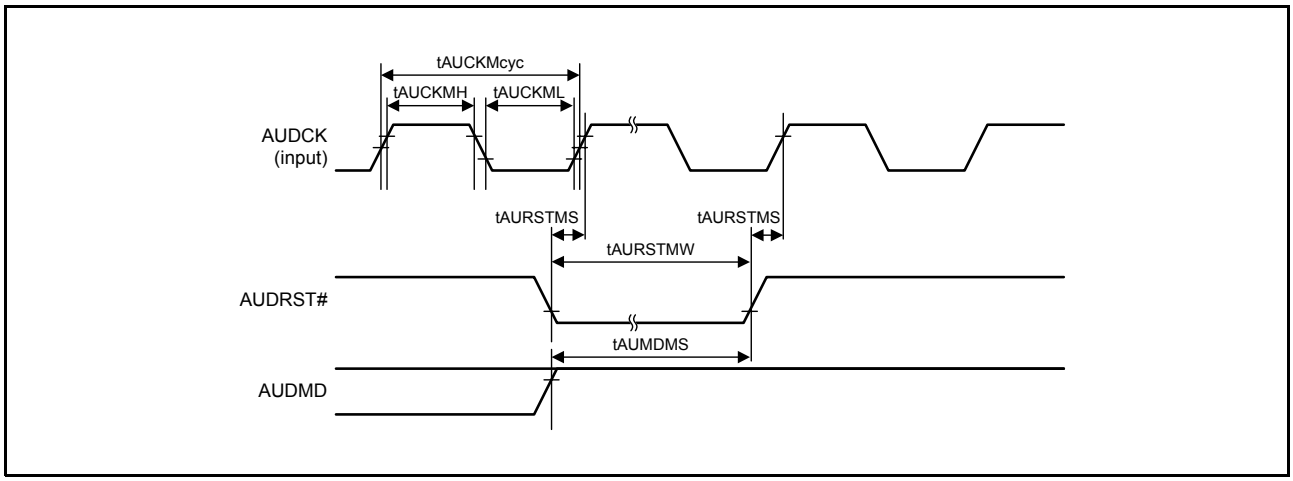


Figure 30.22 Monitor Mode Reset Timing

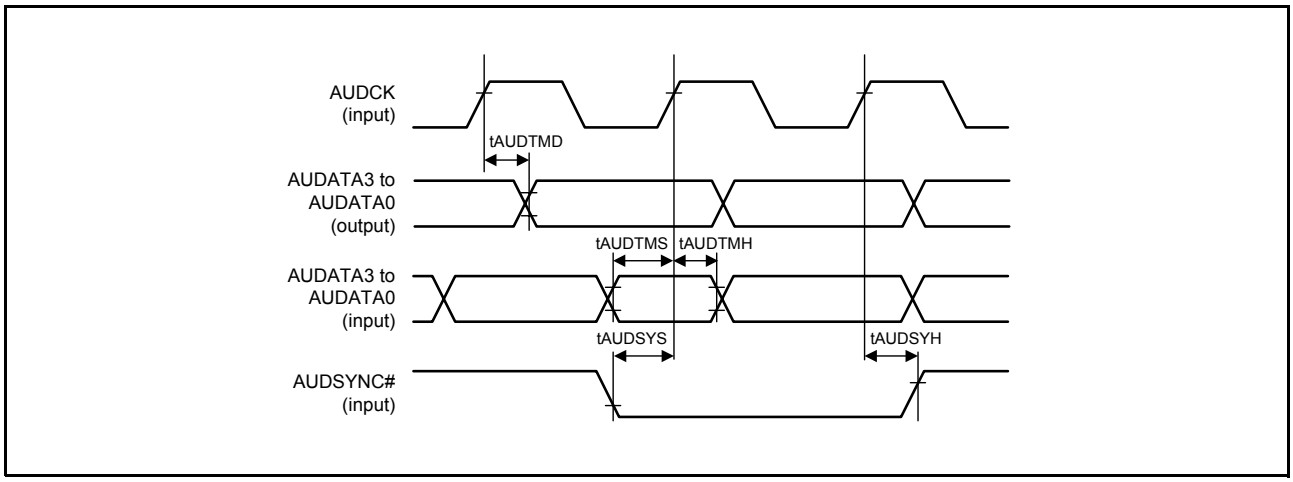


Figure 30.23 Monitor Mode Timing

30.3.15 AC Characteristics Measurement Conditions

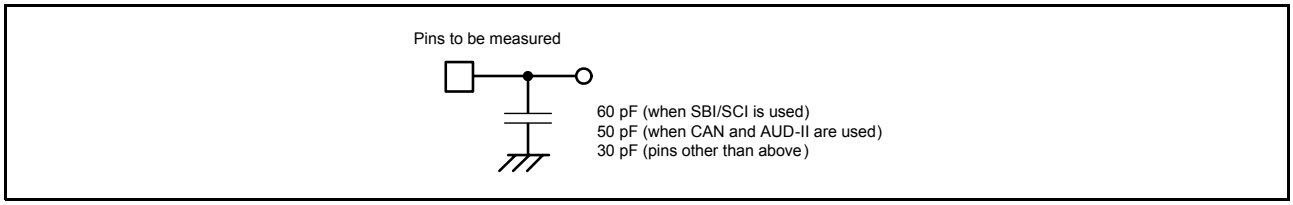


Figure 30.24 Circuit for Measuring Output Switching Characteristics

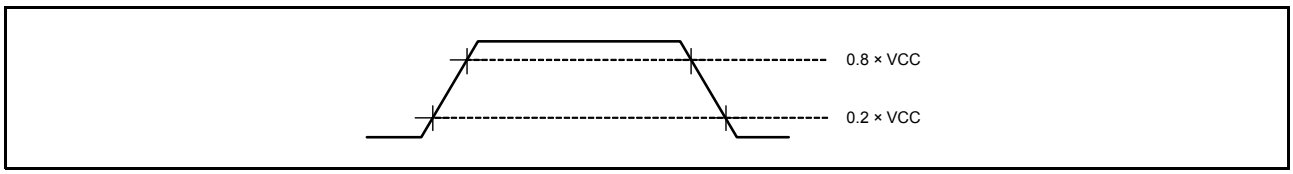


Figure 30.25 Output Determination Voltages in Measurement of the AC Characteristics

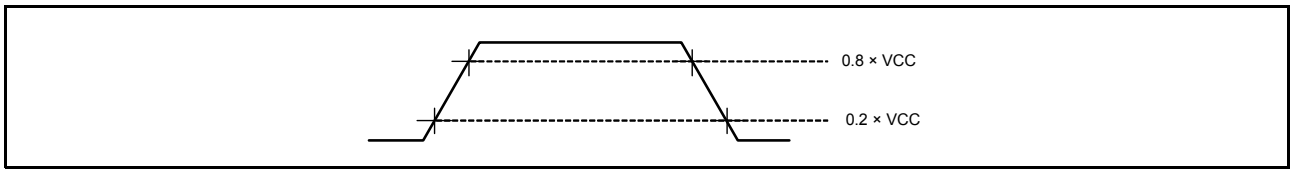


Figure 30.26 Input Determination Voltages in Measurement of the AC Characteristics (Schmitt-Trigger Type)

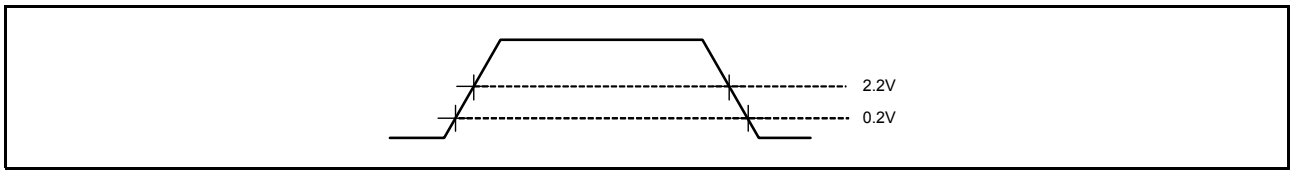


Figure 30.27 Input Determination Voltages in Measurement of the AC Characteristics (TTL Type)

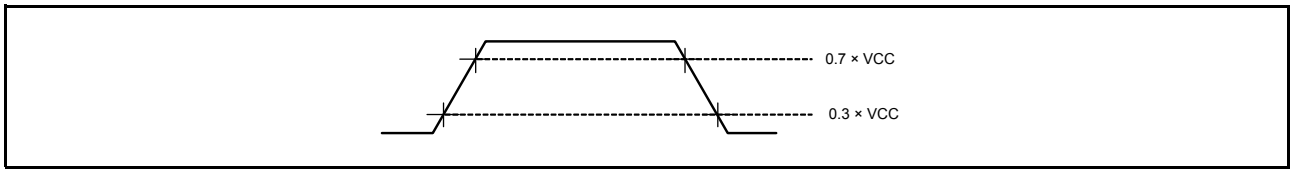


Figure 30.28 Input Determination Voltages in Measurement of the AC Characteristics (CMOS Type with Level Set to (0.5 × VCC))

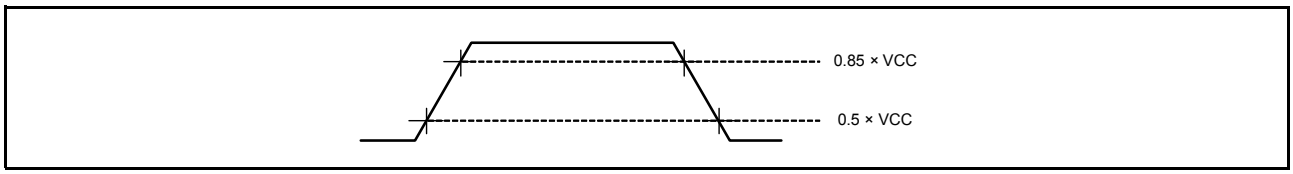


Figure 30.29 Input Determination Voltages in Measurement of the AC Characteristics (CMOS Type with Level Set to (0.7 × VCC))

30.4 ROM Characteristics

Table 30.25 ROM Characteristics

Conditions: VCC = 5.0 V ± 0.5 V, AVCC1 = 5.0 V ± 0.5 V, AVCC0 = 5.0 V ± 0.5 V, VREFH0 = 4.5 V to AVCC0,
VSS = AVSS1 = AVSS0 = VREFL0 = 0 V,
f(PBA) = 8 MHz to 40 MHz (SH72A0 Group) or 10 MHz to 50 MHz (SH72A2 Group), Ta = -40 °C to 125 °C

Item		Symbol	Min.	Typ.	Max.	Unit	Reference Figure
Programming time*1*2*4	256 bytes	t_{P256}	—	2	12	ms	
	8 Kbytes	t_{P8K}	—	45	100	ms	
Erasure time*1*2*4	8 Kbytes	t_{E8K}	—	50	150	ms	
	32 Kbytes*5	t_{E32K}	—	200	560	ms	
	64 Kbytes	t_{E64K}	—	400	1120	ms	
	128 Kbytes	t_{E128K}	—	800	2240	ms	
Number of cycles of programming and erasure		N_{PEC}	100*3	—	—	Cycles	
Suspension delay time during programming		t_{SPD}	—	—	120	µs	Figure 30.30
First suspension delay time during erasing		t_{SESD1}	—	—	120	µs	

Notes: 1. Programming and erasure times depend on the data.
2. Programming and erasure times do not include time taken to transfer the data.
3. This indicates the minimum number for which the characteristics are guaranteed after programming (the guaranteed values cover the range from one to this minimum number.)
4. This indicates the characteristics when programming is performed within the range of specifications, including that for the minimum number of cycles.
5. The size of the user-boot area is 32 Kbytes.

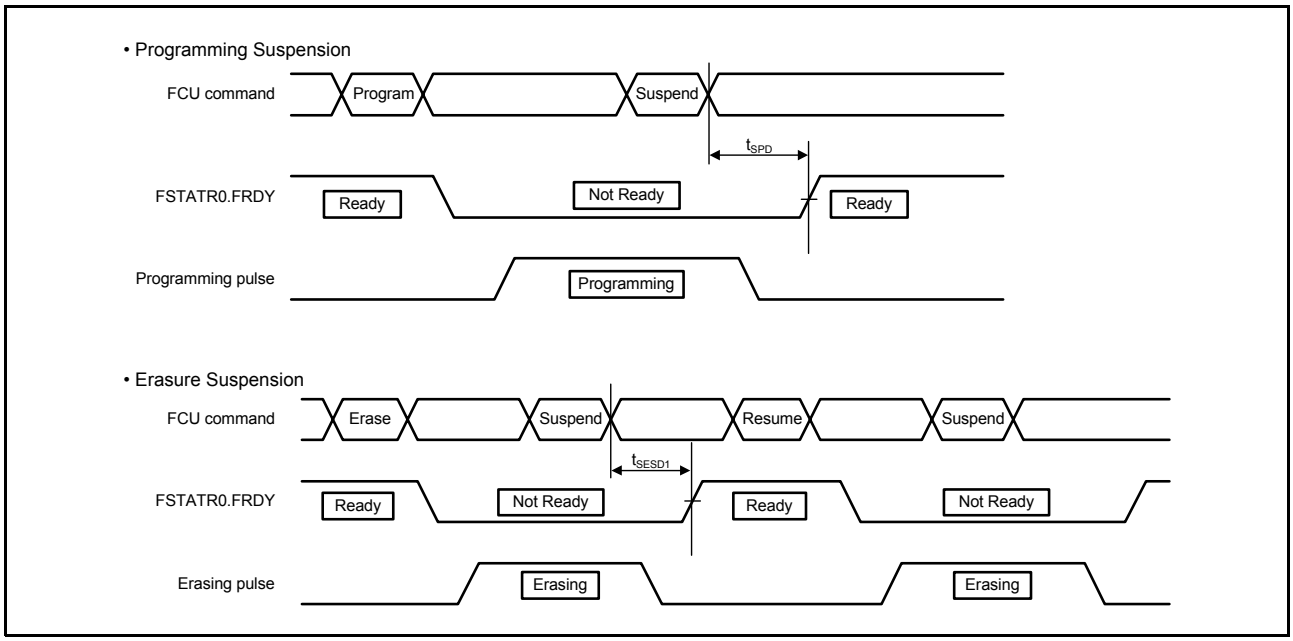


Figure 30.30 Timing of Flash Memory Programming/Erasure Suspension

30.5 EEPROM Characteristics

Table 30.26 shows the EEPROM characteristics.

Table 30.26 EEPROM Characteristics

Conditions: VCC = 5.0 V ± 0.5 V, AVCC1 = 5.0 V ± 0.5 V, AVCC0 = 5.0 V ± 0.5 V, VREFH0 = 4.5 V to AVCC0,
VSS = AVSS1 = AVSS0 = VREFL0 = 0 V,
f(PBA) = 8 MHz to 40 MHz (SH72A0 Group) or 10 MHz to 50 MHz (SH72A2 Group), Ta = -40 °C to 125 °C

Item		Symbol	Min.	Typ.	Max.	Unit	Reference Figure
Programming time*1*2*4	8 bytes	t_{P8_E}	—	0.4	2	ms	
	128 bytes	t_{P128_E}	—	1.0	5.0	ms	
Erasure time*1*2*4	2 Kbytes	t_{E2K_E}	—	70	250	ms	
Blank check time*1*4	8 bytes	t_{BC8_E}	—	—	30	µs	
	2 Kbytes	t_{BC2K_E}	—	—	0.7	ms	
Number of cycles of programming and erasure		N_{PEC_E}	30000 *3	—	—	Cycles	
Suspension delay time during programming		t_{SPD_E}	—	—	120	µs	Figure 30.30
First suspension delay time during erasing		t_{SESD1_E}	—	—	120	µs	
Data retention period*4		t_{DPR_E}	15	—	—	Years	

Notes: 1. Programming and erasure times depend on the data.
2. Programming and erasure times do not include time taken to transfer the data.
3. This indicates the minimum number for which the characteristics are guaranteed after programming (the guaranteed values cover the range from one to this minimum number.)
4. This indicates the characteristics when programming is performed within the range of specifications, including that for the minimum number of cycles.

30.6 12-Bit A/D Converter Characteristics

Table 30.27 shows the 12-Bit A/D converter characteristics.

Table 30.27 12-Bit A/D Converter Characteristics

Conditions: VCC = 5.0 V ± 0.5 V, AVCC1 = 5.0 V ± 0.5 V, AVCC0 = 5.0 V ± 0.5 V, VREFH0 = 4.5 V to AVCC0,
VSS = AVSS1 = AVSS0 = VREFL0 = 0 V, Ta = -40 °C to 125 °C

Item		Symbol	Min.	Max.	Unit	Reference Figure
Digital resolution		—	12	12	bit	
Voltage resolution*		—	1.10	1.34	mV	
Sample & hold sampling cycle		—	30	30	tPBA	
Sample & hold sampling time	SH72A0 Group (f(PBA) = 8 to 40 MHz)	—	0.75	3.75	µs	
	SH72A2 Group (f(PBA) = 10 to 50 MHz)	—	0.60	3.00	µs	
A/D conversion cycle		—	50	50	tPBA	
A/D conversion time	SH72A0 Group (f(PBA) = 8 to 40 MHz)	—	1.25	6.25	µs	
	SH72A2 Group (f(PBA) = 10 to 50 MHz)	—	1.00	5.00	µs	
Analog input voltage range fir guaranteed accuracy of conversion		—	AVSS0 + 0.25	AVCC0 - 0.25	V	
Nonlinearity error		—	—	±4.0	LSB	
Offset error		—	—	±7.5	LSB	
Full scale error		—	—	±7.5	LSB	
Quantization error		—	—	±0.5	LSB	
Absolute error		—	—	±8.0	LSB	
Analog input capacitance	Awaiting	—	—	20	pF	
	Sampling	—	—	40	pF	
Permitted analog signal source impedance		—	—	3	kΩ	
Absolute accuracy during self diagnosis		—	—	±40	LSB	

Notes: * At VREFH0 - VREFL0 = 4.5 V, the resolution is 1.10 mV.
At VREFH0 - VREFL0 = 5.5 V, the resolution is 1.34 mV.

- "tPBA" is the period of peripheral function clock A.

30.7 10-Bit A/D Converter Characteristics

Table 30.28 10-Bit A/D Converter Characteristics

Conditions: VCC = 5.0 V ± 0.5 V, AVCC1 = 5.0 V ± 0.5 V, AVCC0 = 5.0 V ± 0.5 V, VREFH0 = 4.5 V to AVCC0,
VSS = AVSS1 = AVSS0 = VREFL0 = 0 V, Ta = -40 °C to 125 °C

Item		Symbol	Min.	Max.	Unit
Digital resolution		—	10	10	bit
A/D conversion time (50 tPBA cycles)	SH72A0 Group (f(PBA) = 8 to 40 MHz)	—	1.25	6.25	µs
	SH72A2 Group (f(PBA) = 10 to 50 MHz)	—	1.00	5.00	µs
Analog input capacitance		—	—	20	pF
Permitted analog signal source impedance			—	1	KΩ
Nonlinearity error		—	—	±3.5	LSB
Offset error		—	—	±3.5	LSB
Full scale error		—	—	±3.5	LSB
Quantization error		—	—	±0.5	LSB
Absolute accuracy		—	—	±4.0	LSB
Absolute accuracy during self diagnosis		—	—	±16	LSB

Note: “tPBA” is the period of peripheral function clock A.

30.8 Other Characteristics

30.8.1 Voltage Monitor Circuit Characteristics

Table 30.29 Voltage Monitor Circuit Characteristics

Conditions: VCC = 5.0 V ± 0.5 V, AVCC1 = 5.0 V ± 0.5 V, AVCC0 = 5.0 V ± 0.5 V, VREFH0 = 4.5 V to AVCC0,
VSS = AVSS1 = AVSS0 = VREFL0 = 0 V, Ta = -40 °C to 125 °C

Item		Symbol	Min.	Typ.	Max.	Unit	Measurement conditions
Voltage monitor level	Voltage monitor circuit (LVD1)	Vdet1	3.70	3.90	4.10	V	
			3.95	4.15	4.35	V	
			4.10	4.30	4.50	V	
Minimum VCC falling time*		t_{VOFF}	200	—	—	µs	
Response delay time		tdet	—	—	200	µs	

Note: * The time power is off is the time when VCC is below the minimum LVD1 voltage monitor level (Vdet1).

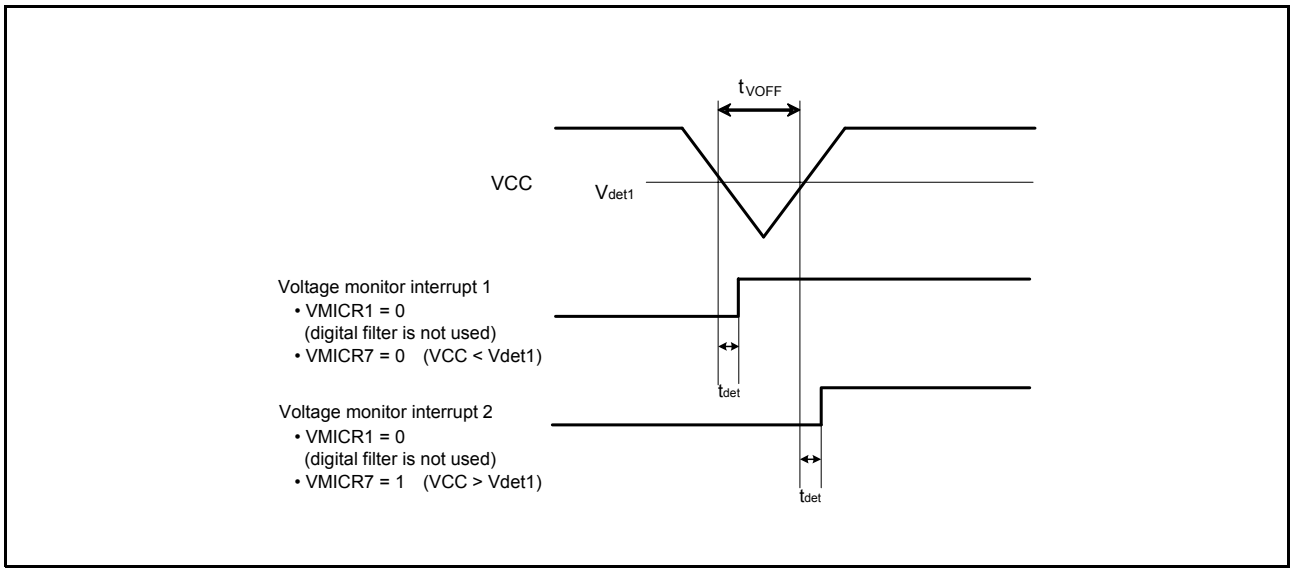


Figure 30.31 Voltage Monitor Circuit Timing (Vdet1)

30.8.2 External Oscillation Stop Detection Conditions

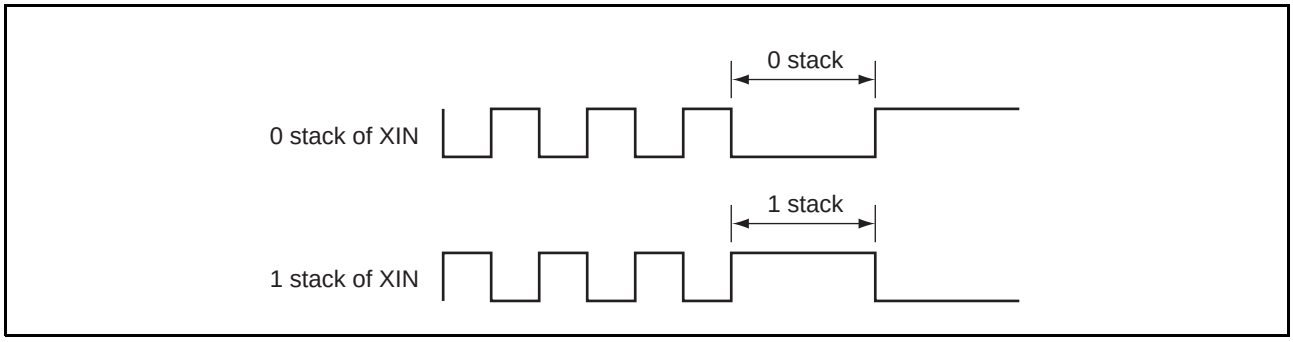


Figure 30.32 External Oscillation Stop Detection Conditions

Table 30.30 External Oscillation Stop Detection Conditions

Conditions: VCC = 5.0 V ± 0.5 V, AVCC1 = 5.0 V ± 0.5 V, AVCC0 = 5.0 V ± 0.5 V, VREFH0 = 4.5 V to AVCC0,
VSS = AVSS1 = AVSS0 = VREFL0 = 0 V, Ta = -40 °C to 125 °C

Item	Measurement Conditions	Reference Figure
Oscillation stop detection conditions	0 or 1 stack	Figure 30.32

30.8.3 External Oscillation Stop Detection Timing

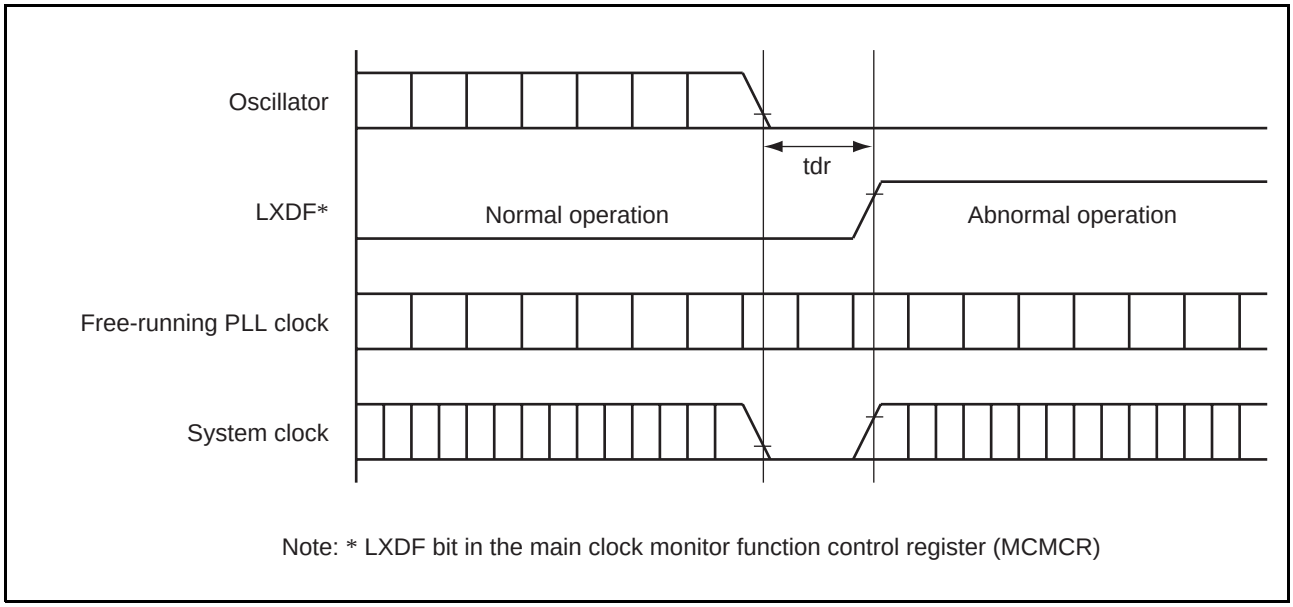


Figure 30.33 External Oscillation Stop Detection Timing

Table 30.31 Detection Time to Determine Abnormal Operation of External Oscillation Stop detection Circuit

Conditions: VCC = 5.0 V ± 0.5 V, AVCC1 = 5.0 V ± 0.5 V, AVCC0 = 5.0 V ± 0.5 V, VREFH0 = 4.5 V to AVCC0,
VSS = AVSS1 = AVSS0 = VREFL0 = 0 V, Ta = -40 °C to 125 °C

Item	Symbol	Min.	Typ.	Max.	Unit	Reference Figure
Detection time	tdr	—	—	10	µs	Figure 30.33

30.8.4 Low-Speed On-Chip Oscillator Clock Frequency

Table 30.32 Low-Speed On-Chip Oscillator Clock Frequency

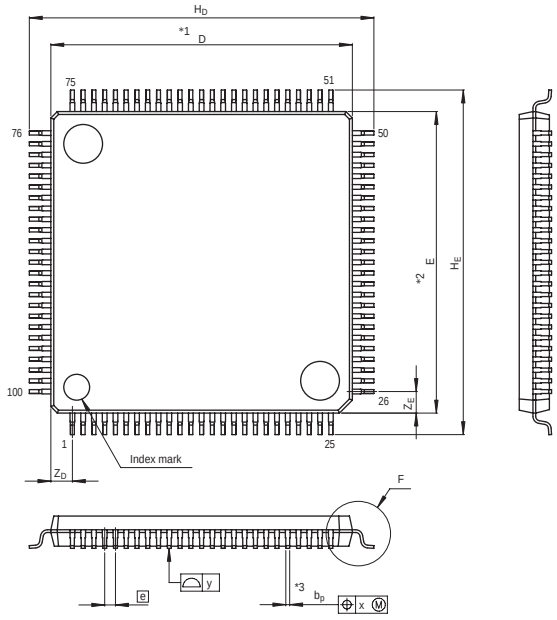
Conditions: VCC = 5.0 V ± 0.5 V, AVCC1 = 5.0 V ± 0.5 V, AVCC0 = 5.0 V ± 0.5 V, VREFH0 = 4.5 V to AVCC0,
VSS = AVSS1 = AVSS0 = VREFL0 = 0 V, Ta = -40 °C to 125 °C

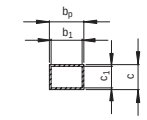
Item	Symbol	Min.	Typ.	Max.	Unit	Reference Figure
Low-speed on-chip oscillator clock frequency	$f_{(LOCO)}$	100	125	150	kHz	Figure 30.33

Appendix 1. Package Dimensions

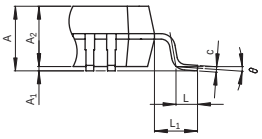
The latest information on package dimensions and details on packaging are available under "Package Items" on the Renesas Electronics website.

JEITA Package Code	RENESAS Code	Previous Code	MASS[Typ.]
P-LQFP100-14x14-0.50	PLQP0100KB-A	100P6Q-A / FP-100U / FP-100UV	0.6g





Terminal cross section

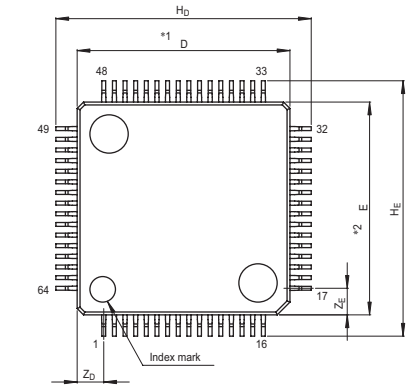


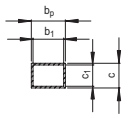
Detail F

NOTE)
1. DIMENSIONS "*1" AND "*2" DO NOT INCLUDE MOLD FLASH.
2. DIMENSION "*3" DOES NOT INCLUDE TRIM OFFSET.

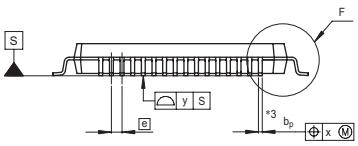
Reference Symbol	Dimension in Millimeters		
	Min	Nom	Max
D	13.9	14.0	14.1
E	13.9	14.0	14.1
A_2	—	1.4	—
H_D	15.8	16.0	16.2
H_E	15.8	16.0	16.2
A	—	—	1.7
A_1	0.05	0.1	0.15
b_p	0.15	0.20	0.25
b_1	—	0.18	—
c	0.09	0.145	0.20
c_1	—	0.125	—
θ	0°	—	8°
e	—	0.5	—
x	—	—	0.08
y	—	—	0.08
Z_D	—	1.0	—
Z_E	—	1.0	—
L	0.35	0.5	0.65
L_1	—	1.0	—

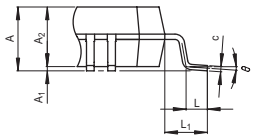
JEITA Package Code	RENESAS Code	Previous Code	MASS[Typ.]
P-LQFP64-10x10-0.50	PLQP0064KB-A	64P6Q-A / FP-64K / FP-64KV	0.3g





Terminal cross section





Detail F

NOTE)
1. DIMENSIONS "*1" AND "*2" DO NOT INCLUDE MOLD FLASH.
2. DIMENSION "*3" DOES NOT INCLUDE TRIM OFFSET.

Reference Symbol	Dimension in Millimeters		
	Min	Nom	Max
D	9.9	10.0	10.1
E	9.9	10.0	10.1
A_2	—	1.4	—
H_D	11.8	12.0	12.2
H_E	11.8	12.0	12.2
A	—	—	1.7
A_1	0.05	0.1	0.15
b_p	0.15	0.20	0.25
b_1	—	0.18	—
c	0.09	0.145	0.20
c_1	—	0.125	—
θ	0°	—	8°
e	—	0.5	—
x	—	—	0.08
y	—	—	0.08
Z_D	—	1.25	—
Z_E	—	1.25	—
L	0.35	0.5	0.65
L_1	—	1.0	—

SH72A2 Group, SH72A0 Group User's Manual: Hardware

Publication Date: Rev.1.00 Apr 20, 2012

Published by: Renesas Electronics Corporation

SALES OFFICES Renesas Electronics Corporation http://www.renesas.com

Refer to "http://www.renesas.com/" for the latest and detailed information.

Renesas Electronics America Inc.
2880 Scott Boulevard Santa Clara, CA 95050-2554, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

Renesas Electronics Canada Limited
1101 Nicholson Road, Newmarket, Ontario L3Y 9C3, Canada
Tel: +1-905-898-5441, Fax: +1-905-898-3220

Renesas Electronics Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-585-100, Fax: +44-1628-585-900

Renesas Electronics Europe GmbH
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-65030, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
7th Floor, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100083, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 204, 205, AZIA Center, No.1233 Lujiazui Ring Rd., Pudong District, Shanghai 200120, China
Tel: +86-21-5877-1818, Fax: +86-21-6887-7858 / -7898

Renesas Electronics Hong Kong Limited
Unit 1601-1613, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2886-9318, Fax: +852 2886-9022/9044

Renesas Electronics Taiwan Co., Ltd.
13F, No. 363, Fu Shing North Road, Taipei, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.
1 harbourFront Avenue, #06-10, keppel Bay Tower, Singapore 098632
Tel: +65-6213-0200, Fax: +65-6278-8001

Renesas Electronics Malaysia Sdn.Bhd.
Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics Korea Co., Ltd.
11F., Samik Lavied' or Bldg., 720-2 Yeoksam-Dong, Kangnam-Ku, Seoul 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

Colophon 1.1

SH72A2 Group, SH72A0 Group

Renesas Electronics Corporation

R01UH0164EJ0100